ABSTRACT
Energy storage systems are becoming a key component in smart grids with increasing renewable penetration. Storage technologies feature diverse capacity, charging, and response specifications. Investment and degradation costs may require charging batteries at multiple timescales, potentially matching the control periods at which grids are dispatched. To this end, a microgrid equipped with slow- and fast-responding batteries is considered here. Energy management decisions are taken at two stages. Slow-responding batteries are dispatched at an hourly resolution with decisions remaining invariant over multiple fast control slots. Building on Lyapunov optimization, slow- and fast-responding batteries are charged based on real-time and data-dependent with quantifiable sub-optimality bounds. Numerical tests using real data demonstrate the advantage of operating heterogeneous batteries.

Index Terms— Lyapunov optimization, stochastic approximation.

1. INTRODUCTION
With lowering costs, electric energy storage systems (ESS) constitute an efficient means for energy arbitrage, balance and reserve, frequency or voltage control, and peak shaving [1]. Built on diverse technologies, ESS exhibit heterogeneous characteristics in terms of capacities (1–100 kWh), response times (milliseconds to seconds), and (de)charging rates ranging from seconds to hours [2]; hence, complementing well power system tasks at various timescales.

Scheduling ESS is challenging due to decision coupling across time and the uncertainty involved in generation and costs. Storage scheduling solutions can be broadly classified into three groups. Approximate dynamic programming solvers typically incur high computational complexity and require the joint probability distribution function (pdf) of the related random processes to be known in advance; see e.g., [3], [4]. The second group comprises model predictive control (MPC)-schemes, where battery charging is tackled in a deterministic or stochastic fashion over a finite horizon that is progressively shifted as time advances [5], [6]; yet there are no performance guarantees. The third group involves real-time solutions stemming from Lyapunov optimization with relatively mild assumptions. Leveraging tools from stochastic networking [7], methods in this group relax time-coupling constraints and apply a modified greedy policy attaining feasible solutions with bounded suboptimality. In the smart grid context, the Lyapunov technique was first applied to harvest price differentials in [8], and to integrate energy storage in data centers [9]. A distributed implementation of online Lyapunov policies is derived in [10] as the coordination protocol between an energy aggregator and multiple storage devices. Coupling storage with load shedding, [11] puts forth a stochastic approximation view of Lyapunov minimization. The Lyapunov technique is modified in [12] to account for battery leakage and charging inefficiencies.

In all previous schemes, battery decisions are synchronized with the control period of the energy system. However, storage technologies operating at slower timescales may have to be employed to lower investment costs. Slower control rules could also be enforced by batteries having high degradation costs [2]. Hence, coordinating batteries at multiple timescales is practically relevant. A double-timescale Lyapunov energy management scheme for data centers is devised in [13]. Reference [14] combines non-ideal batteries with the latter framework for online management of communication stations with renewables and storage. Building on the latter, this work develops a real-time control scheme for coordinating batteries at two timescales. The scheme entails Lyapunov optimization at the fast timescale and stochastic approximation at the slow timescale to yield feasible solutions with bounded suboptimality gap. Numerical tests on real data show the advantage of heterogeneous storage.

2. MICROGRID MODELING
Consider a microgrid consisting of a photovoltaic, a variable load, and two energy storage units, which is coordinated by a controller as shown in Fig. 1. Due to heterogeneous storage technologies and the manner energy is exchanged between the microgrid and the main grid, control operations evolve in two timescales: The control horizon at the fast timescale is discretized into slots of equal duration indexed by $t$. A sequence of $T$ consecutive fast-timescale slots comprises a control period for the slow timescale indexed by $n = \lfloor t/T \rfloor$. Time $t$ can be then expressed as $t = nT + \tau$ to indicate the slow control period it belongs to, and the related offset $\tau$.

At the slow timescale, the microgrid exchanges energy
with the main grid through a time-ahead energy market and it operates the slower battery. The control decisions taken for slow period $n$ remain unchanged over the next $T$ fast control slots. If $S_n$ represent the state of charge (SoC) for the slow battery at the beginning of slow period $n$, and $B_n$ the amount by which the same unit is charged over period $n$, it holds

$$S_{n+1} = S_n + B_n$$  \hspace{1cm} (1a)
$$\underline{S} \leq S_n \leq \overline{S}$$  \hspace{1cm} (1b)
$$\underline{B} \leq B_n \leq \overline{B}$$  \hspace{1cm} (1c)

where (1a) captures battery dynamics; (1b) preserves the SoC within the capacity ($\underline{S}, \overline{S}$); and (1c) enforces charging rates ($\underline{B}, \overline{B}$). The microgrid buys energy $P_n$ from the main grid to charge the slow battery by $B_n$, while the remaining energy

$$E_n = P_n - B_n$$  \hspace{1cm} (2)

serves the load over the next $T$ fast control slots, and it is bounded as $E_n \in [\underline{E}, \overline{E}]$.

At the fast timescale, the controller collects information on solar generation and load demand, operates the fast battery, and exchanges energy with the main grid through a real-time market. Let $\phi_t$ denote the difference between demand and solar generation over fast period $t$. Similarly to (1), the SoC $s_t$ and the charge $b_t$ for the fast battery at time $t$ satisfy

$$s_{t+1} = s_t + b_t$$  \hspace{1cm} (3a)
$$\underline{s} \leq s_t \leq \overline{s}$$  \hspace{1cm} (3b)
$$\underline{b} \leq b_t \leq \overline{b}.$$  \hspace{1cm} (3c)

If $p_t$ is the energy bought from the real-time market at period $t$ and the amount of energy $E_n$ is delivered uniformly across the next $T$ fast times slots, then energy balance implies

$$p_t = \phi_t + b_t - \frac{E_n}{T}.$$  \hspace{1cm} (4)

Energy costs are modeled as convex increasing functions $C_n(P_n)$ and $c_t(p_t)$ for the time-ahead and the real-time market, respectively. Two functions are assumed random. If $G_n = \partial C_0(P_n)$ and $g_t = \partial c_t(p_t)$ denote the cost subgradients, define their extreme values as $\underline{G} := \min_n \{G_n\}$, $\overline{G} := \max_n \{G_n\}$, $\underline{g} := \min_t \{g_t\}$, and $\overline{g} := \max_t \{g_t\}$.

### 3. PROBLEM FORMULATION

Given that the time-ahead cost $C_n(P_n)$ occurs once every $T$ control periods and the real-time cost $c_t(p_t)$ at each control period $t$, the energy management task can be posed as

$$\phi_1^* = \min \lim_{N \to \infty} \frac{1}{NT} \sum_{n=0}^{N-1} \sum_{t=0}^{T-1} \mathbb{E} \left[ \frac{C_n(P_n)}{T} + c_t(p_t) \right]$$  \hspace{1cm} (5)

over $\{P_n, B_n, S_n, E_n\}, \{p_t, s_t, b_t\}$

where the expectation $\mathbb{E}$ is with respect to (wrt) $\{C_n, c_t, \phi_t\}$. Solving (5) is challenging as the randomness and the coupling across successive periods in (1a) and (3a). Conventional solutions based on approximate dynamic programming suffer from the curse of dimensionality and presume the joint pdf to be known [15]. Alternatively, Lyapunov optimization can approximately tackle infinite-horizon problems of particular structure by solving a sequence of relatively simple problems as time proceeds [7].

To transform the energy management task in (5) into the Lyapunov optimization framework, consider the problem

$$\phi_2^* = \min \lim_{N \to \infty} \frac{1}{NT} \sum_{n=0}^{N-1} \sum_{t=0}^{T-1} \mathbb{E} \left[ \frac{C_n(p_n)}{T} + c_t(p_t) \right]$$  \hspace{1cm} (6a)

over $\{P_n, B_n, E_n\}, \{p_t, b_t\}$

s.t. (1c), (2), (3c), (4)

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} B_n = 0$$  \hspace{1cm} (6b)

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=0}^{N-1} \sum_{t=0}^{T-1} b_t = 0$$  \hspace{1cm} (6d)

where constraints (1a)–(1b) and (3a)–(3b) appearing in (5) have been replaced by the time-averaged constraints (6c)–(6d) and variables $\{S_n, s_t\}$ have been eliminated. Problem (6) constitutes a relaxation of (5); see also [9], [11]. To see this, consider sequences $\{P_n, B_n, S_n, E_n\}$ and $\{p_t, s_t, b_t\}$ that are feasible for (5). Unfolding the dynamics (1a) and (3a) yields $S_n = S_0 + \sum_{n=0}^{N-1} B_n$ and $s_{NT} = s_0 + \sum_{n=0}^{N-1} \sum_{t=0}^{T-1} b_t$. Due to (1b) and (3b), the states $S_n$ and $s_{NT}$ are finite at all times. Dividing the previous equations by $N$ and $NT$ respectively, and sending $N$ to infinity provides (6c) and (6d). Therefore, the sequence $\{P_n, B_n, E_n\}$ and $\{p_t, b_t\}$ that are feasible for (5) are also feasible for (6).

Since (6) is a relaxation of (5), it follows that $\phi_2^* \leq \phi_1^*$. If an algorithm solves (6) with a suboptimality gap of $\epsilon$, and yields $\{P_n, B_n, E_n, p_t, b_t\}$ that are feasible for (5), then this algorithm attains an optimal value $\phi_2^*$ for which $\phi_1^* \leq \phi_2^* \leq \phi_2^* + \epsilon$. Combining the latter with $\phi_2^* \leq \phi_1^*$ proves that
algorithm would be $\epsilon$-suboptimal for (5) too, i.e., $\phi_1^* \leq \phi_2^* \leq \phi_1^* + \epsilon$. Since (6) does not involve time-coupling constraints, it is easier to solve than (5). An online approximate solver for (6) yielding control decisions feasible for (5) is derived next.

The Lyapunov technique introduces two queues $X_n$ and $x_t$ and minimizes the drift plus penalty cost for all $n$ [16]:

$$
\min_{X_n, B_n} X_n B_n + V C_n(P_n) + \sum_{\tau=0}^{T-1} \mathbb{E} [x_t b_t + V c_t (p_t)]
$$

over $P_n, B_n, E_n, \{p_t, b_t\}$

s.to (1c), (2), (3c), (4)

where the virtual queues relate to the SoCs as

$$
X_n := S_n + \Gamma, \quad x_t := s_t + \gamma
$$

for constants $\Gamma$ and $\gamma$ to be specified later. Heed that the expectation in the cost of (7) is now only wrt $(c_t, \ell_t)$. Nonetheless, problem (7) is still challenging since it involves expectations over the future values of the queue parameter $x_t$. Similarly to [13], [14], to overcome this difficulty, the queue values $\{x_t\}_{t=0}^{T-1}$ are replaced by $x_n$, and the resultant problem is handled using stochastic approximation.

Upon the aforesaid simplification and if $\{c_t, \ell_t\}$ are iid, problem (7) is substituted by

$$
\min_{E_n \leq E} H(E_n; X_n) + T \mathbb{E} [F_t(E_n; x_n)]
$$

where the function $H(E_n; X_n)$ is defined as

$$
H(E_n; X_n) := \min_{B_n \leq B, \leq \overline{B}} X_n B_n + V C_n(B_n + E_n)
$$

and each $F_t(E_n; x_n)$ relies on a single realization $(c_t, \ell_t)$ as

$$
F_t(E_n; x_n) := \min_{b_t \leq b_t} x_n b_t + V c_t (\ell_t + b_t - E_n / T).
$$

Because functions $H(E_n; X_n)$ and $\{F_t(E_n; x_n)\}$ are convex, the minimization in (9) is convex too.

Minimizing (9) over $E_n$ could be solved via a projected subgradient scheme. A subgradient of $H(E_n; X_n)$ is $V G_n(B_n^j + E_n^j)$ with $B_n^j$ being a minimizer of (10) attaining $H(E_n^j; X_n)$. Likewise, a subgradient of $F_t(E_n; x_n)$ is $-1/T g_t (\ell_t + b_t^j - E_n^j / T)$, where $b_t^j$ is a minimizer of (11) attaining $F_t(E_n^j; x_n)$. The $j$-th subgradient update reads

$$
E_n^{j+1} = E_n^j - \mu_j V G_n(B_n^j + E_n^j) - 1/T \mathbb{E} [g_t (\ell_t + b_t^j - E_n^j / T)]
$$

for $\mu_j > 0$. Observe that updating $E_n$ requires solving (10) once, but also infinitely many problems of the form in (11).

To avoid the computational burden, stochastic approximation surrogate the previous update with a single evaluation of the related stochastic subgradient $F_t(E_n; x_n)$, that is

$$
E_n^{j+1} := E_n^j - \mu_j V G_n(B_n^j + E_n^j)
$$

### Algorithm 1 Two Timescale Storage Management Scheme

```
1: for $t = 0, 1, 2, \ldots$ do
2: if $t/T$ is integer then
3: Set $n = t/T$ and observe cost $C_n$.
4: Set $E_n^0 = E_{n-1}^n$ and $X_n = S_n + \Gamma$.

5: for $j = 0, 1, 2, \ldots$ do
6: Draw sample $(c_t, \ell_t)$ and solve (11) for $E_n^j$.
7: Update $E_n^{j+1}$ from (12).

8: end for
9: Find $B_n^j$ by solving (10) for $E_n^j$.
10: Buy energy $P_n = B_n^j + E_n^j$ from main grid.

11: end if
12: Observe $(c_t, \ell_t)$ and set $x_t = s_t + \gamma$.
13: Find $b_t^j$ from (11) for $E_n^*$. 
14: Buy energy $p_t = \ell_t + b_t^j - E_n^*/T$ from main grid.

15: end for
```

For $\mu_j = \mu / j$ with $\mu > 0$, the stochastic subgradient update of (12) is guaranteed to converge to a minimizer of (7). The charge $B_n^*$ can now be found as the minimizer of (10) for $E_n^*$. Having found $(E_n^*, B_n^*)$, the real-time control decisions $(p_t, b_t)$ for the next $T$ fast time slots can be found by solving (11) for $E_n^*$. Steps 2–11 of Alg. 1 precede the slow control period $n$, and Steps 12–14 correspond to fast control slots.

### 4. ALGORITHM PERFORMANCE

We first provide the conditions under which the charging decisions of Alg. 1 are feasible for problem (5) [9], [13]

**Proposition 1.** Under the mild assumptions that $\mathcal{S} - \mathcal{S} \geq \mathcal{B} - \mathcal{B}$ and $\bar{s} - \underline{s} \geq T(\bar{b} - \underline{b})$, the control decisions $\{B_n^j\}$ and $\{b_t^j\}$ found by Alg. 1 are feasible for problem (5) if the parameters $(\Gamma, \gamma, \gamma)$ satisfy:

$$
-V \mathcal{G} + \mathcal{B} - \mathcal{S} \leq \Gamma \leq -V \mathcal{G} + \mathcal{B} - \mathcal{S}
$$

$$
-V \mathcal{G} + T \bar{b} - \underline{s} \leq \gamma \leq -V \mathcal{G} + T \bar{b} - \underline{s}
$$

$$
0 < \mathcal{V} \leq \mathcal{V}
$$

where $\mathcal{V} = \min \{\frac{\bar{s} - \underline{s} + \bar{b} - \underline{b}}{2}, \frac{\bar{s} + \bar{b} + T(\underline{b} - \underline{b})}{2}\}$.

**Proof of Prop. 1.** Proving (13a) by mathematical induction, it is shown next that if $S_n \in [\underline{S}, \bar{S}]$, the same holds for $S_{n+1}$. From (1a) and (8), it follows that $S_{n+1} = S_n + B_n = X_n - \Gamma + B_n$, where $B_n$ is the minimizer of (10). Depending on the queue value $X_n$, three cases can be considered:

**C1** If $X_n \leq -V \mathcal{G}$, it is easy to see that $B_n = \mathcal{B}$, and hence, $S_{n+1}$ can only increase compared to $S_n$. To ensure $S_{n+1} \leq X_n - \Gamma + \mathcal{B} \leq \mathcal{S}$, it suffices that $\Gamma \geq -V \mathcal{G} + \mathcal{B} - \mathcal{S}$.

...
(C2) If $X_n \geq -V \overline{G}$, it holds that $B_n = B$, and hence, $S_{n+1}$ can only decrease compared to $S_n$. To ensure $S_{n+1} = X_n - \Gamma + B \geq S$, it suffices that $\Gamma \leq -V \overline{G} + B - S$.

(C3) When $-V \overline{G} \leq X_n \leq -V \overline{G}$, the minimizer has to be feasible $B_n \in [B, \overline{B}]$. The previous limits on $X_n$ are valid since $\overline{G} \leq \overline{G}$ and $V > 0$. A sufficient condition ensuring $S_{n+1} \leq S$ is $\Gamma \leq -V \overline{G} + B - S$ and a sufficient condition ensuring $S_{n+1} \geq S$ is $\Gamma \leq -V \overline{G} + B - S$. Claim (13a) follows since the limits under case (C3) are tighter than the respective limits under (C1) and (C2).

Claim (13b) is shown likewise. From (3a) and (8), it holds that $s_n + 1 = s_n + \sum_{\tau=0}^{T-1} b_T = x_n - \gamma + \sum_{\tau=0}^{T-1} b_t$, where $b_T$ is a minimizer of (11). Based on $x_n$, three cases are distinguished:

(c1) If $x_n \leq -V \overline{g}$, then $\{b_T = 0\}_{T=0}^{T-1}$. Thus, $s_{n+1}$ is larger than $s_n$ and $s_{n+1} \geq s$ is ensured if $\gamma \leq -V \overline{g} + T \overline{B} - s$.

(c2) If $x_n \geq -V \overline{g}$, then $\{b_T = 1\}_{T=0}^{T-1}$. Thus, $s_{n+1}$ is smaller than $s_n$ and $s_{n+1} \leq s$ is ensured if $\gamma \geq -V \overline{g} + T \overline{B} - s$.

(c3) When $-V \overline{g} \leq x_n \leq -V \overline{g}$, the minimizers $\{b_T\}_{T=0}^{T-1}$ lie in $[0, 1]$. Guaranteeing $s_{n+1} \in [s, \overline{s}]$ is assured if $\gamma \geq -V \overline{g} + T \overline{B} - s$ and $\gamma \leq -V \overline{g} + T \overline{B} - s$. These two bounds are tighter than those obtained under (c1)-(c2), and (13b) follows. Ensuring $s_n \in [\overline{s}, s]$ implies that $s_T \in [\overline{s}, s]$ at all times.

Bounding $V$ by $\overline{V}$ in (13c) assures that the upper bounds in (13a)-(13b) are larger than the related lower bounds, and hence, $\Gamma$ and $\gamma$ are implementable. □

Lemma 1 ([7]). Let $\{P_n^i, B_n^i, P_n^t, p_n^t, b_n^t\}$ be the decisions under a policy that selects them based solely on the current realization $\{c_n, c_t, \ell_t\}$. If states are iid over time, there exists one such policy satisfying (1c), (3c), (2), and (4), for which:

$E[B_n^i] = 0, E[b_n^t] = 0, E[C_n(P_n^i)/T + c_t(p_n^t)] = \phi^*_i$. (14)

The next result upper bounds $\hat{\phi}_i$ and asserts that $V = \overline{V}$ yields the tightest bound while maintaining feasibility.

Proposition 2. If $\{\ell_t\}$ are iid over time, then $\hat{\phi}_i \leq \phi^*_i + \frac{K_B}{2 T V} + \frac{T K_b}{2}$, where $K_B := \max\{\overline{B}^2, \overline{B}^2\}$ and $K_b := \max\{\overline{b}^2, \overline{b}^2\}$.

Proof of Prop. 2. Define the Lyapunov function $L_t := \frac{1}{2}(X_n^2 + X_n^2)$ and the $T$-slot Lyapunov drift $\Delta_t := E[L_t|T| - L_t|X_t, x_t]$ for $t = n T$. Using (1a), (3a), and (8) yields

$\Delta_n T = \frac{1}{2} E[X_{n+1}^2 - X_n^2 + x_{n+1}^2 - x_n^2 | X_n, x_n]$

$= \frac{1}{2} E \left[ 2 X_n B_n + B_n^2 + 2 x_n \sum_{\tau=0}^{T-1} b_t + \left( \sum_{\tau=0}^{T-1} b_t \right)^2 | X_n, x_n \right]$

$\leq E \left[ X_n B_n + x_n \sum_{\tau=0}^{T-1} b_t | X_n, x_n \right] + \frac{K_B + T^2 K_b}{2}$ (15)

Define $\phi_n^t = C_n(P_n^t)/T + c_t(p_n^t); \Delta t \sum_{\tau=0}^{T-1} E[\phi_n^t | X_n, x_n]$ on both sides of (15); and rearrange to get

$\Delta_n T + V \sum_{\tau=0}^{T-1} E[\phi_n^t | X_n, x_n] \leq E[X_n B_n + x_n \sum_{\tau=0}^{T-1} b_t | X_n, x_n]$

Fig. 2. Time-averaged microgrid operation cost.

$+ V \sum_{\tau=0}^{T-1} E[\phi_n^t | X_n, x_n] + \frac{K_B + T^2 K_b}{2}$. (16)

Notice that the minimization of the right-hand side of (16) coincides with (7). Hence, the value of (16) attained by Alg. 1 would be the minimum over all feasible policies, including the one of Lemma 1. From (14) if follows:

$\Delta_n T \sum_{\tau=0}^{T-1} E[\phi_n^t | X_n, x_n] \leq V T \phi^*_i + \frac{K_B + T^2 K_b}{2}$. (17)

Taking expectations on both sides of (17) wrt $(X_n, x_n)$; applying the law of total expectation; and summing over $N$ consecutive slow intervals yields $V \sum_{n=0}^{N-1} \sum_{\tau=0}^{T-1} E[\phi_n^t] \leq V N T \phi^*_i + \frac{N (K_B + T^2 K_b)}{2} - E[L_{NT} - L_0]$. Prop. 2 is obtained by dividing both sides of the last inequality by $V N T$; taking $N$ to infinity; and noting $E[L_0]$ is finite and $E[L_{NT}] \geq 0$. □

5. NUMERICAL TESTS

Algorithm 1 was tested using 5-min load data from Home C of the Smartp project [17], scaled up by a factor of 10 and repeated to yield 20 weeks of load $\ell_t$ [17]. Costs $\{C_n, c_t\}$ were modeled as convex piecewise linear with different sell- and buying prices. Buying prices for $C_n$ were taken as the hourly day-ahead prices for the Michigan hub in the MISO market over April, 2015, repeated to match the duration of $\ell_t$. Buying prices for $c_t$ were simulated as uniformly distributed having the related $C_n$ as mean and $10/MWh$ as variance. Selling prices for $C_n$ and $c_t$ were set to 0.9 times the buying prices. The faster timescale had $T = 12$ fast intervals. Battery parameters were set to $\overline{S} = 1$ MWh, $\overline{s} = 84$ kWh, $\overline{g} = s = 0$ kWh, and $\overline{B} = B = \overline{b} = \overline{b} = 10$ kW, and $E_{-} = E_{+} = 30$ kWh. Figure 2 depicts the time-averaged operational costs for three microgrid scenarios. The curves demonstrate that adding heterogeneous batteries lowers the operational cost compared to the other two cases.
6. REFERENCES


