

Manuscript version: Author's Accepted Manuscript

The version presented in WRAP is the author's accepted manuscript and may differ from the published version or Version of Record.

Persistent WRAP URL:

<http://wrap.warwick.ac.uk/82131>

How to cite:

Please refer to published version for the most recent bibliographic citation information. If a published version is known of, the repository item page linked to above, will contain details on accessing it.

Copyright and reuse:

The Warwick Research Archive Portal (WRAP) makes this work by researchers of the University of Warwick available open access under the following conditions.

Copyright © and all moral rights to the version of the paper presented here belong to the individual author(s) and/or other copyright owners. To the extent reasonable and practicable the material made available in WRAP has been checked for eligibility before being made available.

Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

Publisher's statement:

Please refer to the repository item page, publisher's statement section, for further information.

For more information, please contact the WRAP Team at: wrap@warwick.ac.uk.

OPTIMAL STRATEGIES FOR OPERATING ENERGY STORAGE IN AN ARBITRAGE OR SMOOTHING MARKET

LISA C FLATLEY* AND ROBERT S MACKAY

Mathematics Institute, University of Warwick
Gibbet Hill Road, Coventry, CV4 7AL, UK

MICHAEL WATERSOON

Economics Department, University of Warwick
Gibbet Hill Road, Coventry, CV4 7AL, UK

(Communicated by Elvio Accinelli)

ABSTRACT. We characterize cost-minimizing operating strategies for an energy store over a given interval of time $[0, T]$. The cost functional here can represent, for example, a traditional economic cost or a penalty for time-variation of the output from a storage-assisted wind farm or more general imbalance between supply and demand. Our analysis allows for leakage, operating inefficiencies and general cost functionals. In the case where the cost of a store depends only on its instantaneous power output (or input), we present an algorithm to determine the optimal strategies. A key feature is that this algorithm is localized in time, in the sense that the action of the store at a time $t \in [0, T]$ requires cost information over only some usually much shorter subinterval of time $[t, t_k] \subset [t, T]$.

1. Introduction. Over the coming decades, world energy-markets face significant challenges as they strive to meet their climate-change targets. Renewable generation is set to play a more dominant role in the future electricity-supply markets but the availability of renewable energy is intermittent. This unreliability means that there will always be a need for a quick-reacting back-up, in order to ensure that demand is met. On the other hand, at off-peak demand times, supply may be curtailed (often at a high cost) if the rest of the system is not flexible enough to respond. On the demand side, future demand profiles are likely to look very different from today. Electric heat pumps, for example, could create a marked seasonal peaking of demand during the colder, winter months. Electric vehicle charging and new smart technologies, on the other hand, could change our daily demand patterns. In this introduction we pay particular attention to the British scene because it is the one with which we are most familiar, but the issues apply world-wide.

Electricity storage is one potential option for improving the flexibility and reliability of our electricity system. It could offer services such as peak-shaving, frequency response, reactive-power regulation and the provision of reserve. In Great Britain (GB), grid-scale electricity storage currently comes only in the form of pumped-hydro power plants, which are implemented largely to meet peak demands. Their

2010 *Mathematics Subject Classification.* Primary: 49K30; Secondary: 93C10.

Key words and phrases. Dynamic optimization, energy storage, optimal localized algorithm.

* Corresponding author: Lisa C Flatley.

main source of revenue is through Short-Term Operating Reserve (STOR) and Fast Reserve, which are funded by National Grid. The stores then replenish their supply during the night, when electricity prices are at their lowest [1].

Much research has been dedicated to analyzing storage viability. Some papers focus on the potential social benefits of storage (see, for example, [2, 3, 4, 5]). A common approach here is to solve the unit dispatch problem, i.e. to select a cost-minimizing combination of generators (including storage) to run at each time, in order to ensure that demand is met (or that demand is met with a given probability). An advantage of this method is that it can incorporate the interactions between all assets of the energy system into a single optimization problem, allowing a comparison between the actions of storage against its competing options. A drawback, however, is that the system optimum does not necessarily coincide with each individual firm's most profitable strategy. In many cases, it is not even clear that each firm would make a profit under these solutions. Therefore, a whole-system approach of this kind is generally better suited to questions where the store is not privately owned, but instead owned and operated by a central controller, such as the system operator.

Other research focuses on the profits available to a store which faces stochastic or probabilistic prices. Some of these papers restrict the behaviour of the store to a pre-defined set of operating strategies (e.g. [6, 7]). Another approach is to implement Dynamic Programming techniques (e.g. [8, 9]). One drawback of this latter approach, however, is that it tends to be computationally heavy. A significant limitation is that it often requires information about prices over the entire time horizon over which the store's operation is to be optimized. Since electricity wholesale prices contain significant idiosyncratic components, this requires a heroic assumption regarding the foresight of its operator. Additionally, such methods do not give much scope for mathematical insight into the dynamics of the solutions.

In this paper, we characterize cost-minimizing strategies of an electricity store, for general cost functionals, in terms of the associated Karush-Kuhn-Tucker multipliers. Further, for a certain natural class of cost functional (see Section 1.2), we present an algorithm to determine these optimal strategies. A natural application here would be to maximize the arbitrage profit available to an electricity store which is operating in the wholesale electricity market. In doing so, we assume that the store can predict an electricity price function $p : [0, T] \rightarrow \mathbb{R}$ where $[0, T]$ is the period of time over which we wish to optimize. However, the method only actually uses price information over a smaller time interval, except in special cases or near T , thus reducing the required prediction horizon and the amount of computation needed. It is worth commenting here that given the presence of future markets at various maturities, which coalesce on the spot wholesale price as they approach maturity (see e.g. [17]), the prediction of short-term future prices in this way is not an unreasonable assumption. Another application, as discussed in Section 5.3, is in the use of a store for "smoothing". This could apply, for example, to cases where an energy store is built in conjunction with a wind farm, with the intention of smoothing the wind output from that farm. Such smoothing could be useful for reducing short-term fluctuations of supply.

Our method is derived from standard Calculus of Variations techniques and is intended as an extension to [10]. The main result is Proposition 2.1. Our work differs from [10] in several respects. Firstly, we present our model in a continuous-time setting, rather than the discrete setting employed in [10]. Even if prices (for

example) are declared at discrete time intervals, in accordance with the current market system, our approach allows for the input of piecewise constant prices but with continuous variation in the operation of the store. Secondly, [10] allows only for convex cost functions, whereas here we allow for much more general operating costs and, in particular, we remove the convexity assumption. This is realistic: there is little reason to believe that the cost of running a motor, for example, is a convex function of the output power. Although these more general conditions do not guarantee the existence of solutions, we prove that optimal operating strategies of the form given in Proposition 2.1 exist if and only if the algorithm yields both a strategy q^* and its associated multiplier function μ^* . Finally, we allow the inclusion of slightly more general operating constraints - namely, time-varying power and capacity constraints.

The structure of the paper is as follows: Over the remainder of this section, we introduce our storage model and its associated costs. In Section 2, we present the main result, which characterizes the optimal strategies via a reference price function (or multiplier). In Section 3, we present an algorithm which determines both the optimal strategy and the reference price for the basic case where the store is constrained by only its power ratings and its capacity constraints. We prove that if an optimal strategy exists of the form of Proposition 2.1, then the algorithm will find it. Section 4 contains a discussion of the time-localization property that is inherent both in the algorithm and in the more general case of the standard storage model (see Section 1.2). In Section 5, we present some applications of the theory and the algorithm. Finally, the conclusions of the paper are summarized in Section 6.

1.1. The storage model. Let $[0, T]$ be the interval of time over which we want to optimize the actions of the store (with $T > 0$) and denote by $U \subset \mathbb{R}$ the set of admissible power outputs associated with the store. The operator of the store then chooses an *operating strategy* $q : [0, T] \rightarrow U$ which allocates, at each time t , the power $q(t)$ that is to enter the store (using the convention that if $q(t) < 0$, then the store discharges power $-q(t)$ at time t). The power taken from the grid may be larger than q , to allow for inefficiency of conversion to storage. Similarly, if q is negative the power delivered to the grid may be between $-q$ and 0. The constraint set U could, if we wish, vary with time, therefore allowing the inclusion of constraints such as planned closures or transmission congestion into the model. All results presented in this paper generalise naturally to this case but, for simplicity of notation, we will from now on refer only to a time-independent constraint set U . Throughout this paper we insist that $0 \in U$. Without this assumption, the store would never be permitted to be at rest.

The *level of stored energy* at time t , associated with the strategy q , is denoted $\ell[q](t)$ and solves the differential equation

$$\begin{cases} \dot{\ell}(t) = -\alpha\ell(t) + q(t) & \forall t \in [0, T] \\ \ell(0) = \ell_0, \end{cases} \quad (1)$$

where $\alpha \in [0, 1)$ is the leakage rate of the store and ℓ_0 is the initial level of stored energy. The level of stored energy thus evolves as

$$\ell[q](t) = e^{-\alpha t} \ell_0 + e^{-\alpha t} \int_0^t e^{\alpha s} q(s) ds \quad \forall t \in [0, T]. \quad (2)$$

The *capacity constraints* of the store are characterized by two functions $E^+, E^- : [0, T] \rightarrow \mathbb{R}$, so that any strategy q is constrained by the inequalities

$$E^-(t) \leq \ell[q](t) \leq E^+(t) \quad \forall t \in [0, T].$$

We assume that the initial and terminal levels of the store, denoted ℓ_0 and ℓ_T respectively, are pre-specified, so that

$$\ell_0 := E^-(0) = E^+(0) \quad \text{and} \quad \ell_T := E^-(T) = E^+(T).$$

We also assume, without loss of generality, that $\{\dot{E}^-(t) + \alpha E^-(t), \dot{E}^+(t) + \alpha E^+(t)\} \subset U$ for all $t \in [0, T]$, since otherwise E could be adjusted so that this is true, without changing the set of admissible strategies (cf. Definition 1.1).

The set of possible levels of the store, at each time, is represented by the *admissible energy domain* $E \subset [0, T] \times \mathbb{R}$, which is the set of all pairs $(t, m) \in [0, T] \times [E^-(t), E^+(t)]$. Often, one may choose to replace the capacity constraints E^+ and E^- with constants (except at the initial and final times), so that $E^+ > 0$ is the physical size of the store and $E^- \geq 0$ is the minimum technically feasible level of the store. Choosing to represent E in this more general form allows the store, for example, to participate in multiple markets (such as forward or futures markets).

The total cost associated with a strategy q is denoted $C[q] \in \mathbb{R}$. One natural interpretation is as a variable cost in the usual economic sense, which may incorporate factors such as running costs, warming-up costs and costs of conversion to and from stored form, as well as the cost of purchasing power and the payments received for providing power. Later, in Section 5.3, we will consider the use of a store for smoothing the output of a wind farm and, in this case, C will be the total variation of net wind farm output. The aim of this paper is to identify those operating strategies q which minimize the total cost $C[q]$, whilst adhering to all of the physical constraints outlined above.

Definition 1.1 (Admissible and optimal strategies). We say that $q : [0, T] \rightarrow U$ is an admissible strategy if q is piecewise continuous and satisfies the capacity constraints

$$\ell[q](t) \in [E^-(t), E^+(t)] \quad \forall t \in [0, T]$$

and the terminal conditions $\ell[q](0) = \ell_0$ and $\ell[q](T) = \ell_T$. We denote by X the set of all admissible strategies and say that $q^* \in X$ is an *optimal strategy* if

$$C[q^*] \leq C[q] \quad \forall q \in X. \quad (3)$$

1.2. The “standard storage model”. Our standard storage model, which will be investigated in detail in Section 3 and in the examples of Sections 5.2 and 5.3, is a store whose cost may be written in the general form

$$C[q] = \int_0^T L(t, q(t)) \, dt \quad (4)$$

for some cost rate function $L : [0, T] \times U \rightarrow \mathbb{R}$ with $L(t, 0) = 0$ for all $t \in [0, T]$. In a price-arbitrage model, for example, $L(t, q(t))$ may be the cost per unit time of buying enough electrical power from the grid at time t in order to input power $q(t)$ into the store at that time (using the convention that if $q(t) < 0$, then this corresponds to outputting power $-q(t)$ from the store and selling). If the store is profitable to run, then the minimal cost should, of course, be negative. There may well be costs that are fixed or sunk and so do not vary with how the plant is operating (e.g. paying employees, rent and utilities for the building, interest on loans

etc.) but since these do not change the optimisation problem, we have subtracted them off, without loss of generality.

For technical reasons we will require that, for each $t \in [0, T]$, the map $\lambda \mapsto L(t, \lambda)$ is piecewise differentiable and that the associated partial derivative $\partial L(t, \lambda) / \partial \lambda$ has a continuous inverse at almost every $\lambda \in U$. The piecewise differentiability assumption allows for jumps in the cost which may relate, for example, to the cost of switching on an additional motor in order to provide a higher power output.

2. Characterization of optimal strategies. The following proposition provides a characterization of optimal strategies in terms of a multiplier function μ^* . Owing to its relation to the arbitrage model outline in Section 1.2, we will often refer to this as a “reference cost”.

Proposition 2.1 (Characterization of optimal strategies). *Suppose there exists a piecewise differentiable function $\mu^* : [0, T] \rightarrow \mathbb{R}$ and a strategy $q^* \in X$ with the following properties:*

(i) *The strategy q^* is a minimizer of*

$$C[q] - \int_0^T e^{\alpha t} \mu^*(t) q(t) dt \quad (5)$$

over all piecewise continuous functions $q : [0, T] \rightarrow U$.

(ii) *If μ^* is differentiable at $t \in [0, T]$, then the following complementary slackness conditions are satisfied:*

$$\dot{\mu}^*(t) = 0 \quad \text{if} \quad E^-(t) < \ell[q^*](t) < E^+(t), \quad (6)$$

$$\dot{\mu}^*(t) \geq 0 \quad \text{if} \quad \ell[q^*](t) = E^+(t), \quad (7)$$

$$\dot{\mu}^*(t) \leq 0 \quad \text{if} \quad \ell[q^*](t) = E^-(t). \quad (8)$$

(iii) *If μ^* is not differentiable at $t \in [0, T]$, then the following “jump” complementary slackness conditions are satisfied:*

$$\mu^*(t^+) - \mu^*(t^-) = 0 \quad \text{if} \quad E^-(t) < \ell[q^*](t) < E^+(t), \quad (9)$$

$$\mu^*(t^+) - \mu^*(t^-) \geq 0 \quad \text{if} \quad \ell[q^*](t) = E^+(t), \quad (10)$$

$$\mu^*(t^+) - \mu^*(t^-) \leq 0 \quad \text{if} \quad \ell[q^*](t) = E^-(t), \quad (11)$$

where $\mu^(t^-)$ and $\mu^*(t^+)$ are the left and right limits respectively of μ^* at t .*

Then q^ is an optimal strategy.*

Proof. Let $q \in X$ and let $0 = a_1 < \dots < a_n < a_{n+1} = T$ be a partition such that $\cup_{i=1}^n [a_i, a_{i+1}] = [0, T]$ and q^*, q are continuous and μ^* is differentiable over each (a_i, a_{i+1}) . Notice that (1) can be equivalently written as

$$\frac{d}{dt} (e^{\alpha t} \ell[q](t)) = e^{\alpha t} q(t) \quad \forall t \in [0, T].$$

Hence, (5) implies

$$\begin{aligned} C[q^*] - C[q] &\leq \int_0^T e^{\alpha t} \mu^*(t) (q^*(t) - q(t)) dt \\ &\stackrel{(1)}{=} \sum_{i=1}^n \int_{a_i}^{a_{i+1}} \mu^*(t) \left\{ \frac{d}{dt} (e^{\alpha t} \ell[q^*](t) - e^{\alpha t} \ell[q](t)) \right\} dt \end{aligned}$$

$$\begin{aligned}
&= \sum_{i=1}^n \left[e^{\alpha a_{i+1}} \mu^*(a_{i+1}^-) (\ell[q^*] - \ell[q])(a_{i+1}) - e^{\alpha a_i} \mu^*(a_i^+) (\ell[q^*] - \ell[q])(a_i) \right] \\
&\quad - \sum_{i=1}^n \int_{a_i}^{a_{i+1}} e^{\alpha t} \dot{\mu}^*(t) (\ell[q^*](t) - \ell[q](t)) dt \\
&= \sum_{i=2}^n e^{\alpha a_i} (\mu^*(a_i^-) - \mu^*(a_i^+)) (\ell[q^*] - \ell[q])(a_i) \\
&\quad - \sum_{i=1}^n \int_{a_i}^{b_i} e^{\alpha(t)} \dot{\mu}^*(t) (\ell[q^*] - \ell[q])(t) dt.
\end{aligned}$$

The second equality follows from integration by parts and the final equality results from a rearrangement of the first sum on the previous line, together with the condition that $\ell[q](0) = \ell[q^*](0) = E^-(0)$ and $\ell[q](T) = \ell[q^*](T) = E^-(T)$ if $q \in X$. Applying the complementary slackness conditions (6)-(11) to the final equality above, we obtain $C[q^*] - C[q] \leq 0$. \square

In a simple price-arbitrage model, as discussed in Section 1.2, the multiplier function μ^* provides a reference cost per unit of energy. Roughly, if the cost per unit of energy for operating the store is lower than the reference cost, then this indicates the store should charge at that time (and similarly for discharging).

It is worth mentioning here that exactly the same result follows if we apply the Karush-Kuhn-Tucker (KKT) conditions to the capacity constraints of the original minimization problem (3). By following such an approach, one instead looks for minimizers of the relaxed functional

$$C[q] - \int_0^T \left(\lambda_1(t) (\ell[q](t) - E^-(t)) + \lambda_2(t) (E^+(t) - \ell[q](t)) \right) dt$$

over all piecewise-continuous $q : [0, T] \rightarrow U$, where $\lambda_1, \lambda_2 : [0, T] \rightarrow \mathbb{R}$ are piecewise-continuous functions which satisfy the complementary slackness conditions

$$\lambda_1(t) (\ell[q](t) - E^-(t)) = \lambda_2(t) (E^+(t) - \ell[q](t)) = 0 \quad \forall t \in [0, T].$$

The relation between the two approaches is that the reference cost function μ^* in Proposition 2.1 can be expressed as an integral of the KKT functions:

$$\mu^*(t) = \int_t^T (\lambda_1(s) - \lambda_2(s)) ds,$$

and, in particular, $\dot{\mu}^*(t) = -\lambda_1(t) + \lambda_2(t)$.

3. Optimal strategies for the standard storage model. In Section 1.2 we introduced our standard storage model. We present here an algorithm to determine μ^* , and consequently the optimal strategy q^* , in this case. Our algorithm relies on Proposition 2.1 which we recall states that, if we can find the appropriate reference price function μ^* , then the optimal strategy $q^* \in X$ solves

$$\mathcal{L}[q^*, \mu^*] \leq \mathcal{L}[q, \mu^*] \quad \forall q : [0, T] \rightarrow U, \quad (12)$$

where

$$\mathcal{L}[q, \mu] := \int_0^T \left(L(t, q(t)) - e^{\alpha t} \mu(t) q(t) \right) dt \quad \forall q, \mu : [0, T] \rightarrow \mathbb{R}. \quad (13)$$

The regularity assumptions on L , q and μ imply that (12) reduces to a separate minimization problem for each time. Specifically, the optimal strategy $q^* \in X$ solves

$$L(t, q^*(t)) - e^{\alpha t} \mu^*(t) q^*(t) \leq L(t, w) - e^{\alpha t} \mu^*(t) w \quad \forall w \in U, \text{ almost all } t \in [0, T]. \quad (14)$$

Thus, as long as μ^* is known, the problem has now simplified into a collection of minimizations over bounded scalars, rather than the original minimization over a set of constrained functions. The challenge therefore is to determine μ^* . This is the objective of the algorithm.

An important result is Proposition 3.5, which states that if the algorithm does not terminate early, then it does indeed provide an optimal strategy q^* .

3.1. Preliminary results and definitions. We aim to find a piecewise differentiable $\mu^* : [0, T] \rightarrow \mathbb{R}$ such that the optimal strategy $q^* \in X$ solves (12). The conditions of Proposition 2.1 state that μ^* should be constant over intervals of time where the level of the store is away from the capacity constraints. This corresponds to intervals of the form $[\tau_k, \sigma_k)$ in the algorithm of Section 3.2 and, together with (14), motivates the following construction:

Given $t \in [0, T]$ and $\lambda \in \mathbb{R}$, we want to define a quantity $u(t, \lambda) \in \mathbb{R}$ as a solution of

$$u(t, \lambda) = \operatorname{argmin}_{w \in U} (L(t, w) - e^{\alpha t} \lambda w). \quad (15)$$

If we know that μ^* is constant over some interval $[\tau, \sigma] \subset [0, T]$, then the task of the algorithm reduces to finding the correct λ and setting $q^*(t) = u(t, \lambda)$ for almost all $t \in [\tau, \sigma]$. Care needs to be taken, however, because there may be multiple minimizers $u(t, \lambda)$ associated with λ . With this in mind, we denote by $\mathcal{M}_t \subset \mathbb{R}$ the set of λ which admit more than one minimizer in (15), and deal with this technicality in the following lemma. Note that, if L is assumed to be strictly convex in its second argument, then minimizers of (5) are of course unique, implying that each $\mathcal{M}_t = \emptyset$. The following lemma states the important result that $u(t, \lambda)$ is monotone increasing with λ . This property will be key to the algorithm.

Lemma 3.1 (Monotonicity and regularity properties). *For each $(t, \lambda) \in [0, T] \times \mathbb{R}$, let $u(t, \lambda)$ be any solution of (15). Then, at each $t \in [0, T]$, the mapping $\lambda \mapsto u(t, \lambda)$ is monotone increasing and piecewise continuous. Moreover, the set \mathcal{M}_t is finite and discontinuities in $\lambda \rightarrow u(t, \lambda)$ occur only at points in \mathcal{M}_t .*

Proof. Let $t \in [0, T]$ and suppose that the map $\lambda \mapsto u(t, \lambda)$ is not monotone increasing. Then, there exists $\lambda_1 < \lambda_2$ such that $u_1 := u(t, \lambda_1) > u(t, \lambda_2) =: u_2$. But then,

$$L(t, u_1) - e^{\alpha t} \lambda_2 u_1 \leq L(t, u_2) - e^{\alpha t} \lambda_1 u_2 - e^{\alpha t} (\lambda_2 - \lambda_1) u_1 < L(t, u_2) - e^{\alpha t} \lambda_2 u_2.$$

The first inequality follows from the definition of u_1 and u_2 as solutions to (15), and the second inequality follows from the supposition. However, the above contradicts the definition of u_2 , and we conclude that the map $\lambda \mapsto u(t, \lambda)$ is monotone increasing at all $t \in [\tau, T]$.

The piecewise continuity of the map $\lambda \mapsto u(t, \lambda)$ follows immediately from the regularity assumptions on L . Precisely, if for almost all $\lambda \in \mathbb{R}$, the minimizers $u(t, \lambda)$

lie away from discontinuities of L , then they must satisfy

$$\left. \frac{\partial L}{\partial w}(t, w) \right|_{w=u(t, \lambda)} - e^{\alpha t} \lambda = 0 \quad \text{a.e. } \lambda \in \mathbb{R}.$$

The continuous invertibility assumption on the partial derivative of L therefore implies the piecewise continuity of the map $\lambda \mapsto u(t, \lambda)$. If, on the other hand, there is a non-degenerate subset $W \subset \mathbb{R}$ such that $\lambda \in W$ implies $u(t, \lambda)$ lies at a discontinuity of L , then the above monotonicity property and the piecewise continuity of L imply that the map $\lambda \mapsto u(t, \lambda)$ must be piecewise constant over W . In either case, the map $\lambda \mapsto u(t, \lambda)$ is piecewise continuous.

Finally, we prove the finiteness of the set \mathcal{M}_t by supposing that the opposite is true. To this end, let $\mathcal{A} \subset \mathbb{R}$ be an infinite set such that, for each $\lambda \in \mathcal{A}$, there are two distinct local solutions $x(\lambda), y(\lambda) \in U \setminus \partial U$ to (15), where ∂U denotes the boundary of the set U . Let $\lambda_0 \in \mathcal{A}$ be such that there exists a sequence $(\lambda_n)_{n \in \mathbb{N}}$ in \mathbb{R} with $\lim_{n \rightarrow \infty} \lambda_n = \lambda_0$, and set $x(\lambda_0) = x_0$ and $y(\lambda_0) = y_0$. Assume without loss of generality that

$$L_u(t, x(\lambda)) = L_u(t, y(\lambda)) = e^{\alpha t} \lambda$$

is satisfied at $\lambda = \lambda_0$, where $L_u(t, \cdot)$ denotes the partial derivative of L with respect to the second argument, and that $x(\lambda)$ and $y(\lambda)$ lie away from any discontinuity of L . Setting $g(\lambda) := L(t, x(\lambda)) - \lambda x(\lambda)$ and $h(\lambda) := L(t, y(\lambda)) - \lambda y(\lambda)$, we obtain

$$g'(\lambda_0) = L_u(t, x_0)x'(\lambda_0) - x'(\lambda_0)\lambda_0 - x_0 = (L_u(t, x_0) - \lambda_0)x'(\lambda_0) - x_0 = -x_0,$$

and similarly, $h'(\lambda_0) = -y_0$. Hence, if x_0 and y_0 are both global minimizers solving (15) but $x_0 \neq y_0$, then there exists $N \in \mathbb{N}$ such that $g(\lambda_n) \neq h(\lambda_n)$ for all $n \geq N$. In particular, $x(\lambda_n)$ and $y(\lambda_n)$ cannot both be minimizers for $n \geq N$. This contradicts the assumption that there exists such a set \mathcal{A} . \square

The definitions which follow will be employed in the algorithm and are written under the assumption that there exists a pair (μ^*, q^*) which satisfies the conditions of Proposition 2.1. Condition (6) and inequality (14) motivate the following definition.

Definition 3.2 ((λ, τ) -admissibility). For any choice of $\lambda \in \mathbb{R}$ and $\tau \in [0, T]$, we say a piecewise continuous function $q : [0, T] \rightarrow U$ is (λ, τ) -admissible if $q \equiv 0$ over $[0, \tau]$ and if

$$q(t) \in \operatorname{argmin}_{w \in U} \left(L(t, w) - e^{\alpha t} \lambda w \right) \quad \text{for almost all } t \in [\tau, T]. \quad (16)$$

We say that q is τ -admissible if there exists a $\lambda \in \mathbb{R}$ such that q is (λ, τ) -admissible.

Note that, in general, a (λ, τ) -admissible strategy q is not admissible: for a general choice of λ , one of the capacity constraints is likely to be broken at some time in $(\tau, T]$. The idea of the algorithm is to piece together (λ, τ) -admissible strategies, adjusting λ at times when the store is full or empty, in order to construct the optimal strategy $q^* \in X$. The reference cost μ^* attains the appropriate value of λ over intervals of time when μ^* is constant. In the special case that one can find a $(\lambda, 0)$ -admissible strategy which is also admissible, then of course we are done, since we may set $\mu^* \equiv \lambda$. Such a strategy satisfies condition (6) at every $t \in [0, T]$.

The key feature of the algorithm is to determine: (i) the intervals of time (t_k, σ_k) over which μ^* is constant and (ii) the value λ_k that μ^* attains over these intervals.

We may then set q^* to coincide with a (λ_k, τ_k) -admissible strategy over these intervals. The correct choice of μ^* relies on the following characterization of strategies.

Definition 3.3 (Characterization of strategies). Let $\tau \in [0, T]$, $m \in [E^-(\tau), E^+(\tau)]$. We characterize each τ -admissible q as follows:

- (i) We write $q \in X^-(\tau, m)$ if there exists $e \in (\tau, T)$ and $\varepsilon_0 > 0$ such that

$$E^-(t') \leq m + \ell[q](t') \leq E^+(t') \quad \forall t' \in (\tau, e) \quad (17)$$

and $m + \ell[q](e + \varepsilon) < E^-(e + \varepsilon)$ for all $\varepsilon \in (0, \varepsilon_0)$. We call e the exit time associated with q .

- (ii) Similarly, we write $q \in X^+(\tau, m)$ if there exists $e \in (\tau, T)$ and $\varepsilon_0 > 0$ such that condition (17) holds and $m + \ell[q](e + \varepsilon) > E^+(e + \varepsilon)$ for all $\varepsilon \in (0, \varepsilon_0)$. Again, we call e the exit time associated with q .

We write $\lambda \in \Lambda^-(\tau, m)$ if there exists a (λ, τ) -admissible $q \in X^-(\tau, m)$, and $\lambda \in \Lambda^+(\tau, m)$ if there exists a (λ, τ) -admissible $q \in X^+(\tau, m)$.

In particular, the set $X^-(\tau, m)$ contains all τ -admissible q whose first violation of the capacity constraints would occur at the lower boundary E^- if the level of the store at time τ were known to be m (and similarly for $X^+(\tau, m)$). The monotonicity property of Lemma 3.1 implies that $\Lambda^-(\tau, m)$ and $\Lambda^+(\tau, m)$ are connected subintervals of \mathbb{R} satisfying

$$\sup \Lambda^-(\tau, m) \leq \inf \Lambda^+(\tau, m) \quad \forall (\tau, m) \in [0, T] \times [0, M]. \quad (18)$$

3.2. The algorithm. We are now in a position to present the algorithm, which will identify an $N \in \mathbb{N}$ and two increasing sequences of times $\{\tau_i\}_{i=1}^N, \{\sigma_i\}_{i=1}^N$ such that $\tau_i < \sigma_i < \tau_{i+1}$ for each i , and such that $\mu^*(t) = 0$ whenever $t \in (\tau_i, \sigma_i)$. The multiplier μ^* is only allowed to jump in value at times τ_i or σ_i . Fixing $\tau \in [0, T]$, we suppose that μ^* and q^* are known over $[0, \tau]$ and that the algorithm has already identified all of the times $\tau_1, \dots, \tau_{k-1}$ and $\sigma_1, \dots, \sigma_{k-1}$ which apply over this interval, with $\tau_k < \sigma_k \leq \tau$. We further assume without loss of generality that $m := \ell[q^*](\tau) \in \{E^-(\tau), E^+(\tau)\}$ (otherwise, move τ backwards until this is satisfied).

Step 1. Set $\lambda = \sup \Lambda^-(\tau, m)$ (Defn 3.3). If there exists a (λ, τ) -admissible q such that $q \notin X^-(\tau, m) \cup X^+(\tau, m)$, then set $\tau_k = \tau$, $\sigma_k = e_k = T$ and $\mu^*(t) = \lambda$ for all $t \in (\tau, T]$ and define the restriction of q^* to $(\tau_k, T]$ to be q . **The algorithm is complete.**

If there is no such q , proceed to step 2.

Step 2. Again, set $\lambda := \sup \Lambda^-(\tau, m)$. There are three cases. For each case, we either set $\tau_k = \tau$ and define the pair (μ^*, q^*) restricted to an interval $(\tau_k, \sigma_k]$, or we proceed to the next step in order to identify $\tau_k > \tau$.

- a) If $\lambda \in \Lambda^-(\tau, m) \setminus \Lambda^+(\tau, m)$: Consider the set Λ of (λ, τ) -admissible q such that $m + \ell[q](\sigma) = E^+(\sigma)$ for some $\sigma \in [\tau, e)$, where e is the exit time associated with q (cf. Definition 3.3). Select the latest such σ and let $q \in \Lambda$ be the (λ, τ) -admissible strategy associated with it.

If $\sigma = \tau$, proceed to step 3a.

If $\sigma > \tau$, then set $\tau_k = \tau$, $\sigma_k = \sigma$ and $e_k = e$. Set $\mu^*(t) = \lambda$ for all $t \in (\tau_k, \sigma_k]$ and define the restriction of q^* to $(\tau_k, \sigma_k]$ to coincide with q . If $\sigma_k < T$, proceed to step 3b; if $\sigma_k = T$, proceed to step 4.

- b) If $\lambda \in \Lambda^+(\tau, m) \setminus \Lambda^-(\tau, m)$: Consider the set Λ of (λ, τ) -admissible $q \in X^+(\tau, m)$ such that $m + \ell[q](\sigma) = E^-(\sigma)$ for some $\sigma \in [\tau, e]$, where e is the exit time associated with q . Select the latest such σ and let $q \in \Lambda$ be the (λ, τ) -admissible strategy associated with it.

If $\sigma = \tau$, proceed to step 3a.

If $\sigma > \tau$, then set $\tau_k = \tau$, $\sigma_k = \sigma$ and $e_k = e$. Set $\mu^*(t) = \lambda$ for all $t \in (\tau_k, \sigma_k]$ and define the restriction of q^* to $(\tau_k, \sigma_k]$ to coincide with q . If $\sigma_k < T$, proceed to step 3b; if $\sigma_k = T$, proceed to step 4.

- c) If $\lambda \in \Lambda^-(\tau, m) \cap \Lambda^+(\tau, m)$: Consider the set Λ of (λ, τ) -admissible q such that either $m + \ell[q](\sigma) = E^+(\sigma)$ or $m + \ell[q](\sigma) = E^-(\sigma)$ for some $\sigma \in [\tau, e]$, where e is the exit time associated with q . Select the latest such σ and let $q \in \Lambda$ be the (λ, τ) -admissible strategy associated with it.

If $\sigma = \tau$, proceed to step 3a.

If $\sigma > \tau$, then set $\tau_k = \tau$, $\sigma_k = \sigma$ and $e_k = e$. Set $\mu^*(t) = \lambda$ for all $t \in (\tau_k, \sigma_k]$ and define the restriction of q^* to $(\tau_k, \sigma_k]$ to coincide with q . If $\sigma_k < T$, proceed to step 3b; if $\sigma_k = T$, proceed to step 4.

Step 3a. For each $s > \tau$, let $m(s) = E^-(s)$ if $m = E^-(\tau)$; and $m(s) = E^+(s)$ if $m = E^+(\tau)$. Select the first time $s \in (\tau, T]$ such that, on relabelling $m(s)$ as m , and s as τ , and on determining the new associated σ from steps 1 and 2, we have $\sigma > \tau$. Set $\tau_k = s$ and return to step 1.

Step 3b. For each $s > \tau$, let $m(s) = E^-(s)$ if $m = E^-(\tau)$; and $m(s) = E^+(s)$ if $m = E^+(\tau)$. Select the first time $s \in (\tau, T]$ such that, on relabelling $m(s)$ as m , and s as τ , and on determining the new associated σ from steps 1 and 2, we have $\sigma > \tau$. Set $\tau_{k+1} = s$ and return to step 1.

If there is no such τ_{k+1} , then we have found all intervals (τ_i, σ_i) over which μ^* is constant and have defined the restriction of q^* over these intervals. Proceed to step 4.

Step 4. Over each interval $(\sigma_i, \tau_{i+1}]$, corresponding to the times found in the steps above, and for each $t \in (\sigma_i, \tau_{i+1}]$, define

$$q(t) = \begin{cases} \dot{E}^+(t) + \alpha E^+(t) & \text{if } \ell[q^*](\sigma_i) = E^+(\sigma_i) \\ \dot{E}^-(t) + \alpha E^-(t) & \text{if } \ell[q^*](\sigma_i) = E^-(\sigma_i). \end{cases}$$

Define the restriction of q^* to the union of these intervals to coincide with q . We have now constructed a strategy $q^* \in X$.

Step 5. For each $i \in \{1, \dots, N\}$, check if there exists a piecewise differentiable $\mu_i : (\sigma_i, \tau_{i+1}) \rightarrow \mathbb{R}$ such that the following conditions hold:

- $q^*(t)$ is a minimizer of $L(t, x) - \mu_i(t)x$ over all $x \in \mathbb{R}$.
- If $\ell[q^*](\sigma_i) = E^+(\sigma_i)$, then μ_i is non-decreasing over (σ_i, τ_{i+1}) and satisfies $\mu_i(\sigma_i^+) \geq \mu^*(\sigma_i^-)$ and $\mu_i(\tau_{i+1}^-) \leq \mu^*(\tau_{i+1}^+)$.
- If $\ell[q^*](\sigma_i) = E^-(\sigma_i)$, then μ^* is non-increasing over $(\sigma_i, \tau_{i+1}]$ and satisfies $\mu_i(\sigma_i^+) \leq \mu^*(\sigma_i^-)$ and $\mu_i(\tau_{i+1}^-) \geq \mu^*(\tau_{i+1}^+)$.

If such a μ_i exists for each $i \in \{1, \dots, N\}$, then define the restriction of μ^* to each of the intervals (σ_i, τ_i) to coincide with μ_i . We have now constructed a pair (μ^*, q^*) such that q^* is optimal (see Proposition 3.4).

If, on the other hand, there exists $i \in \{1, \dots, N\}$ with no such μ_i , then we have constructed a strategy $q^* \in X$ but no corresponding multiplier $\mu^* : [0, T] \rightarrow \mathbb{R}$. In this case, q^* may not be optimal, and we say **the algorithm terminated early**.

End of algorithm

3.3. Optimality of the strategies generated by the algorithm. The following result states that, whenever the algorithm yields a pair (μ^*, q^*) , then q^* is an optimal strategy, according to Proposition 2.1. The converse is then given by Proposition 3.5: if there is an optimal strategy which satisfies the conditions of Proposition 2.1, then the algorithm will find it.

Proposition 3.4. *If the above algorithm yields a pair (μ^*, q^*) , then q^* is optimal.*

Proof. Let (μ^*, q^*) be the pair which is determined through the algorithm and let $\{\tau_i\}_{i=1}^N$ and $\{\sigma_i\}_{i=1}^N$ be the associated sequences of times, so that μ^* is constant over each interval (τ_i, σ_i) . Assume that the conditions of Proposition 2.1 are satisfied up until time $\tau_k \in [0, T)$. At each $i \in \{1, \dots, N\}$, we have

$$\sup \Lambda^-(\tau_i, \ell[q^*](\tau_i)) \leq \mu^*(\tau_i^+) \leq \inf \Lambda^+(\tau_i, \ell[q^*](\tau_i)). \quad (19)$$

If the conditions of step 1 are satisfied at τ_k , then q^* is clearly admissible and satisfies the properties of the proposition. Hence, q^* is optimal and the algorithm is complete.

Assume therefore that the conditions of step 1 do not hold at τ_k so that $\sup \Lambda^-(\tau_k, m) = \inf \Lambda^+(\tau_k, m)$. It is clear that the strategy defined by the algorithm is admissible, and it remains to check that conditions (6)-(11) are satisfied by μ^* .

To this end, note that, the algorithm ensures that the conditions are satisfied over the intervals (τ_k, σ_k) and $(\sigma_k, \tau_{k+1}]$, and it remains to check that the conditions hold at σ_k . Suppose first that case a) of step 2 is satisfied at time τ_k , so that $\ell[q^*](\sigma_k) = E^+(\sigma_k)$. Then, by construction, we have $\mu^*(\sigma_k^-) \in \Lambda^-(\sigma_k, E^+(\sigma_k))$. Thus, if $\sigma_k = \tau_{k+1}$ then, together with (19), this implies

$$\mu^*(\sigma_k^-) \leq \sup \Lambda^-(\sigma_k, E^+(\sigma_k)) \leq \mu^*(\sigma_k^+) \quad (20)$$

and condition (10) is satisfied at σ_k . If, on the other hand, $\sigma_k < \tau_{k+1}$, then the construction of μ^* through the algorithm immediately implies that $\mu^*(\sigma_k^-) \leq \mu^*(\sigma_k^+)$.

Similarly, if case b) of step 2 holds at τ_k , then $\ell[q^*](\sigma_k) = E^-(\sigma_k)$ and $\mu^*(\sigma_k) \in \Lambda^+(\sigma_k, E^-(\sigma_k))$. Thus, if $\sigma_k = \tau_{k+1}$ then, together with (19), this implies

$$\mu^*(\sigma_k) \geq \inf \Lambda^+(\sigma_k, E^-(\sigma_k)) \geq \mu^*(\sigma_k^+) \quad (21)$$

and condition (11) is satisfied at σ_k . If, on the other hand, $\sigma_k < \tau_{k+1}$, then the construction of μ^* through the algorithm immediately implies that $\mu^*(\sigma_k^-) \geq \mu^*(\sigma_k^+)$.

Finally, if case c) holds at τ , then the above two cases together imply that the conditions of Proposition 2.1 are satisfied. \square

Proposition 3.5. *If there is a pair (μ, q) which satisfies the conditions of Proposition 2.1, then the strategy $q^* \in X$ generated by the algorithm is optimal.*

Proof. Let (μ, q) be a pair satisfying the conditions of Proposition 2.1 and let q^* be the strategy generated by the algorithm. At the initial time $t_0 = 0$, there exists an interval $[t_0, s) \subset [0, T]$ with two associated options: either (i) μ is constant over this interval, or (ii) μ is varying over this interval, in the sense that $\dot{\mu}(t) \neq 0$ or $\mu(t^+) - \mu(t^-) \neq 0$ for almost all $t \in (t_0, s)$. We will prove the following two claims:

Claim 1. *If case (i) holds and $[t_0, s)$ is the largest connected interval over which μ is constant, then the algorithm yields a constant μ^* over this interval and $\ell[q](s) = \ell[q^*](s) \in \{E^-(s), E^+(s)\}$. Moreover, we have $\mu(t) = \mu^*(t)$ for all $t \in [t_0, s)$ (or,*

we may adjust μ so that this is correct, without violating the conditions of Proposition 2.1).

Claim 2. *If case (ii) holds, then $q(t) = q^*(t)$ for almost all $t \in [t_0, s)$. In particular, we may set $\mu^*(t) = \mu(t)$ for all $t \in [t_0, s)$ in step 5 of the algorithm and we have $\ell[q](s) = \ell[q^*](s)$.*

This will complete the proof of Proposition 3.5 since, as long as $\ell[q](s) = \ell[q^*](s)$, we may re-set the start time as $t_0 = s$ and re-apply these claims to find another interval $[t_0, s')$ over which the above results hold. Since μ^* coincides with μ over $[t_0, s')$, we conclude that the pair (μ^*, q^*) generated by the algorithm satisfies the conditions of Proposition 2.1 over this interval (and consequently over all of $[0, T]$ by iteration).

We will employ the notation $\lambda \sim \lambda'$ if the set of (λ, t_0) -admissible functions coincides exactly with the set of (λ', t_0) -admissible functions over the interval (t_0, s) and we introduce the ordering $\lambda \prec \lambda'$ if $\lambda \approx \lambda'$ and $\lambda < \lambda'$. We set $m_0 = \ell[q](t_0)$ and $\lambda^* = \sup \Lambda^-(t_0, m_0)$.

To prove Claim 1, suppose that case (i) holds. The monotonicity and continuity properties of Lemma 3.1 imply for $\theta \in \mathbb{R}$ the relations:

$$\theta \prec \lambda^* \quad \Rightarrow \quad \theta \in \Lambda^-(t_0, m_0) \setminus \Lambda^+(t_0, m_0) \quad (22)$$

$$\theta \succ \lambda^* \quad \Rightarrow \quad \theta \in \Lambda^+(t_0, m_0) \setminus \Lambda^-(t_0, m_0). \quad (23)$$

Setting $\lambda = \mu(t)$ for all $t \in [t_0, s)$, suppose that $\lambda \prec \lambda^*$. Then we can choose θ such that $\lambda \prec \theta \prec \lambda^*$. If $\ell[q](s) = E^+(s)$, then the left-hand inequality $\lambda \prec \theta$, together with the continuity and monotonicity of the map u from Lemma 3.1, guarantees the existence of a (θ, t_0) -admissible q' (cf. Definition 3.2) such that $m_0 + \ell[q'](t) \geq E^+(t)$ for all $t \in [t_0, s]$, with strict inequality at $t = s$. This implies that $\theta \in \Lambda^+(t_0, m_0)$, which contradicts (22).

If, on the other hand, $\ell[q](s) = E^-(s)$, then Proposition 2.1 requires that $\mu(s^+) - \mu(s^-) \leq 0$. If μ is constant over an interval $(s, t') \subset (s, T]$, then (22) implies that the lower capacity constraint is broken by q , which contradicts the assumption that $q \in X$. If μ is not constant over any such interval, then Proposition 2.1 requires that $q(t) = u(t, \mu(t)) = \dot{E}^-(t) + \alpha E^-(t)$ for all $t \in (s, t')$, where u is the map of Lemma 3.1. By the monotonicity of u , this is possible only if $\mu(t^+) - \mu(t^-) > 0$ at some $t \in (s, t')$, since otherwise the fact that $\lambda \in \Lambda^-(t_0, m_0)$ implies that q will break the lower capacity constraint. However, this contradicts condition (8) of the proposition. Hence, we must have $\lambda \sim \lambda^*$ or $\lambda \succ \lambda^*$. If $\lambda \succ \lambda^*$, then similar arguments as above lead us to analogous contradictions. We conclude that $\lambda \sim \lambda^*$, as stated in Claim 1 above.

The remainder of Claim 1 states that $\ell[q](s) = \ell[q^*](s)$. Since $\lambda \sim \lambda^*$ (and therefore μ and μ^* coincide over $[t_0, s)$), this is always true if (16) admits a unique minimizer at each $t \in [t_0, s)$. However, if this uniqueness does not hold, then there are potentially two (or more) strategies $q^1, q^2 \in X$ which satisfy the conditions of Proposition 2.1, and we need to make sure that Claim 1 holds regardless of which strategy we choose. Let x^1, x^2 be the corresponding (λ, t_0) -admissible strategies and relabel our previous value of s as s^1, s^2 , so that $q^i(t) = x^i(t)$ for all $t \in [t_0, s^i)$, for $i = 1, 2$. Let e^1 and e^2 be the corresponding exit times for x^1 and x^2 respectively (cf. Definition 3.3), with $s^1 < e^1$ and $s^2 < e^2$. As shown above, the corresponding multipliers μ^1, μ^2 both satisfy that $\mu^i(t) \sim \lambda^*$ for all $t \in [0, s^i)$, $i = 1, 2$, but according to our definition for s , the value of each μ^i must change at time s^i , so

that $\lambda_+^i := \mu((s^i)^+) \approx \lambda$. In particular, this means that if $x^i \in X^-(t_0, m_0)$, then we must have $\ell[q^i](s^i) = E^+(s^i)$ and $\lambda_+^i \succ \lambda$, since otherwise Proposition 2.1 tells us that $\ell[q^i](s^i) = E^-(s^i)$ and $\lambda_+^i \prec \lambda$, which would be a contradiction because, by the monotonicity property of Lemma 3.1, this would mean that q^i breaks the lower capacity constraint. Similarly, if $x^i \in X^+(t_0, m_0)$, then we conclude that $\ell[q^i](s^i) = E^-(s^i)$ and $\lambda_+^i \prec \lambda$. Assuming for now that $x^1 \in X^-(t_0, m_0)$, we consider the following (exhaustive list of) cases: (a) $x^2 \in X^+(t_0, m_0)$ and $s^1 \leq e^2$ and $s^2 \leq e^1$; (b) $x^2 \in X^+(t_0, m_0)$ and either $s^1 > e^2$ or $s^2 > e^1$; (c) $x^2 \in X^-(t_0, m_0)$; or (d) $x^2 \notin X^-(t_0, m_0) \cup X^+(t_0, m_0)$.

If case (a) holds, then $\ell[q^1](s^1) = E^+(s^1)$ and $\lambda_+^1 \succ \lambda$. However, we can construct a new (λ, t_0) -admissible x which agrees with x^1 over $[t_0, s^1]$ and which agrees with x^2 over $[s^1, s^2]$. By construction, $x \in X^+(t_0, m_0)$. Thus, the monotonicity property of Lemma 3.1 implies that if $\lambda_+^1 \succ \lambda$, then q^1 must break the upper capacity constraint at some time, which contradicts $q^1 \in X$. A similar argument reveals an analogous contradiction for q^2 . Thus, q cannot coincide with x^i over $[t_0, s^i]$, for $i = 1, 2$ but rather must coincide with some other (λ, t_0) -admissible x over its corresponding interval $[t_0, s]$.

If case (b) holds and $s^1 > e^2$, then $\ell[q^2](s^2) = E^-(s^2)$ and $\lambda_+^2 \prec \lambda$. By constructing another (λ, t_0) -admissible x which agrees with x^2 over $[t_0, e^2]$ and agrees with x^1 over $[e^2, s^1]$, we have $x \in X^-(t_0, m_0)$ and conclude that q^2 must break the lower capacity constraint at some time. Thus, q cannot coincide with x^2 over $[t_0, s^2]$. Similarly, q cannot coincide with x^1 over $[t_0, s^1]$ if instead $s^2 > e^1$.

Similarly, if case (c) holds and $s^1 < s^2$, then $\ell[q^1](s^1) = E^+(s^1)$ and $\lambda_+^1 \prec \lambda$. Constructing a (λ, t_0) -admissible $x \in X^+(t_0, m_0)$ which agrees with x^1 over $[t_0, s^1]$ and agrees with x^2 over $[s^1, s^2]$, again implies that q^1 must break the upper capacity constraint at some time. Thus, q cannot coincide with x^1 over $[t_0, s^1]$. A similar argument holds if $s^2 < s^1$.

Finally, if case (d) holds, we again have that $\ell[q^1](s^1) = E^+(s^1)$ and $\lambda_+^1 \prec \lambda$, and we can construct a (λ, t_0) -admissible $x \in X^+(t_0, m_0)$ which agrees with x^1 over $[t_0, s^1]$ and agrees with x^2 over $[s^1, T]$. Thus q^1 must break the upper capacity constraint at time T and we conclude that q cannot coincide with x^1 over $[t_0, s^1]$.

A unifying feature of the above cases (a)–(d) is that, if $s^1 < s^2$, then we arrive at a contradiction. Thus, the pair (μ, q) can satisfy the conditions of Proposition 2.1 only if q agrees with the (λ, t_0) -admissible x which has the latest corresponding update time s . This means that, not only does the algorithm choose $\lambda^* \sim \lambda$ over the interval $[t_0, \sigma_1]$ over which μ^* is constant (with σ_1 being the first update time, as defined in the algorithm), but also that $\sigma_1 = s$. Analogous results hold if we assume instead that $x^1 \in X^+(t_0, m_0)$. Thus, the only case not covered above (after relabelling x^1 and x^2 if necessary) is where $x^1, x^2 \in X$ are both admissible. However, by definition and by step 1 of the algorithm, we here have $s = \sigma_1 = T$, $\lambda \sim \lambda^*$ and $\ell[q](T) = \ell[q^*](T) = \ell_T$. Thus, the same conclusions hold.

The only remaining issue now, to complete the proof of Claim 1, is to show that $\ell[q^1](s^1) = \ell[q^2](s^2)$ whenever $s^1 = s^2$. However, we can only have $s^1 = s^2$ in cases (a) or (c), or if $x^1, x^2 \in X$ are both admissible (in which case, $s^1 = s^2 = T$ and we have already shown that we are done). In case (a), we have shown that neither q^1 nor q^2 are admissible, and so must search for a new (λ, t_0) -admissible x with a later update time s ; whilst in case (c), we have by definition $\ell[q^1](s^1) = \ell[q^2](s^2) = E^+(s^2)$. This completes the proof of Claim 1.

To prove Claim 2, suppose that case (ii) holds. Then, by Proposition 2.1, the level of stored energy remains fixed at a capacity constraint throughout $[0, s)$ and we may assume, without loss of generality, that $\ell[q](t) = E^-(t)$ for all $t \in [0, s)$ and consequently that μ is decreasing over this interval. Suppose now that there is a connected interval $I \subset [t_0, s)$, containing t_0 , such that $q(t) \neq q^*(t)$ for almost all $t \in I$. Then, the algorithm tells us that μ^* attains a constant value λ^* over all of I and that there is a (λ^*, t_0) -admissible x which coincides with q^* over $[t_0, \sigma_1)$. Since q remains at the lower capacity constraint over $[t_0, s)$, we have $q(t) \leq x(t)$ and $\mu(t) \prec \lambda^*$ for all $t \in I \cap [t_0, s)$. (If, instead, $\mu(t) \sim \lambda^*$ for all $t \in I \cap [t_0, s)$, then we have already dealt with this case in Claim 1.) If $x \notin X^+(t_0, m_0)$, then (22) together with conditions (8) and (11), imply that q breaks the lower capacity constraint before the exit time associated with μ^* , thus contradicting the assumption that $q \in X$. If, on the other hand, $x \in X^+(t_0, m_0)$, then conditions (8) and (11) imply that q breaks the lower capacity constraint before μ^* changes value, again contradicting the assumption that $q \in X$. This completes the proof of Claim 2. \square

Proposition 3.6. *If the map $x \mapsto L(t, x)$ is convex for all $t \in [0, T]$, then there exists a solution of the form of Proposition 2.1.*

Proof. The proof is a direct analogue of that presented in [10], so we do not replicate it here. \square

4. The time-localization property. Consider the standard storage model of Section 1.2. The algorithm presented in Section 3.2 reduces the original problem - an optimization over a constrained set of functions - to a series of new simpler optimizations over a set of bounded scalars. Moreover, each of these new optimizations are localized in time, in the sense that at each time τ_k from the algorithm, one needs cost information only up to the associated exit time e_k to know how to operate the store over (τ_k, σ_k) . Indeed, the same is true for each time $t \in (\tau_k, \sigma_k)$. We call this property the “time-localization property”.

4.1. An intuitive illustration. As an illustration of both the algorithm and the time-localization property, consider a simple arbitrage model of the standard storage model form, with no leakage ($\alpha = 0$) and with cost rate function

$$L(t, x) = \begin{cases} p(t)x/\epsilon_1 & \text{if } x \geq 0 \\ \epsilon_2 p(t)x & \text{if } x < 0 \end{cases} \quad \forall t \in [0, T], x \in U, \quad (24)$$

where, for each $t \in [0, T]$, $p(t) \in [0, \infty)$ is the price of unit energy. Let $U = [-q^-, q^+]$ be the range of admissible powers, with $q^-, q^+ \in (0, \infty)$ denoting the discharge and charge power ratings of the store respectively. $\epsilon_1, \epsilon_2 \in (0, 1]$ are efficiency factors: in order to put one unit of electrical energy into the store, the operator must buy more than this to account for losses during the charging process; similarly, when discharging one unit of energy from the store, some energy is lost during the discharge process and the operator can only sell a fraction ϵ_2 of this energy. We also assume that the store starts and finishes empty so that $\ell_0 = \ell_T = 0$.

Both Proposition 2.1 and the algorithm say that the optimal strategy consists of intervals of time over which the multiplier is constant. These intervals are each of one of the following forms: \mathcal{F} (the store is full at the end of the interval, without touching the capacity constraints in between the start and end time of the interval); and \mathcal{E} (the store is empty at the end of the interval, again not touching the capacity constraints in between the start and end times of the interval). Sometimes the

algorithm may yield a multiplier which is constant over longer intervals of time, so that a capacity constraint is attained somewhere away from the end-times, but the same categorization holds on splitting such intervals up into smaller ones. In the complement of these intervals of constant multiplier, the value of the multiplier is allowed to vary but the level of stored energy must remain at one of the capacity constraints. In the case of our simple arbitrage model above, a multiplier λ can be interpreted as a reference cost which dictates whether the store should be charging, discharging or doing nothing at any given time, according to (15), with

$$q^*(t) = \begin{cases} q^+ & \text{if } p(t)/\epsilon_1 < \lambda \\ 0 & \text{if } \epsilon_2 p(t) < \lambda < p(t)/\epsilon_1 \\ -q^- & \text{if } \lambda < \epsilon_2 p(t). \end{cases} \quad (25)$$

Proposition 2.1 states that the reference cost can increase only if the store is full. In terms of (25), this can be interpreted as the operator waiting for prices to rise to a high enough level that it is worth selling energy. Similarly, the reference cost can fall only when the store is empty and the operator is waiting for low buying prices. If $\lambda = p(t)/\epsilon_1$ or $\lambda = \epsilon_2 p(t)$, then we have to choose the appropriate value for $q^*(t)$ via the algorithm but, for simplicity, let us assume that the price is always varying so that the set of times at which one of these conditions occurs is of zero measure and therefore the choice of q^* at these times does not matter. An alternative approach is to make (24) strictly convex by appending a small strictly convex function to L , e.g. replacing $L(t, x)$ with $L(t, x) + \delta x^2$ for some small $\delta > 0$. Then, the solution to (16) is unique for each multiplier λ and the optimal strategy obtained with this modified cost rate function will converge to q^* as δ tends to zero. This approximation is useful for practical computations where $p(t)$ might be constant over intervals.

The purpose of steps 1–3b of the algorithm is to identify these intervals (which are labelled (τ_k, σ_k)) of constant multiplier and to choose the correct value for the multiplier over these times. To illustrate the time-localization property, suppose that the multiplier attains a constant value λ over some interval (τ, σ) and that this interval is of type \mathcal{E} . Suppose now that we can accurately forecast prices up to some time $\bar{h} > \sigma$. If, on keeping the multiplier constant and extending the associated strategy (as defined by (25)) up to time \bar{h} , we find that the store becomes full at some time $h \in (\sigma, \bar{h}]$, then any price information beyond time h is irrelevant to the actions of the store over (τ, σ) : if prices are relatively high after time h , then the store is empty at time σ but is able to charge up at suitably low prices over the interval (σ, h) , in preparation for discharging after time h ; if prices are relatively low after time e , then the store can adjust its strategy over (σ, h) by waiting for the cheapest prices at which to charge after time σ . In such a way, we can argue that a store operator needs forecast prices only up to time h in order to determine the optimal strategy over (τ, σ) .

In a similar way, if instead the interval (τ, σ) is of type \mathcal{F} , then we look for the first time h at which the store would be empty if the multiplier were kept constant and the associated strategy extended up to time h . As above, we find that price forecasts over the interval (τ, e) provide enough information in order to optimally operate the store over (τ, σ) . Thus, in the case of either type \mathcal{E} or type \mathcal{F} , the determination of the optimal strategy over (τ, σ) requires price forecasts up until the first time that the store would have reached the opposite capacity constraint to that attained at time σ , if the multiplier were kept constant from time τ onwards.

Moreover, (τ, h) is the shortest interval of time over which we need to forecast prices in order to optimally operate the store over (τ, σ) . If, for example, (τ, σ) is of type \mathcal{E} but we could only forecast prices up to a time $\underline{h} < h$, then prices beyond time \underline{h} might be exceptionally high but, as the store is not full at time \underline{h} , we may not be able to take full advantage of these high prices. In this situation, it would have been better to increase the multiplier at time τ in order to charge the store in preparation for the subsequent high prices. Similarly, if (τ, σ) is of type \mathcal{F} , then prices beyond time \underline{h} might be very low, so that it would be best to choose a low multiplier which allows us to empty the store in preparation for charging during these times. However, if prices beyond \underline{h} are very high, then it is better to maintain a high-valued multiplier in order to ensure that the store is full prior to these high prices.

We call $h = h(\tau)$ the “look-ahead time” associated with time τ . Price information over (τ, h) suffices in order to determine the optimal strategy at time τ (and indeed over the entire interval (τ, σ)). Moreover, (τ, h) is the shortest interval with this property - price forecasts only up to an earlier time $\underline{h} < h$ will not be sufficient to determine the best strategy for the store at time τ .

4.2. The look-ahead time associated with the algorithm and Proposition 2.1.

For more general versions of the standard storage model, an alternative viewpoint of the time-localization property comes directly from Proposition 2.1 and the algorithm. If the optimal strategy q^* is of the form of Proposition 2.1, then steps 1-3 of the algorithm say that, over an interval (τ, σ) of constant multiplier λ , we have $\lambda = \sup \Lambda^-(\tau, m)$, where m is the optimal level of stored energy at time τ . To identify λ , choose any $\lambda_1 \in \mathbb{R}$ and consider the set of corresponding (λ_1, t) -admissible q_1 (cf. Definition 3.2). If each q_1 breaks the lower capacity constraint of the store and does not hit the top of the store before doing so, as shown in Figure 1, then λ_1 is too small (by Lemma 3.1). On the other hand, if we choose λ_2 such that each corresponding (λ_2, t) -admissible q_2 breaks the upper capacity constraint without hitting the bottom of the store beforehand, then λ_2 is too large. The correct choice is $\lambda \in (\lambda_1, \lambda_2)$, with a (λ, t) -admissible q which either hits the top of the store (at some time after t) before hitting the lower constraint, or vice versa, at the look-ahead time h . Without changing the multiplier λ at some subsequent time, it is possible that the associated q may break a capacity constraint at time h . Now notice that, to recognise that λ_1 and λ_2 are not the correct choice of multiplier, we only need to determine the corresponding q_1 and q_2 up until the exit times where the capacity constraints are first broken (labelled e_1 and e_2 respectively in Figure 1). As we increase λ_1 (or decrease λ_2) to converge to λ , the corresponding time horizons e_1 and e_2 also increase monotonically and are bounded from above by $h = h(\tau)$. Step 4 of the algorithm corresponds to the case where the corresponding look-ahead time is $h = \tau$. In any case, $[\tau, h]$ is the shortest interval of time over which cost information is required in order to decide how to optimally operate the store at time τ .

4.3. The look-ahead time associated with the general standard storage model.

Even when the optimal solution is not of the form of Proposition 2.1, time-localization still holds in the sense that the determination of the optimiser $q^*(t)$ at a time t is independent of the value of the cost function $L(s, \cdot)$ for all times $s > H(t)$, for some identifiable time horizon $H(t)$ (see Proposition 4.1). This time horizon can be determined in the following way: supposing we know the optimal

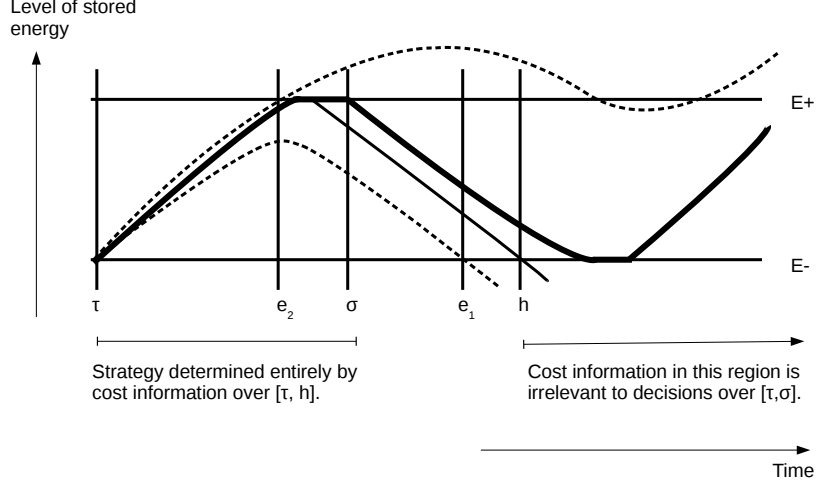


FIGURE 1. The thin solid line is the level of stored energy $\ell[q](t)$, plotted over times $t \geq \tau$, where q is the strategy associated with the constant multiplier λ . The thick solid line represents the level of stored energy associated with the optimal strategy q^* , where q^* agrees with q over (τ, σ) . The look-ahead time associated with time τ is h . The lower and upper dashed lines are respectively $\ell[q_1](t)$ and $\ell[q_2](t)$. These correspond to λ_1 and λ_2 , and only need determining until times e_1 and e_2 respectively in order to discard these as incorrect choices for the multiplier.

strategy q^* , then the time horizon $H(t)$ is the first time at which the store either (i) hits full capacity (at some time after t) then discharges to hit the lower capacity constraint, or (ii) hits empty (at some time after t) then charges to hit the upper capacity constraint. The assumption that we may need to know the entire strategy q^* before we can calculate $H(t)$ is obviously not ideal, but as long as our optimisation algorithm moves forwards in time (as it does in Section 3.2), we will know that we have reached our time horizon $H(t)$ as soon as we have met one of these criteria. In a similar way to the look-ahead time, any cost information after time $H(t)$ is irrelevant to the action of the store not only at time t , but at all times within the interval $[t, \sigma(t))$, where $\sigma(t)$ denotes the final time prior to $H(t)$ at which the store is full in case (i) or empty in case (ii).

Even if we were to choose an algorithm which works backwards in time, this property could still be useful since we may try solving the optimization problem over some shortened interval of time $[0, H'] \subset [0, T]$. Applying our criteria will then reveal whether or not $H(0) \leq H'$. If the inequality holds, then we know that our solution is correct up to time $\sigma(0)$. Relabelling $H(0)$ as the initial time and setting the level of the store at that time to agree with the level given by our previous optimization, we repeat this procedure until the whole interval $[0, T]$ has been covered. The efficiency of this procedure of course relies upon making good guesses in advance for the value of each $H(t)$.

Proposition 4.1 (Time-localization). *Let $E : [0, T] \rightarrow \mathbb{R}$ be an admissible domain and $U \subset \mathbb{R}$ be a set of admissible power outputs. Let C and \bar{C} be cost functionals of the form (4), where the associated cost rate functions L, \bar{L} respectively satisfy the conditions of Section 1.2. Assume that there exist optimal strategies $q^*, \bar{q}^* \in X$ associated with C and \bar{C} respectively and let $t \in [0, T)$ be such that $\ell[q^*](t) = \ell[\bar{q}^*](t)$.*

If there exist times $t_1 = \sigma(t)$ and $t_2 = H(t)$ such that $t < t_1 < t_2 \leq T$ and such that the pair of store energy levels $(\ell[q^](t_1), \ell[q^*](t_2))$ coincides with either $(E^-(t_1), E^+(t_2))$ or $(E^+(t_1), E^-(t_2))$ then, as long as $L(s, \cdot) = \bar{L}(s, \cdot)$ for almost all $s \in [t, t_2]$, there exists an optimal strategy $\bar{q} \in X$, associated with the cost functional \bar{C} , such that $\bar{q}(s) = q^*(s)$ for all $s \in [t, t_1]$.*

Proof. Suppose that there exist times t_1, t_2 with the required properties and that $L(s, \cdot) = \bar{L}(s, \cdot)$ for almost all $s \in [t, t_2]$. Assume first that $\ell[q^*](t_1) = E^-(t_1)$ and suppose that the lemma is not true. By the continuity of the map $s \mapsto \ell[q](s)$ for each admissible $q \in X$, there must exist at least one interval $(a, b) \subset [t, t_2]$ such that $\ell[q^*](a) - \ell[\bar{q}^*](a) = \ell[q^*](b) - \ell[\bar{q}^*](b) = 0$ and $\ell[q^*](s) \neq \ell[\bar{q}^*](s)$ for all $s \in (a, b)$. However, this implies that both q^* and \bar{q}^* must be minimizers of $\int_a^b L(t, q(s)) ds$ over all admissible $q : [a, b] \rightarrow U$ such that $\ell[q](a) = \ell[q^*](a)$ and $\ell[q](b) = \ell[q^*](b)$. In particular, we may adapt \bar{q}^* if necessary so that $\bar{q}^*(s) = q^*(s)$ for all $s \in (a, b)$. Repeating this argument for all such intervals (a, b) , and letting \bar{b} be the supremum of all such end-points b , implies that the optimal strategy associated with \bar{C} can be chosen to coincide with q^* over (t, \bar{b}) . The condition that $\ell[q^*](t_1) = E^+(t_1)$, together with the continuity of each map $s \mapsto L[q](s)$, further imply that $\bar{b} \geq t_1$, hence completing the proof for the case $\ell[q^*](t_1) = E^-(t_1)$. A similar argument holds for the case where $\ell[q^*](t_1) = E^+(t_1)$. \square

4.4. A comparison of update times, exit times, look-ahead times and time horizons. Consider again the algorithm of Section 3.2 and suppose that we have identified an interval (τ, σ) of constant multiplier λ . Then σ can be chosen to be the first time after τ at which the store is either full or empty (or the level of the store is ℓ_T if $\sigma = T$). As defined above, the look-ahead time $h = h(\tau)$ associated with τ is such that (τ, h) is the shortest time-interval over which cost information is required in order to operate the store optimally at time τ (and, in fact, over the entire interval (τ, σ)). The look-ahead time has the property that the level of stored energy at time h lies on the opposite capacity constraint to that at the update time σ .

The time horizon $H = H(\tau)$ associated with τ serves a similar purpose to the look-ahead time, in the sense that cost information over (τ, H) is sufficient to determine the optimal strategy over (τ, σ) . However, unlike with the look-ahead time, there is no guarantee that (τ, H) is the shortest such interval. In fact, $h = \min(H, e)$, where $e = e(\tau)$ is the exit time associated with τ (cf. Definition 3.2). This is the first time after τ at which a capacity constraint would be broken if the multiplier were not updated before this time. As with the look-ahead time h , both H and e share the characteristic that the level of stored energy at times H and e lies on the opposite capacity constraint to that at the update time σ .

Finally, consider extending the interval (τ, σ) , if possible, to the maximum interval $(\tau, \bar{\sigma})$ over which the multiplier is constant. We call $\bar{\sigma} = \bar{\sigma}(\tau)$ the “update time” associated with τ , since at this time the multiplier must be updated to a new value. The level of stored energy at time $\bar{\sigma}$ lies on the same capacity constraint as at time σ . In order to evaluate the optimal strategy over $(\tau, \bar{\sigma})$, cost information is required

up until the associated exit time e . Thus, whilst the look-ahead time indicates the minimum amount of cost information required in order to operate the store at any time $t \in [\tau, \sigma]$, the strategy actually remains completely determined by the multiplier λ over the (potentially) longer period of time $[\tau, \bar{\sigma}]$. On the one hand, the look-ahead times provide a useful method for the real-time operation of the store when uncertainty is introduced into the cost functions, as discussed below; on the other hand, a deterministic optimisation benefits from the fact that the operation of the store can be characterized by a simple rule, determined by λ , up until the update time $\bar{\sigma}$.

5. Examples.

5.1. Real-time stochastic optimization. Suppose that the cost rate function $L(t, \cdot)$ of Section 1.2 is not perfectly known in advance for all times t , but rather that it is being frequently updated - at times $0 < s_1 < s_2 < \dots < T$, say - as new information becomes available. Then, at time t , having decided on the action of the store over $[0, t)$ and thus having determined the corresponding level of stored energy ℓ_t at time t , we wish to choose $q_r^* : [t, T] \rightarrow U$ in order to minimise the expected cost

$$\mathbb{E} \left[\int_t^T L(u, q(u)) \, du \mid \ell[q](t) = \ell_t \right] \quad (26)$$

over all piecewise continuous $q : [t, T] \rightarrow U$ that satisfy the capacity constraints of the store.

The look-ahead times $h(t)$ associated with each time t suggest that a rolling time-horizon approach should be an appropriate way of dealing with uncertainty of this kind. Beginning at time $t = 0$, as long as cost predictions are good up until the look-ahead time $h(0)$, then the algorithm of Section 3.2 should provide a good estimate of the best course of action $q_r : [0, s_1) \rightarrow U$ for the store up until the first information update time s_1 . One can then restart the algorithm at time $t = s_1$, with the level of stored energy at this new initial time given by $\ell[q_r](s_1)$. Again, as long as cost predictions are good from time s_1 up until the look-ahead time $h(s_1)$, we can apply the algorithm to extend the strategy q_r into $[0, s_2]$. Continuing in this way, we can decide on an action for the store at each time t , with the action depending only on cost information that is available from time t until the look-ahead time $h(t)$. This approach has a number of benefits:

- (i) Cost information can be frequently updated. This is in contrast to some other approaches in which the cost functions are assumed to follow a fixed probability distribution over all future times.
- (ii) Computations need to be carried out only over times $[t, h(t)]$ in order to decide on the best action for the store at time t . This method should, therefore, compare favourably with other rolling time approaches, as well as with methods based on dynamic programming, in terms of the number of computations needed to operate the store in real-time.
- (iii) This method contrasts favourably with other rolling-time approaches which explicitly introduce look-ahead times, perhaps based on an assumption of underlying periodicity in the cost rate function L . With the algorithm of Section 3.2, we can avoid making long-term predictions and save on computations in a similar way, but we need not make an a priori periodicity assumption, since the time horizons $h(t)$ are implicit in the analysis.

5.2. Maximizing arbitrage profits. We consider here a compressed air energy store (CAES) which operates purely within a single wholesale market. Currently, there are only two commercially operating stores of this kind (one in Alabama USA, and the other in Germany), both of which require an input of gas at the discharge stage. Research into heat storage technology is striving to reduce or even remove this reliance on gas, by replacing the energy provided by igniting gas at the discharge stage with heat energy which is restored from charging the store. The case where the heat of compression is reused and no gas is burnt is called “adiabatic”.

Our model of a compressed air store has charge and discharge power rates q^+, q^- respectively and a fixed energy capacity M , so that the set of admissible powers is $U = [-q^-, q^+]$ and the admissible domain satisfies $E^-(t) = 0$ and $E^+(t) = M$ for all $t \in (0, T)$. We further assume that the store is constrained to start and finish empty, so that $E^-(0) = E^+(0) = E^-(T) = E^+(T) = 0$. Typically, leakage from stores of this type is considered to be very low, and we therefore set $\alpha = 0$. The cost functional C is of the “standard storage model” form introduced in Section 1.1, with a cost rate function L (cf. (4)) defined by

$$L(t, x) := \begin{cases} p(t)x & \text{if } x \geq 0 \\ \frac{1}{\beta} (p(t) - \gamma g(t)) x & \text{if } x < 0 \end{cases} \quad \forall x \in U, t \in [0, T], \quad (27)$$

where $p(t), g(t) \in \mathbb{R}$ is the price per unit energy at time t and β, γ is the amount of electrical and gas power respectively required to generate and sell 1 unit of electrical power. Setting $\gamma = 0$ and $\beta > 1$ gives us an adiabatic compressed air store with round-trip efficiency $1/\beta$. This model, with $\gamma = 0$ and $\beta > 1$, would also be a suitable approximation for other storage types, including pumped hydro (assuming that the effects of rainfall and seepage on stored energy in the upper reservoir is negligible), thermal and liquid air energy storage. An additional multiplicative factor can easily be incorporated into the charge side of the cost function if we wish to model charge and discharge efficiencies separately, rather than a single round-trip efficiency. We are assuming here that the store is a price-taker so that its actions do not impact on the price of electricity. However, market impact could be incorporated by replacing each $p(t)$ with a suitable price of the form $\tilde{p}(t, q(t))$ (see [10] and [12] for a further discussion of this). If the price of electricity is always low enough that $p(t)(1 - \beta) < \gamma\beta g(t)$ for all $t \in [0, T]$, then the cost functional (27) is convex and an optimal strategy exists. By Propositions 3.5 and 3.6, an optimal strategy can therefore be determined via the algorithm of Section 3.2.

For a price-taking arbitrage situation, as we have here, the multiplier μ^* can be interpreted as a reference cost: the strategy of the store is to charge at full power if the cost of electricity is sufficiently low that $p(t) < \mu^*(t)$; to discharge at full power if the cost of electricity (or gas) is sufficiently high (or low) that $(1/\beta)(p(t) - \gamma g(t)) > \mu^*(t)$; and to do nothing if prices are not extreme enough to satisfy either of these conditions. The reference cost should not change whilst the store is neither full nor empty; intuitively, there is no incentive to do so. There are analogies here with the literature on investment under uncertainty [18]. When the store is full, however, the reference cost is permitted to increase. At these times, the store operator is waiting until the price of electricity becomes high enough (or the price of gas low enough, but in our illustration we have taken the price of gas to be constant), that it is worth selling power. If the reference cost were instead decreasing at these times, then this would indicate that the store should not have

	Gross Profit (£m)	Look-ahead timelengths (h)		
		10th perc.	Av.	90th perc.
Huntorf	2.39	17.0	87.9	211.0
Adiabatic 75%	2.74	11.0	31.9	61.0
Adiabatic 85%	4.14	8.0	21.3	35.0

TABLE 1. The annual 2014 profits of each store (where Huntorf has parameters $\beta = 0.8$ and $\gamma = 1.6$), together with the 10th percentile, mean and 90th percentile of all look-ahead timelengths required in order to determine the optimal strategy at each time of the year.

charged up so soon, but rather it should have waited for these more favourable buying conditions. Similar arguments hold when the store is empty.

Figure 2 shows the optimal strategies for a compressed air store with parameters modelled on those of the Huntorf store in Germany: $M = 580\text{MWh}$, $q^- = 290\text{MW}$, $q^+ = 72.5\text{MW}$. The left-hand plot takes parameters $\beta = 0.8$, $\gamma = 1.6$, which coincides with data available for Huntorf station [14]. The right-hand plot represents two adiabatic stores with $\beta = \gamma = 0$ and with different efficiencies. We have chosen here round-trip efficiencies $1/\beta$ of 0.75 and 0.85 which, according to [15] are the lower and upper efficiencies to be expected from the recent adiabatic research CAES plant, ADELE. The actions of the store are optimized over the course of one year. Electricity prices are APX half-hourly 2014 mid-price index prices (the data are available from [16]). These are a weighted average of all traded prices at any given time and represent an approximation to the spot price at that time. We have used a constant gas price throughout the year, which is set to the average 2014 gas price of £17.56/MWh, as reported by DECC (the UK Department of Energy and Climate Change).

An immediate observation is that each store follows a similar pattern of optimal storage levels - charging at times of low prices and discharging at times of high prices. As the round-trip efficiency of the adiabatic store increases, it operates over longer periods as the lower energy losses mean that the store is able to take greater advantage of price differences. The gas-fired CAES plant follows a strategy which is more closely related to the adiabatic store with a 0.75 round trip efficiency and this is further reflected in the similarity of the profits of these two stores (see Table 1). The CAES plant is slightly less active than either adiabatic plant here since the cost of discharging power is usually higher for the former than the latter. More precisely, the cost $L(t, x)$ associated with discharging x MW at time t is higher for an adiabatic store with round-trip efficiency ε than our CAES plant if and only if $2g(t) < p(t)(1/0.8 - \varepsilon)$. With our chosen parameters of $g(t) = 17.56$ for all t and $\varepsilon = 0.75$, this condition becomes $p(t) > 70.24$.

We highlight here that, although the price of electricity follows a reasonably periodic pattern (with periods of roughly a day in length), it does not follow that the level of the store is the same at the start and end of each day (see, for example, [13] for a periodic setting in which the optimal solutions are not necessarily periodic). This underlines point (iii) made in Section 5.1, that if we are interested in the real-time operation of the store, then we are likely to obtain a better strategy by implementing the implicit look-ahead times of the algorithm of Section 3.2, rather than introducing explicit look-ahead times with extra constraints imposed at those times at each stage of the optimization.

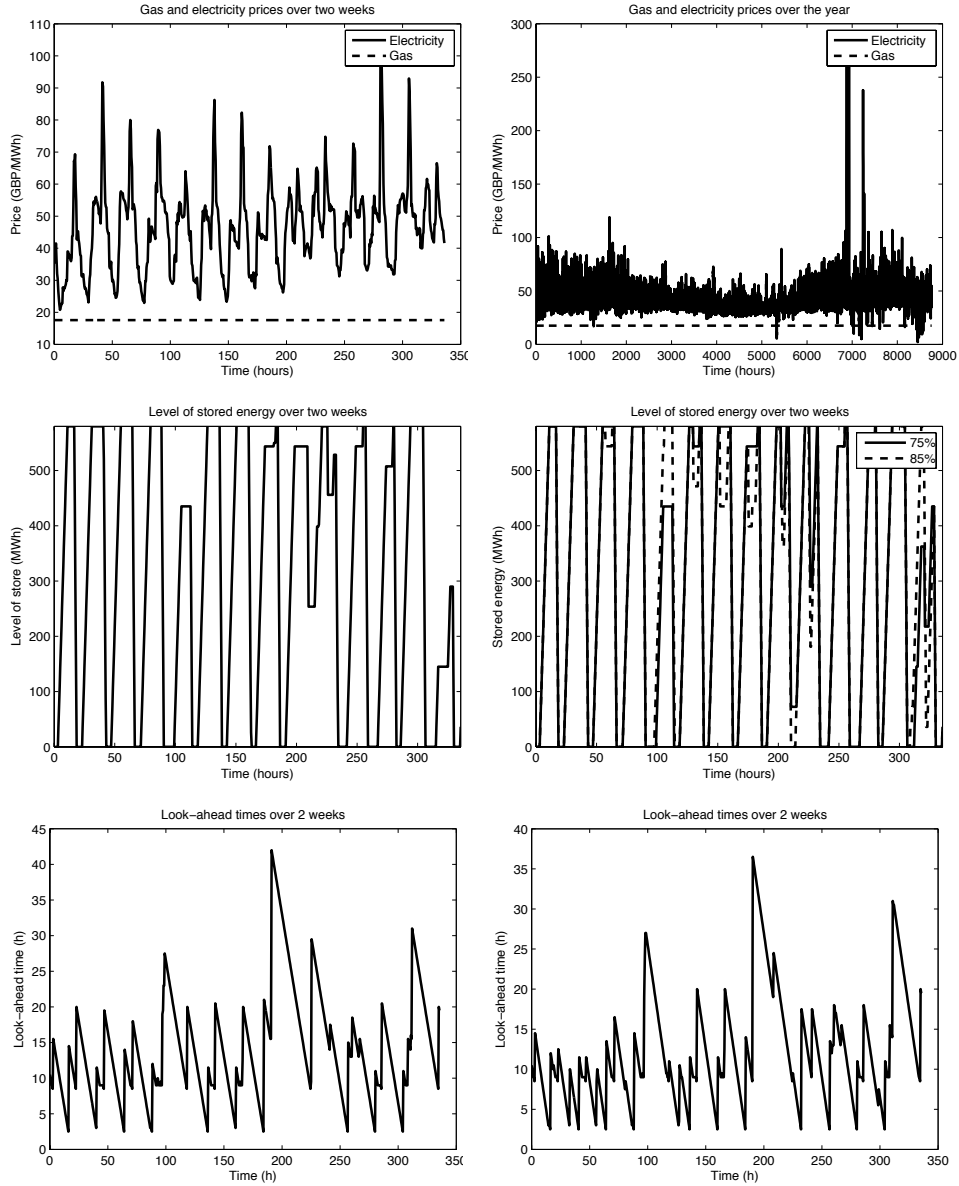


FIGURE 2. Top row: electricity mid-index prices and the average gas price over the first two weeks of 2014 (left) and over the entire year (right). Middle row: Optimal stored energy levels over the first two weeks of 2014 for a gas-fired CAES plant (left) and adiabatic stores with round-trip efficiencies 0.75 and 0.85 (right). Bottom row: the length of look-ahead time required to decide on the optimal strategy at each time over the first two weeks of 2014 for a gas-fired CAES plant (left) and an adiabatic store with round-trip efficiency 0.75 (right).

The length of look-ahead times required in order to determine the best strategy at each time is generally longer for the gas-fired CAES plant than for either adiabatic store. In fact, Table 1 indicates that, on average, the operator of a CAES plant needs to predict electricity prices over periods of 87.9 hours, as opposed to 31.9 hours or 21.3 hours for a 0.75 or 0.85 efficient adiabatic store respectively. The lower look-ahead times for the higher efficiency adiabatic store can be explained in a similar way to the difference in optimal strategies - a higher efficiency plant does not need to look as far ahead in time as a lower efficiency plant since it is able to take advantage of smaller price differences, thus resulting in quicker cycling times. Similarly, as observed above, the gas-fired CAES plant is usually less able than a 0.75 efficient adiabatic store to take advantage of smaller price differences, and therefore needs to look further ahead in order to identify price differences which are sufficiently great to encourage the store to cycle.

5.3. Storage for smoothing. Consider the same store that was introduced in Section 5.2 but suppose now that the store operator wishes to “smooth” a supply function $g : [0, T] \rightarrow \mathbb{R}$. The supply here might be the power output of a wind farm, or perhaps the difference between supply and demand at some location. Such smoothing could be useful, for example, in controlling the variability of renewable outputs or demand, thus potentially reducing balancing costs and the need for peaking plants. For this example, we will assume no rate constraints, so that the set of admissible powers for the store is $U = \mathbb{R}$, and to simplify exposition we will initially restrict attention to the case of a perfectly efficient store in which there are no energy losses during operation.

In the perfectly efficient case, an optimal smoothing strategy $q^* \in X$ is now defined as a minimizer of the total variation $\text{TV}(y)$ over all functions $y : [0, T] \rightarrow \mathbb{R}$ of the form $y(t) = q(t) - g(t)$, for all $t \in [0, T]$ and some admissible strategy $q \in X$, where

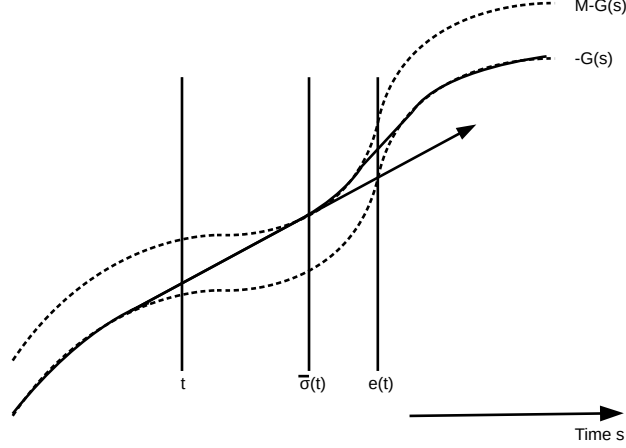
$$\text{TV}(y) := \min_{P \in \mathcal{P}} \sum_{i=1}^{n(P)} \int_{t_i}^{t_{i+1}} |y(t_{i+1}) - y(t_i)| dt, \quad (28)$$

and where \mathcal{P} is the set of all partitions P of the form $0 = t_1 < \dots < t_n < t_{n+1} = T$, with $n = n(P)$. If y is differentiable (or absolutely continuous, thus differentiable in a distributional sense and satisfying the fundamental theorem of calculus), then (28) becomes

$$\text{TV}(y) := \int_0^T |\dot{y}(t)| dt. \quad (29)$$

Alternatively, let $G(t) = \int_0^t g(s) ds$ be the accumulated supply output up until time $t \in [0, T]$ and consider the diagram in Figure 3. Let \mathcal{T} denote the tunnel bounded by $-G(t)$ and $M - G(t)$. The curve through \mathcal{T} represents the trajectory $Y^*(t) := \ell[q^*](t) - G(t)$ associated with the optimal smoothing strategy q^* and is the same as the path taken by a string that is stretched taut through the tunnel, with fixed start and end positions $\ell_0 - G(0)$ and $\ell_T - G(T)$ respectively (with $\ell_0 = \ell_T = 0$ in the diagram). In light of the taut string analogy, it is not difficult to see that q^* is completely characterized by the following conditions:

- (i) The net output y^* is constant over intervals of time when the store is neither full nor empty.
- (ii) y^* can increase only at times when the store is full.

FIGURE 3. The exit time $e(t)$ and update time $\bar{\sigma}(t)$ associated with y^* .

(iii) y^* can decrease only at times when the store is empty.

Relabelling $-G(t)$ as $E^-(t)$ and $M - G(t)$ as $E^+(t)$, we see that the conditions (i)–(iii) on y^* coincide exactly with the conditions (6)–(11) on μ^* . Hence, the algorithm of Section 3.2 can be used to determine y^* and consequently the optimal smoothing strategy q^* . In particular, even though the objective function (28) is clearly not of the same form as the standard storage model of Section 1.2, the same time-localization properties continue to hold here. In fact, the update time $\bar{\sigma}(t)$ and exit time $h(t)$ associated with a time t can now be illustrated clearly as in Figure 3. The exit time $e(t)$ corresponds to the latest point on the tunnel that can be seen by projecting a straight line from (t, ℓ_t) , where $\ell_t = \ell[q^*](t)$ is the level of stored energy at time t , so that

$$e(t) = \max_{x \in \mathbb{R}} \left\{ s_0 \geq t : \ell_t + \int_t^{s_0} (x + g(u)) du \in [0, M] \quad \forall s \in [t, s_0] \right\}. \quad (30)$$

The update time $\bar{\sigma}(t)$ is the latest time prior to $e(t)$ at which the line coincides tangentially with the boundary of the tunnel which is opposite to the boundary through which the line exits at $e(t)$. If the supply function g is known over the interval $[t, e(t)]$, then this straight line corresponds to x^* over $[t, \sigma(t)]$, and thus uniquely determines the optimal strategy q^* over that interval.

Consider now a more general case in which the store has a round-trip efficiency $\epsilon \in (0, 1]$, in the sense that for each unit of energy discharged from the store, only ϵ units are dispatched to the grid, due to operational losses. (For simplicity of exposition, we assume no losses during the charging process, although our arguments can be adjusted for more general cases.) As a direct analogue of the above, an optimal smoothing strategy is now one which minimizes (28) over all functions $y : [0, T] \rightarrow \mathbb{R}$ of the form $y(t) = \epsilon q(t) - g(t)$ if $q(t) < 0$ and $y(t) = q(t) - g(t)$ if $q(t) \geq 0$, for all $t \in [0, T]$ and some admissible strategy $q \in X$. Although it is difficult to extend Figure 3 to this more general case, conditions (i)–(iii) still hold. We omit a formal proof here for brevity of exposition, although some intuition can be gained from the following discussion: If condition (i) does not hold, so that y^* is varying at times when the store is neither full nor empty, then the local lack of

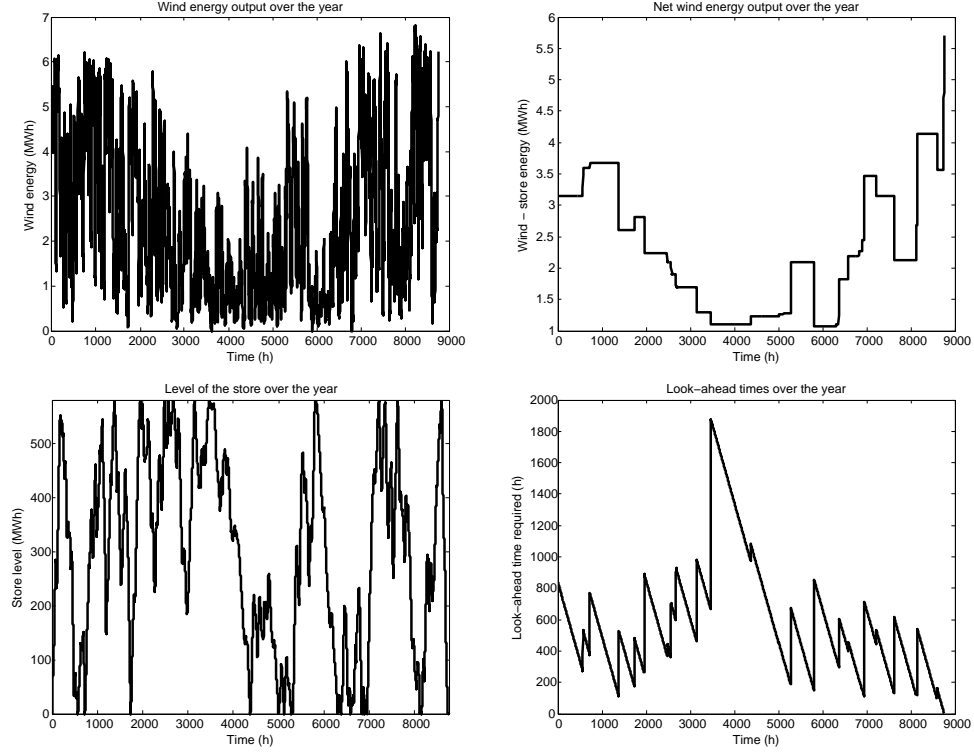


FIGURE 4. Top row: 2014 wind energy half-hourly output (left) and the half-hourly net wind energy output when using a 580MWh, 0.75 efficient store for smoothing the wind energy (right). Bottom row: Optimal stored energy levels when using this store for smoothing (left) and the length of look-ahead time required to decide on the optimal strategy at each time over the year (right).

capacity constraints means that there is freedom for the storage operator to slightly adjust the store's power output in order to improve smoothing over these times. If condition (ii) does not hold, so that the store is full over some period of time but y^* is decreasing over that time (so that q^* is constant but g is increasing), then the storage operator can improve smoothing by discharging a small amount of energy at the start of this time interval and then refilling back to full by the end of the time interval. (Similarly for case (iii).)

To illustrate, consider an adiabatic store with a capacity of 580MWh and round-trip efficiency of $\epsilon = 0.75$, with no rate constraints. Suppose that the store is designed to smooth the wind output of a wind farm whose capacity is approximately 1/500 of the total installed wind capacity for the UK. This corresponds to an installed capacity of approximately 22MW. Taking 2014 half-hourly wind output data for our supply function g (available from Elexon [16]) and using our algorithm to minimize the total variation (28), we obtain a total variation of 14.0. This compares with a total variation of 1515.9 associated with the wind farm output alone, with no storage attached for smoothing. To achieve this level of smoothing, the store requires a maximum charging rate of 7.6MW and a maximum discharging rate of 6.8MW.

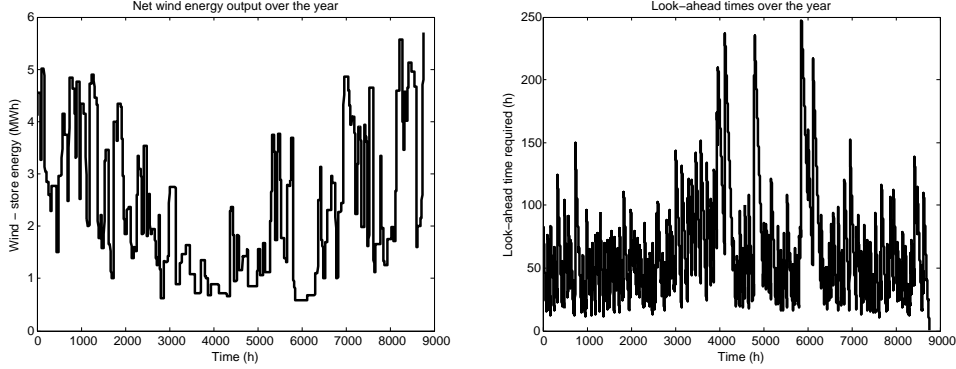


FIGURE 5. Half-hourly net wind energy output when using a 60MWh, 0.75 efficient store for smoothing 2014 wind energy (left) and the length of look-ahead time required to decide on the optimal strategy at each time over the year (right).

The strategy here is very different from the revenue-maximizing strategy of Section 5.2. Firstly, the store no longer operates only at maximum powers but rather continuously adjusts its power output in line with wind generation. Moreover, whilst the prices in Figure 2 loosely follow a daily cycle, here the wind output displays no such periodicity. As such, the optimal smoothing strategy loses the approximate diurnal cycling that was displayed in our arbitrage model of Section 5.2. The average look-ahead timelength required over the course of the year is 1177 hours, with 10th and 90th percentile look-ahead timelengths of 436 hours and 2084 hours respectively. These durations are much higher than those indicated in Table 1. This can be attributed to the large capacity of the store. As seen in Figure 4, the level of stored energy rarely reaches full capacity - a similar level of smoothing could most likely be attained with a smaller capacity store. Thus, for large intervals of time, the store maintains the net wind-farm power output at a constant level, and there is no incentive to change this level whilst the store is neither full nor empty. This in turn means that the storage operator needs to look a long way into the future in order to determine the next constant level of net wind output it needs to aim for. Reducing the capacity of the store would in turn reduce the look-ahead timelengths. Taking instead a 60MWh store, for example, we obtain an average look-ahead timelength of 130 hours (approximately 5 and a half days). This order of timelength is much better suited to the times over which we can currently make reasonable wind forecasts and so, for practical purposes of real-time operation, a smaller store could well be better here. The maximum charge and discharge rates required in this case are 6.1MW and 4.7MW respectively. The disadvantage of using a smaller capacity store, however, is a lower level of wind-smoothing, as seen in Figure 5.

6. Conclusions. We have presented a method to determine how to operate an energy store in order to minimize a given cost functional. This method could apply, for example, to revenue maximization or wind-power smoothing problems. Our setting allows for leakage, inefficiencies, time-varying power constraints and general operating costs which are functions of the power output.

A significant benefit associated with this method is the implicit localization in time of the solution, meaning that the operator of the store typically needs to have only limited foresight of future cost information in order to decide on the optimal real-time operation of the store. We have indicated that this property should render our methods useful for stochastic optimization problems.

Another property of our method comes from the Karush-Kuhn-Tucker multipliers associated with the optimization problem. We have shown that the solution can be characterized entirely by the associated multiplier which, over intervals of time when the store is filling or emptying, is constant. In terms of an arbitrage revenue maximization problem, this multiplier may be interpreted as a reference price which indicates whether the store should be charging or discharging energy at each time.

Software implementing the method will be available via <http://estoolbox.org>

An intended future publication [12] will report on extension to cases where the store is large enough to influence prices or with multiple competing stores.

Acknowledgments. This work was made possible by the IMAGES (Integrated Market-fit Affordable Grid-Scale Energy Storage) research group. The authors would like to thank the other members of the group for their suggestions and insight, with particular thanks to Monica Giuliatti. The authors would also like to express their gratitude to Stan Zachary, James Cruise and Richard Gibbens for their many helpful discussions related to this topic. The work was supported by EPSRC (grant EP/K002228/1) and the Alfred P. Sloan Foundation New York.

REFERENCES

- [1] D. J. C MacKay, [Sustainable Energy - without the hot air](#), *Am. J. Phys.*, **78** (2010), p222.
- [2] *Energy Storage and Management Study*, AEA, 2010.
- [3] *Options for low-carbon power sector flexibility to 2050 - a report to the Committee on Climate Change*, Pöry, 2010.
- [4] M. Black and G. Strbac, [Value of bulk energy storage for managing wind power fluctuations](#), *IEEE Transactions on Energy Conversion*, **22** (2007), 197–205.
- [5] M. Aunedi, N. Brandon, D. Jackravut, D. Predrag, D. Pujianto, R. Sansom, G. Strbac, A. Sturt, F. Teng and V. Yufit, *Strategic Assessment of the Role and Value of Energy Storage Systems in the UK Low Carbon Energy Future - Report for Carbon Trust*, 2012.
- [6] J. P. Barton and D. G. Infield, [Energy storage and its use with intermittent renewable energy](#), *IEEE Transactions on Energy Conversion*, **19** (2004), 441–448.
- [7] P. Grünwald, T. Cockerill, M. Contestabile and P. Pearson, The role of large scale storage in a GB low carbon energy future: Issues and policy challenges, *Energy Policy*, **39** (2011), 4807–4815.
- [8] *Study of Compressed Air Energy Storage with Grid and Photovoltaic Energy Generation, Draft Final Report - for Arizona Public Service Company*, Arizona Research Institute for Solar Energy (AZTISE), 2010.
- [9] N. Löhdorf and S. Minner, Optimal day-ahead trading and storage of renewable energies - an approximate dynamic programming approach, *Energy Syst.*, **1** (2010), 61–77.
- [10] J. Cruise, L. C. Flatley, R. Gibbens and S. Zachary, *Optimal Control of Storage Incorporating Market Impact and with Energy Applications*, 2015, arXiv:1406.3653v1
- [11] A. J. Pimm and S. D. Garvey, [The economics of hybrid energy storage plant](#), *International Journal of Environmental Studies*, (2014), 787–795.
- [12] J. Cruise, L. C. Flatley and S. Zachary, *Impact of Storage on Energy Markets*, working paper, 2016).
- [13] R. S. MacKay, S. Slijepčević and J. Stark, [Optimal scheduling in a periodic environment](#), *Nonlinearity*, **13** (2000), 257–297.
- [14] *Compressed Air Energy Storage Power Plants*, BINE Informationsdienst projektinfo 05/07.
- [15] *ADELE - Adiabatic Compressed-Air Energy Storage for Electricity Supply*, RWE Power.
- [16] <https://www.elexonportal.co.uk/>

- [17] M. Giuliatti, L. Grossi and M. Waterson, [Price transmission in the UK electricity market: Was NETA beneficial?](#), *Energy Economics*, **32** (2010), 1165–1174.
- [18] A. K. Dixit and R. S. Pindyck, *Investment Under Uncertainty*, Princeton UP, 1994.

Received March 2016; revised September 2016.

E-mail address: L.Flatley@warwick.ac.uk

E-mail address: R.S.MacKay@warwick.ac.uk

E-mail address: Michael.Watson@warwick.ac.uk