TU Delft

Delft University of Technology

A 1-dimensional continuous and smooth model for thermally stratified storage tanks including mixing and buoyancy

Lago, Jesus; De Ridder, Fjo; Mazairac, Wiet; De Schutter, Bart

DOI
10.1016/j.apenergy.2019.04.139

Publication date
2019

Document Version
Final published version

Published in
Applied Energy

Citation (APA)
Lago, J., De Ridder, F., Mazairac, W., & De Schutter, B. (2019). A 1-dimensional continuous and smooth model for thermally stratified storage tanks including mixing and buoyancy. *Applied Energy*, *248*, 640-655. https://doi.org/10.1016/j.apenergy.2019.04.139

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

Applied Energy 248 (2019) 640–655

Contents lists available at ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

A 1-dimensional continuous and smooth model for thermally stratified storage tanks including mixing and buoyancy

Jesus Lago[a,b,*], Fjo De Ridder[b], Wiet Mazairac[c], Bart De Schutter[a]

[a] Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, Delft, The Netherlands
[b] Algorithms, Modeling, and Optimization, VITO, Energyville, ThorPark, Genk, Belgium
[c] ECOVAT, Poort van Veghel 4946, Veghel, The Netherlands

HIGHLIGHTS

- We propose a smooth and continuous 1D model for stratified thermal storage vessels.
- The model includes buoyancy and mixing effects through a smooth formulation.
- The smoothness property is critical to integrate the model in optimization problems.
- The new model solves some computational issues of existing models from literature.
- Buoyancy is modeled by distinguishing between slow and fast buoyancy effects.

ARTICLE INFO

Keywords:
Thermal Storage
Stratified Tank
Modeling
Parameter Estimation
Optimal Control
Numerical Optimization

ABSTRACT

To mitigate the effects of the intermittent generation of renewable energy sources, reliable and efficient energy storage is critical. Since nearly 80% of households energy consumption is destined to water and space heating, thermal energy storage is particularly important. In this context, we propose and validate a new model for one of the most efficient heat storage systems: stratified thermal storage tanks. The novelty of the model is twofold: first, unlike the non-smooth models from the literature, it identifies the mixing and buoyancy dynamics using a smooth and continuous function. This smoothness property is critical to efficiently integrate thermal storage vessels in optimization and control problems. Second, unlike models from literature, it considers two types of buoyancy: slow, linked to naturally occurring buoyancy, and fast, associated with charging/discharging effects. As we show, this distinction is paramount to identify accurate models. To show the relevance of the model, we consider a real tank that can satisfy heat demands up to 100 kW. Using real data from this vessel, we validate the proposed model and show that the estimated parameters correctly identify the physical properties of the vessel. Then, we employ the model in a control problem where the vessel is operated to minimize the cost of providing a given heat demand and we compare the model performance against that of a non-smooth model from literature. We show that: (1) the smooth model obtains the best optimal solutions; (2) its computation costs are 100 times cheaper; (3) it is the best alternative for use in real-time model- based control strategies, e.g. model predictive control.

1. Introduction

In the last decade, as the integration of renewable energy sources into the electrical grid has steadily increased, energy storage has emerged as one of the key components in this change. In particular, due to the increasing uncertainty associated with renewable source generation, imbalances between production and consumption of electricity have become more common. As the amount of variable renewable electricity is expected to increase in future electrical systems [1], these problems will become worse. In this context, energy storage is paramount to tackle these imbalances as it shifts consumption and generation and keeps the grid stable. In addition to grid stability, energy storage has also become extremely important for profit maximization. More specifically, under positive imbalances, i.e. generation larger than consumption, electricity prices are usually lower (and vice versa). In this situation, utility companies would ideally like to buy and store energy under positive imbalances and use/sell the stored energy under negative imbalances. To optimally carry out this strategy, utilities need

* Corresponding author.
E-mail address: j.lagogarcia@tudelft.nl (J. Lago).

https://doi.org/10.1016/j.apenergy.2019.04.139
Received 2 October 2018; Received in revised form 3 April 2019; Accepted 17 April 2019
0306-2619/ © 2019 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/BY/4.0/).

cost-effective storage systems that are also fast and efficient.

While the perfect energy storage system does not exist, *thermal energy storage (TES)* systems partially fulfill the increasing demand for cost-effective and efficient storage. While using electricity to generate heat was not efficient in fossil fuel-based power systems, the flexibility of using electricity for heating purposes combined with TES has recently received increasing attention [2,3]. Particularly, considering that 26.3% of the electricity consumption in EU households is destined to water and space heating, and that water and space heating accounts for 79.2% of the total energy consumption in the same households [4] and one-third of related greenhouse gas emissions [5], TES systems might help fulfill some of the energy storage requirements. The new class of systems that exploit the interaction between different energy carriers are usually called multi-energy systems, and as they try integrate diverse energy systems to achieve a higher energy utilization efficiency [6], they have become in a central point of research [7,8]. Another application where TES systems are a key component is concentrated solar power plants; there, they help to smooth the production, to maximize earnings from the electricity market fluctuation, and to increase the lifespan of the power block [9].

1.1. Stratified fluid tank

One of the most important TES systems are stratified fluid tanks [10], which store energy by keeping fluid layers stratified at different temperatures. In detail, exploiting the fact that fluid density decreases as temperature increases, they are able to stratify fluid layers where the warmest layers are displaced to the top of the tank and the coldest layers to the bottom. This type of heat storage systems are widely used, and multiple applications can be found in the literature [3,5,11–13].

One of their main advantages is that, in comparison with a regular mixed fluid tank, a stratified tank improves the average net energy and exergy efficiencies by up to 60% [10]. To maximize the energy efficiency, the system is built such that the mixing between the stratified layers is minimized.

Within the family of stratified fluid tanks, there are several possible configurations depending on how the tank is charged, i.e. how heat is introduced in the tank, and on how the tank is discharged. In terms of charging, the fluid in the tank can be heated directly using a fluid flow or indirectly using a heat exchanger [10]. Likewise, when discharging the tank, heat can be directly extracted as a fluid flow or indirectly extracted using a heat exchanger. The four possible combinations are depicted in Fig. 1.

The main advantage of indirect charge/discharge is that, as no flow is introduced in the tank, stratification is more easily maintained. However, since heat is indirectly transferred, the energy efficiency is lowered. In contrast, while direct heating introduces turbulences in the tank that might destroy the thermal stratification, it has a larger efficiency as heat is directly transferred [10].

1.2. Modeling of storage tanks

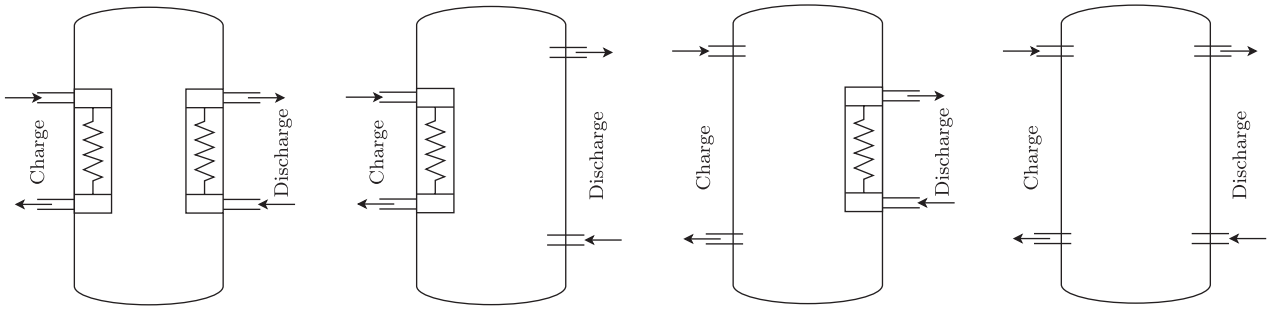
The scientific literature regarding modeling of stratified thermal storage vessels is very large and diverse. Typically, the proposed models can be divided into three categories: 1-dimensional models [14–18], 2-dimensional models [19,20], and 3-dimensional computational fluid dynamics models [21,22]. Although 1D models are less accurate, they are the preferred choice in several applications: while 2D and 3D models are more accurate, their computational complexity makes them unsuitable for process optimization or long-term simulation of the storage tank [10,19]. By contrast, to analyze the behavior of the fluid within the tank or the effect of new configurations, 2D and 3D models give detailed information that a 1D model cannot provide.

One of the most important reasons behind the reduced accuracy of 1D models is the approximation they make to model the mixing of layers due to buoyancy effects [18,19]. In more detail, given two consecutive layers in a real tank, the lower layer might achieve a temperature higher than the top layer; in this scenario, the bottom layer would rise, the top layer would sink, and during this process both layers would mix. In a real tank, there are several scenarios when this effect might occur:

1. Due to its larger contact area with the environment, the top layer in the tank would normally suffer larger heat losses. Therefore, as the top layer loses more heat, there is a point when the layer below reaches a higher temperature and both layers mix.
2. When the tank is directly charged/discharged, new fluid enters the tank. If the temperature of the incoming fluid is higher than the temperature of any layer above the entrance point, the incoming fluid will rise and mix.
3. When the tank is indirectly charged/discharged, a lower layer might be heated more than a top layer. In this scenario, the fluid in the lower layer will rise and mix.

When considering 1D models for thermal stratified storage vessels from the literature, none of them can physically model this effect. More specifically, while 1D models consider heat transfers between fluid layers and input/output flows, they do not model the effect of gravity in the tank. To address this, the 1D models proposed in the literature usually include a post-processing step after each simulation step that approximates the mixing of layers due to buoyancy effects [16,19]. This post-processing algorithm has the following structure:

1. Check the temperature of each layer and evaluate whether buoyancy effects are present.
2. If buoyancy effects are present, mix the corresponding layers.
3. Repeat steps 1–2 until buoyancy effects are removed.



(a) Indirect charge - Indirect discharge. (b) Indirect charge - Direct discharge. (c) Direct charge - Indirect discharge. (d) Direct charge - Direct discharge.

Fig. 1. Simplified configurations of stratified tanks when considering the two types of exchanging heat (direct and indirect) and the two directions of the heat (charge and discharge).

1.3. Motivation

The post-processing method described before has a large disadvantage: the dynamics of the tank are not defined by a single continuous equation, but by a first continuous equation that models the heat transfers and the input/output flows and a second non-smooth algorithm that models the buoyancy effects. Due to this structure, the 1D dynamical models proposed in the literature cannot be used with derivative-based optimization algorithms that make use of analytical expressions; instead, they either require the use of heuristic optimization methods or of finite differences if derivative-based optimization methods are to be used. This is especially critical for control applications: if the optimal set of controls to steer the tank have to be computed, the optimization problem that computes these controls needs to be based on a heuristic method, e.g. genetic algorithms [23,24] or particle swarm optimization [25], or on a derivative-based optimization algorithm that relies on finite differences. As both heuristic methods and derivative-based optimization via finite differences require large computation times, the controller time step might not be enough for these methods to find a solution. In addition, as heuristic methods cannot guarantee that the obtained solution is a local minimum, the quality of the obtained solution might be worse. This last issue could be solved using dynamic programming [26], which could compute the global solution using the two-step non-continuous dynamics; however, due to the curse of dimensionality [26], the applications would be limited to thermal storage tanks with a low-dimensional state space, i.e. a small number of controls and a small number of stratified layers.

1.4. Contributions and organization of the paper

The main goal of this paper is to solve the described problem by deriving a 1D continuous and smooth dynamical model that can accurately model the buoyancy effects and that can be used in derivative-based optimization algorithms that use analytical methods to compute derivative information, e.g. automatic differentiation. While a model with similar characteristics has been proposed in [12], that approach only modeled the buoyancy effects due to input/output flows, i.e. due to direct charging and discharging. In the current paper, the aim is different as we propose a more general model that can represent the buoyancy effects due to heat losses, indirect charging and discharging, and direct charging and discharging. In addition, a second difference w.r.t. [12] is the fact that, while the model of [12] included non-smooth expressions given by the min and max functions, the model proposed in the current paper is completely smooth.

As a second contribution, we take an approach that differs from the existing literature and we explicitly model and identify two different types of buoyancy: slow buoyancy effects that are linked to naturally occurring processes, and fast buoyancy effects that are associated with charging and discharging the vessel. As we show, this distinction is very important to obtain a smooth model that can accurately represent the buoyancy dynamics.

Finally, as a third contribution, to show the benefits of using the smooth model in optimization problems, we compare the smooth model against the non-smooth model in a real-life optimization setup: an optimal control application where a 1500 m^3 real commercial storage vessel needs to satisfy a given heat demand over some time horizon and, knowing the electricity prices over the given period, it has to minimize the cost of charging the vessel while satisfying the demand. Using this case study we show that the smooth model does not only obtain the best optimal solutions, but its computation costs are 100 times cheaper.

As a final remark, it is important to note that the model estimation and validation is done using a real thermal storage vessel. Having a real-life setup is important to strengthen the conclusions of the study as the model is validated in a real noisy environment.

The remainder of the paper is organized as follows: Section 2 introduces a real thermal vessel used as a case study to verify and study the proposed model. Section 3 presents and discusses the proposed model. Next, Section 4 estimates the parameters of the model when applied to the real system and validates the model. Then, Section 5 illustrates the benefits of the model in an optimal control set-up by comparing the model performance against non-smooth models from the literature. Finally, Section 6 concludes the paper and discusses future research.

2. Real thermal vessel

In this paper, to illustrate the proposed general model, we consider a real stratified thermal storage tank: the Ecovat vessel [27]. This system will be used as a case study to validate the proposed general model for thermal storage vessels and it is briefly presented here as a short introduction to this type of storage systems. Note however that the proposed approach is generic and can easily be applied to other types of stratified thermal storage tanks.

The considered thermal vessel is a large subterranean thermal storage vessel with capabilities for seasonal thermal storage and with the ability to supply heat demand to a cluster of buildings. The thermal vessel is divided into several segments that can be charged and discharged separately. Due to this property and due to thermal stratification, each of these segments acts as a different heat buffer. The vessel employs indirect charging/discharging via heat exchangers located in the vessel walls that can receive heat from two difference sources: a heat pump or resistance heaters. The insulation structure of the vessel is such that it can very efficiently store energy between seasons. In particular, the heat losses of the vessel are about 25% in a period of 6 months. Another advantage of the vessel is that, due to its indirect charging/discharging structure, the stratification of the layers, i.e. the exergy of the system, can be better maintained. Fig. 2 provides a schematic overview of the vessel and Fig. 3 depicts the real system when it was under construction.

2.1. Technical specifications

Depending on the specific use case, the commercial versions of the Ecovat vessel can be built with different volumes and different numbers of heat buffers, with volumes and numbers of heat buffers respectively ranging from 20,000 m^3 to 100,000 m^3 and from 8 to 15. For our case study, as the first commercial vessel is still under construction, we consider the Ecovat prototype, a thermal storage system of 1500 m^3 and 5 heat buffers that has been operational for approximately one year.

A technical schematic diagram of the prototype is depicted in Fig. 4, where the gray parts represent concrete elements, the blue ones water, and the yellow ones the insulation material. In the schematic diagram, all dimensions are expressed in millimeters, the reference point of the system is located approximately 1 meter above the ground, and the water level starts 4 meters below the reference point (i.e. approximately 3 meters below the ground level). The diameter of the vessel is approximately 11 meters, and the water depth is 15.3 meters (i.e. 19.3 meters deep from the reference point). Moreover, the considered vessel has 5 heat buffers, each one of them with different isolation thickness.

The vessel consists of an external concrete wall, an intermediate isolation layer, and an internal concrete wall. While it cannot be seen from the figure, the heat exchangers are not inside the water but embedded in the internal concrete walls. The thermal properties of the building materials, i.e. concrete and foam glass, are only known within a range of values. These parameters are listed in Table 1, which summarizes the thermal parameters of the concrete, the foam glass, and the fluid (water) used in the Ecovat prototype.

2.2. Working principle

As can be seen from Fig. 4, the vessel consists of 5 different heat

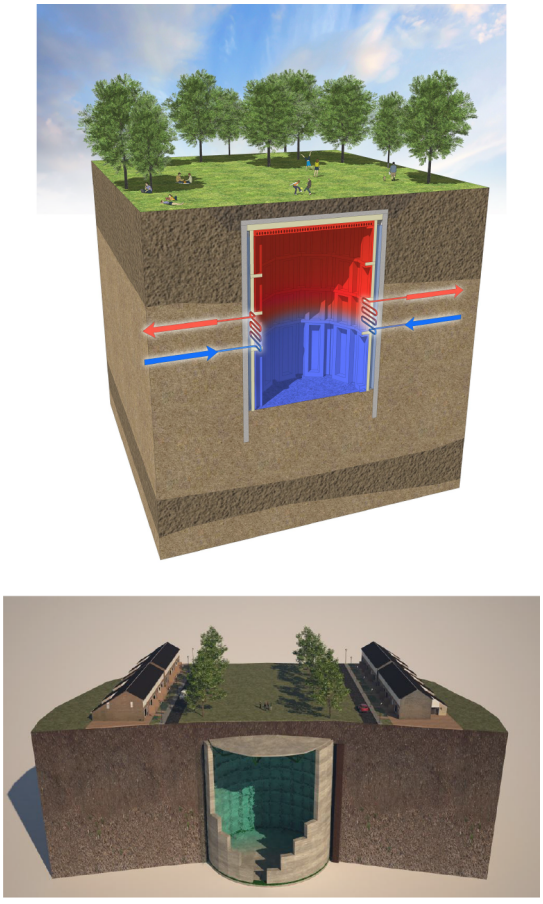


Fig. 2. Schematic representation of the Ecovat system.

Fig. 3. Construction of the real Ecovat system.

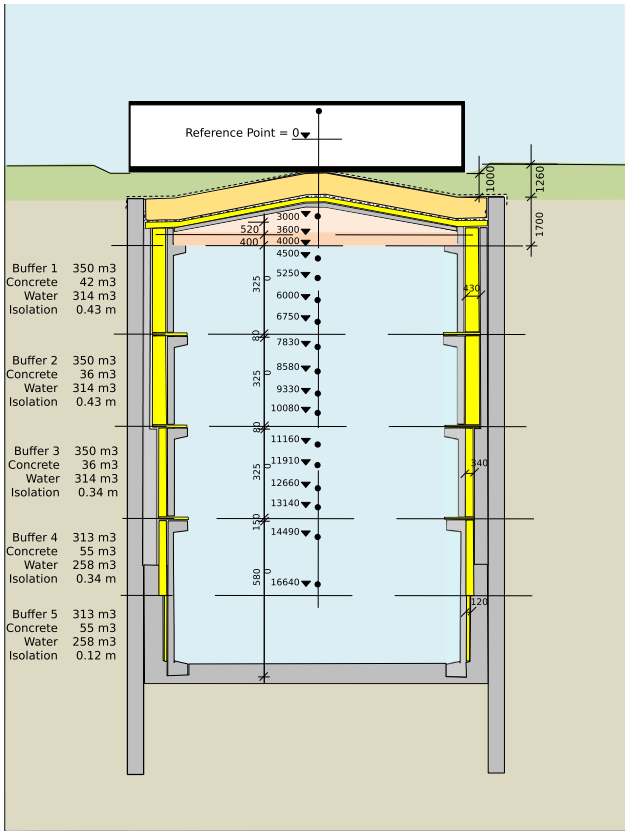


Fig. 4. Technical schematic diagram of the Ecovat. All dimensions are given in mm.

Table 1
Thermal properties of the materials.

Parameter	Value/Range
Density of concrete	2360 kg/m^3
Thermal conductivity of concrete	1.8 W/m K
Specific heat of concrete	[750, 1170] J/kg K
Thermal diffusivity of concrete	[10^{-6}, $6.5{\cdot}10^{-7}$] m^2/s
Thermal conductivity of foam glass	0.041 W/m K
Density of water	1000 kg/m^3
Specific heat of water	4181.3 J/kg K
Thermal diffusivity of water	$0.143{\cdot}10^{-6}$ m^2/s

buffers. These heat buffers are separated along the wall by a horizontal layer of isolation material so that the heat conductance across the walls is restricted to each buffer. This structure prevents the destruction of stratification due to wall conductance between the heat buffers and helps the vessel to store energy more efficiently.

In addition of being isolated, each of the heat buffers can be charged and discharged independently. Particularly, as mentioned in the previous section, each heat buffer has heat exchangers embedded in the internal walls, which allows to charge and discharge the heat buffer independently. This specific location of the heat exchangers will play a very important role when estimating the system parameters: as explained in Section 4.2, this characteristic implies that the medium where the heat is stored is a mixture of fluid and concrete.

3. Mathematical model

Before explaining and deriving the proposed mathematical model, it is necessary to introduce the models from the literature and to discuss their limitations.

3.1. Traditional model

The standard 1D model [10,17] for a heat storage vessel divides the tank in M segments/layers. Then, it models each layer with a *partial differential equation (PDE)* based on the heat transfer equation. In its most general case, each layer i is characterized by a state T_i representing the temperature of the layer; this state can be controlled by the input flow $\dot{m}_i$ and its temperature T_i^{in} or by the external input heat $\dot{Q}_i$ (heat sink or hear source) in the layer.

3.1.1. Partial differential equation

The PDE that models the state evolution of layer i is given by:

$$\frac{\partial T_i}{\partial t} = \alpha \frac{\partial^2 T_i}{\partial z^2} + \frac{P_i k_i}{\rho c_p A_i}\left(T_\infty - T_i\right) + \frac{\dot{Q}_i}{\rho c_p A_i \Delta z_i} + \frac{\dot{m}_i (T_i^{\text{in}} - T_i)}{\rho A_i \Delta z_i}, \tag{1}$$

where α, ρ, and c_p respectively represent the fluid diffusivity, density, and specific heat; A_i, P_i, and Δz_i the cross-sectional area, perimeter, and thickness of layer i;k_i the thermal conductance of the isolation wall of layer i; and T_∞ the ambient temperature (ground temperature if the vessel is underground).

It is important to note that not all vessel architectures make use of all input controls: in case of direct charging and discharging, only the input controls $\dot{m}_i$ and T_i^{in} are used. Likewise, in case of indirect charging and discharging, only $\dot{Q}_i$ is required. In addition, it is also important to remark that T_i^{in} might be T_{i-1} or T_{i+1} depending on whether the flow $\dot{m}_i$ comes from the bottom or top layer.

3.1.2. Mixing and inversion of layers

As briefly introduced in Section 1.3, this 1D model has an important drawback: as it is solely based on heat transfer, it cannot model the mixing of layers due to buoyancy effects. To address this issue, the traditional models from the literature perform, after each simulation time step, a *non-smooth* post-processing algorithm. In this post-processing step, the temperature of all layers is checked to detect buoyancy effects; if buoyancy is present, the layers involved are mixed; this process is repeated until all buoyancy effects are removed.

A simple example of this traditional simulation scheme is given in Algorithm 1. On this example, the vessel is divided into M layers, where layer M is at the top and layer 1 at the bottom of the tank, and the tank is steered using a generic control vector $\mathbf{u}$, where $\mathbf{u}$ can comprise the input flow $\dot{m}$, the input temperature T^{in} of the flow, and/or the heat sink/source $\dot{Q}$. The algorithm simulates the system using a simulation time step of length Δt and for a total of N time steps. As can be seen in lines 4 and 5, the simulation routine involves two steps: a first part where the PDE is solved and a second part where the buoyancy effects are included as an iterative algorithm.

Algorithm 1. Traditional Simulation Scheme

```
1:  function SIMULATOR T_1, [u_1, ..., u_N], Δt, V
2:      T_t ← T_1
3:      for t ∈ {1, 2, ..., N} do
4:          T_{t+1} ← simulatePDEStep(T_t, u_t, Δt)
5:          T_{t+1} ← correctBuoyancy(T_{t+1}, V)
6:      end for
7:      return T_1, T_2, ..., T_N
8:  end function
9:  function CORRECTBUOYANCY T, V
10:     while max_{i=2,...,N} T[i] − T[i − 1] < 0 do
11:         for i ∈ {2, ..., M} do
12:             if T[i] < T[i − 1] then
13:                 T[i], T[i − 1] ← mixLayers(T[i], T[i − 1], V[i], V[i − 1])
14:             end if
15:         end for
16:     end while
17: end function
18: function MIXLAYERS T_i, T_{i−1}, V_i, V_{i−1}
19:     ΔT = T_{i−1} − T_i
20:     T_{i−1} = T_{i−1} − (A_i Δz_i)/(A_i Δz_i + A_{i−1} Δz_{i−1}) ΔT
21:     T_i = T_i + (A_{i−1} Δz_{i−1})/(A_i Δz_i + A_{i−1} Δz_{i−1}) ΔT
22: end function
```

3.1.3. Model drawback

As explained in Section 1.3, while this traditional scheme is a very good approximation when simulating the dynamics of the heat storage vessel, it is not so suitable for use in derivative-based optimization problems. In particular, this limitation becomes very important when controlling the vessel and/or estimating its parameters: to solve the related optimization problems either heuristic methods or derivative-based optimization methods with finite differences are needed. As explained before, these methods have two problems:

1. Their computational requirements can easily become unacceptable.
2. They can compromise the quality of the solution.

3.2. Modeling slow buoyancy effects

In order to tackle these issues, derivative-based optimization methods with analytical derivative computations, i.e. automatic differentiation [28], can be used. However, in order to use them, the dynamics of the system need to be modeled by smooth expressions. In this paper, we propose a possible solution to include the buoyancy effects within the dynamics of the system using a continuous and smooth function. In particular, in this first section, we propose a general methodology for buoyancy effects whose time span is larger than the simulation time step, i.e. slow buoyancy effects. Next, in Section 3.3, we expand the model to also include fast buoyancy effects, i.e. buoyancy effects with a time span shorter than the simulation time step.

3.2.1. Discretized dynamics

In a simulation framework, the PDE defined by (1) is normally discretized and integrated on time using an expression of the following form:

$$T_{t+1,i} = F_i(\mathbf{T}_t,\ \dot{Q}_{t,i},\ \dot{m}_{t,i},\ T_i^{\text{in}},\ \Delta t), \tag{2}$$

where $T_{t,i}$ is the temperature of layer i at time step t, $\mathbf{T}_t$ the vector of temperatures in the M layers at time step t, $\dot{Q}_{t,i}$, $\dot{m}_{t,i}$, and T_i^{in} the control inputs of layer i at time step t, and Δt the length of the time step.

As an example, we can derive F_i in two steps using a simple integration method:

1. Transforming the PDE into an *ordinary differential equation (ODE)* by approximating the second-order spatial derivatives by finite differences.
2. Using a numerical integration method to perform integration of the ODE.

In this case, if a forward Euler method is used for the numerical integration, (2) is equivalent to:

$$T_{t+1,i} = T_{t,i} + \left(\alpha \frac{T_{t,i+1} + T_{t,i-1} - 2T_{t,i}}{\Delta z_i^2} + \beta_i\left(T_\infty - T_{t,i}\right) + \frac{\lambda_i}{\Delta z_i}\dot{Q}_{t,i} + \frac{\phi}{\Delta z_i}\dot{m}_{t,i}\left(T_{t,i}^{\text{in}} - T_{t,i}\right)\right)\Delta t \tag{3}$$

where Δz_i is the thickness of layer i and:

$$\beta_i = \frac{P_i k}{\rho c_p A_i}, \quad \lambda_i = \frac{1}{\rho c_p A_i}, \quad \phi = \frac{1}{\rho A_i}. \tag{4}$$

3.2.2. Buoyancy effects via the max function

As shown in [16], a single mixing iteration in the standard approach, i.e. line 13 in Algorithm 1, can be easily modeled using the *max* function by replacing line 13 by the following two expressions:

$$T[i] = T[i] + \theta_{i,i-1}\max(0,\, T[i-1] - T[i]), \tag{5}$$

$$T[i-1] = T[i-1] - (1-\theta_{i,i-1})\max(0,\, T[i-1] - T[i]), \tag{6}$$

where

$$\theta_{i,i-1} = \frac{A_{i-1}\Delta z_{i-1}}{A_i \Delta z_i + A_{i-1}\Delta z_{i-1}} \in \left[0,\, 1\right] \tag{7}$$

is the ratio between the volume of layer $i-1$ and the sum of the volumes of both layers.

It can be argued that if the dynamics of the buoyancy effects are slow, a single mixing iteration might suffice. In particular, if the simulation time step Δt is small in comparison with the time span where the effects of buoyancy start to be noticed, a single mixing iteration can keep up with the changes that occur due to buoyancy. In this scenario, using (5) and (6), the mixing algorithm can be directly integrated in the discrete dynamics. More specifically, (2) can be expanded as follows:

$$\begin{aligned} T_{t+1,i} = F_i(\mathbf{T}_t,\ \dot{Q}_{t,i},\ \dot{m}_{t,i},\ T_i^{\text{in}},\ \Delta t) + \theta_{i,i-1}\max(0,\ T_{t,i-1} - T_{t,i}) \\ - \theta_{i,i+1}\max(0,\ T_{t,i} - T_{t,i+1}) \end{aligned} \tag{8}$$

3.2.3. Buoyancy effects via a smooth function

While the *max* approximation allows to integrate the buoyancy effects directly within the dynamics, the resulting equations cannot easily be handled by a derivative-based optimization method as the *max* function is non-smooth. However, as indicated in [29], the *max* function can be approximated by the convex *log-sum-exp* function; in that case, (8) can be approximated by the following smooth expression:

$$\begin{aligned} T_{t+1,i} = F_i\left(\mathbf{T}_t,\ \dot{Q}_{t,i},\ \dot{m}_{t,i},\ \Delta t\right) + \theta_{i,i-1}\frac{1}{\mu}\log(e^0 + e^{\mu(T_{t,i-1}-T_{t,i})}) \\ - \theta_{i,i+1}\frac{1}{\mu}\log(e^0 + e^{\mu(T_{t,i}-T_{t,i+1})}), \end{aligned} \tag{9}$$

where the parameter μ is a scaling factor to make the *max* approximation sharper. The specific value of μ should be selected according to the specific application so that, while the approximation of the *max* function is sharp, there are no numerical issues. In the case of heat storage vessels, considering the range of temperature differences between consecutive layers, we have found that $\mu = 10$ is a reasonable value.

With this new approximation, the model for the vessel dynamics includes buoyancy, is represented by a smooth function, and can be integrated in any derivative-based optimization framework.

3.2.4. Empirical analysis

As indicated in the previous section, an important requirement for the proposed approximation to work is to have buoyancy effects whose time span is large in comparison with the simulation time step.

An example of a relative slow buoyancy effect is the natural recirculation of the fluid due to the higher heat losses in the top layer. In particular, as the top layer has a larger contact area with the environment, it suffers larger heat losses than the layers below it; as a result, the top layer reaches lower temperatures and the fluid from the lower layers rises to the top. As this effect depends on heat losses, it is a very slow process; in the case of the Ecovat vessel, it can be shown that, even for time steps Δt of 2 h, the proposed model can accurately model this natural buoyancy effect.

This can be observed in Fig. 5, where the simulated system using the slow buoyancy model is compared at different time steps Δt against the real data in the vessel during a 2-months cycle when the system was undisturbed, i.e. no heat was added or extracted (note that the simulation was done after estimating the model parameters; for the details of the parameter estimation we refer to Section 4). As can be observed, both $\Delta t = 30$ min and $\Delta t = 2$ h are small enough and a single mixing iteration per time step suffices to correct the buoyancy effect: the larger heat losses of the top layer cannot be noticed as the model correctly represents the mixing of the lower layers. However, as Δt increases to 4–6 h, the effect starts to be noticeable: the rate of heat losses in the top layer is larger than the update frequency of the buoyancy effect and the upper layers appear to have a lower temperature than the layers below it.

While the proposed model is very accurate for slow buoyancy effects, it is not for the faster buoyancy effects that appear due to charging/discharging. As indirect/direct charging introduces a much larger heat rate than heat losses, the corresponding buoyancy effects take place in a much shorter time span. This can be observed in Fig. 6, where the simulated slow model is compared for different time steps Δt against the real evolution of the vessel during a 3-week charging period. During that period, the vessel is charged via its heat buffer 3, i.e. right in the middle of the vessel; during that time there is a moment where the temperature in the middle of the tank is equal to the temperature of the layers above and mixing due to buoyancy starts to occur. While the proposed slow model still works, it only does so accurately for small time steps Δt. Particularly, while $\Delta t = 5$ min can correctly represent the buoyancy effects, $\Delta t = 30$ min already leads to an inconsistent system state where the middle temperatures are higher than the top ones. For even larger Δt the situation worsens: not only does the difference in temperature between the middle and top layers increase, but numerical artifacts such as oscillations appear in the simulated trajectory.

3.3. Modeling fast buoyancy effects

Based on the previous results, it is clear that to model fast buoyancy effects a different approach is needed. In this section, we propose a model for the buoyancy effects that appear when charging and discharging the vessel. For the sake of simplicity, the model explanation will focus on the case of indirect charging. However, in Sections 3.3.4 and 3.3.5 we briefly describe how the model also applies to the case of indirect discharging and direct charging/discharging. Similarly, during the model derivation, we will assume that all layers have the same volume; nonetheless, in Section 3.3.6 we will briefly explain the extension to layers with different volumes.

3.3.1. Empirical observation

When observing the buoyancy effect due to indirect charging, what we can effectively see is that any layer on top of the charged layer with a lower or equal temperature is also charged. More specifically, what we empirically observe is that, when applying heat to a specific layer, the heat is homogeneously distributed across the charged layer and any layer above it with an equal or lower temperature.

This concept is better understood with a simple example: consider a simple vessel with 4 layers as represented in Fig. 7. Let us analyze what the above effect means in terms of heating the third layer: if T_3 is equal or lower than T_1 or T_2, any heat applied to the first or second layer will be equally distributed to the third layer; similarly, if T_3 is equal or larger than T_4, any heat applied to the third layer will be effectively divided between the third and the fourth layer. Mathematically, this means that at any given time step t, the third layer is indirectly heated by the following amount:

$$\alpha_3 \dot{Q}_{t,3} + \alpha_2 \dot{Q}_{t,2} + \alpha_1 \dot{Q}_{t,1} \tag{10}$$

with

$$\alpha_3 = \begin{cases} 1 & \text{if } T_3 \leqslant T_4 \\ 0.5 & \text{if } T_3 > T_4 \end{cases} \tag{11}$$

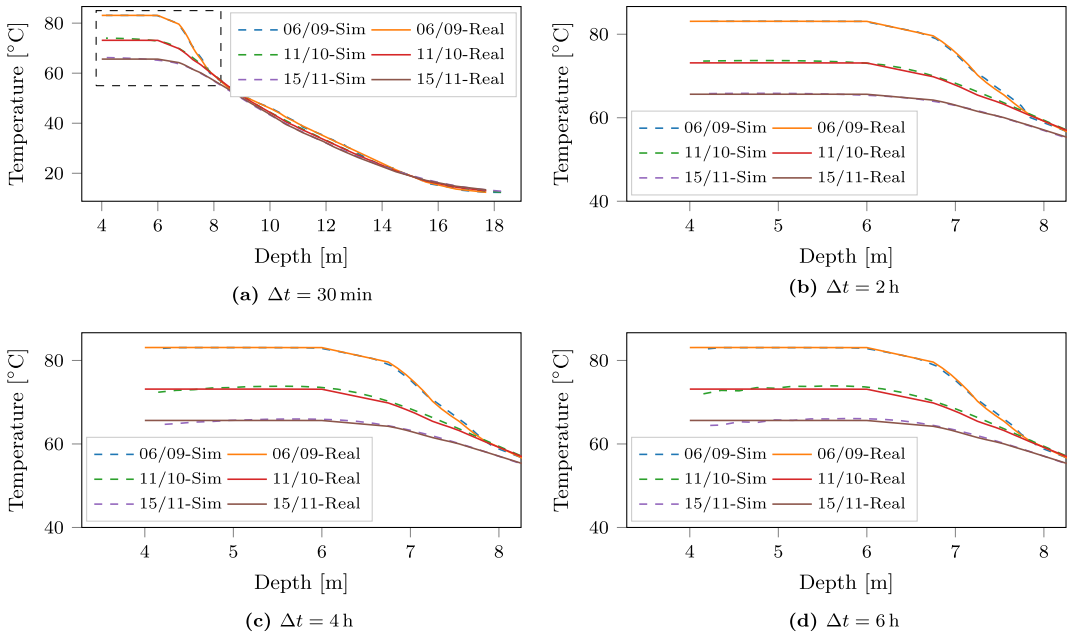


Fig. 5. Effect of Δt when using the slow buoyancy model for slow buoyancy effects. The plot compares real vs. simulated trajectories of the Ecovat vessel during a 2-month cycle where the vessel is undisturbed. As buoyancy is slow, a medium-size Δt, e.g. 30 min or 2 h, can model the buoyancy effect. However, as Δt increases, to e.g. 4–6 h, the effect starts to be noticeable.

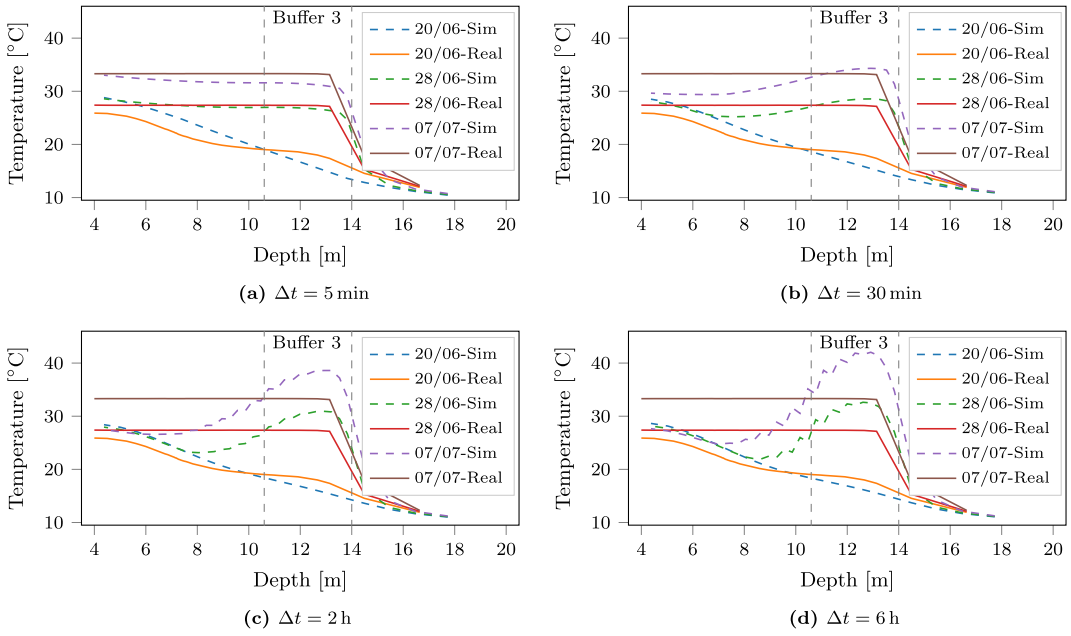


Fig. 6. Effect of Δt when using the slow buoyancy model for fast buoyancy effects. The plot compares real vs. simulated trajectories of the Ecovat vessel during a 3-week cycle where the third heat buffer is charged. In this scenario, the slow model only captures buoyancy effects when using a small Δt, e.g. 5 min.

$$\alpha_2 = \begin{cases} 0 & \text{if } T_2 \leqslant T_3 \text{ and } T_2 \leqslant T_4 \\ 0.5 & \text{if } T_3 < T_2 \leqslant T_4 \\ 1/3 & \text{if } T_3 < T_2 \text{ and } T_4 < T_2 \end{cases} \tag{12}$$

$$\alpha_1 = \begin{cases} 0 & \text{if } T_1 \leqslant T_3 \\ 1/3 & \text{if } T_3 < T_1 \leqslant T_4 \\ 0.25 & \text{if } T_3 < T_1 \text{ and } T_4 < T_1 \end{cases} \tag{13}$$

3.3.2. Mathematical modeling

Formally, the above observation means that, at any time step t, the ith layer in the vessel is heated by the following amount:

$$\sum_{l=0}^{i} \dot{Q}_{t,l} \cdot \begin{cases} \frac{1}{\sum_{j=l}^{M} \mathbb{1}_{T_{t,l} \geqslant T_{t,j}}} & \text{if } T_{t,l} \geqslant T_{t,i} \\ 0 & \text{else} \end{cases} \tag{14}$$

with $\mathbb{1}_S$ defining the indicator function:

$$\mathbb{1}_S = \begin{cases} 1 & \text{if } S \text{ is true,} \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

Using this empirical observation, we can now extend (2) to include the buoyancy effects due to indirect charging. In particular, by replacing $\dot{Q}_{t,i}$ by (14), these effects can be included in the model.

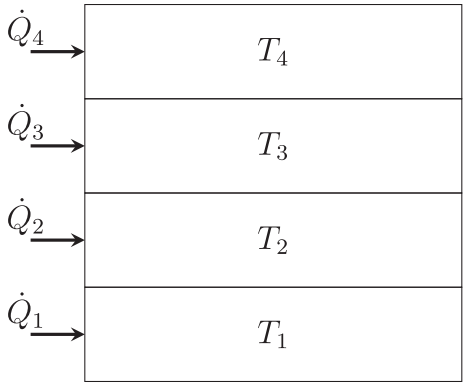


Fig. 7. Simplified heat storage vessel with 4 layers and indirect charge/discharge.

3.3.3. Smooth approximation

The model proposed in the previous section is non-smooth and non-continuous. As with the initial model proposed for the slow buoyancy effects, an approximation is needed to transform (14) into a smooth and continuous expression. Analyzing (14), it can be seen that its non-smoothness and its non-continuity come from the inclusion of step functions:

$$\text{step}(T_1 - T_2) = \begin{cases} 1 & \text{if } T_1 - T_2 \geqslant 0 \\ 0 & \text{else.} \end{cases} \tag{16}$$

Therefore, the only requirement to obtain a smooth and continuous model is to replace the step functions by a smooth and continuous approximation. One of the most popular approximations [30, Chapter 11] of the step function is the *logistic* function:

$$S(T_1 - T_2) = \frac{1}{1 + e^{-\mu(T_1 - T_2)}}, \tag{17}$$

where μ is a scaling parameter that indicates how sharp the logistic function is, i.e. the larger the μ the closer the logistic function is to the real step function. In all the experiments carried out in this research, $\mu = 1$ was selected as it was empirically shown to be an appropriate value.

Using this approximation, we can finally establish the smooth and continuous approximation of the buoyancy effects due to indirect charging. Particularly, by replacing $\dot{Q}_{t,i}$ in (14) by:

$$\dot{Q}'_{t,i} = \sum_{l=0}^{i} \dot{Q}_{t,l} \cdot \frac{S(T_{t,l} - T_{t,i})}{\sum_{j=l}^{M} S\left(T_{t,l} - T_{t,j}\right)}, \tag{18}$$

the buoyancy effects due to indirect charging can be included within the discrete system dynamics.

3.3.4. Modeling indirect discharging

While the proposed model was derived for the case of indirect charging, it can be easily extended to the case of indirect discharging. In detail, when discharging the vessel, the effect is the opposite: any layer below the layer that is discharged that has a temperature higher or equal than the discharged layer will also be discharged in an equally distributed manner.

This effect can be easily modeled using the same approximations as for the charging case, i.e. for the indirect discharging, $\dot{Q}_{t,i}$ should be replaced by:

$$\dot{Q}'_{t,i} = \sum_{l=i}^{M} \dot{Q}_{t,l} \cdot \frac{S(T_{t,i} - T_{t,l})}{\sum_{j=0}^{l} S\left(T_{t,j} - T_{t,l}\right)}, \tag{19}$$

3.3.5. Modeling direct charging and discharging

In the case of direct charging, the empirical observation is very similar: the heat of the incoming flow will be distributed across any layer on top of the input layer that has a temperature equal to or lower than the temperature of the incoming flow (note that for indirect charging it was any layer on top with a temperature lower than or equal to the temperature of the charged layer). Therefore, defining by $T^{\text{in}}_{t,i}$ the temperature of the incoming flow at time t and layer i, the approximation for direct charging can be easily derived from (18) with a minor modification: the replacement of the temperature of layer l at time step t, i.e. $T_{t,l}$, by the temperature of the incoming flow at layer l at time step t, i.e. $T^{\text{in}}_{t,l}$:

$$\dot{m}'_{t,i} = \sum_{l=0}^{i} \dot{m}_{t,l} \cdot \frac{S(T^{\text{in}}_{t,l} - T_{t,i})}{\sum_{j=l}^{M} S\left(T^{\text{in}}_{t,l} - T_{t,j}\right)}, \tag{20}$$

Using the same reasoning as for indirect discharging, the approximating for direct discharging can be similarly derived as:

$$\dot{m}'_{t,i} = \sum_{l=i}^{M} \dot{m}_{t,l} \cdot \frac{S(T_{t,i} - T^{\text{in}}_{t,l})}{\sum_{j=0}^{l} S\left(T_{t,j} - T^{\text{in}}_{t,l}\right)}, \tag{21}$$

3.3.6. Layers with different volume

For the sake of simplicity, all the derivations have been performed assuming constant volume across the layers, i.e. assuming that $A_i \cdot \Delta z_i$ was constant for each layer i. However, the model can be easily extended to the case where the volumes are not constant. In this case, it can be shown that (18) is equivalent to:

$$\dot{Q}'_{t,i} = \sum_{l=0}^{i} \dot{Q}_{t,l} \cdot \frac{A_i \Delta z_i S(T_{t,l} - T_{t,i})}{\sum_{j=l}^{M} A_j \Delta z_j S\left(T_{t,l} - T_{t,j}\right)}, \tag{22}$$

4. Parameter estimation and validation of the model

In order to validate the proposed model, we use it to estimate the parameters of the real stratified thermal storage vessel introduced in Section 2. First, we estimate the model parameters using a training dataset, and then evaluate the model performance in an out-of-sample dataset. Next, we compare the obtained parameters with their ideal value considering the structure and materials of the vessel.

4.1. Data

The employed data for validating the model consists of a period of 7 months divided in two different cycles:

1. A 5-month cycle (15/04/2017–10/09/2017) where the buffer 3, 4, and 5 (the top 3 buffers of the vessel) are charged interchangeably.
2. A 2-month cycle (10/09/2017–15/11/2017) where the system is left in a steady state and natural discharge occurs.

From this period, there are two types of measurements available:

1. The temperatures T inside the vessel at different depths; these are sampled with a frequency of 1 week.
2. The heat values $\dot{Q}$ introduced in each of the 5 heat buffers; these are sampled with a frequency of 15 min.

These measurements of the input heats and the temperatures are respectively depicted in Fig. 8 and Fig. 9.

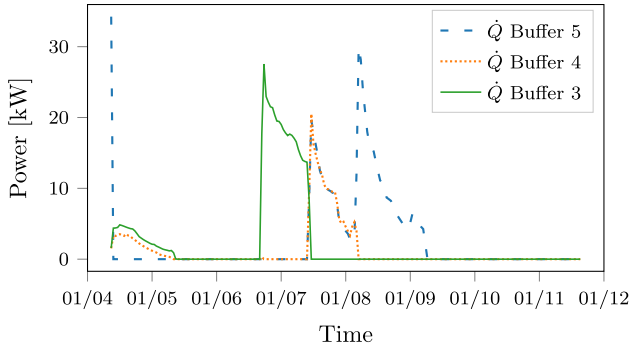


Fig. 8. Heat introduced in the vessel during the 7-month period considered.

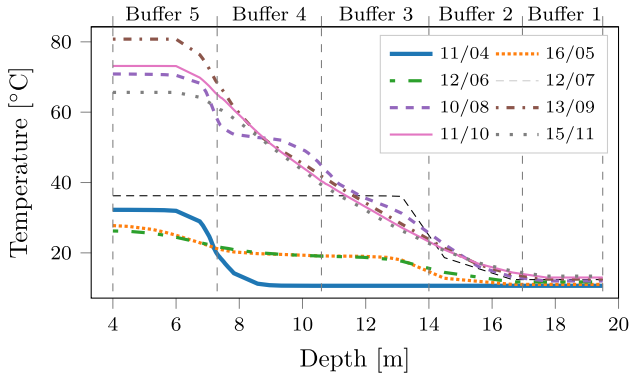


Fig. 9. Temperature evolution in the vessel at intervals of 1 month during the 7-month period considered. The black dashed lines represent the division between the 5 heat buffers within the vessel. The depth is measured using the reference system defined in Fig. 4.

4.2. Model parameters

Unlike most heat storage vessels, the considered vessel has its heat exchangers embedded into the concrete wall. As a result, the storage medium for the heat is a mixture of fluid and concrete and, unlike regular heat storage tanks, the parameters that define the dynamics of the vessel are not specific to the fluid but to a mixture of fluid and concrete. More specifically, the density ρ, the specific heat c_p, and the diffusivity α are defined by the properties of the fluid (water in the case of the considered vessel) and the concrete in the walls. Considering this observation, the fact that the considered vessel works with indirect charge/discharge, as well as Eqs. (1) and (3), it can be shown that there are 6 unknown parameters defining the system dynamics; these are listed and described in Table 2.

4.3. Estimation problem

Define by $\mathbf{T}_k = [T_{k,1}, ..., T_{k,M}]^\top$ the temperature distribution at time step k, by $\dot{\mathbf{Q}}_k = [\dot{Q}_{k,1}, ..., \dot{Q}_{k,M}]^\top$ the input heat at the same time step, by $\boldsymbol{\varphi} = [\alpha,\ \lambda,\ \beta,\ \beta_1,\ \beta_M,\ T_\infty]^\top$ the vector of unknown parameters, and by M the number of layers used to model the vessel dynamics. Define also the discrete system dynamics by $T_{k+1,i} = G_i(\mathbf{T}_k,\ \dot{\mathbf{Q}}_k,\ \Delta t,\ \boldsymbol{\varphi})$, with Δt the length of the discretization interval and with $G_i(\cdot)$ the numerical integration of (1) including the two proposed buoyancy models; i.e. $G_i(\cdot)$ is equivalent to (9) but replacing $\dot{Q}_{k,i}$ by (18) and (19). Then, the numerical optimization problem that is solved to estimate the model parameters is given by:

$$\underset{\boldsymbol{\varphi}, \mathbf{T}_0, ..., \mathbf{T}_N}{\text{minimize}} \sum_{i \in \mathscr{S}} \left\| \bar{\mathbf{T}}_i - \mathbf{T}_i \right\|_2^2 \tag{23a}$$

subject to

$$T_{k+1,i} = G_i(\mathbf{T}_k,\ \dot{\mathbf{Q}}_k,\ \Delta t,\ \boldsymbol{\varphi}) \quad \text{for } k = 0, ..., N-1, \text{ for } i = 1, ..., M \tag{23b}$$

where $\bar{\mathbf{T}}_i$ represents the temperature measurement at time step i; $\mathscr{S}$ the set of time indices when temperature measurements are available, i.e. $\mathscr{S} = \{i | i = 0,\ ..., N,\ \bar{\mathbf{T}}_i \text{exists}\}$[1]; and $N+1$ the number of discrete time points.

Note that the above formulation could also include the input heat $\dot{Q}$ as optimization variable. However, for the sake of simplicity, it is assumed that the measurements of $\dot{Q}$ are error-free. In addition, to avoid interpolations in $\dot{Q}$, Δt is selected as a multiple of the sampling period of $\dot{Q}$, i.e. $\Delta t = a \cdot 15\,\text{min}$ with $a \in \mathbb{N}$.

4.4. Experimental setup

To estimate and validate the model, the experimental setup is divided into two distinguishable parts: model validation and parameter estimation.

4.4.1. Model Validation

In a first part, the model is estimated using a training dataset and validated with an out-of-sample test dataset. The goal of this part is to ensure that the estimated model does not overfit the training dataset and to ensure that its accuracy is in agreement with the existing literature. To verify that the model validation is independent from the training/test dataset split, i.e. that it does not overfit, the parameter estimation is done 50 times and each time a random training/test dataset split is considered. Then, the average performance is compared with the performance of the existing models from the literature. To perform the splits, the training/test datasets are built using the following convention:

1. Training dataset: a dataset that spans 5.5 months and includes part of the charging period and part of the natural discharging period.
2. Test dataset: an out-of-sample dataset of 1.5 months that includes three weeks of the charging period and three weeks of the natural discharging period.

To randomize the splits, we consider that there are 25 weeks during the charging period and we randomly select an integer i between 1 and 23. Then, weeks i, $i+1$, and $i+2$ are used for the test dataset and the remaining 22 for the training dataset. Next, the process is repeated for the natural discharging period considering that in this case the number of weeks is 9, i.e. the randomly selected integer varies between 1 and 7. The results of this experiment are described in detail in Section 4.5.

4.4.2. Parameter estimation

After validating the model, the parameter estimation is performed on the full dataset to ensure that the parameters are estimated using as much information as possible. In addition, as the estimation problem (23a) is non-convex, the problem is solved using multi-start optimization. In particular, the optimization problem is solved multiple times using different initial guesses for the optimization variables and the optimal parameters are selected from the optimization run that obtains the best optimization cost. For this application, the optimization problem is solved 30 times[2] and the initial guesses are randomly generated using Gaussian distributions. For the six model parameters, i.e. α, λ, β, β_1, β_M, T_∞, the means of their Gaussian distributions are selected as their theoretical values assuming water as storage medium (see the first column of Table 4). For the temperature variables, at those time points where measurements are available, the means of the

[1] Note that the temperature sampling period of one week will always be larger than Δt.

[2] For the considered application, it was empirically observed that after 20–30 iterations the best optimal solution did not vary much anymore.

Table 2
Model parameters to be estimated.

Parameter	Description
α [m^2/s]	Diffusivity of the storage medium.
λ [m K/J]	Coefficient of the input heat. As the cross-sectional area A does not vary along the tank, the same $\lambda_i = \frac{1}{\rho c_p A_i}$ applies to each layer i.
β [1/s]	Coefficient of heat losses of the inner layers. As the cross-sectional A and the perimeter P do not vary along the tank, the same $\beta_i = \frac{P_i k}{\rho c_p A_i}$ applies to each inner layer i.
β_1 [1/s]	Coefficient of heat losses of the bottom layer, which differs from the general β as the bottom layer has a larger surrounding wall area.
β_M [1/s]	Coefficient of heat losses of the top layer, which differs from the general β as the top layer has a larger surrounding wall area.
T_∞ [°C]	Temperature of the surrounding terrain. While it is known that the surrounding ground has a temperature of 11–13 °C, the exact value is unknown.

Gaussian distributions are selected as the measured temperatures; for the time points where measurements are not available, the means are selected by a linear interpolation using the closest measurements in time. The standard deviations of the distributions are selected as half the value of the means. The results of this experiment are described in Section 4.6.

4.4.3. Implementation details

To estimate the model, we consider a discrete time step Δt of 2 h and an explicit Euler scheme to perform the numerical integration of the dynamics. In terms of the spatial discretization, we consider different thicknesses for the different heat buffers of the vessel. Particularly, as the top buffers are more often charged and discharged, they employ a coarser spatial discretization. The discretization uses 23 layers distributed as follows:

- 6 layers of 550 mm for the top heat buffer (buffer 5).
- 6 layers of 550 mm for the heat buffer 4.
- 6 layers of 550 mm for the heat buffer 3.
- 3 layers of 967 mm for the heat buffer 2.
- 2 layers of 1450 mm for the bottom heat buffer (buffer 1).

The problem is modeled in `python` using the mathematical modeling framework `CasADi` [31] and solved using the interior point solver IPOPT [32].

4.5. Model validation

To validate the model, the parameters are estimated for 50 different training/test dataset splits and, for each split, the model is evaluated in terms of the mean and maximum absolute errors for both the training and the test dataset. Then, the model is considered to be valid if its performance is independent of the training and test datasets and if its accuracy is in agreement with the existing literature. A summary of the results is listed in Table 3, which presents the average and standard deviation of the mean and maximum absolute errors across the 50 splits. In addition, the distribution of the two error metrics is depicted in Fig. 10. As can be observed, the average of both metrics in the training and test datasets is of the same order of magnitude, with the average errors in the test datasets being smaller. This last result is expected as the test datasets comprise a shorter time span and should therefore have lower errors.

Another observation that can be made is that the variance of the errors in the training datasets is smaller than the variance in the test datasets. This effect is also expected as the size of the training datasets is much larger than the size of the test datasets (a training dataset spans 5.5 months while a test dataset only 1.5 months). More specifically, as two training datasets can be at most 27% different (over the 5.5 months two training datasets can at most differ by a month and a half), it is normal for them to have similar errors. In contrast with that, two test datasets can show more variable errors as they can be 100% different (a test dataset only contains 1.5 months of the total 7 months of data).

Based on the obtained results it is clear that no significant differences exist between the performance of the model in the training datasets and the test datasets. Moreover, considering that over a period of 5.5 months the mean absolute errors are below 1 °C and the maximum errors are below 5 °C, it can be stated that the accuracy of the proposed model is in agreement with the existing literature [16]. Using these two observations it can be concluded that: (1) the proposed model is valid; (2) the model is accurate; (3) the model does not overfit the data.

Table 3
Comparison between training and test errors across the 50 training/test dataset splits in terms of the average and standard deviation of the mean absolute errors and the average and standard deviation of the maximum absolute errors.

Metric	Training set	Test set
Average mean absolute error	0.84 °C	0.64 °C
Std. mean absolute error	0.07 °C	0.39 °C
Average maximum absolute error	4.14 °C	2.88 °C
Std. maximum absolute error	0.50 °C	1.24 °C

4.6. Parameter estimation

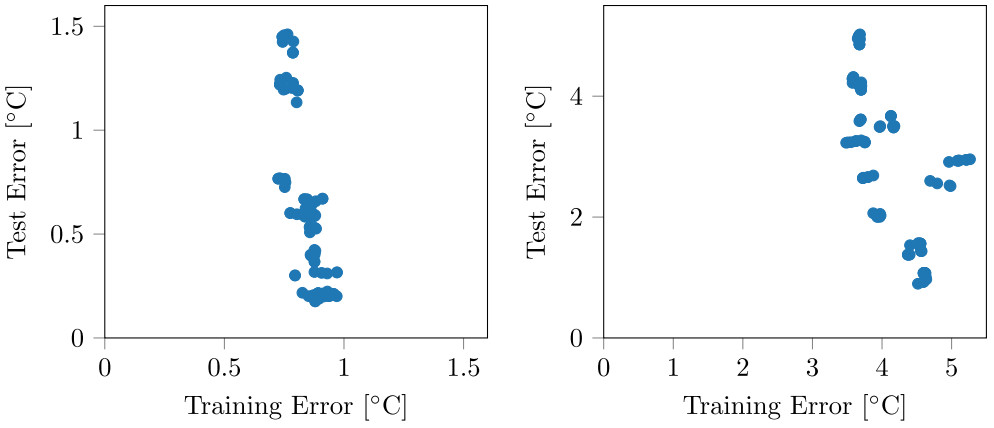
As described in Section 4.4.2, after the model has been validated, the parameters of the model are estimated using the full dataset and multi-start optimization. In addition, to further validate that the obtained parameters are reasonable, two comparisons are done:

1. The estimated parameters are compared for two different spatial discretizations: 23 layers (as before) and 31 layers. As the parameters are independent from the number of layers, a valid model should have similar parameters in both estimations.
2. The estimated parameters are compared against the theoretical parameters in the case that the heat storage medium is only water or only concrete. As motivated in Section 4.2, since the considered vessel has its heat exchangers embedded in the concrete wall, the parameters of the dynamics are not specific to the fluid but to a mixture of fluid and concrete. Therefore, if the model is valid, the estimated parameter are expected[3] to be in the range of the parameter values in the case of having only water as storage medium and the parameter values in the case of having only concrete.

The results of this experiment are listed in Table 4. As can be observed, the estimated parameters satisfy the two conditions for the model to be valid:

1. The difference in the estimated parameters for different numbers of layers is very small. Particularly, except for the heat loss parameter β_1 in the lowest layer, all other parameters show very small variations. In the case of β_1, this variation can be easily explained: as can be observed from Fig. 9, the temperature at the bottom of the vessel

[3] Assuming that, if the model parameters depend on two materials, their value is a linear combination of the parameters defined for each material.



(a) Mean absolute error (b) Maximum absolute error

Fig. 10. Mean and maximum absolute errors of the training and test datasets for the 50 different dataset random splits. As the average of both metrics in the training and test datasets is of the same order of magnitude, it is clear that no significant differences exist between the performance of the model in the training and test datasets.

Table 4
Results of the parameter estimation for different numbers of layers. The estimation is compared against the ideal theoretical parameters assuming that the construction dimensions are perfectly known and considering that the material for storing heat in the vessel is either only water or only concrete.

	Theoretical values		Estimated parameters	
Parameter	Water	Concrete	23 Layers	31 Layers
α [m^2/s]	$1.4{\cdot}10^{-07}$	$52.0{\cdot}10^{-07}$	$2.32{\cdot}10^{-07}$	$2.76{\cdot}10^{-07}$
λ [m K/J]	$2.4{\cdot}10^{-09}$	$5.7{\cdot}10^{-09}$	$3.49{\cdot}10^{-09}$	$3.48{\cdot}10^{-09}$
β [1/s]	$0.8{\cdot}10^{-08}$	$1.9{\cdot}10^{-08}$	$1.60{\cdot}10^{-08}$	$1.68{\cdot}10^{-08}$
β_1 [1/s]	$2.7{\cdot}10^{-07}$	$6.3{\cdot}10^{-07}$	$3.99{\cdot}10^{-07}$	$2.55{\cdot}10^{-04}$
β_M [1/s]	$5.7{\cdot}10^{-08}$	$13.5{\cdot}10^{-08}$	$9.62{\cdot}10^{-08}$	$10.15{\cdot}10^{-08}$
T_∞ [°C]	[11, 13]		13.03	13.09

only varies between 11 °C and 13 °C, which in turn is in the same range as the ground temperature T_∞. As a result, the heat losses in the lower layer have an almost negligible effect on the overall accuracy of the model and, as the problem is non-convex, there might exist several local minima with very similar accuracies but very different β_1 values.

2. All the estimated parameters are within the range of theoretical parameters corresponding to the cases of having only water as storage medium and having only concrete.

4.7. Discussion

Based on the obtained results, it can be claimed that the proposed model is a valid model that correctly identifies the system dynamics. In particular, the model accuracy in out-of-sample datasets is similar to the accuracy in training datasets and this accuracy is within the range of expected accuracy for a 1D model. In addition, the estimated parameters are within the range of values to be expected to correctly identify the physics behind the system dynamics. These two observations are clear indications that the model seems to be correct and that correctly represents the real physical behavior of heat storage vessels.

5. Model comparison

As motivated in the introduction, the goal of the proposed model is to provide a smooth representation of the dynamics of heat storage vessels so that the model can be employed in derivative-based optimization problems. In this section, we show the benefits obtained in optimization problems when using the proposed smooth model instead of the non-smooth models from the literature. In detail, we compare the quality of the optimal solution and the computation cost of the smooth and non-smooth models in the same optimization setup: an optimal control application where a stratified thermal storage vessel needs to satisfy a given heat demand over some time horizon while the electricity prices over the same period are known. The goal of the controller is to find the optimal charging strategy that minimizes the cost while satisfying the heat demand. As before, we consider as a case study the real stratified thermal storage vessel introduced in Section 2.

5.1. Smooth vs non-smooth models

As introduced in Sections 1.3 and 3.1, because the 1D dynamical models proposed in the literature are non-smooth, they cannot be employed in optimization setups using derivative-based optimization methods that employ automatic differentiation. Instead, they either need to employ heuristic methods or derivative-based methods that use finite differences.

5.1.1. Heuristic methods

As heuristic methods can optimize black-box functions, they are a good fit for optimization problems where the gradient and Hessian matrix of the problem cannot be computed analytically. However, while they can be used to integrate non-smooth models in optimization setups, they have two major problems: (1) they provide no guarantee of finding a local minimum or even a feasible solution; (2) they are usually computationally more expensive than derivative-based optimization methods.

5.1.2. Finite differences

Another option for non-smooth models is to employ finite differences to compute the derivative-based information and then use the same derivative-based optimization algorithms that smooth methods can use. However, unlike smooth models for which automatic differentiation can be used, the cost of computing second-order and first-order derivative information via finite differences respectively grows quadratically and linearly with the number of optimization variables. As a result, finite differences can quickly become computationally infeasible for many optimization problems.

More specifically, denoting the cost of evaluating some function $f\colon x \in \mathbb{R}^n \to \mathbb{R}$ by c_f, computing the gradient $\nabla_x f$ and the Hessian matrix $\nabla^2_x f$ via finite differences have approximate costs of $n{\cdot}c_f$ and $n{\cdot}(n+1){\cdot}c_f$ respectively[4]. By contrast, using automatic differentiation, the cost of computing $\nabla_x f$ is not only independent of the number of variables n but lower than $4{\cdot}c_f$; likewise, the cost of computing $\nabla^2_x f$ not only grows linearly with n but is bounded by $8{\cdot}n{\cdot}c_f$ [28]. This

[4] Note that the cost of computing the Hessian is derived for the general case of having a non-symmetric Hessian, i.e. the Hessian of a non-smooth function, as this represents the real cost of the non-smooth models from the literature.

Table 5
Comparison of the computation cost of using finite differences versus using automatic differentiation.

Function	Finite differences	Automatic differentiation
Cost(f: $x \in \mathbb{R}^n \to \mathbb{R}$)	c_f	c_f
Cost($\nabla_x f$)	$(n+1)\cdot c_f$	$<4\cdot c_f$
Cost($\nabla_x^2 f$)	$(n+1)^2\cdot c_f$	$<8\cdot n\cdot c_f$

Table 6
Parameters and variables of the optimal control problem considered for evaluating the proposed smooth model.

Parameter	Units	Description
N		Number of steps for the time horizon of the OCP.
$\mathbf{T}_k = [T_{k,1},...,T_{k,23}]^\top$	K	Temperature distribution at time step k.
$\dot{\bar{Q}}_k^{\mathrm{d}}$	W	Given heat demand at time step k.
p_k,	€/J	Given heat price at time step k.
$\dot{\mathbf{Q}}_k^{\mathrm{c}} = [\dot{Q}_{k,2}^{\mathrm{c}},...,\dot{Q}_{k,5}^{\mathrm{c}}]^\top$	W	Heat added to the four buffers at time step k.
$\bar{\mathbf{T}}_0$	K	Initial observed temperature.
$T_{k+1,i} = G_i(\mathbf{T}_k, \dot{\mathbf{Q}}_k^{\mathrm{c}}, \dot{\mathbf{Q}}_k^{\mathrm{d}})$	K	Discrete system dynamics.
$\mathscr{S}_i$		Set of indices of the layers in heat buffer i
$T_{\max}$	K	Maximum temperature in the vessel.
$\dot{m}_{\max}$	kg/s	Maximum water flow through the heat exchangers.
$T_{\mathrm{in}}^{\mathrm{d}}$	K	Minimum input temperature of the discharge heat exchanger.
$T_{\mathrm{in}}^{\mathrm{c}}$	K	Maximum input temperature of the charge heat exchanger.
c_{p}	J/(K·kg)	Specific heat of water.
k_{he}	W/K	Heat exchanger coefficient.

comparison is summarized in Table 5.

A second disadvantage of using finite differences is the accuracy: computing the gradient and Hessian matrix via finite differences reduces the numerical precision w.r.t. to the original function f and introduces truncation errors [28].

5.2. Optimal control problem

To evaluate and compare the performance of the proposed smooth model vs. non-smooth models, we consider an *optimal control problem (OCP)* where a stratified thermal storage vessel needs to satisfy a given heat demand over a horizon while ensuring that its costs are minimized. For this, the spatial discretization of 23 layers described in Section 4 is considered again. In addition, it is assumed that only the top 4 buffers (buffer 2 to buffer 5) can be charged and discharged and that each buffer has independent heat exchangers for charging and discharging. The definition of the different OCP variables and parameters are listed in Table 6. Using these definitions the OCP can then be defined as:

$$\underset{\substack{\mathbf{T}_0,...,\mathbf{T}_N,\ \dot{\mathbf{Q}}_0^{\mathrm{c}},...,\\ \dot{\mathbf{Q}}_{N-1}^{\mathrm{c}},\ \dot{\mathbf{Q}}_0^{\mathrm{d}},...,\dot{\mathbf{Q}}_{N-1}^{\mathrm{d}}}}{\text{minimize}} \quad \sum_{k=0}^{N-1} p_k\cdot\Delta t\cdot\sum_{i=2}^{5}\dot{Q}_{k,i}^{\mathrm{c}} + \gamma\cdot\left\|\mathbf{T}_0-\mathbf{T}_N\right\|_2^2 \tag{24a}$$

subject to

$$\mathbf{T}_0 = \bar{\mathbf{T}}_0 \tag{24b}$$

$$T_{k+1,i} = G_i(\mathbf{T}_k,\ \dot{\mathbf{Q}}_k^{\mathrm{c}},\ \dot{\mathbf{Q}}_k^{\mathrm{d}}),\quad \text{for } k=0,...,N-1,\ \text{for } i=1,...,M, \tag{24c}$$

$$\dot{\mathbf{Q}}_k^{\mathrm{c}} \geqslant 0 \quad \text{for } k=0,...,N-1 \tag{24d}$$

$$\dot{\mathbf{Q}}_k^{\mathrm{d}} \leqslant 0 \quad \text{for } k=0,...,N-1 \tag{24e}$$

$$T_\infty \leqslant \mathbf{T}_k \leqslant T_{\max},\quad \text{for } k=0,...,N \tag{24f}$$

$$\dot{\bar{Q}}_k^{\mathrm{d}} = \sum_{i=2}^{5}\dot{Q}_{k,i}^{\mathrm{d}},\quad \text{for } k=0,...,N-1 \tag{24g}$$

$$\dot{Q}_{k,i}^{\mathrm{c}} \leqslant \dot{m}_{\max}c_{\mathrm{p}}\left(T_{\mathrm{in}}^{\mathrm{c}} - \frac{1}{M_i}\sum_{j\in\mathscr{S}_i}T_{k,j}\right)\left(1-e^{\frac{-k_{\mathrm{he}}}{\dot{m}_{\max}c_{\mathrm{p}}}}\right)$$
$$\text{for } k=0,...,N-1,\quad \text{for } i=2,3,4,5 \tag{24h}$$

$$\dot{Q}_{k,i}^{\mathrm{d}} \geqslant \dot{m}_{\max}c_{\mathrm{p}}\left(\frac{1}{M_i}\sum_{j\in\mathscr{S}_i}T_{k,j} - T_{\mathrm{in}}^{\mathrm{d}}\right)\left(1-e^{\frac{-k_{\mathrm{he}}}{\dot{m}_{\max}c_{\mathrm{p}}}}\right)$$
$$\text{for } k=0,...,N-1,\quad \text{for } i=2,3,4,5. \tag{24i}$$

From the optimization problem above, several facts are to be noted:

- The penalty cost $\gamma\cdot\|\mathbf{T}_0-\mathbf{T}_N\|_2^2$ ensures that the optimal solution does not leave the vessel discharged.
- The constraint (24g) ensures that at each time step k the heat extracted from the 4 buffers is equal to the heat demand.
- In (24h) and (24i), $\frac{1}{M_i}\sum_{j\in\mathscr{S}_i}T_{k,j}$ is simply the average temperature of heat buffer i.
- The constraint (24h) ensures that the input heat is lower than the maximum input heat considering the maximum temperature and flow in the heat exchanger and the actual temperature in the vessel.
- Eq. (24i) ensures that the average temperature in a given heat buffer is enough to satisfy the heat demand required from that buffer.

5.3. Comparison setup

To compare the performance of the smooth model vs the non-smooth models from the literature, the OCP is first solved using the proposed smooth model and a second-order Newton-based optimization method with automatic differentiation. Then, the same OCP is solved using the equivalent non-smooth model and the following optimization techniques:

1. The *particle swarm optimization (PSO)* [33], one of the most widely used heuristic algorithms that has often been used for optimizing different types of energy storage systems [25,34,35]. The algorithm is run for 3 different numbers of particles: 100, 1000, and 10,000 particles.
2. The Markov chain Monte Carlo (MCMC) using the Metropolis–Hastings algorithm [36], a method used in the literature to estimate the parameters of a non-smooth heat vessel model [16], and also used for other energy demand applications [37].
3. The tree-structured Parzen estimator (TPE) [38], a black-box optimization algorithm previously used in energy-related applications [39–41].
4. A second-order Newton method where the Hessian is computed using finite differences.
5. A first-order Newton method where the Hessian is approximated

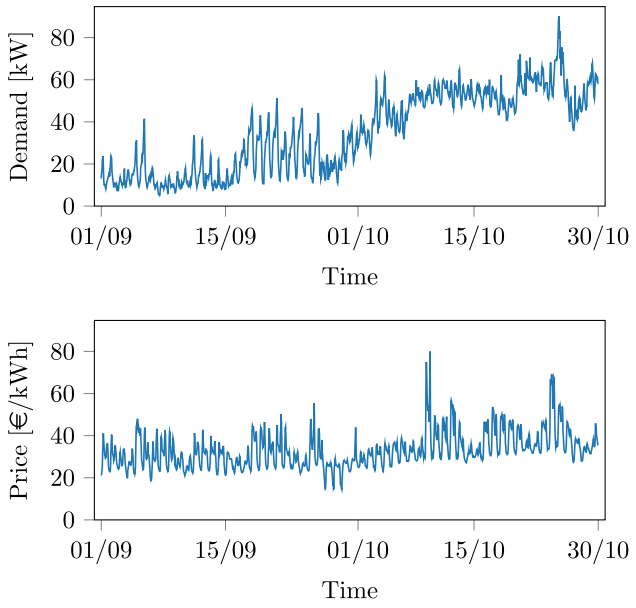


Fig. 11. Considered heat demand (top) and heat prices (bottom) in the OCP. The heat demand is the real heat consumption of a group of buildings. The prices are the real market prices of the day-ahead electricity market in The Netherlands.

using the gradient through the *Broyden–Fletcher–Goldfarb–Shanno (BFGS)* algorithm [42], and the gradient is computed using finite differences.

5.4. Implementation details

The heat demand used in the OCP is the realistic heat demand profile from a cluster of buildings in The Netherlands from 01/09/2016 to 31/10/2016. Similarly, the price of buying heat is assumed to be the price of buying the equivalent amount of electricity in the Dutch day-ahead electricity market in the same time period. Both quantities are depicted in Fig. 11.

All the OCPs are solved using a time resolution of 1 h, and to study the effect of the number of optimization variables, the OCP is solved for 4 different horizons: 1 day, 1 week, 1 month, and 2 months. In addition, as the constraints (24g)–(24i) cannot be explicitly included in the heuristic methods, they are modeled as penalty costs in the objective function.

For the Newton-based method using the smooth model, the system dynamics are defined as in the OCP above: using a multiple shooting algorithm [43] where the system state (temperature profile) is an optimization variable and the dynamics are ensured via constraint (24c). For the heuristic methods, in order to reduce the size of the search space, only the input heats $\dot{Q}^{c}$ and $\dot{Q}^{d}$ are considered as optimization variables. More specifically, the system dynamics are implicitly defined by modeling each temperature profile T_k as a function of the initial temperature T_0 and the previous heat inputs. As the cost of computing the Hessian via finite differences scales quadratically with the number of variables, the reduced OCP formulation is also employed for the Newton-based method that uses finite differences.

It is important to note that, as can be seen from (24a), the OCP is solved via a direct optimal control method where the controls and states are discretized in time and the problem is transformed into a nonlinear optimization problem. Thus, when compared with indirect methods, this approach has the disadvantage that, as it discretizes the controls and states, the obtained solution is just an approximation [44]. However, unlike indirect methods, this approach is robust to the initial approximation and requires a much lower amount of work since the user does not need to derive the adjoint differential equations [44].

Table 7
Comparison of the OCP optimal solution/cost (in EUR) for different time horizons using different optimization methods. The best solutions per time horizon are marked in bold. The first row represents the cost of buying directly the heat without the heat buffer, i.e. buying the heat demand at the actual market price. Cells with an x represent cases where the method was unable to find a feasible solution either because the algorithm converged to an infeasible solution (in the case of heuristic methods) or because the algorithm did not converge within 3 days (in the case of finite differences methods).

	OCP horizon [h]			
Optimization method	24	168	720	1440
No buffer	10.1	73.3	397.0	1761.1
Smooth model	**0**	**16.2**	**215.9**	**1108.9**
Finite-diff.: BFGS	**0**	**16.2**	235.1	x
MCMC	**0**	19.6	343.8	x
PSO 10000	**0**	38.1	2570.4	x
PSO 1000	**0**	361.6	3305.0	x
PSO 100	73.5	783.9	x	x
TPE	41.6	x	x	x
Finite-diff.: 2nd order	**0**	x	x	x

Table 8
Comparison of the computation time (in minutes) required to solve the OCP for different time horizons and using different optimization methods. The most cost efficient solutions at each time horizon are marked in bold. Cells with an x represent cases where the method was unable to find a feasible solution either because the algorithm converged to an infeasible solution (in the case of heuristic methods) or because the algorithm did not converge within 3 days (in the case of finite differences methods).

	OCP horizon [h]			
Optimization method	24	168	720	1440
Smooth-model	**0.1**	**0.6**	**5.1**	**34.9**
Finite-diff.: BFGS	0.3	47.6	708.3	x
MCMC	26.4	137.6	571.0	x
P-Swarm 10000	4.2	194.3	1203.6	x
P-Swarm 1000	0.8	18.8	165.6	x
P-Swarm 100	0.2	15.9	x	x
TPE	182.3	x	x	x
Finite-diff.: 2nd order	816.6	x	x	x

These two advantages together with the fact that the errors of direct methods are normally lower than 1% [44], make the direct approach a more suitable option for the current case study.

All experiments are implemented in `python`. The OCP using the smooth model is solved using CasADi [31] and IPOPT [32]. The same applies for the OCP using the non-smooth model and finite differences. For the heuristic methods, we consider the hyperopt [45] library for the TPE algorithm, the pyswarm library [46] for PSO, and an in-house library for MCMC.

5.5. Results

The comparison results of solving the OCP via the different methods are listed in Tables 7 and 8. Table 7 compares the quality of the optimal solution as the cost of buying heat and uses a baseline that represents the cost of buying heat without using the heat buffer, i.e. buying the heat demand at the actual market price. Table 8 compares the computation time required for each of the methods. In both tables the combinations of algorithm horizon that were unable to compute a solution were marked with an x; this issue always occurred for one of two reasons:

1. In the case of heuristic methods, the problem was always the same: the solver was unable to find a feasible solution. Particularly, the constraint (24h) limiting the maximum input heat was usually

violated.
2. In the case of finite differences, the algorithm was unable to find a solution after 3 days. It is important to note that this bound of 3 days was randomly selected to avoid situations where the algorithms would run indefinitely.

After analyzing the obtained results, the superiority of using a smooth model with a Newton-based optimization method and automatic differentiation becomes clear. Particularly, the following observations can be made:

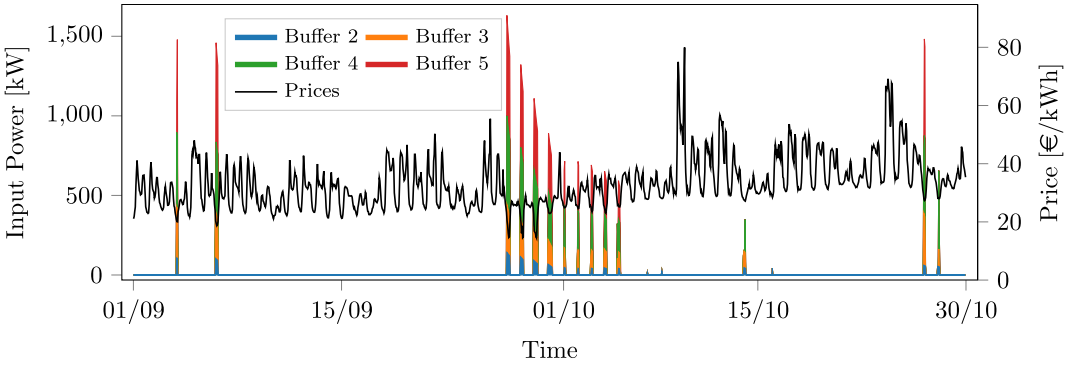
- The OCP solved with the smooth model is able to obtain the best solution for all possible horizons.
- Not only is the proposed approach the one with the best optimal solutions, but also the only one that outperforms the baseline across all horizons. In terms of economic savings, the proposed approach respectively reduces the baseline cost by 100%, 77%, 46%, and 37% for the 1-day, 1-week, 1-month, and 2-months horizons.
- In terms of accuracy, the majority of the alternative methods do not perform well: except for the shortest time horizon of 24 h, 5 of the 7 alternative methods are outperformed by the baseline and the remaining 2 directly fail to find any feasible solution.
- The best alternative methods are MCMC and BFGS, which can find better solutions than the baseline for 3 of the 4 horizons. However, they still fail to find a solution for the longest time horizon and they are outperformed by the proposed approach in terms of both computation time and the quality of the solution.
- In terms of computation cost, the method using the smooth model is by far the best: in comparison with all the other alternatives, the method using the smooth model finds the optimal solution between 10 and 100 times faster.
- As it could be expected, as the number of optimization variables increases, all the methods using the non-smooth model struggle to find optimal solutions. In the case of a 1-month horizon, only MCMC and BFGS are able to find a solution. In the case of a 2-months horizon, none of these methods can.

To further illustrate the good results of the smooth model, its optimal solution obtained for the longest horizon is depicted in Figs. 12 and 13. Fig. 12 illustrates the optimal charging and discharging strategy considering the heat price and the heat demand. Fig. 13 depicts the optimal temperature evolution of vessel when applying this optimal charging/discharging strategy. As could be expected, the optimal solution is to fully charge the vessel when prices are lowest, i.e. between the last week of September and the first week of October, and then to discharge the system to follow the heat demand.

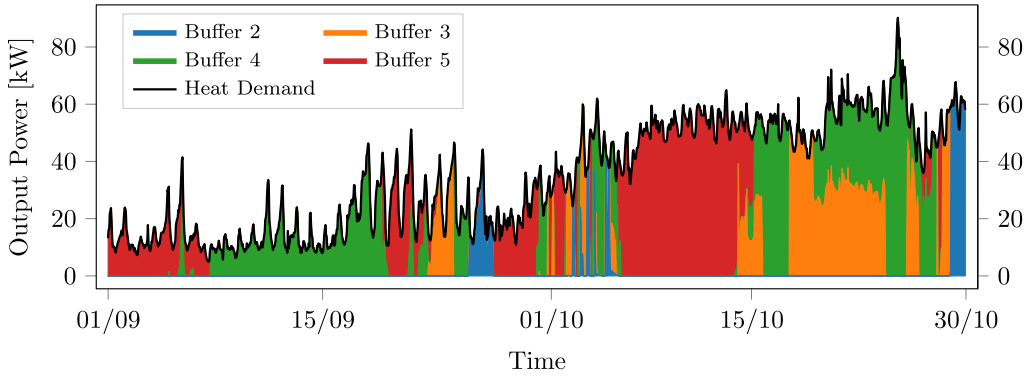
5.6. Discussion

Based on the obtained results, it is clear that smooth models are very important if heat storage vessels are used in optimization contexts, e.g. if heat storage vessels are to be controlled optimally. In particular, when solving the OCP that provides the best charging strategy for the vessel, the smooth model provides the best yet fastest optimal solutions by using derivative-based optimization with automatic differentiation.

This gain becomes more significant for optimization problems with a large number of variables; in those situations, both heuristic methods and derivative-based optimization methods using finite differences struggle to solve the optimization problem and they use significant amounts of computational resources. Particularly, looking at the case of a 2-months horizon OCP, none of these methods is able to find a feasible solution either because the algorithm converges to an infeasible



(a) Charging strategy



(b) Discharging strategy

Fig. 12. Optimal charging and discharging of the vessel over the 2-months period. As it is expected, the optimal solution charges the vessel when prices are low and discharges the system to follow the heat demand.

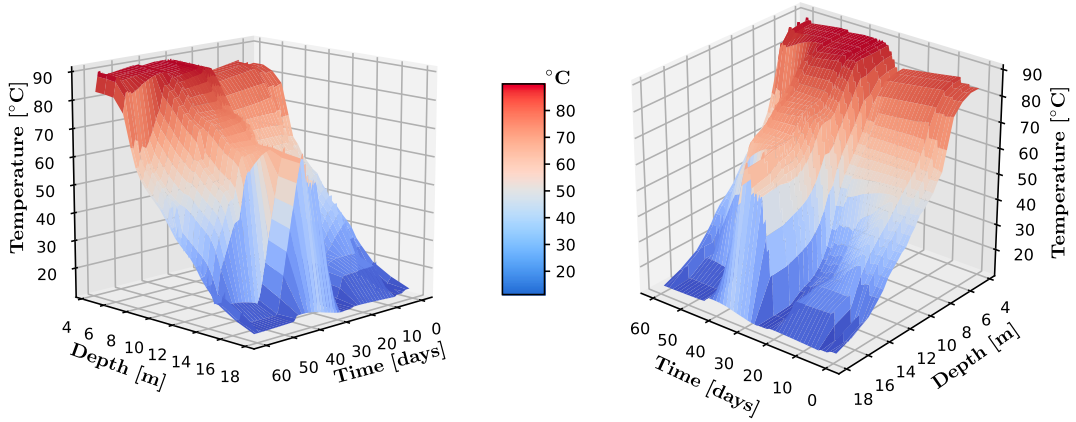


Fig. 13. Optimal temperature evolution in the vessel over the 2-months period.

Table 9
Comparison of the cost of using automatic differentiation versus using finite differences (via first-order and second-order methods) to compute the Hessian of the Lagrangian in the OCP.

	Computational cost		
	Finite differences		Automatic
Horizon	Exact Hessian	Hessian-BFGS	Differentiation
24 h	7.4 s[a]	78 ms[a]	1.5 ms
168 h	43 min[a]	5 s[a]	11 ms
720 h	73 h[a,b]	32 s[a]	65 ms
1440 h	405 h[a,b]	130 s[a]	100 ms

[a] To reduce the computational cost of finite differences and list their best case scenario, these costs consider the single shooting formulation where only the control inputs are optimization variables.

[b] This cost is an approximation: it is computed as the cost of evaluating the Lagrangian times $(n + 1)^2$, where n is the number of variables in the OCP.

solution (in the case of heuristic methods) or because the algorithm does not converge within 3 days (the case of finite differences). Similarly, for the 1-month horizon, only MCMC and BFGS can find a solution; however, even in that case, the optimal solutions of MCMC and BFGS are worse and 100 times more computationally expensive than the one provided by the smooth model via derivative-based optimization.

While they might seem surprising, these results are in fact not unusual as the search space for the heuristic methods becomes very large for the longer horizons: the number of variables to be optimized are 192, 1344, 5760, and 11,520 respectively for the 1-day, 1-week, 1-month, and 2-months horizons. In the case of numerical optimization with finite differences, a similar problem occurs: the computation time quickly becomes unacceptable. This property can be seen from Table 9, which lists the time to compute the Hessian of the Lagrangian for the 4 different horizons and for the first-order and second-order methods: for the longest horizon, the computation cost difference between using finite differences and automatic differentiation is a factor of 10^6 for the second-order method and a factor of 10^3 for the first-order method. Considering that computing the Hessian is done at each iteration of the optimization process, it becomes clear why the finite differences approaches do not scale well with the OCP horizon.

In addition to the quality of the solution and the cheap computational cost, another clear advantage of the smooth model is that it provides the only feasible alternative to be run in real time, e.g. in a model predictive control setup. In particular, a real-time control application would require computation times below the time step $\Delta t = 1$ h, and as can be seen from Table 8, only the proposed model satisfies that.

6. Conclusions

In this research, a new 1-dimensional model for stratified heat storage vessels has been proposed. The model overcomes the shortcomings of the existing models from literature by providing a smooth and continuous 1-dimensional representation of the system dynamics while including buoyancy effects. More specifically, this is the first model that, while remaining 1-dimensional, it is able to model buoyancy using a smooth and continuous function. The combination of the smoothness property and the 1-dimensionality of the model is critical to efficiently integrate the model in optimization problems and to obtain better optimal solutions while using less computational resources. These properties allow the use of state-of-the-art derivate-based methods which, in comparison with the optimization methods available for the non-smooth methods from the literature, are computationally much more efficient and lead to more optimal solutions.

In addition, the model further innovates the state-of-the-art in the field via a second contribution. In particular, by explicitly distinguishing between slow and fast buoyancy effects, the model obtains a more accurate smooth representation of the buoyancy dynamics.

To show the benefits and the accuracy of the model, we have considered a real commercial stratified storage vessel so that the model is evaluated in a real and noisy environment. In detail, two experiments were carried out: first, the model was validated using real data from a large stratified thermal storage vessel. During the estimation, the obtained parameters were shown to correctly identify the physical properties behind the system dynamics.

In a second experiment, the benefits of using the smooth model in optimization problems were demonstrated: the performance of the proposed smooth model was compared against that of non-smooth models from literature considering an optimal control problem where the stratified thermal storage vessel was controlled to minimize its costs while satisfying a given heat demand. In this case study, it was shown that the smooth model did not only result in the best optimal solutions, but it also required computation costs that were 100 times less.

In future research, other uses for the proposed model will be explored. A possible line of research will be the inclusion of the model in a model predictive control set-up where the heat storage vessel has to interact with multiple markets and systems.

Acknowledgment

This research has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No. 675318 (INCITE). The computational resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the Research Foundation Flanders (FWO) and the Flemish Government – department EWI.

References

[1] Campana PE, Quan SJ, Robbio FI, Lundblad A, Zhang Y, Ma T, et al. Optimization of a residential district with special consideration on energy and water reliability. Appl Energy 2017;194:751–64. https://doi.org/10.1016/j.apenergy.2016.10.005.

[2] Bloess A, Schill W-P, Zerrahn A. Power-to-heat for renewable energy integration: a review of technologies, modeling approaches, and flexibility potentials. Appl Energy 2018;212:1611–26. https://doi.org/10.1016/j.apenergy.2017.12.073.

[3] Stinner S, Huchtemann K, Müller D. Quantifying the operational flexibility of building energy systems with thermal energy storages. Appl Energy 2016;181:140–54. https://doi.org/10.1016/j.apenergy.2016.08.055.

[4] Energy consumption in households. In: Statistics Explained. Eurostat; 2018 < http://ec.europa.eu/eurostat/statistics-explained/index.php/Energy_consumption_in_households > .

[5] Chang C, Wu Z, Navarro H, Li C, Leng G, Li X, et al. Comparative study of the transient natural convection in an underground water pit thermal storage. Appl Energy 2017;208:1162–73. https://doi.org/10.1016/j.apenergy.2017.09.036.

[6] Li Z, Xu Y. Optimal coordinated energy dispatch of a multi-energy microgrid in grid-connected and islanded modes. Appl Energy 2018;210:974–86. https://doi.org/10.1016/j.apenergy.2017.08.197.

[7] Wang J, Zhong H, Ma Z, Xia Q, Kang C. Review and prospect of integrated demand response in the multi-energy system. Appl Energy 2017;202:772–82. https://doi.org/10.1016/j.apenergy.2017.05.150.

[8] Gabrielli P, Gazzani M, Martelli E, Mazzotti M. Optimal design of multi-energy systems with seasonal storage. Appl Energy 2018;219:408–24. https://doi.org/10.1016/j.apenergy.2017.07.142.

[9] Fasquelle T, Falcoz Q, Neveu P, Hoffmann JF. A temperature threshold evaluation for thermocline energy storage in concentrated solar power plants. Appl Energy 2018;212:1153–64. https://doi.org/10.1016/j.apenergy.2017.12.105.

[10] Han YM, Wang RZ, Dai YJ. Thermal stratification within the water tank. Renew Sustain Energy Rev 2009;13(5):1014–26. https://doi.org/10.1016/j.rser.2008.03.001.

[11] Renaldi R, Kiprakis A, Friedrich D. An optimisation framework for thermal energy storage integration in a residential heat pump heating system. Appl Energy 2017;186:520–9. https://doi.org/10.1016/j.apenergy.2016.02.067.

[12] Baeten B, Confrey T, Pecceu S, Rogiers F, Helsen L. A validated model for mixing and buoyancy in stratified hot water storage tanks for use in building energy simulations. Appl Energy 2016;172:217–29. https://doi.org/10.1016/j.apenergy.2016.03.118.

[13] Al-Habaibeh A, Shakmak B, Fanshawe S. Assessment of a novel technology for a stratified hot water energy storage – the water snake. Appl Energy 2018;222:189–98. https://doi.org/10.1016/j.apenergy.2018.04.014.

[14] Paradis P-L, Rousse DR, Lamarche L, Nesreddine H, Talbot M-H. One-dimensional model of a stratified thermal storage tank with supercritical coiled heat exchanger. Appl Therm Eng 2018;134:379–95. https://doi.org/10.1016/j.applthermaleng.2018.02.015.

[15] Rahman A, Smith AD, Fumo N. Performance modeling and parametric study of a stratified water thermal storage tank. Appl Therm Eng 2016;100:668–79. https://doi.org/10.1016/j.applthermaleng.2016.01.163.

[16] De Ridder F, Coomans M. Grey-box model and identification procedure for domestic thermal storage vessels. Appl Therm Eng 2014;67(1):147–58. https://doi.org/10.1016/j.applthermaleng.2014.03.003.

[17] Kleinbach EM, Beckman WA, Klein SA. Performance study of one-dimensional models for stratified thermal storage tanks. Solar Energy 1993;50(2):155–66. https://doi.org/10.1016/0038-092X(93)90087-5.

[18] Saloux E, Candanedo JA. Modelling stratified thermal energy storage tanks using an advanced flowrate distribution of the received flow. Appl Energy 2019;241:34–45. https://doi.org/10.1016/j.apenergy.2019.02.075.

[19] De Césaro Oliveski R, Krenzinger A, Vielmo HA. Comparison between models for the simulation of hot water storage tanks. Solar Energy 2003;75(2):121–34. https://doi.org/10.1016/j.solener.2003.07.009.

[20] Homan KO, Soo SL. Model of the transient stratified flow into a chilled-water storage tank. Int J Heat Mass Transfer 1997;40(18):4367–77. https://doi.org/10.1016/S0017-9310(97)00063-X.

[21] Pizzolato A, Donato F, Verda V, Santarelli M. CFD-based reduced model for the simulation of thermocline thermal energy storage systems. Appl Therm Eng 2015;76:391–9. https://doi.org/10.1016/j.applthermaleng.2014.11.029.

[22] Yaïci W, Ghorab M, Entchev E, Hayden S. Three-dimensional unsteady CFD simulations of a thermal storage tank performance for optimum design. Appl Therm Eng 2013;60(1):152–63. https://doi.org/10.1016/j.applthermaleng.2013.07.001.

[23] Hu M, Xiao F. Price-responsive model-based optimal demand response control of inverter air conditioners using genetic algorithm. Appl Energy 2018;219:151–64. https://doi.org/10.1016/j.apenergy.2018.03.036.

[24] Cox SJ, Kim D, Cho H, Mago P. Real time optimal control of district cooling system with thermal energy storage using neural networks. Appl Energy 2019;238:466–80. https://doi.org/10.1016/j.apenergy.2019.01.093.

[25] Liu Z, Chen Y, Zhuo R, Jia H. Energy storage capacity optimization for autonomy microgrid considering CHP and EV scheduling. Appl Energy 2018;210:1113–25. https://doi.org/10.1016/j.apenergy.2017.07.002.

[26] Bellman R. Dynamic programming. Princeton University Press; 1957.

[27] Ecovat. Ecovat renewable energy technologies BV < http://www.ecovat.eu > .

[28] Griewank A, Walther A. Evaluating derivatives: principles and techniques of algorithmic differentiation. vol. 105 Siam; 2008.

[29] Boyd S, Vandenberghe L. Convex optimization. Cambridge: University Press; 2004.

[30] Hastie T, Tibshirani R, Friedman J. The elements of statistical learning, springer series in statistics. New York, NY, USA: Springer, New York Inc.; 2001. doi:10.1007/978-0-387-21606-5.

[31] Andersson JAE, Gillis J, Horn G, Rawlings JB, Diehl M. CasADi – a software framework for nonlinear optimization and optimal control. Math Program Comput 2018:1–36. https://doi.org/10.1007/s12532-018-0139-4.

[32] Wächter A, Biegler LT. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. Math Program 2006;106(1):25–57.

[33] Kennedy J. Particle swarm optimization. Encyclopedia of machine learning. Springer; 2011. p. 760–6.

[34] Yan Y, Zhang C, Li K, Wang Z. An integrated design for hybrid combined cooling, heating and power system with compressed air energy storage. Appl Energy 2018;210:1151–66. https://doi.org/10.1016/j.apenergy.2017.07.005.

[35] Han X, Liu D, Liu J, Kong L. Sensitivity analysis of acquisition granularity of photovoltaic output power to capacity configuration of energy storage systems. Appl Energy 2017;203:794–807. https://doi.org/10.1016/j.apenergy.2017.06.062.

[36] Hastings WK. Monte Carlo sampling methods using markov chains and their applications. Biometrika 1970;57(1):97–109. https://doi.org/10.2307/2334940.

[37] Xie S, He H, Peng J. An energy management strategy based on stochastic model predictive control for plug-in hybrid electric buses. Appl Energy 2017;196:279–88. https://doi.org/10.1016/j.apenergy.2016.12.112.

[38] Bergstra J, Bardenet R, Bengio Y, Kégl B. Algorithms for hyper-parameter optimization. In: Advances in neural information processing systems; 2011. p. 2546–54.

[39] Lago J, De Ridder F, Vrancx P, De Schutter B. Forecasting day-ahead electricity prices in Europe: the importance of considering market integration. Appl Energy 2018;211:890–903. https://doi.org/10.1016/j.apenergy.2017.11.098.

[40] Lago J, De Ridder F, De Schutter B. Forecasting spot electricity prices: deep learning approaches and empirical comparison of traditional algorithms. Appl Energy 2018;221:386–405. https://doi.org/10.1016/j.apenergy.2018.02.069.

[41] Lago J, De Brabandere K, De Ridder F, De Schutter B. Short-term forecasting of solar irradiance without local telemetry: a generalized model using satellite data. Solar Energy 2018;173:566–77. https://doi.org/10.1016/j.solener.2018.07.050.

[42] Liu DC, Nocedal J. On the limited memory BFGs method for large scale optimization. Math Program 1989;45(1–3):503–28.

[43] Bock HG, Plitt K-J. A multiple shooting algorithm for direct solution of optimal control problems. IFAC Proc Vol 1984;17(2):1603–8.

[44] Von Stryk O, Bulirsch R. Direct and indirect methods for trajectory optimization. Ann Oper Res 1992;37(1):357–73. https://doi.org/10.1007/bf02071065.

[45] Bergstra J, Yamins D, Cox DD. Making a science of model search: hyperparameter optimization in hundreds of dimensions for vision architectures. Proceedings of the 30th international conference on machine learning. 2013. p. 115–23.

[46] Particle swarm optimization with constraint support. Python library < https://pythonhosted.org/pyswarm/ > .