Peak Power Reduction in Hybrid Energy Systems with Limited Load Forecasts

Yash V. Pant, Truong X. Nghiem and Rahul Mangharam
Department of Electrical and Systems Engineering
University of Pennsylvania
{yashpant, nghiem, rahulm}@seas.upenn.edu

***Abstract*— Hybrid energy systems, which consist of a load powered by a source and a form of energy storage, find applications in many systems, e.g., the electric grid and electric vehicles. A key problem for hybrid energy systems is the reduction of peak power consumption to ensure cost-efficient operation as peak power draws require additional resources and adversely affect the system reliability and lifetime. Furthermore, in some cases such as electric vehicles, the load dynamics are fast, not perfectly known in advance and the on-board computation power is often limited, making the implementation of traditional optimal control difficult. We aim to develop a control scheme to reduce the peak power drawn from the source for hybrid energy systems with limited computation power and limited load forecasts. We propose a scheme with two control levels and provide a sufficient condition for control of the different energy storage/generation components to meet the instantaneous load while satisfying a peak power threshold. The scheme provides performance comparable to Model Predictive Control, while requiring less computation power and only coarse-grained load predictions. For a case study, we implement the scheme for a battery-supercapacitor-powered electric vehicle with real world drive cycles to demonstrate the low execution time and effective reduction of the battery power (hence temperature), which is crucial to the lifetime of the battery.**

I. Introduction

Hybrid energy systems (HES) consist of a load powered by a source and a renewable form of energy storage. They find use in many applications due to their unique properties (e.g., fast response time, peak power reduction, supplementary energy source) and the potential advantages (e.g., cost-efficient operation, longer battery lifetime) they offer over homogeneous energy systems. A common problem across different HES is peak power demands which have an adverse effect on the system, e.g., increased operating charges in electric grids, reduced battery lifetime and capacity in electric vehicles. Smart control of the energy flows in HES can lead to peak power reduction on the energy source, which has been studied for many systems ([1], [2], [3]).

In this paper, we aim to develop a control scheme for the peak power problem in HES with limited computation power and limited load forecasts. Our main contributions are:

1) We propose a control scheme with two levels: a simple low-level control algorithm running at a fast sampling rate to directly actuate the plant, and a more complex high-level control algorithm running at a slow sampling rate to compute optimal parameters for the low-level control to operate under a peak threshold. This architecture makes the scheme suitable for systems with fast dynamics because all the complex computations are not affected by the low-level sampling rate.
2) The scheme does not require fine grained load predictions at every time step.
3) We provide a sufficient condition for controlling the HES to meet the instantaneous load while satisfying a peak power threshold and other energy constraints.

The proposed control scheme operates with good performance while being computationally efficient, which lends itself to an online implementation.

A. The hybrid energy system model

We consider a HES consisting of three components: a **load**, a **power source** and an **energy storage**. An example of a HES is shown in Fig. 1. The load has an instantaneous energy demand at every time $t \geq 0$, denoted $d(t)$, that needs to be supplied exactly. Examples of loads are electric appliances in a building, and electric motors in an electric vehicle (EV). Typically d is non-negative, which means that energy needs to be supplied to the load. However, in certain applications such as EVs, d can be negative, which means the load has regeneration energy that can be used by the other components.

The power source, or source for short, is the main energy supplier in the system, e.g., the electric grid for buildings and the battery in an EV. The source has high energy density and can provide a large amount of energy over time. However, as discussed earlier, it is desirable to avoid high peaks in the power drawn from the source. Let $u(t)$ denote the power rate of the source at time t, which is positive (negative) if power flows out of (into) the source.

The energy storage, or storage for short, has the capability to store short-term energy, e.g., supercapacitors. It is used to alleviate the peak power issues by reducing, or flattening, the power drawn from the source. Similar to the source, $v(t)$, which denotes the power rate of the storage at time t, is positive (negative) if power flows out of (into) the storage. The state of charge (SoC) of the storage is denoted by x. Because of its limited capability to store energy, x is constrained between $x_{\min}$ (fully discharged) and $x_{\max} >$

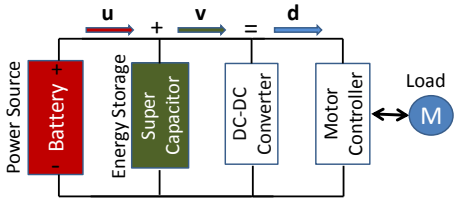

Fig. 1. A simplified view of an architecture for a battery and supercapacitor hybrid energy source powering an EV motor.

This work was supported by STARnet – a Semiconductor Research Corporation program sponsored by MARCO and DARPA – and by the US Department of Transportation University Transportation Center Program.

$x_{\min}$ (fully charged). We assume an ideal storage with no charge and discharge losses, therefore x has a first-order dynamics: $\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -v(t)$ with saturations at $x_{\min}$ and $x_{\max}$.

The relation between the three components is specified by a balance equation which states that at any time t, the load's demand must be matched by the source and the storage:

$$u(t) + v(t) = d(t), \qquad \forall t \geq 0. \tag{1}$$

B. Organization of the paper

The rest of this paper is organized as follows. Section II discusses the standard predictive control approach and its limitations. In Section III we introduce the basic idea and structure of our control scheme, and a key theorem that is central to the scheme. In Section IV we present the high-level control optimization of the scheme. An adaptive thresholding algorithm for a finer time scale is discussed in Section IV-D. Section V consolidates the previous sections and highlights the overall detailed structure of the proposed scheme. We evaluate our scheme when applied to a battery-supercapacitor system on a EV and compare it to other schemes to show its benefits in Section VI. Finally, in Section VII we conclude and discuss future work and potential improvements.

II. Standard Predictive Control Approach

Model Predictive Control (MPC) has been a popular approach for industrial control systems [4]. It involves optimizing a cost function subject to the dynamics of the system, over a finite horizon of time. The first computed input is applied, and at the next step the optimization is solved again. With predictions for future load, it is possible to use MPC to minimize the peak power demand on the energy source of a HES. For example, [5] applied finite-horizon control to a battery-supercapacitor system with linear capacitor (energy storage) dynamics and a log-barrier cost function to smooth the battery (power source) current profile over a time-varying load demand. This resulted in a continuous control of both the battery and the capacitor together, but required exact knowledge of the future demand of load.

Even with perfect predictions of the load demand, optimization horizons cannot be arbitrarily long. With this in mind, we can apply existing MPC techniques to control the HES with a norm cost on the power supplied from the power source. Eq. (2) shows the MPC formulation for a HES with discretized dynamics and upper and lower energy storage limits. Note that the power source is treated as an infinite energy source but has a cost of use while the energy storage has limited energy capacity but has no cost of use.

$$\text{minimize } \|\boldsymbol{u}\|_p \text{ subject to} \tag{2a}$$

$$u(t+i) + v(t+i) = d(t+i),\ \forall i = 0, \ldots, M \tag{2b}$$

$$x(t+i+1) = x(t+i) - hv(t+i),\ \forall i = 0, \ldots, M \tag{2c}$$

$$x_{\min} \leq x(t+i) \leq x_{\max},\ \forall i = 0, \ldots, M+1 \tag{2d}$$

Here M is the control horizon, $\boldsymbol{u} = [u(t), \ldots, u(t+M)]^T$, and h the sampling time. Eq. (2b) is the balance equation (1). Eq. (2c) is the discretized energy dynamics of the energy storage and we assume there is no loss term, although linear loss terms can be easily incorporated. Eq. (2d) implies that the SoC of the storage must be within some bounds at all times. In many applications the load demand for every time instant cannot be known perfectly in advance. A stochastic MPC formulation can be obtained by treating the load demand as a multivariate normal variable where load demands at every time instant are uncorrelated. For the case study (Section VI), we compare our scheme to such a stochastic MPC, which can be easily formulated as an LP with a slight variation of the procedure in [6, Section 4.4.2].

A. Limitations of standard MPC

While MPC is a good control approach for peak minimization in HES, there are some drawbacks to using MPC:

1) MPC needs fine-grained information about the load demand, i.e., either the exact load demand, or its distribution (for stochastic MPC), at every time instant, which is difficult to obtain for many practical systems.
2) For a system with fast dynamics, the computational requirements for MPC may make it impractical for implementation, especially given limited computational capability. Move-blocking MPC [7] has been developed as a technique to reduce the computational overhead by reducing the number of control variables to be solved for. However it assumes a constant control signal during each blocking window, which consists of multiple time steps. This may result in infeasibility given some constraints on the system's state variables, e.g., x in Eq. (2d) may violate its upper or lower limits if the control signal is not free to change at every time step.

III. Multi-level Control Approach

To overcome the aforementioned drawbacks of move-blocking MPC we propose a multi-level control approach based on the following fundamental idea. Consider the load curve from the current time t_0 to some time instant $t_f > t_0$ as depicted by the solid line in Fig. 2. The interval $[t_0, t_f]$ is divided into $N \geq 1$ equidistant subintervals: $t_0 < t_1 < \cdots < t_N = t_f$. During each $[t_i, t_{i+1})$, $0 \leq i \leq N-1$, we set a peak threshold $\overline{P}_i$ on u and control the source and the storage so that u does not exceed $\overline{P}_i$ at any time during the subinterval, i.e., $u(t) \leq \overline{P}_i$, $\forall t \in [t_i, t_{i+1})$. In Fig. 2, the peak thresholds for the subintervals are illustrated by the dashed lines. The residuals between the load and the source will be picked up by the storage. In particular, in Fig. 2, the blue regions when the source power exceeds the load are charged to the storage, while the red regions when the load exceeds the source power are supplied by the storage.

To realize the above idea, we design a control structure consisting of two levels, as illustrated in Fig. 3:

- **High-level control** determines the peak threshold for each time subinterval such that:

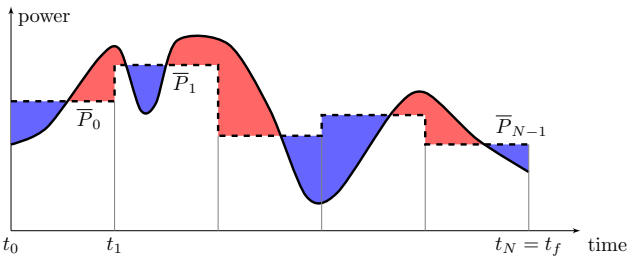

Fig. 2. The fundamental idea of the proposed control approach.

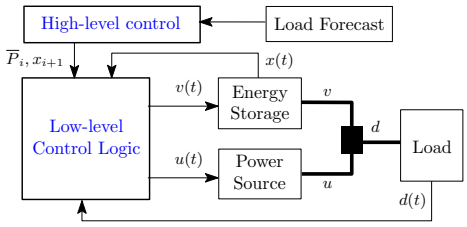

Fig. 3. Structure of the proposed control approach.

  - a global objective is attained, for example the overall peak of u is reduced, and
  - each individual peak threshold can be satisfied while ensuring that the instantaneous load is provided.
- **Low-level control** determines, at any time t, the powers $u(t)$ and $v(t)$ so that the required load is served, the peak threshold for the current subinterval (as determined by the high-level control) is satisfied, and the storage is neither over-charged or over-exhausted.

The two control levels are linked by the peak thresholds $\overline{P}_i$, which are computed by the high-level control and executed by the low-level control.

A. Low-level Control

Let $\overline{P}(t)$ denote the peak threshold in effect at time t, i.e., $\overline{P}(t) = \overline{P}_i$ if $t \in [t_i, t_{i+1})$. The general rule for the low-level control is to keep u at the peak threshold for the current subinterval as long as it is feasible to do so.

- If the storage is not exhausted ($x(t) > x_{\min}$) and the load exceeds the peak threshold ($d(t) > \overline{P}(t)$), the storage is discharged to supply for the residual.
- If the storage is not fully charged ($x(t) < x_{\max}$) and the peak threshold exceeds the load ($d(t) < \overline{P}(t)$), we keep $u(t) = \overline{P}(t)$ to charge the storage with the surplus energy. Should the demand spike later on, the stored energy can be used to alleviate the peak.

If this rule is infeasible then the source must track the load, i.e., $u(t) = d(t)$. Specifically, the low-level control logic is described in Alg. 1, where we omit the time t for brevity.

Algorithm 1 Low-level control logic.

1: **if** $\left(d \le \overline{P} \wedge x < x_{\max}\right) \vee \left(d \ge \overline{P} \wedge x > x_{\min}\right)$ **then**
2: $\quad u \leftarrow \overline{P}$, $v \leftarrow d - \overline{P}$
3: **else**
4: $\quad u \leftarrow d$, $v \leftarrow 0$
5: **end if**

A *top saturation* is an interval $[t_1, t_2]$ during which the storage is saturated at the maximum SoC $x_{\max}$, i.e., $x(t) = x_{\max}$ and $d(t) < \overline{P}(t)\ \forall t \in [t_1, t_2]$. Similarly, a *bottom saturation* is an interval $[t_1, t_2]$ during which the storage is saturated at the minimum SoC $x_{\min}$, i.e., $x(t) = x_{\min}$ and $d(t) > \overline{P}(t)\ \forall t \in [t_1, t_2]$. These cases correspond to line 4 in Alg. 1. If no saturation happens at time t, the system is *saturation-free*. This case corresponds to line 2 in Alg. 1.

B. Interface Between High and Low Control Levels

The key result of this paper, presented in Theorem 1, regards the interface between the high-level control and the low-level control. It specifies a sufficient condition on the peak thresholds $\overline{P}_i$, as computed by the high-level control, so that with the low-level control logic in Alg. 1 the set peak thresholds are always honored. Let the notation d^+ denote the non-negative portion of the load's demand, i.e., $d^+(t) = d(t)$ if $d(t) \ge 0$ and $d^+(t) = 0$ otherwise.

Theorem 1: Given an interval $[0, T)$, $T > 0$, a desired final SoC $x_{\mathrm{f}} \in [x_{\min}, x_{\max}]$, and E^+, E, $\bar{d}$ such that

$$E^+ \ge \int_0^T d^+(t)\,\mathrm{d}t \ge 0 \tag{3a}$$

$$E \ge \int_0^T d(t)\,\mathrm{d}t \tag{3b}$$

$$\bar{d} \ge \max\nolimits_{0 \le t < T} d(t). \tag{3c}$$

Choose any peak threshold $\overline{P} \ge 0$ satisfying

$$\overline{P} \ge \left(x_{\mathrm{f}} - x(0) + E\right)/T \tag{4}$$

$$\text{if } E^+ > 0\text{: } \overline{P} \ge \bar{d}\left(1 - \left(x(0) - x_{\min}\right)/E^+\right) \tag{5}$$

$$\text{if } E^+ > 0\text{: } \overline{P} \ge \bar{d}\left(1 - \left(x_{\max} - x_{\mathrm{f}}\right)/E^+\right). \tag{6}$$

Then with the low-level control algorithm in Alg. 1, the following statements hold:

a) For all $t \in [0, T)$: $u(t) \le \overline{P}$.

b) $x(T) \ge x_{\mathrm{f}}$.

Statement (a) essentially confirms that, with the chosen peak threshold $\overline{P}$, the source power never exceeds the peak threshold. Furthermore, by statement (b), the desired final SoC x_{f} is achieved. For brevity, the proof of Theorem 1 is omitted here, but can be found in [8].

IV. High-level Control

The key result in Theorem 1 connects the high-level control and the low-level control. Essentially, to ensure that the peak threshold set by the high-level control is always honored by the low-level control, it should satisfy the sufficient conditions given in the Theorem. In this section, we formulate the high-level control problem using a receding-horizon control (RHC) approach [4].

Let us recall the fundamental idea of our approach in Section III. We divide a given time horizon $[t_0, t_f]$ into N subintervals. In the high-level control, we determine for each subinterval i a peak threshold $\overline{P}_i$, which is then used by the low-level control in Alg. 1. The computed peak thresholds are good if they not only satisfy the sufficient conditions in Theorem 1 but also optimize a certain global objective for the horizon $[t_0, t_f]$. The latter can be represented in a RHC formulation. In the next subsection, we discuss the load forecast information required by the high-level control to compute the peak thresholds.

A. Required Load Forecast Information

From Theorem 1, to determine a peak threshold for each subinterval $[t_i, t_{i+1})$, $0 \le i \le N-1$, the high-level control requires the three values defined in Eq. (3), namely

- $E_i^+ \ge \int_{t_i}^{t_{i+1}} d^+(t)\,\mathrm{d}t$ is an upper bound of the total non-negative energy demand for the subinterval;
- $E_i \ge \int_{t_i}^{t_{i+1}} d(t)\,\mathrm{d}t$ is an upper bound of the total net energy demand for the subinterval;
- $\bar{d}_i \ge \max_{t_i \le t < t_{i+1}} d(t)$ is an upper bound of the maximum power demand during the subinterval.

Note that E_i^+, E_i and $\bar{d}_i$ are upper-bound estimates of the corresponding values on the right-hand side, so exact estimates of these values are preferred but not required. To obtain these estimates for each time subinterval requires a

prediction of the load during the subinterval. Depending on the actual application, the level of technical effort involved in the prediction might vary widely. For example, if the load is generated by electric appliances in a building, the upper bound estimates can be calculated based on the schedule or the history of appliance use, which is relatively easy. However, if the load is generated by the motors of an EV, obtaining these estimates will be much more difficult as it involves predicting the road condition, the traffic condition, and the driving habits to name a few. For that reason, we assume that these estimates are available and ignore for now the technical details of obtaining them.

B. High-level Control Optimization

Given a horizon $[t_0, t_f]$ divided into N subintervals, the high-level control determines the peak thresholds such that the conditions in Theorem 1 are satisfied and a global objective function $F(\boldsymbol{P})$ is minimized. Here $\boldsymbol{P}$ is the vector of the peak thresholds: $\boldsymbol{P} = \left[\overline{P}_0, \ldots, \overline{P}_{N-1}\right]^T$. The choice of the objective function depends on the goals of the application. A possible choice is the l_p-norm function $F(\boldsymbol{P}) = \|\boldsymbol{P}\|_p$, where $\|\cdot\|_p$ denotes the l_p-norm of a vector, for $p \geq 1$ or $p = \infty$. When the l_∞-norm is used, the overall peak during $[t_0, t_f]$ is minimized because $\|\boldsymbol{P}\|_\infty = \max_{0 \leq i \leq N-1} \overline{P}_i$. In our case study in Section VI we found that the l_2-norm worked best to both smoothen the u curve and reduce its overall peak. To further smoothen u we can penalize the variations in $\boldsymbol{P}$ by adding a term $\sum_{i=0}^{N-2} \left|\overline{P}_{i+1} - \overline{P}_i\right|$. For example $F(\boldsymbol{P}) = \|\boldsymbol{P}\|_p + \epsilon \sum_{i=0}^{N-2} \left|\overline{P}_{i+1} - \overline{P}_i\right|$ where $\epsilon > 0$ is a predefined constant.

Let x_i, $1 \leq i \leq N$, be the desired SoC of the storage at the end of the subinterval $[t_i, t_{i+1}]$. These values are also determined by the high-level control. The final desired SoC x_N can be either free or fixed at a predefined value $x_\mathrm{f} \in [x_\mathrm{min}, x_\mathrm{max}]$. We can now formulate the high-level control optimization for the interval $[t_0, t_f]$ as follows:

$$\text{minimize}_{\boldsymbol{P}, x_1, \ldots, x_N} \; F(\boldsymbol{P}) \text{ subject to}$$

$$\overline{P}_i \geq \left(x_{i+1} - x_i + E_i\right) / \left(t_{i+1} - t_i\right) \tag{7a}$$

$$\text{if } E_i^+ > 0\text{: } \overline{P}_i \geq \bar{d}_i \left(1 - \left(x_i - x_\mathrm{min}\right) / E_i^+\right) \tag{7b}$$

$$\text{if } E_i^+ > 0\text{: } \overline{P}_i \geq \bar{d}_i \left(1 - \left(x_\mathrm{max} - x_{i+1}\right) / E_i^+\right) \tag{7c}$$

$$\overline{P}_i \geq 0 \tag{7d}$$

$$x_\mathrm{min} \leq x_{i+1} \leq x_\mathrm{max} \tag{7e}$$

$$x_0 = x(t_0), \quad x_N = x_\mathrm{f} \tag{7f}$$

in which the constraints (7a) to (7e) are satisfied for all $i = 0, \ldots, N-1$. Eq. (7e) constraints the SoC between x_min and x_max. Eq. (7f) specifies the initial condition and the (optional) final condition on the SoC of the storage. We remark that all the constraints are linear, therefore if F is convex the optimization (7) can be solved efficiently [6].

C. Receding-Horizon High-level Control Algorithm

The optimization (7) only solves the peak thresholds for a finite time horizon, which is usually due to the limitation that we cannot predict the load too far into the future. However, the system typically operates far beyond that time horizon, possibly forever. In order to compute the peak thresholds continuously as time progresses, we employ the *receding-horizon control* approach [4].

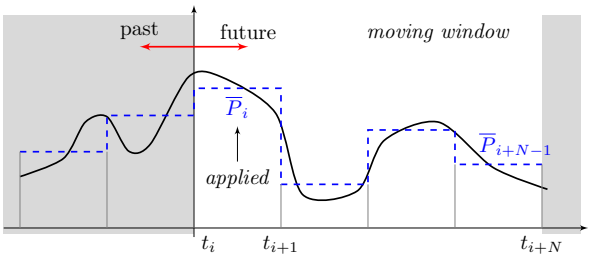

Fig. 4. RHC approach for high-level control: The solid line is the load. The dashed line is the peak threshold determined by the high-level control. Optimization (7) is solved for a moving time window of N sampling steps.

Let $T_s > 0$ be the sampling time step and $t_i = iT_s$, $i \in \mathbb{N}$, be the sampling instants of the high-level control. We assume that at each t_i we can obtain the predictions $(E_j^+, E_j, \bar{d}_j)$ for N future steps: $i \leq j \leq i + N - 1$. The RHC approach is illustrated in Fig. 4. At time t_i, we solve Eq. (7) for the time window $[t_i, t_{i+N}]$ consisting of N sampling intervals. We then apply the resulting optimal peak thresholds and desired SoCs for only the first interval $[t_i, t_{i+1}]$. At the next time instant t_{i+1}, the N-step horizon window is advanced by one step to $[t_{i+1}, t_{i+1+N}]$, for which new predictions are obtained and the optimization process is repeated. The high-level control algorithm is summarized in Alg. 2.

Algorithm 2 RH high-level control algorithm.

for $i = 0, 1, 2, \ldots$ **do**
  Obtain $(E_j^+, E_j, \bar{d}_j)$ for $j = i, \ldots, i + N - 1$
  Solve optimization (7) and apply $\overline{P}_i$ and x_{i+1}
  Wait until t_{i+1}
end for

D. Adaptive Peak Threshold

The low-level control algorithm uses a constant peak threshold $\overline{P}_i$ throughout the interval $[t_i, t_{i+1}]$. The calculation of $\overline{P}_i$ by the high-level control is often over-conservative due to our lack of knowledge of the actual future load during the interval. However, as time progresses, at time $t \in [t_i, t_{i+1})$, we have knowledge of the past load from t_i to t, hence we can adapt the peak threshold to be less conservative. Specifically, define $\mathcal{E}_i(t) = E_i - \int_{t_i}^{t} d(\tau)\,\mathrm{d}\tau$ and $\mathcal{E}_i^+(t) = E_i^+ - \int_{t_i}^{t} d^+(\tau)\,\mathrm{d}\tau$, which can be calculated by simple integrators. Note that $\mathcal{E}_i(t_i) = E_i$ and $\mathcal{E}_i^+(t_i) = E_i^+$. Then, by considering $[t, t_{i+1})$ as the time interval in Theorem 1, we can re-compute the peak threshold for the rest of the interval, as in Alg. 3. The adaptive peak threshold computation is executed periodically after every $T_a > 0$ time units during each interval $[t_i, t_{i+1})$.

Algorithm 3 Adaptive peak threshold algorithm.

repeat every instant $t = t_i, t_i + T_a, \ldots$ during $[t_i, t_{i+1}]$
  Obtain $\mathcal{E}_i(t)$ and $\mathcal{E}_i^+(t)$
  $\overline{P} \leftarrow \max\left\{0, \left(x_{i+1} - x(t) + \mathcal{E}_i(t)\right) / \left(t_{i+1} - t\right)\right\}$
  if $\mathcal{E}_i^+(t) > 0$ **then**
    $\overline{P} \leftarrow \max\big\{\overline{P}, \bar{d}_i \left(1 - \left(x(t) - x_\mathrm{min}\right) / \mathcal{E}_i^+(t)\right),$
    $\bar{d}_i \left(1 - \left(x_\mathrm{max} - x_{i+1}\right) / \mathcal{E}_i^+(t)\right)\big\}$
  end if
end repeat

It can be shown, using Lemma 1 in [8], that for all $t_i \leq t < t' \leq t_{i+1}$, if $\mathcal{E}_i^+(t) > 0$ and $\mathcal{E}_i^+(t') > 0$ then $\bar{d}_i \Big(1 -$

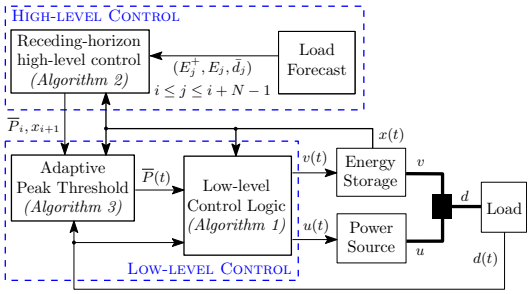

Fig. 5. Overall detailed structure of the proposed scheme.

$\frac{x(t)-x_{\min}}{\mathcal{E}_i^+(t)}\Big) \geq \bar{d}_i\Big(1-\frac{x(t')-x_{\min}}{\mathcal{E}_i^+(t')}\Big)$. Hence, the lower-bound on $\overline{P}$ in Eq. (5) is non-increasing during the interval. Similarly, we can show that the lower-bounds on $\overline{P}$ in Eqs. (4) and (6) are also non-increasing. Therefore, it is guaranteed that the adjusted peak threshold computed by Alg. 3 will not increase towards the end of the interval.

V. Overall Control Structure

We have discussed in Sections III to IV the essential components of the proposed control scheme: the *low-level control*, the *high-level control* and the *adaptive peak threshold algorithm*. These components are put together in the overall control structure in Fig. 5. At the bottom right is the plant consisting of the source, the storage and the load. The *high-level control* at the top carries out load forecasting and the receding-horizon control algorithm (Alg. 2). The calculated peak thresholds and desired SoCs are then provided to the *low-level control*. The *adaptive peak threshold* algorithm (Alg. 3) periodically adjusts the peak constraint $\overline{P}(t)$.

Fig. 5 highlights an important feature of the proposed scheme: the separation between the executions of the high-level control and the low-level control. In particular, the low-level control consists of simple algorithms, therefore its implementation is simple and it can be executed at a very fast rate. On the other hand, the high-level control, which consists of more complex algorithms involving load predictions and an optimization, can be executed at a much slower rate. An implication of this feature is that any change in the execution (sampling) rate of the low-level control does not affect the high-level control. This helps reduce the development and implementation cost of the control scheme.

VI. Case Study : Power Management in EV

To demonstrate the proposed scheme we consider the case of an EV with a battery and supercapacitor (SC) HES. A battery has a relatively high energy density but a low power density compared to an SC. SCs, with their low-energy storage and high charge/discharge rate capacities [3], can be used to partially or completely meet the peak power demands from the load (for a short period of time), effectively reducing the power drawn from the battery and hence reducing heat generation in the battery [9], [10]. Figure 1 shows a general architecture for a battery supercapacitor energy source delivering power to a motor. Details on the architecture can be found in [3]. Predictive control for such systems has also been studied with finite horizon control involving perfect knowledge of future load demand [5] and also with data-driven model predictive control [11]. For the scope of this study, we obtain predictions from historic data.

A. Generating predictions for the proposed scheme

In order to get predictions of the load demand parameters $(E, E+, \bar{d})$ of Theorem 1, we rely on historical data. For the simulations in this paper, we use data from the ChargeCar project [12] to give us real world drive cycles and also to predict the required parameters. To get predictions for our scheme and Stochastic MPC, we limit ourselves to trips along a particular route (assumed that the route is known *a priori*). This is done to reduce the amount of data to be processed and also since intuitively, the driving profile is expected to be similar for trips along a particular route. For the current horizon (at time T) of length M, given the dataset along a route, $d_i(t), \forall i = 1, \ldots,$ number of trips, $t = T, \ldots, T+M-1$, we can compute the parameters required for our scheme as follows: $E = \max_i \sum_t d_i(t)$, $E+ = \max_i \sum_{t|d_i(t)>0} d_i(t)$ and $\bar{d} = \max_i \max_t d_i(t)$. For the scope of this study, we take all the trips to generate the predictions and then pick one of the trips as the simulation trip. While this follows the non-ideal practice of having a testing set which is a subset of the training set, this is necessary to make the parameters $E, E+, \bar{d}$ be upper bounded by the worst case for all trips. Since the focus of this paper is on the control algorithm, the development of advanced prediction schemes is left for future work.

B. Control schemes for the battery supercapacitor system

In order to show the effectiveness of the proposed scheme in reducing peak power drawn from the battery (and hence reducing battery temperature) we implemented and evaluated the following schemes for comparison:

1) **Battery-only**: The load demand is met by the battery only. This serves as a worst-case (for schemes with a SC).

2) **Naive scheduling**: This greedy scheme uses the SC to meet the load whenever it has charge, and charges the SC with regeneration (unless it is full).

3) **Stochastic MPC**: As knowing the future load demand perfectly is not easily possible for an EV, a stochastic MPC formulation can be obtained as briefly described in Section II. The mean and covariance of the multivariate distribution assumed for the load demand can be obtained from historic data using maximum likelihood estimation. More details on stochastic MPC can be found in [13].

4) **Optimal scheme**: The optimal control formulation is similar to the deterministic MPC formulation in Eq. (2), except that the horizon M is the length of the entire simulation. The optimization is solved in one shot, and is done offline. Note that the optimal scheme gives the best-case performance (since the optimization is done with perfect *a priori* knowledge load demand) which we use as a baseline to evaluate any scheme. In practice, this optimal scheme is intractable for lengthy drive cycles because the entire driving period has to be known in advance.

The proposed control scheme needs comparatively less information than stochastic MPC and the optimal scheme: only the upper bounds on the total energy demanded by the load and the maximum power demand in the horizon. Also, due to the separation of the two control levels, the high-level optimization is solved at a low rate, irrespective of the

Scheme	MaxTemp (C)(%)	MeanTemp (C)(%)	Exec. Time (s)
Proposed scheme	**47.1733** (9.57)	**27.5373** (6.24)	**29.50**
Stochastic MPC	**47.3732** (9.18)	**28.0696** (4.43)	**579.29**
Optimal	38.5115 (26.16)	24.6182 (16.15)	0.49
Naive	50.6186 (2.96)	28.9097 (1.57)	N/A
Battery-only	**52.1639**	**29.3711**	N/A

TABLE I

SIMULATION RESULTS (AVERAGED OVER 9 TRIPS).

sampling rate for the low-level control. These features make the proposed scheme attractive in practice.

C. Simulation setup and results

To evaluate the effectiveness of the proposed scheme and the battery-SC control schemes outlined above, we use the Advanced Vehicle Simulator (ADVISOR) [14]. The load power demands are generated for an EV going through nine drive cycles (corresponding to different trips) on the route chosen from the ChargeCar data. The battery temperature and SoC profiles generated from the simulation for ADVISOR's Li-Ion model are used to compare the performance of the different schemes. The SC has ratings of 50F and 60V and is half full initially. For our scheme we used $N = 8$, $T_s = 5\,\text{s}$ and $T_a = 1\,\text{s}$. For the stochastic MPC, the horizon was limited to $8\,\text{s}$. Both schemes used a l_2-norm cost function for the optimizations. For more details, refer to [8].

Figure 6 shows the mean temperature profiles for a Li-Ion battery subject to the 9 drive cycle from ChargeCar data and controlled by the different schemes. The simulation results are summarized in Table I, which presents the relevant battery parameters, averaged over the 9 trips. Our scheme performs as well as stochastic MPC and leads to a marked reduction in the average and the maximum battery temperatures compared to the battery-only and naive schemes. This shows the effectiveness of the proposed scheme in reducing the battery operating temperature, with only historic data.

In terms of computation time, for a drive cycle length of 965s and at 1Hz sampling rate, stochastic MPC took an average of 579s while our scheme took only 29s on average. This improvement is because the stochastic MPC implementation repeats its optimization at every time step of 1s, while our scheme only needs to solve the high-level optimization every 5s. The computation time of the low-level control is negligible due to its simplicity. Moreover, because of the separation of the two control levels in our scheme, increasing the low-level sampling rate of the system does not affect the computation time of the high-level control.

VII. CONCLUSION AND FUTURE WORK

In this paper, we developed a multi-level control scheme for peak power reduction in HES. The basic idea is to cap the power drawn from the source by a peak threshold while using the energy storage for the residual between the load and the source power. We provided a sufficient condition for the peak threshold calculated by the high-level control to be feasible for the low-level control logic. This architecture allows the complex computations be decoupled from the low-level sampling rate, hence making the proposed scheme computationally efficient and applicable to systems with fast dynamics. A notable advantage of this scheme is that it does not require exact and fine grained predictions of the load at every time step. Using a case study of a battery and supercapacitor energy system for electric vehicles, we evaluated the proposed scheme and showed its effectiveness in reducing peak battery power (hence battery temperature) as well as its applicability to fast dynamical systems. A focus of ongoing research is to overcome the necessity of having deterministic upper bounds on the parameters needed for the scheme, knowing which can be hard for some applications, by having a stochastic version of this scheme.

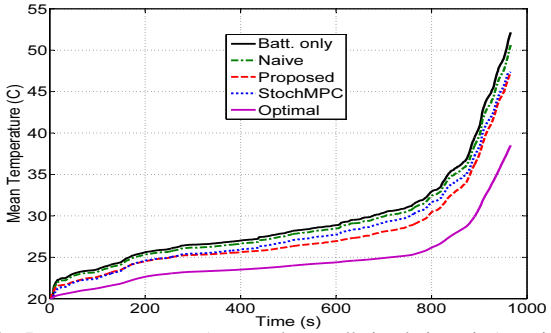

Fig. 6. Battery temperature (averaged over all simulation trips) vs time for different schemes. A lower battery temperature increases battery lifetime.

REFERENCES

[1] Y. Ma, F. Borrelli, B. Hencey, B. Coffey, S. Bengea, and P. Haves, "Model predictive control for the operation of building cooling systems," in *Proceedings of the American Control Conference*, 2010.

[2] J. Liu, P. H. Chou, N. Bagherzadeh, and F. Kurdahi, "Power-aware scheduling under timing constraints for mission-critical embedded systems," in *Proc. Design Automation Conference*. ACM, 2001.

[3] A. Khaligh and Z. Li, "Battery, ultracapacitor, fuel cell, and hybrid energy storage systems for electric, hybrid electric, fuel cell and plug-in hybrid electric vehicles: State of the art," *IEEE Tran. on Vehicular Tech.*, 2008.

[4] E. Camacho and C. Bordons, *Model predictive control*. Springer Verlag, 2004.

[5] M. Choi, S. Kim, and S. Seo, "Energy management optimization in a battery/supercapacitor hybrid energy storage system," *IEEE Transactions on Smart Grid*, 2012.

[6] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2006.

[7] R. Cagienard, P. Grieder, E. Kerrigan, and M.Morari, "Move blocking strategies in receding horizon control," *IEEE CDC*, 2004.

[8] Y. V. Pant, T. X. Nghiem, and R. Mangharam, "Peak power control of battery and super-capacitor energy systems in electric vehicles," University of Pennsylvania, Tech. Rep. TR-2014-ESE01, Feb 2014.

[9] A. Pesaran, A. Vlahinos, and S. Burch, "Thermal performance of ev and hev battery modules and packs," *Proceedings of the 14th International Electric Vehicle Symposium*, 1997.

[10] M. Zahrand and A. Atef, "Electrical and thermal properties of nicd battery for low earth orbit satellite's applications," *Proceedings of the 6th WSEAS International Conference on Power Systems*, 2006.

[11] A. Styler and I. Nourbakhsh, "Model predictive control with uncertainty in human driven systems," *AAAI Conf. on Aritif. Intel.*, 2013.

[12] ChargeCar, http://www.chargecar.org, accessed: 2013-09-23.

[13] F. Oldewurtel, C. Jones, and M. Morari, "A tractable approximation of chance constrained stochastic mpc based on affine disturbance feedback," *IEEE Conf. on Dec. and Control*, 2008.

[14] K. Wipke, M. Cuddy, D. Bharathan, S. Burch, V. Johnson, A. Markel, and S. Sprik, "Advisor 2.0: A secondgeneration advanced vehicle simulator for systems analysis," NREL, Tech. Rep., March 1999.