Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 7954–7959

A novel learning-based MPC with embedded profiles prediction for microgrid energy management ⋆

V. Casagrande * **F. Boem** *

* *Dept. of Electronic and Electrical Engineering, University College London, UK (e-mail: vittorio.casagrande.19@ucl.ac.uk, f.boem@ucl.ac.uk.).*

Abstract: This paper presents a novel algorithm for microgrid energy management based on a differentiable learning-based Model Predictive Control (MPC) for jointly optimising profiles prediction and control performance. Specifically, we propose an algorithm for the online training of a Neural Network (NN) that predicts the unknown parameters of the MPC optimisation problem during control operation. Since the training is performed online at each time step the controller adapts to possible changes in the system parameters, while avoiding the offline training phase. Differently to standard methods in the literature, the proposed NN is trained by minimising a performance-based loss, i.e. the total cost of the energy trading with the utility grid. Simulation results show that the proposed approach outperforms the traditional approach minimising an estimation-only MSE loss, both when the model parameters are perfectly known and when they are uncertain.

Keywords: Model predictive control, Microgrid, Energy management, Convex optimisation, Energy systems, Learn-based control, Neural-network

1. INTRODUCTION

A microgrid Energy Management System (EMS) is the controller that computes the power flows to provide stable delivery of power to loads, while guaranteeing a cost-effective microgrid operation and other operational goals. One of the main challenges that arise in the design of an EMS is dealing with the uncertainty of electricity prices, renewable power production and load demand profiles. The standard approach to solve this problem is to first predict the unknown profiles, and secondly to compute the microgrid power scheduling solving an optimisation problem. In contrast to this, we take inspiration from the *performance-based* approach where the NN is trained by minimising the ultimate criterion the model is evaluated on. In the scheduling context, this means that optimisation objective is used to compute the loss function used for the NN training. Such performance-based approach has proved its effectiveness for scheduling problems in finance (Bengio (1997)) and offline battery management (Donti et al. (2017)). In order to train a NN embedding an optimisation problem, it is necessary to compute the gradients of its output (the optimal value) with respect to the NN parameters. Recent papers (for example Agrawal et al. (2019)) show that this can be done by implicitly differentiating the optimality conditions for convex optimisation problems. Starting from these results, the main contribution of this paper is an algorithm that allows to adopt the performance-based approach in an online way (i.e. in real-time at each time step) to schedule the microgrid power flows and learn from the efficacy of the past control actions. The method combines MPC and differentiable optimisation layers for energy management purposes in the case of unknown/uncertain future power profiles and electricity price. In particular, we propose an appropriate loss function for this scope and we develop an algorithm that allows to online train the obtained NN based on the optimality of the past decisions. We then apply such algorithm to a microgrid EMS and we show its superior performance with respect to the standard approach minimising a profile prediction loss. The advantages of this method are twofold: it takes advantage of NNs that have the ability to approximate functions with high accuracy (Ljung et al. (2020)) and exploits the MPC algorithm to compensate for system uncertainty and enforce constraints. We now provide a brief literature review to frame our work.

Learning-based MPC Machine learning methods can be used to improve the performance of a MPC (Al-Saadi et al. (2023)). In this paper, differentiable optimisation layers (Agrawal et al. (2019)) are exploited to improve the MPC by embedding the optimisation problem in a NN and training it end-to-end (E2E). The advantages of differentiating through an implicit optimisation problem have been explored in the MPC setting for the controller auto-tuning problem in Agrawal et al. (2021); Amos et al. (2018), where the MPC policy is trained by minimising an imitation loss, given an expert system running alongside. Conversely, in this paper we propose an algorithm that learns from the optimality of its past control actions without the necessity of an expert system and hence can be run in real-time.

⋆ This work has been supported by the UK Engineering and Physical Sciences Research Council (grant reference: EP/R513143/1, EP/W024411/1).

10.1016/j.ifacol.2023.10.915

Microgrid EMS Uncertainty is one of the main challenge in the EMS, which requires the prediction of unknown future profiles such as load, renewable power and electricity price. Many prediction tools have been used to deal with it: in Parisio et al. (2014) the unknown profiles are estimated using SVM regression, in Guo et al. (2015) a seasonal auto regressive moving average method is used, in Hans et al. (2015) Monte Carlo simulations are used to generate a set of future possible scenarios in the stochastic MPC framework. NNs have also been broadly used to predict unknown profiles, for example in Motevasel and Seifi (2014); Solanki et al. (2015); Wang et al. (2019). The main issue of the aforementioned methods is that the training is performed offline before the deployment of the controller, hence they do not take into account possible changes in the system. In order to overcome this limitation, Reinforcement Learning (RL) algorithms have been proposed for energy management, for example in Venayagamoorthy et al. (2016), Ji et al. (2019) and Liu et al. (2018). However there are some issues in the deployment of a RL algorithm in real systems since exploration can lead to unsafe situations and hence it requires a simulation environment with a lot of available data. Finally, performance-based E2E learning have been used for battery scheduling purposes in Donti et al. (2017) and it shows its superior performance with respect to networks trained by minimising the prediction error (MSE loss). In that paper, the NN is trained offline on a large dataset and it is then used to schedule the battery charge for the next 24 hours. In contrast to this approach, in this paper we propose an algorithm that can be trained online performing a training step at each time step, hence it does not require an offline dataset and the parameters can be adapted to a possibly time-varying system.

Contributions In this paper we propose an EMS that schedules the microgrid operation using a differentiable NN-based MPC algorithm. We introduce a novel algorithm to online train the NN and define a loss function. In particular, we make use of a performance-based (or task-based) loss and we show its advantages with respect to a standard MSE loss. The effectiveness of the proposed method is proved through extensive simulation results. Moreover, we test the algorithm in the case of uncertain model parameters. Specifically, we assume we do not know precisely the self-discharge rate and energy conversion efficiency of the battery. Simulation results show that the performance drop (compared to the completely known parameters scenario) is smaller when the performance-based loss is used with respect to an MSE loss. Preliminary results are presented in Casagrande and Boem (2023), where we assume to know the future power profiles of renewable generators and loads and we consider an exact model of the storage system. In this paper we further investigate the proposed method under storage system parameters uncertainty and including the estimation of the future renewable and load power profiles.

The rest of the paper is organised as follows. In Section 2 we outline the proposed learning-based MPC algorithm, the training procedure and the required dataset. In Section 3 we describe the microgrid model and we tailor the proposed method for the energy management problem. In Section 4 we show the results of the simulations and in Section 5 we draw the conclusions.

Notation The transpose operator is denoted by the superscript $^\top$, for example the vector $v^\top$ is the transpose of the vector v. We use the subscript to denote time instants, i.e. v_t is the vector v at time t. We denote the value of the variable v, k steps ahead of the time step t (i.e. at $t+k$) as $v_{k|t}$. The estimation of the variable v, available at time t, k steps ahead of the time step t (i.e. at $t+k$) is denoted as $\hat{v}_{k|t}$. We use bold variables to denote time sequences of N samples, namely $\mathbf{v}_{N|t} = \{v_{k|t}\}_{k\in\{0,\dots,N-1\}}$ is the v sequence computed at time t for the next N steps.

2. LEARNING-BASED MPC

Background. Optimisation is used in many decision-making applications, however some of the parameters of the problems are often uncertain. Formally, given an optimisation problem

$$\min_{\xi} \quad f(\xi, \omega) \tag{1a}$$

$$s.t. \quad \xi \in C(\omega), \tag{1b}$$

where ξ is the decision variable, ω is an unknown parameter vector, f is the objective function and C is the constraint set, the optimal solution $\xi^*(\hat{\omega})$ depends on the available estimation $\hat{\omega}$ of the unknown parameters and may not match the solution $\xi^*(\omega)$ based on the actual value of the unknown parameters. In some cases the unknown variable can be estimated from data, i.e. if a dataset $\mathcal{X} = \{\psi_i, \omega_i\}_{i=1}^N$ of input features ψ_i and corresponding parameter ω_i is available, then a statistical method can be used to estimate $\hat{\omega}$ (for example a NN trained minimising the MSE loss). In this context a *two-stage approach* can be used where estimation and optimisation are separately executed: first the unknown parameters are estimated, then such estimation is used in the decision-making optimisation problem (Mandi and Guns (2020)). In the Predict and Optimise framework (Kotary et al. (2021)), the goal is to used supervised learning to compute the estimate $\hat{\omega}$ so optimise $\xi^*(\hat{\omega})$ with respect to the optimisation problem cost function f. This is done by introducing a *regret* function as Mandi and Guns (2020):

$$\mathcal{L}(\hat{\omega}, \omega) = f_{\hat{\omega}}(\xi^*(\hat{\omega})) - f_{\omega}(\xi^*(\omega)), \tag{2}$$

where the first term is the optimal cost obtained using the estimate $\hat{\omega}$ and the second term is the optimal cost computed using the actual value ω. The first challenge of the performance-based approach is the implicit differentiation of the optimisation problem (1a)-(1b) required to find the gradients of the loss with respect to the network parameters θ ($\frac{\partial \mathcal{L}}{\partial \theta} = \frac{\partial \mathcal{L}}{\partial \xi^*}\frac{\partial \xi^*}{\partial \hat{\omega}}\frac{\partial \hat{\omega}}{\partial \theta}$) and the solution is provided in literature for various types of problems (Amos and Kolter, 2017; Elmachtoub and Grigas, 2021; Ferber et al., 2020). The second challenge is the online implementation, since the future values of the parameter ω are not known in advance, thus not making possible to calculate the regret or loss function (2). In this paper we propose a solution to this second challenge.

Algorithm overview. We consider a scheduling problem solved by an MPC algorithm. The MPC optimisation problem is formulated as follows:

$$\min_{\mathbf{u}_{T|t}} \quad \sum_{k=0}^{T-1} J(x_{k|t}, u_{k|t}, \hat{\omega}_{k|t}) \tag{3a}$$
$$\text{s.t.} \quad x_{k+1|t} = Ax_{k|t} + Bu_{k|t} \tag{3b}$$
$$x_{k|t} \in X, u_{k|t} \in U \tag{3c}$$
$$x_{0|t} = x_t \tag{3d}$$

where T is the prediction horizon, $x_t \in \mathbb{R}^n$ is the system state, $u_t \in \mathbb{R}^m$ is the system input, $X \subset \mathbb{R}^n$ is the state constraint set, $U \subset \mathbb{R}^m$ is the input constraint set and $\hat{\omega}_{k|t} \in \mathbb{R}^p$ is the estimation of the unknown parameter $\omega_{k|t}$. We assume the cost function is affine and constraints are convex leading to a convex optimisation problem. In a traditional MPC framework, at time t, Problem (3a)-(3d) is firstly solved to find the optimal sequence solution $\mathbf{u}^*_{T|t}(\hat{\boldsymbol{\omega}}_{T|t})$, then the input to the system is defined as $u_t = u^*_{0|t}$. At time step $t+1$ the optimisation is repeated updating the current state value (3d). In the performance-based framework the optimisation problem (3a)-(3d) is included as the last layer of a NN employed to jointly estimate the unknown parameters of the problem. We consider a look-back time window of length L, and the number F of input features. The input feature tensor at time t is denoted as $\psi_t \in \mathbb{R}^{L\times F}$ which includes the sequence of all the possible features that are used to compute the optimal solution of the optimisation problem $\mathbf{u}_{T|t}$. Input features may include the past values of the unknown parameter $\boldsymbol{\omega}_{L|t-L}$ and past state observations $\mathbf{x}_{L|t-L}$. The input ψ_t and predicted $\hat{\omega}$ samples in the forward pass (i.e. the computation of the output of the NN given the input tensor) are highlighted in red in the upper part of Fig. 1. The data collected during the controller operation is then used to improve the future controller performance. We define the following loss function:

$$\mathcal{L}(\hat{\boldsymbol{\omega}}_{T|t-T}, \boldsymbol{\omega}_{T|t-T}) = \sum_{k=0}^{T-1} \Big[J(u^*_{k|t-T}(\hat{\boldsymbol{\omega}}_{T|t-T}), \hat{\omega}_{k|t-T}) - J(u^*_{k|t-T}(\boldsymbol{\omega}_{T|t-T}), \omega_{k|t-T}) \Big] \tag{4}$$

where the first term is computed using the NN with input feature tensor ψ_{t-T} and the second term is computed by solving the optimisation problem (3a)-(3d) using the actual past values of the unknown parameter $\boldsymbol{\omega}_{T|t-T}$. Hence, the dataset used to train the network at time step t is $\mathcal{X}_t = \{\psi_{t-T}, \boldsymbol{\omega}_{T|t-T}\}$. The samples used for the training step are highlighted in blue in the lower part of Fig. 1. Each training step can be seen as a mini-batch update of the network, hence performances will not be good for some initial iterations.

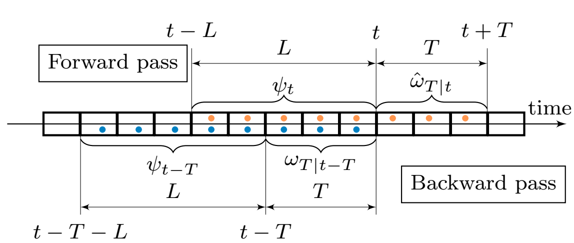

Fig. 1. Data structure for online training. In this example, $L=5$ and $T=3$.

3. MICROGRID ENERGY MANAGEMENT

In this Section we describe the microgrid model and the architecture of the EMS shown in Fig. 2.

3.1 Microgrid model

We consider a microgrid model similar to Casagrande et al. (2022a,b). We consider 4 types of agents connected to the microgrid: (i) load; (ii) renewable generator; (iii) energy storage system; (iv) connection to the utility grid. Loads and renewable generators are power sinks and power sources in the microgrid and they are both characterized by a power profile, $\bar{P}^l_t$ and $\bar{P}^r_t$ respectively. We denote such profiles as external power profiles since they cannot be changed nor controlled. We model the storage system as in Parisio et al. (2014) as a first order linear system denoting the level of charge of the storage as s_t:

$$s_{t+1} = (1-\sigma)s_t + \eta T_s P^s_t \tag{5}$$

where $\sigma \in [0,1]$ is the self-discharge decay, $\eta \in [0,1]$ is the energy conversion efficiency, T_s is the controller sample time and P^s_t is the power exchanged with the microgrid. Such power is limited as:

$$-\bar{P}^s \le P^s_t \le \bar{P}^s \tag{6}$$

and it is positive when it flows from the microgrid to the storage system. The limits on the storage capacity are denoted as $\underline{s}$ and $\bar{s}$:

$$\underline{s} \le s_t \le \bar{s} \tag{7}$$

The microgrid exchanges with the utility grid an amount of power denoted by P^g_t. The power balance constraint is used to ensure that power that is injected in the microgrid is equal to the power that is drawn from the grid at each time step:

$$P^g_t + P^s_t = \bar{P}^r_t - \bar{P}^l_t \tag{8}$$

One goal of the EMS is to ensure the economically efficient operation of the microgrid, i.e. we want to minimise the cost of buying energy from the utility grid. The total cost is expressed as:

$$\sum_{t=0}^{\infty} -p_t P^g_t \tag{9}$$

where p_t is the electricity price and the minus sign is necessary since the power is assumed to be positive when it is sold to the utility grid. It is not possible to use (9) as an objective function for the controller, since the resulting optimisation problem would be intractable and the future electricity price is not known in advance, hence Eq. (9) will only be used to compare the performance of different controllers. In the proposed MPC framework such objective is approximated by a finite horizon cost:

$$\sum_{k=0}^{T-1} -\hat{p}_{k|t} P^g_{k|t} \tag{10}$$

where $\hat{p}_{k|t}$ denotes the k steps ahead prediction of the electricity price computed at time t.

3.2 Prediction of the external power profiles

By external power profiles we mean the load and renewable generation power profiles. These profiles are not known in advance and need to be estimated. Tools used in the literature to estimate these profiles include support vector machines or NNs (Yafeng et al. (2008); Parisio et al. (2014)). Since we are considering Eq. (8), we are only interested in the estimation of $\bar{P}_t^d = \bar{P}_t^r - \bar{P}_t^l$. In this paper, we adopt recurrent NNs due to their efficacy and ease of use. The network developed to predict the disturbance power has a standard LSTM architecture: (i) LSTM layers; (ii) dense layer. The main hyperparameters are the number of LSTM layers and the dimension of the hidden state which are chosen making a trade-off between complexity and estimation results as explained in Section 4.

Forward pass: the network is fed at each time step with the past values of the disturbance power profile $\mathbf{P}^d_{L|t-L}$ and it predicts its future values $\hat{\mathbf{P}}^d_{T|t}$. Such values are used to solve the scheduling optimisation problem (12a)-(12f).

Backward pass: at each time step the NN is trained using the method described in Section 2. The input feature tensor is composed of the past disturbance profile samples in the look-back window $\mathbf{P}^d_{L|t-T-L}$. The network is trained by minimising the MSE loss between the predicted and true profile:

$$\mathcal{L}^{\text{MSE}} = \frac{1}{T}\sum_{k=0}^{T-1}\left[P^d_{k|t-T} - \hat{P}^d_{k|t-T}\right]^2 \tag{11}$$

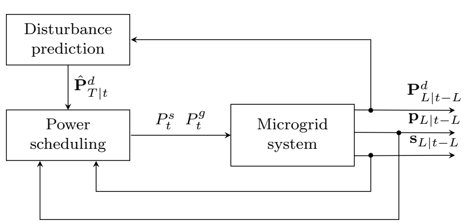

Fig. 2. Architecture of the EMS.

3.3 Power scheduling

Since the unknown parameters of the optimisation problem are time series we use an LSTM-based network designed as follows: (i) LSTM layers; (ii) dense layer; (iii) convex optimisation layer. Given an input tensor $\psi_t \in \mathbb{R}^{L\times F}$, the last LSTM layer outputs a tensor of dimension $h_t \in \mathbb{R}^{n_h}$ where n_h is the hidden state dimension. The dense layer applies to such vector a linear transformation and outputs $\hat{\mathbf{p}}_{T|t}$. This estimate is then passed to the convex optimisation layer that computes the power profile for the storage system and the connection to the utility grid. The problem solved at each time step is:

$$\min_{\mathbf{P}^s_t,\mathbf{P}^g_t} \quad \sum_{k=0}^{T-1} -\hat{p}_{k|t}P^g_{k|t} \tag{12a}$$

$$\text{s.t.} \quad s_{k+1|t} = (1-\sigma)s_{k|t} + \eta T_s P^s_{k|t} \tag{12b}$$

$$-\bar{P}^s \leq P^s_{k|t} \leq \bar{P}^s \tag{12c}$$

$$\underline{s} \leq s_{k|t} \leq \bar{s} \tag{12d}$$

$$P^g_{k|t} + P^s_{k|t} = \hat{P}^d_{k|t} \tag{12e}$$

$$s_{0|t} = s_t \tag{12f}$$

where (12a) is the finite horizon cost (10), (12b) represents the dynamics of the system as in Equation (5), (12c) (12d) are the constraints on the storage power and charge, respectively and (12f) is the current state. The power balance constraint (12e) uses the prediction of the future disturbance profiles obtained by the network described in Section 3.2. We point out that the disturbance power profile is a parameter for the optimisation problem, hence it could be jointly estimated with the price profile using a performance-based approach. However, since this power profile estimate appear as a constraint in the problem, this approach could lead to feasibility issues. Therefore we propose to estimate this profile only using the MSE loss. We assume to know the current value of the disturbance profile $\hat{P}^d_{k|t} = \bar{P}^d_t$

Forward pass: the network is fed with the input tensor ψ_t composed of the past electricity prices and the battery charge over the look-back window $\psi_t = \left[\mathbf{p}_{L|t-L}\ \mathbf{s}_{L|t-L}\right]$. The network then computes the power schedule for the battery and the electricity grid connection. Once the optimal solution $\mathbf{P}^{s,*}_{T,t}, \mathbf{P}^{g,*}_{T,t}$ is found, the control law is defined as $P^s_t(\hat{\mathbf{p}}_{T,t}) = P^{s,*}_{0|t}$ and $P^g_t(\hat{\mathbf{p}}_{T,t}) = P^{g,*}_{0|t}$. At the next time step, $t+1$ the network is fed with the tensor ψ_{t+1} and the optimisation problem is updated with the new feedback measurement s_{t+1} in (12f).

Backward pass: We denote the NN weights at time step t as θ_t and train the NN using a performance-based loss as in Eq. (2). In particular, we define the performance-loss for the energy management problem as:

$$\mathcal{L}^{\text{task}}_t = \mathbf{P}^g_{T|t-T}(\mathbf{p}_{T|t-T})\mathbf{p}^\top_{T|t-T} - \mathbf{P}^g_{T|t-T}(\hat{\mathbf{p}}_{T|t-T})\hat{\mathbf{p}}^\top_{T|t-T} \tag{13}$$

where $\mathbf{P}^g_{T|t-T}(\mathbf{p}_{T|t-T})$ is the solution of (12a)-(12f), defined at $t-T$, computed at time step t when the full sequence of the electricity price is known and $\mathbf{P}^g_{T|t-T}(\hat{\mathbf{p}}_{T|t-T})$ is the output of the NN with weights θ_t and input tensor ψ_{t-T}.

4. NUMERICAL RESULTS

We now present the results obtained by applying the proposed method to a microgrid composed of a load, a renewable generator, a storage system and a connection to the utility grid. We considered site 15 of the EMSx benchmark dataset (Le Franc et al. (2021)). This dataset contains historical data of renewable energy production and load demand. The electricity price profile has been downloaded from the ENTSO-E Transparency Platform (ENTSO-E, 2008), described in Hirth et al. (2018), in particular we considered the *IT-Centre-North* bidding zone in 2017. The profiles are shown in Fig. 3. We simulate the controller using three different loss functions to train the NN: (i) the MSE loss (11) denoted as “MSE”; (ii) the performance-based loss (13) denoted as “task”; (iii) the hybrid loss defined as:

$$\mathcal{L}^{\text{hybrid}}_t = w_1\mathcal{L}^{\text{task}}_t + w_2\mathcal{L}^{\text{MSE}}_t \tag{14}$$

where $w_1, w_2 \in \mathbb{R}$. In this paper we tuned the hyperparameters of each network considering combinations of the following: (i) LSTM number of layers $n_l \in \{1,2\}$; (ii) hidden dimension of LSTM layers $n_h \in \{2,10\}$; (iii) weight w_1 and w_2 of (14), $(w_1,w_2) = \{(1,0),(0,1),(1,1),(10,1),(0.1,1)\}$. In Casagrande et al. (2023) we further investigate the online learning algorithm and we address the problem of hyperparameters optimisation. We then selected the best

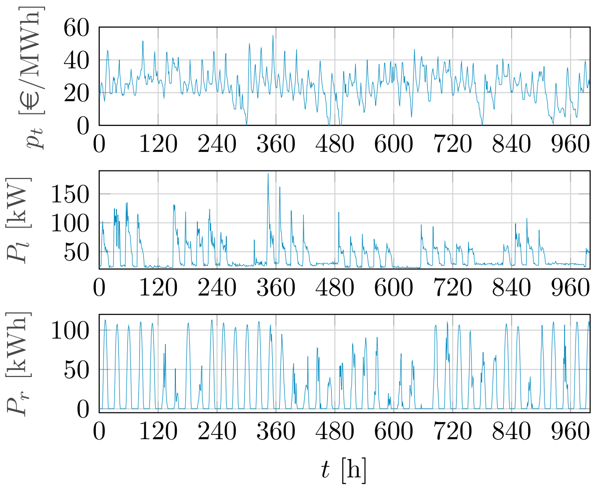

Fig. 3. Price, load and renewable generator profiles.

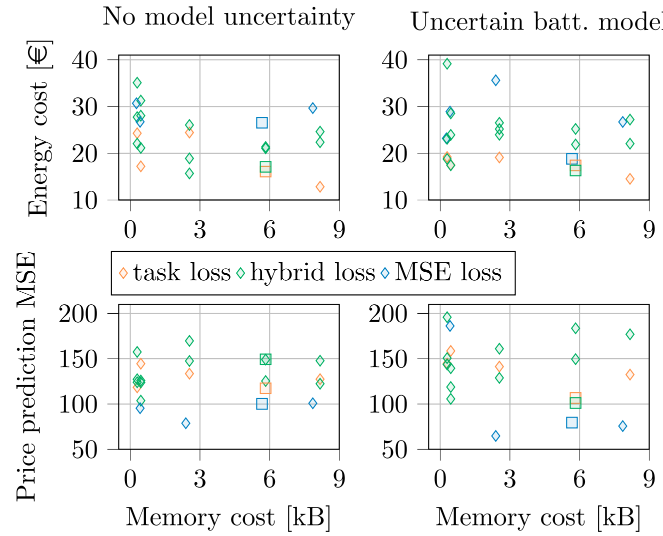

Fig. 4. Electricity cost, memory and price prediction MSE obtained for different hyperparameters. The total cost of the prescient controller is −8.83 €.

network for each loss type and ran the simulations to assess the controller. The obtained results are assessed on the last 200 steps of the simulation on the basis of three performance indicators: (i) memory cost, i.e. the memory required to store the network weights; (ii) MSE of the price profiles prediction; (iii) total electricity cost of the microgrid as in Eq. (9). The required memory is a measure of the network complexity, a network with less parameters is preferred since it requires less memory to store the weights and less computational power (Casagrande et al., 2021). The price estimation MSE is computed as:

$$\mathrm{MSE} = \frac{1}{200}\sum_{t=800}^{1000}\left(p_{t+1} - \hat{p}_{1|t}\right)^2. \tag{15}$$

The electricity cost is computed using Eq. (9) as:

$$\mathrm{Cost} = \sum_{t=800}^{1000} -p_t P_t^g. \tag{16}$$

The other parameters are set as $T_s = 1\,\mathrm{h}$, $T = 12$, $L = 24$, $\underline{s} = 0\,\mathrm{kW\,h}$, $\bar{s} = 400\,\mathrm{kW\,h}$, $\bar{P}^s = 100\,\mathrm{kW}$, $\sigma = 0.0042$, $\eta = 0.95$. Finally, we considered two scenarios: (i) in the first the controller has a perfect knowledge of the battery model (5), i.e. constraint (12b) is implemented with the real values of σ and η; (ii) in the second, the controller uses an approximated battery model by setting $\sigma = 0$ and $\eta = 1$ in (12b). We denote as "Prescient" the controller that has a perfect knowledge of the model parameters, as well as of all the profiles, and computes the control actions by solving the optimisation problem (12a)-(12f) and we use this controller as a benchmark. Fig. 4 shows the results of the simulations for different hyperparameters combinations and adopted loss functions comparing the energy cost and MSE. On the left we consider the controller with no uncertainty on battery parameters whereas on the right the controller uses given incorrect values of σ and η. On the x-axis there is the required memory size in kilobytes of the trained network. As expected we see that the network trained with MSE loss has the lowest MSE, however networks trained with task and hybrid loss have a lower energy cost. We further analyse the best network for each loss denoted by a square in Fig. 4. We ran 10 experiments (with different random seed) for the best network obtained for each loss, which all have 2 LSTM layers and 10 hidden units. Results are shown in Table 1. We see that in the case of model uncertainty overall performance is worse than the prescient case for all the controllers and especially for the controller trained with the MSE-based loss. We see that the latter controller achieves a lower price prediction MSE but a higher electricity cost. Indeed, controllers trained with hybrid and task-based loss functions allow to achieve better performance in terms of electricity cost even though the MSE on the price profile is higher. In the case of exact battery model knowledge the controller trained with hybrid loss achieves the lowest cost and lowest variability over different runs. In the case of model uncertainty the same controller achieves the best performance however it has a high variability. We point out that by using our method it is not necessary to specify which parameters of the system are uncertain: the network adapts the price prediction in order to maximise the control performance.

Loss	No model uncertainty			
	w_1	w_2	MSE	Cost
MSE	0	1	99.15 ± 14.64	28.00 ± 13.02
task	1	0	143.78 ± 33.57	23.79 ± 7.00
hybrid	10	1	141.21 ± 29.27	21.69 ± 5.36
	Uncertain battery model			
	w_1	w_2	MSE	Cost
MSE	0	1	104.53 ± 19.86	33.10 ± 10.23
task	1	0	134.40 ± 36.30	24.62 ± 4.05
hybrid	0.1	1	126.98 ± 23.11	22.84 ± 11.44

Table 1. Results of 10 runs for each controller.

5. CONCLUSION

In this paper we presented a novel algorithm to online train a NN embedding a MPC controller together with profiles prediction for the microgrid energy management problem. Such method is based on differentiable optimisation layers, hence the network is trained E2E based on the performances of the past control actions. The network is trained online so a dataset for training prior to the deployment of the controller is not required. The network parameters are adapted online to maximise the control performance. Simulation results show that taking into account the performance of the controller in the loss function allows to outperform the MSE loss both, with perfect knowledge of the battery model and in the case of model uncertainty. As a future work, we will consider robustness and stability properties of the controller, taking into account the uncertainty on the estimates of the unknown parameters.

REFERENCES

Agrawal, A., Amos, B., Barratt, S., Boyd, S., Diamond, S., and Kolter, J.Z. (2019). Differentiable convex optimization layers. *Advances in neural information processing systems*, 32.

Agrawal, A., Barratt, S., and Boyd, S. (2021). Learning convex optimization models. *IEEE/CAA Journal of Automatica Sinica*, 8(8), 1355–1364.

Al-Saadi, M., Al-Greer, M., and Short, M. (2023). Reinforcement learning-based intelligent control strategies for optimal power management in advanced power distribution systems: A survey. *Energies*, 16(4), 1608.

Amos, B., Jimenez, I., Sacks, J., Boots, B., and Kolter, J.Z. (2018). Differentiable mpc for end-to-end planning and control. *Advances in neural information processing systems*, 31.

Amos, B. and Kolter, J.Z. (2017). Optnet: Differentiable optimization as a layer in neural networks. In *International Conference on Machine Learning*, 136–145. PMLR.

Bengio, Y. (1997). Using a financial training criterion rather than a prediction criterion. *International journal of neural systems*, 8(04), 433–443.

Casagrande, V. and Boem, F. (2023). Model predictive control based on differentiable optimisation layers for microgrid energy management. In *2023 European Control Conference (ECC) (Accepted)*. IEEE.

Casagrande, V., Fenu, G., Pellegrino, F.A., Pin, G., Salvato, E., and Zorzenon, D. (2021). Machine learning for computationally efficient electrical loads estimation in consumer washing machines. *Neural Computing and Applications*, 33(22), 15159–15170.

Casagrande, V., Martin, F., Rodrigues, M., and Boem, F. (2023). An online learning framework for microgrid energy management control. In *2023 Mediterranean Conference on Control and Automation (MED) (Submitted)*. IEEE.

Casagrande, V., Prodan, I., Spurgeon, S.K., and Boem, F. (2022a). Resilient distributed mpc algorithm for microgrid energy management under uncertainties. In *2022 European Control Conference (ECC)*, 602–607. IEEE.

Casagrande, V., Prodan, I., Spurgeon, S.K., and Boem, F. (2022b). Resilient microgrid energy management algorithm based on distributed optimization. *IEEE Systems Journal*.

Donti, P.L., Amos, B., and Kolter, J.Z. (2017). Task-based end-to-end model learning in stochastic optimization. *arXiv preprint arXiv:1703.04529*.

Elmachtoub, A.N. and Grigas, P. (2021). Smart "predict, then optimize". *Management Science*.

ENTSO-E (2008). Transparency platform. URL < `https://transparency.entsoe.eu` >.

Ferber, A., Wilder, B., Dilkina, B., and Tambe, M. (2020). Mipaal: Mixed integer program as a layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 1504–1511.

Guo, Y., Xiong, J., Xu, S., and Su, W. (2015). Two-stage economic operation of microgrid-like electric vehicle parking deck. *IEEE Transactions on Smart Grid*, 7(3), 1703–1712.

Hans, C.A., Sopasakis, P., Bemporad, A., Raisch, J., and Reincke-Collon, C. (2015). Scenario-based model predictive operation control of islanded microgrids. *2015 54th IEEE conference on decision and control (CDC)*, 3272–3277.

Hirth, L., Mühlenpfordt, J., and Bulkeley, M. (2018). The entso-e transparency platform–a review of europe's most ambitious electricity data platform. *Applied energy*, 225, 1054–1067.

Ji, Y., Wang, J., Xu, J., Fang, X., and Zhang, H. (2019). Real-time energy management of a microgrid using deep reinforcement learning. *Energies*, 12(12), 2291.

Kotary, J., Fioretto, F., Van Hentenryck, P., and Wilder, B. (2021). End-to-end constrained optimization learning: A survey. *arXiv preprint arXiv:2103.16378*.

Le Franc, A., Carpentier, P., Chancelier, J.P., and De Lara, M. (2021). Emsx: a numerical benchmark for energy management systems. *Energy Systems*, 1–27.

Liu, W., Zhuang, P., Liang, H., Peng, J., and Huang, Z. (2018). Distributed economic dispatch in microgrids based on cooperative reinforcement learning. *IEEE transactions on neural networks and learning systems*, 29(6), 2192–2203.

Ljung, L., Andersson, C., Tiels, K., and Schön, T.B. (2020). Deep learning and system identification. *IFAC-PapersOnLine*, 53(2), 1175–1181.

Mandi, J. and Guns, T. (2020). Interior point solving for lp-based prediction+ optimisation. *Advances in Neural Information Processing Systems*, 33, 7272–7282.

Motevasel, M. and Seifi, A.R. (2014). Expert energy management of a micro-grid considering wind energy uncertainty. *Energy Conversion and Management*, 83, 58–72.

Parisio, A., Rikos, E., and Glielmo, L. (2014). A model predictive control approach to microgrid operation optimization. *IEEE Transactions on Control Systems Technology*, 22(5), 1813–1827.

Solanki, B.V., Raghurajan, A., Bhattacharya, K., and Canizares, C.A. (2015). Including smart loads for optimal demand response in integrated energy management systems for isolated microgrids. *IEEE Transactions on Smart Grid*, 8(4), 1739–1748.

Venayagamoorthy, G.K., Sharma, R.K., Gautam, P.K., and Ahmadi, A. (2016). Dynamic energy management system for a smart microgrid. *IEEE transactions on neural networks and learning systems*, 27(8), 1643–1656.

Wang, T., He, X., and Deng, T. (2019). Neural networks for power management optimal strategy in hybrid microgrid. *Neural Computing and Applications*, 31(7), 2635–2647.

Yafeng, Y., Yue, L., Junjun, G., and Chongli, T. (2008). A new fuzzy neural networks model for demand forecasting. In *2008 IEEE International Conference on Automation and Logistics*, 372–376. IEEE.