IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 September 2023, accepted 10 October 2023, date of publication 16 October 2023, date of current version 23 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3325194

RESEARCH ARTICLE

Imitation Learning for Agnostic Battery Charging: A DAGGER-Based Approach

ANDREA POZZI[1], (Member, IEEE), AND DANIELE TOTI[1,2], (Member, IEEE)
[1]Faculty of Mathematical, Physical and Natural Sciences, Catholic University of Sacred Heart, 25133 Brescia, Italy
[2]Department of Sciences, Roma Tre University, 00146 Rome, Italy

Corresponding author: Daniele Toti (daniele.toti@unicatt.it; daniele.toti@uniroma3.it)

ABSTRACT This work presents a novel approach to the challenge of battery charging under real-world constraints, related to uncertainties in system parameters and unmeasurable internal states of batteries. By leveraging the imitation learning paradigm, this study introduces an innovative solution to address the inherent challenges associated with traditional predictive control strategies. A key contribution of this work is the successful application and adaptation of the Dataset Aggregation (DAGGER) algorithm to an ‘‘agnostic scenario’’, characterized by uncertain battery parameters and unobservable internal states. Furthermore, this work is, to the authors’ best knowledge, the first attempt to amalgamate deep predictive control within the imitation learning framework, offering a fresh perspective and broadening the array of possible solutions to the difficulties in battery charging. Results derived from a realistic battery simulator implementing an electrochemical model demonstrate marked enhancements in battery charging performance, particularly in satisfying temperature constraints. The performance of the proposed algorithm surpasses that of existing approaches, including a benchmark behavioral cloning method based on supervised learning. These advancements highlight the potential of the imitation learning paradigm in tackling complex control problems in battery management systems.

INDEX TERMS Dataset aggregation, deep neural networks, imitation learning, optimal battery charging, predictive control.

I. INTRODUCTION

In recent years, the ecological transition has been gaining momentum, with batteries at the heart of this transformation. Their role is particularly central in the realm of sustainable mobility, given the burgeoning prevalence of electric vehicles equipped with lithium-ion technology [1]. As these vehicles become increasingly commonplace, the importance of enhancing battery efficiency, longevity, and safety has heightened. In the midst of these developments, the charging phase warrants particular attention. If not managed properly, it can result in underutilization of the battery, compromised safety, and premature aging [2]. Hence, the use of sophisticated control strategies has become increasingly prevalent in both industrial applications and academic literature. These are often implemented in the so-called advanced battery management systems, which typically leverage a mathematical model of the battery to optimize its charging [3]. Among the most frequently employed strategies is Model Predictive Control (MPC) [4]. By solving an optimal control problem, MPC encapsulates the intricacies of battery dynamics and operational constraints, thereby meeting the intended objectives. This approach’s effectiveness is apparent in several studies such as those proposed by the work in [5], [6], [7], and [8].

Nevertheless, the deployment of MPC strategies comes with its fair share of hurdles. A key challenge lies in the online computation of a constrained optimization at every time step that MPC requires, a process that can be computationally intensive, particularly when dealing with nonlinear battery models. Furthermore, predictive control encounters limitations when dealing with uncertainties in system parameters and difficulties associated with measuring specific internal states of the battery. These limitations arise

The associate editor coordinating the review of this manuscript and approving it for publication was Longzhi Yang.

from the assumption that MPC operates under perfect knowledge of system dynamics, an assumption that frequently falls short in real-world situations.

The computational issues are mainly tackled in two ways. One approach involves the use of reduced-order models. These models strike a delicate balance between ensuring model accuracy and managing computational complexity [9], [10], [11]. An alternative approach is to rely on the so-called explicit MPC, which aims to simplify real-time operations to a basic function evaluation [12]. Specifically, explicit MPC precomputes the optimal control action as a piece-wise function of state and reference vectors, meaning its real-time computational cost is reduced to identifying the region in which the states are located. However, this method is not a universal remedy; the computational cost can still surge when dealing with a large number of constraints, leading to a drastic increase in prediction horizons as the number of regions also increases [13]. Many scientific studies have focused on reducing these computational costs (for instance, see works by [14], [15], [16]). One common strategy, proposed by most of these studies, is to approximate the predictive control law in some manner. Lately, the use of deep neural networks for such approximation has garnered significant attention, thanks to their high representation capabilities, leading to what is now known as the deep model predictive control framework [17]. Past research in the realm of deep MPC includes studies like the one proposed in [18], in which the authors present a model that exhibits robustness against input errors. Another notable work is by the authors in [13], where they demonstrate a deep predictive controller's capability to accurately represent an explicit MPC control law, provided that an ample number of neurons and layers are employed. Within the specific sphere of battery charging, deep MPC was proposed in [19] as a solution to reduce the online computational cost to manageable levels.

As previously stated, another significant challenge beyond the computational complexity lies in the inherent assumptions of MPC that all system parameters are certain and all internal battery states are measurable. In practical settings, these assumptions often fail, as parameters need to be inferred from available measurements like voltage, current, and surface temperature, and only the structure of the model can be assumed known a priori. The literature has widely discussed these issues, particularly the problem of parameters and state estimation for lithium-ion batteries. These parameters, essential for reliable controller development, often need to be estimated from specially designed experiments [20], [21], [22], [23] and through the use of suitable observers for online state trajectory reconstruction [24]. However, these methods are intrinsically linked to the accuracy of the estimated electrochemical parameters, which may vary greatly, even among cells of the same type, and may change as the battery ages, often necessitating extensive and intrusive experiments. Furthermore, while stochastic control algorithms have been proposed in literature for battery management under uncertain parameters [25], [26], [27], their adoption in real-time settings is limited due to their high computational complexity. On the other hand, deep MPC approaches, which are well-suited for real-time applications due to their low online computational demand, still grapple with the challenges related to state and parameter estimation. In order to overcome this issue, the techniques proposed in [28] and [29] have adapted the use of deep predictive control to a more realistic scenario with knowledge of the system restricted to the model structure. In particular, an output-based algorithm has been proposed where only current, voltage, and temperature measurements are assumed to be available, with the battery parameters remaining unknown.

All the deep predictive controllers discussed above can be easily seen from the perspective of imitation learning, which is a machine learning framework where the goal is to mimic expert behavior without explicit knowledge of the underlying dynamics of the system [30]. In other words, the learning model seeks to imitate the actions of an expert, in this case, a battery charging strategy, based on the observations of its states and actions. Particularly, all the aforementioned methodologies can be seen as specific examples of behavioral cloning, a subtype of imitation learning which directly maps observations to actions, aiming to replicate the expert's policy, relying on supervised learning techniques, such as the use of a deep neural network as a regression model. However, behavioral cloning has its own limitations. One significant issue is the distributional shift, where the learning model might encounter states that are not present or are underrepresented in the training data. When the model begins to deviate from the expert's trajectory, it may continue to make errors that lead it into unfamiliar states, thereby exacerbating the initial mistake [31]. An illustrative example of this challenge can be found in [32], where behavioral cloning is utilized to learn a policy for car driving. Since the human demonstrations only encompassed instances of "good driving" without any crashes or near misses, the model encounters difficulties when errors arise. If the car strays from the demonstrated trajectories, the learner lacks the knowledge to recover due to the absence of such scenarios in the training data. Dataset Aggregation (DAGGER), has been proposed in [33] as a potential solution to this issue of distributional shift. DAGGER is an iterative algorithm designed to mitigate the accumulation of mistakes from the distributional shift. It operates by gradually blending the actions taken by the learning model and the expert policy, which prevents the model from venturing into unexplored state spaces. By repeatedly incorporating expert guidance during the learning process, DAGGER ensures that the learning model stays close to the expert's trajectory, thereby reducing errors due to the distributional shift.

The principal contribution of this paper lies in addressing the inherent challenges associated with traditional MPC and deep MPC strategies when faced with uncertain system parameters and unmeasurable internal battery states. Drawing on the imitation learning paradigm and, specifically, the DAGGER approach, this work offers a novel solution to battery charging under real-world constraints. This paper is the first to propose and test the use of dataset aggregation in the context of battery charging, with the aim of mitigating the distributional shift issue prevalent in behavioral cloning methods. The work's significant innovation lies in the adaptation of DAGGER to accommodate uncertain parameters and unobservable internal states, a situation referred to as an agnostic scenario in this context. The results derived from a realistic battery simulator implementing an electrochemical model demonstrate marked enhancements in battery charging performance, particularly in satisfying temperature constraints. This performance surpasses that of the existing algorithm proposed in [29], here considered as a benchmark, which applies a behavioral cloning approach grounded in supervised learning. These contributions underscore the potential of the imitation learning paradigm and, specifically, the DAGGER approach in addressing complex control problems in battery management systems. Adding to its contributions, this work, to the best of the available knowledge, represents the pioneering effort to integrate deep predictive control within the imitation learning framework. By unifying these two domains, it presents a novel perspective that broadens the horizons of potential solutions to the challenges encountered in battery charging.

The rest of the paper is structured as follows. In Section II, a thorough background on the battery charging problem and the model predictive control strategy is provided. The imitation learning paradigm, the DAGGER algorithm, and the proposed adaptation for uncertain parameters and unobservable states are introduced in Section III. Discussion on the outcomes of simulation experiments takes place in Section IV. The paper concludes in Section V, summarizing the key findings and suggesting potential directions for future research.

II. BATTERY CHARGING TASK

This section dives into the intricate landscape of battery charging. Firstly, Subsection II-A is introduced to provide a comprehensive depiction of a suitable battery model. This model is utilized throughout the paper, acting both as a simulator in the results section and as an informative tool to elucidate the principal variables and phenomena integral to the battery charging process. Through this model, readers gain an in-depth understanding of the parameters involved and how they interact with one another within the charging process.

Furthermore, Subsection II-B meticulously explores the multifaceted requirements intrinsic to batteries, encompassing various aspects such as battery chemistry, capacity, charging rate, and thermal management. These requirements, in their collective entirety, shape a complex landscape for the battery charging problem, instigating the necessity to explore advanced methods for efficient and optimal charging.

Finally, Subsection II-C revisits the concept of predictive control, forming the bedrock of cutting-edge algorithms employed in battery charging. In this context, rather than serving as a simple benchmark, the predictive control algorithm assumes the role of an ''expert agent''. It is the expert agent that the DAGGER-based approach, proposed in Section III of this paper, seeks to emulate within an imitation learning paradigm.

A. BATTERY MODEL

Among the different mathematical descriptions of a lithium-ion battery available in the literature, the simplified electrochemical model proposed in [34] has been largely adopted for battery control since it achieves a reasonable trade-off between accuracy and computational complexity [35], [36]. Such a model, known as Single-Particle Model (SPM), is obtained from the well-known Doyle-Fuller-Newman model [37] by considering the two electrodes as spherical particles. The accuracy of such a model was demonstrated in [38], among others. The model considered in this paper integrates the dual-state thermal dynamics outlined by the authors in [39], enabling the accurate depiction of thermal behaviors within the battery system.

For a more granular exploration of the model equations, readers are advised to consult the reference [19]. Here, only the primary variables that significantly influence the model are covered. Of primary importance is the battery's state of charge, $\mathrm{soc}(t) \in [0, 1]$, which evolves over time as per the following equation:

$$\frac{d\,\mathrm{soc}(t)}{d\,t} = \frac{I(t)}{3600C} \tag{1}$$

In this representation, t denotes time, $I(t)$ represents the current applied to the battery (with positive current indicating charging), and C symbolizes the battery's capacity in [Ah]. The state of charge reaches its maximum value of $\mathrm{soc}(t) = 1$ when the battery is fully charged, and falls to $\mathrm{soc}(t) = 0$ when completely discharged.

The voltage across the battery, represented by $V(t)$, is modeled by the equation:

$$V(t) = U_p(t) - U_n(t) + \eta_p(t) - \eta_n(t) + R_{sei}I(t) \tag{2}$$

where the $U_i(t)$ and $\eta_i(t)$ terms, for $i \in n, p$, account for open circuit potential and overpotential respectively, as outlined in [19]. The $R_{sei}I(t)$ term reflects the voltage drop across the Solid Electrolyte Interphase (SEI) resistance. In the aforementioned equations, it should be clarified that the subscript p is employed to denote parameters or conditions related to the cathode, while the subscript n is used to signify those corresponding to the anode of the battery.

Temperature dynamics within the battery are captured using the two-state model proposed in [39], which encompasses both the core and surface temperatures, represented by $T_c(t)$ and $T_s(t)$ respectively. The thermal dynamics are represented by:

$$C_c \frac{d\,T_c(t)}{d\,t} = Q(t) - \frac{T_c(t) - T_s(t)}{R_{c,s}} \tag{3a}$$

$$C_s \frac{d\,T_s(t)}{d\,t} = \frac{T_c(t) - T_s(t)}{R_{c,s}} - \frac{T_s(t) - T_{env}}{R_{s,e}} \tag{3b}$$

here, $R_{c,s}$ and $R_{s,e}$ are the thermal resistances between the core and the surface and between the surface and the environment, respectively. The heat capacities of the core and surface are indicated by C_c and C_s, while $Q(t)$ shows the generated heat, computed as:

$$Q(t) = |I(t)(V(t) - U_p(t) + U_n(t))|. \tag{4}$$

In this context, it's important to note that the nominal electrochemical parameters have been derived from the experimental characterization of a commercial cell, the Kokam SLPB 75106100, as detailed in [40] and [41]. The thermal parameters are based on those presented in [39].

B. BATTERY CHARGING REQUIREMENTS

This subsection delineates the optimal control problem related to fast battery charging. The problem is multidimensional, aiming not only to track the state of charge with minimal effort in terms of applied current, but also to fulfill safety considerations. Safety constraints are crucial in the context of battery charging, since charging a battery too quickly can lead to overheating and damage to the battery, or even safety risks. Therefore, constraints on voltage, temperature, and other factors are integral to the charging process.

The aforementioned scenario can be formulated more mathematically as a constrained optimization problem, with the solution representing the optimal charging protocol. Here, the battery is considered as a discrete-time system operating under a digital controller that applies piece-wise constant inputs at discrete times t_k, $k \in \mathbb{N}$, with a sampling time of t_s. In this context, the supplied current serves as the input variable, influencing various state variables such as the battery's state of charge, temperature, and voltage among others.

In particular, at a specific time instant t_k the sequence of optimal currents $\mathbf{I}^\star_{[t_k,\, t_{k+H}]}$ to be applied over the next H time steps $[t_k,\, t_{k+1},\, \ldots,\, t_{k+H-1}]$ can be retrieved by solving the following optimization problem:

$$\mathbf{I}^\star_{[t_k,\, t_{k+H}]} = \underset{\mathbf{I}_{[t_k, t_{k+H}]}}{\operatorname{argmin}} \quad q_{\text{soc}} \sum_{i=k+1}^{k+H} (\text{soc}(t_k) - \text{soc}_{\text{ref}})^2 + r \sum_{i=k}^{k+H-1} I(t_k)^2 \tag{5}$$

that for $i = k,\ k+1,\ \cdots,\ k+H-1$ is subject to:

$$\text{battery dynamics in (1)–(4)} \tag{6a}$$

$$I_{\min} \leq I(t_i) \leq I_{\max} \tag{6b}$$

$$\text{soc}_{\min} \leq \text{soc}(t_i) \leq \text{soc}_{\max} \tag{6c}$$

$$T_c(t_i) \leq T_{\max} \tag{6d}$$

$$T_s(t_i) \leq T_{\max} \tag{6e}$$

$$V(t_i) \leq V_{\max} \tag{6f}$$

TABLE 1. Minimum and maximum values for the main battery variables.

Parameter	Minimum	Maximum
Current	-10 A	10 A
Voltage	n.d.	4.2 V
State of Charge	0	1
Temperature	n.d.	313.15 K

where H is a prediction horizon, while the coefficients q_{soc} and r allow the designer to specify what is most important in the control problem: getting to the desired state quickly, minimizing control effort, or finding a balance between the two. Moreover, the intervals $[I_{\min},\ I_{\max}]$ and $[\text{soc}_{\min},\ \text{soc}_{\max}]$ represent the feasible regions for the applied current and for the state of charge, respectively, while $T_{\max}$ and $V_{\max}$ are the upper bounds for the temperature and voltage. For the subsequent analysis, the prediction horizon is established with a value of $H = 4$, and the time step is set at $t_s = 10s$. The weights for the cost function are configured as $q_{\text{soc}} = 1$ and $r = 10^{-6}$. The specific lower and upper bounds applied to the variables of interest in this research are detailed in Table 1. Notably, the reference state of charge (soc_{ref}) remains unspecified, given its dependency on individual charging preferences.

C. PREDICTIVE CONTROL FOR BATTERY CHARGING

Building upon the discussion in Subsection II-B, the formulation of an optimization problem such as the one described can be significantly enhanced through the adoption of a model predictive control approach. MPC methodologies have demonstrated considerable efficacy in handling nonlinear processes that are subject to input and state constraints, chiefly due to their adoption of a receding horizon framework [42]. In this context, at each time-step t_k, the MPC scheme computes the optimal control sequence $\mathbf{I}^\star_{[t_k,\, ,t_{k+H}]}$ over the prediction horizon H. This is achieved by solving the constrained optimization problem in (5) - (6), the cost function of which is dependent on the predictions offered by a mathematical model of the battery. Subsequently, in alignment with the receding horizon paradigm, only the first element $I^\star(t_k)$ of the resulting optimal input sequence is applied. The future optimal moves, although calculated, are discarded. This iterative process affords a high degree of flexibility and adaptability, making it particularly suitable for nonlinear and constrained systems.

At time-step t_k, the optimal action $I^\star(t_k)$ that the MPC algorithm applies is contingent on the parameters $\mathbf{p}$ of the battery model and the battery states vector $\mathbf{s}_{t_k}$. The parameters $\mathbf{p}$ are unique to each battery cell and can even vary over time due to factors such as aging. The states vector $\mathbf{s}_{t_k}$ encapsulates an array of macroscopic and electrochemical variables, crucial among which are the state of charge and the core temperature of the battery. These parameters and states collectively inform the MPC algorithm, enabling it to determine and implement the optimal charging action at each time-step. However, it is crucial to underscore that in a realistic scenario, the internal states and the electrochemical parameters are typically not directly available to the controller. Instead, only voltage, surface temperature, and current can be directly measured.

The optimal charging strategy applied by the available state-of-the-art methodologies in the literature under the assumption of states and parameters knowledge is therefore the policy $\pi^\star$ that maps the battery states and the battery parameters into the optimal current computed by the aforementioned MPC algorithm, i.e., :

$$\pi^\star : \; (\mathbf{s}_{t_k}, \; \mathbf{p}) \rightarrow I^\star(t_k). \tag{7}$$

Such optimal policy will be considered as the ''expert agent'' in the imitation learning framework proposed in this paper.

III. IMITATION LEARNING APPROACH

Despite the potential advantages, the application of MPC to battery charging is not without its limitations and challenges. Primarily, implementing MPC in real-time requires the solution of a constrained optimization problem at every time step, which can prove to be computationally expensive especially when dealing with large state spaces or complex dynamics. This issue becomes even more pressing when the control frequency is high, as in many battery charging applications. Furthermore, the performance of MPC is inherently dependent on the precision of the model parameters and the accuracy of the state measurements. This underlines the necessity of robust system identification and state estimation methodologies. However, in practical scenarios, acquiring precise measurements of the parameters and the states, especially the state of charge and core temperature among other internal variables, is seldom straightforward. These need to be often estimated from available measurements, which typically include voltage, applied current, and surface temperature. Compounding this challenge is the reality that these measurements are often subject to noise, potentially leading to inaccuracies in the estimated states and parameters. Therefore, while MPC presents a promising approach for battery charging, the practical implementation requires careful consideration and effective handling of these complexities.

In light of these complexities, this section proposes the employment of an imitation learning paradigm. This approach primarily strives to decrease the computational burden of the control algorithm and enhance the performance of battery charging under conditions where the battery parameters are indeterminable and states are unmeasurable directly. Particularly, the real-time operational demands of an imitation learning framework are markedly lower as it only requires the execution of a prediction step using a machine learning model instead of resolving an optimal control problem. Furthermore, the issues of parameters and states inaccuracy can be aptly addressed by adopting a variant of the imitation learning paradigm, inspired by techniques prevalent in the realm of Partially Observable Markov Decision Processes (POMDP) [43].

Subsection III-A delves into a conventional approach to imitation learning, which is rooted in the supervised learning paradigm. Despite its simplicity and intuitive appeal, this methodology is burdened with certain limitations, chief among them being the problem of distributional shift. This issue arises when the state distribution generated by the learner's policy diverges from the one produced by the expert's policy, often leading to a degradation in performance. To circumvent this predicament, we turn our attention to the DAGGER approach in Subsection III-B. This strategy effectively mitigates the adverse effects of the distributional shift, thereby improving the robustness and efficacy of the imitation learning process. Finally, the application of the DAGGER approach to the task of optimal battery charging and its adaptation for the case of uncertain states and parameters is presented in detail in Subsection III-C. This methodological adjustment opens up novel possibilities for achieving efficient and safe battery charging even in the face of system uncertainties.

A. SUPERVISED IMITATION LEARNING

Imitation learning is a machine learning approach aimed at mimicking expert behavior without explicit knowledge of the underlying dynamics of the system. The idea is that a machine learning model learns from demonstrations or examples provided by a knowledgeable or skilled agent (the expert), and attempts to replicate the behavior of this expert as closely as possible.

Specifically, imitation learning can be construed as a subclass of supervised learning, with the primary objective of establishing a policy, denoted $\pi_{\boldsymbol{\theta}}$, that faithfully replicates the behavior of a given expert policy $\pi^\star$. The policy $\pi_{\boldsymbol{\theta}} : \mathcal{S} \rightarrow \mathcal{A}$, parameterized by $\boldsymbol{\theta}$, is a functional mapping from a state space $\mathcal{S}$ to an action space $\mathcal{A}$, which can be realized through a machine learning model.

The ultimate aim of the imitation learning paradigm is to determine an optimal parameter vector $\hat{\boldsymbol{\theta}}$ such that the resulting policy $\pi_{\hat{\boldsymbol{\theta}}}$ mirrors the expert's strategy with high fidelity. This learning process revolves around considering pairs of states and actions $(s, a) \in \mathcal{S} \times \mathcal{A}$ which are collected by facilitating the interaction of the expert policy with either the environment or an appropriate environmental simulator

This can be viewed as a supervised learning problem, where a mapping from states (inputs to the machine learning model) to actions (targets) is learned, given a dataset $\mathcal{D} = \{(\mathbf{s}_k, \pi^\star(s_k))\}_{k=1}^N$ of state-action pairs collected by executing the expert policy. To align the behavior of the imitation learning policy with the expert, a loss function is minimized over the dataset $\mathcal{D}$. A common choice for the loss function is the mean square error between the policy $\pi_{\boldsymbol{\theta}}$ and the expert policy $\pi^\star$, given by

$$L(\boldsymbol{\theta}) = \frac{1}{N} \sum_{k=1}^{N} \left[\pi_{\boldsymbol{\theta}}(\mathbf{s}_k) - \pi^\star(\mathbf{s}_k)\right]^2. \tag{8}$$

This loss function encourages the imitation learning policy to output actions similar to those of the expert in the states encountered by executing the expert policy.

However, one significant limitation of this approach, often called behavioral cloning, is that it does not account for the distributional shift in the states encountered by the learned policy as compared to the expert policy. Specifically, distributional shift refers to the change in the distribution of states encountered by the learned policy as it starts to deviate from the expert policy. When training, the learned policy only has access to the states that the expert encounters, which represents a certain distribution in the state space. However, as soon as the learned policy starts acting in the environment and potentially makes mistakes, it can start encountering states that the expert would not, which may lie outside the distribution of states the model was trained on. This shift in distribution can lead to poor performance, as the learned policy has not been trained on how to act in these states. This is especially problematic because the probability of the learned policy encountering these off-distribution states can increase over time, as each mistake can lead to more unfamiliar states, a situation often referred to as ''compounding errors''.

B. DAGGER APPROACH

The challenge posed by the distributional shift in behavioral cloning has catalyzed the research community to devise effective countermeasures. Among these solutions, Dataset Aggregation has emerged as one of the most prevalent approaches.

Dataset Aggregation, also known as DAGGER, as proposed by the authors in [33], is an innovative algorithmic approach specifically tailored to counteract the notorious issue of distributional shift inherent in behavioral cloning. This iterative method combines the expertise of both the learned policy and the expert policy in an ingenious way, thereby enriching the training data over each iteration. In each iteration of the DAGGER algorithm, the learning system amasses state-action pairs from two sources: the current learned policy, denoted as $\pi_{\theta_{i-1}}$, and the expert policy, $\pi^\star$. These pairs are then integrated into the existing training set to create an augmented dataset, which serves as the foundation for the subsequent policy update. The newly acquired data is gathered under a mixed policy, π_i, that blends the actions from the expert policy and the current learned policy. The probability of choosing the expert policy, quantified as $\beta_i \in [0,\ 1]$, balances against the likelihood of choosing the learned policy, represented by $1-\beta_i$, i.e. the policy π_i is defined as:

$$\pi_i(\mathbf{s}_{t_k}) = \begin{cases} \pi^\star(\mathbf{s}_{t_k}), & \text{with probability } \beta_i \\ \pi_{\theta_{i-1}}(\mathbf{s}_{t_k}), & \text{with probability } 1-\beta_i \end{cases} \tag{9}$$

Over the course of iterations, the contribution of the expert policy typically reduces, allowing the learned policy to progressively take more autonomous decisions. Once the mixed policy data is collected, the next task is to update the learned policy. The updated policy π_{θ_i} is derived by minimizing the empirical loss in (8) on the aggregated dataset up to the i-th iteration. Note that, for each iteration i, the labels of the data added to the aggregated dataset correspond to the actions that the expert agent would execute in the state encountered by following the mixed policy π_i. As a result, the learned policy adjusts its behavior to more closely emulate the expert, using the rich, diverse set of state-action experiences gathered so far. An algorithmic description of the DAGGER methodology is provided in Algorithm 1.

This systematic, iteratively refined method offers a dynamic learning platform that gradually aligns the distribution of states experienced by the learned policy to the distribution of states the expert policy operates in. Through this process, the learned policy continually improves, both in terms of its ability to mimic the expert and in its competence to handle the wider state space. The resulting performance enhancement marks a considerable leap over the standard behavioral cloning approach, highlighting the effectiveness of DAGGER in tackling the challenge of distributional shift.

Algorithm 1 DAGGER Algorithm

```
Require: dataset $\mathcal{D}_0$, iterations number $n_D$, decay factor $\beta$
for $i = 1$ to $n_D$ do
    Train policy $\pi_{\theta_{i-1}}$ on $\mathcal{D}_{i-1}$
    Generate new dataset $\mathcal{D}_i$ by following policy $\pi_i$
    Aggregate the datasets: $\mathcal{D}_i = \mathcal{D}_i \cup \mathcal{D}_{i-1}$
end for
Return final policy $\pi_{\theta_{n_D}}$
```

C. DAGGER FOR AGNOSTIC BATTERY CHARGING

The exposition of the imitation learning paradigm thus far presumes full accessibility to the system's state vector **s**. Additionally, an implicit assumption made is the adherence to the Markov property, whereby the expert policy is solely contingent on the states, although it is well understood that, in the battery charging problem under consideration, the MPC strategy is also influenced by the battery parameters.

This nuanced situation, where only a fraction of system variables (tainted by Gaussian noise) are accessible and where the system's parameters may undergo changes, necessitates the development of an agnostic battery charging methodology. Such an approach should be capable of achieving satisfactory performance in terms of state of charge tracking and safety constraint satisfaction, irrespective of prior knowledge of the battery.

This challenge can be reconceptualized and addressed through an approach inspired by partially observable Markov decision processes [43]. In POMDPs, prior measurements (observations) are employed to practically reinstate the Markov property. A similar methodology can be adopted for the optimal charging task, thereby enhancing the robustness and adaptability of the proposed approach.

In the context of POMDPs, an agent doesn't have direct access to the underlying state $\mathbf{s} \in \mathcal{S}$ of the system but can instead make observations $\mathbf{o} \in \mathcal{O}$ that bear some statistical relationship to this state. Note that $\mathcal{O}$ represents all possible sensor readings that the agent can receive. This notion aligns well with the situation at hand where the complete state of the battery is not directly measurable, but only voltage and surface temperature are available.

To recover the Markov property, which assumes complete knowledge of the state, a history of observations can be used. The history at a particular time step t_k, represented by $\mathbf{h}_{t_k}$, consists of past measurements up to the current time step. Formally, it can be defined as:

$$\mathbf{h}_{t_k} = \{\mathbf{o}_\tau,\ \mathbf{a}_\tau\}_{\tau=t_0}^{t_k-1}, \tag{10}$$

where $\mathbf{o}_\tau$ and $\mathbf{a}_\tau$ are the observation and the action at time τ, respectively. The history $\mathbf{h}_{t_k}$ effectively encodes all available information up to time t_k. As such, an agent using the history to make decisions is adhering to the Markov property. In practice, only a sufficiently large window of previous measurements is considered, whose size is denoted by n_w:

$$\mathbf{h}_{t_k}^{n_W} = \{\mathbf{o}_\tau,\ \mathbf{a}_\tau\}_{\tau=t_{k-n_W}}^{t_k-1}. \tag{11}$$

With this adaptation, the optimal charging problem can be reframed as a POMDP, allowing us to effectively handle the uncertainties associated with the battery's parameters and states.

In fact, utilizing a window of past measurements enables the model to effectively filter out Gaussian noise, reconstruct the system state, and potentially recover the system dynamics that need to be controlled. Indeed, this approach is feasible provided that a sufficiently large window of prior measurements is available and a machine learning model with a suitable level of complexity is utilized to represent the policy. The machine learning model thereby learns to interpret the window of past observations and leverages this knowledge to make effective control decisions, in spite of initial system uncertainties and noise.

For the case of the battery charging the available measurements are represented by voltage and surface temperature, and therefore the vector of observations $\mathbf{o}_{t_k}$ at a particular time-step t_k is given by:

$$\mathbf{o}_{t_k} = [V(t_k),\ T_s(t_k)] \tag{12}$$

while the action a_{t_k} is represented by the applied current $I(t_k)$.

To conclude, it's crucial to note that in the optimal battery charging scenario, the actions of the expert agent are also influenced by the chosen reference for the state of charge (soc_{ref}). This reference can vary based on the user's preferences. As a result, we incorporate the reference information into the historical data utilized by the agnostic DAGGER algorithm for decision-making, given that this information is generally assumed to be accessible to the controller.

IV. EXPERIMENTS AND RESULTS

This section presents a comprehensive account of the experimental procedures undertaken to substantiate the validity of the proposed methodology.

The foundational pillar of these experiments is the mathematical model described in II-A, which serves as a realistic simulator of a lithium-ion battery. Utilizing this model brings forth numerous advantages, primarily allowing the synthetic generation of a dataset, based on the expert agent's strategy, for the training of both the behavioral cloning policy and the DAGGER-based policy. It's important to note that the expert agent's operation necessitates the exact availability of both states and parameters, hence precluding its execution on a real battery. Furthermore, the use of this simulator facilitates the generation of data encompassing various battery parameters and initial conditions, a task that would be prohibitively costly with real batteries. The core assumption underpinning the ideas presented in this paper is the belief that the analytical structure of the model used for battery simulation is sufficiently accurate to capture real-world phenomena. Compared to the requirement of precise parameters and state measurements, this is a relatively mild assumption. Should this assumption hold true, the policy derived from the application of the DAGGER algorithm can be seamlessly implemented on a real battery with parameters within the range considered during the data generation phase.

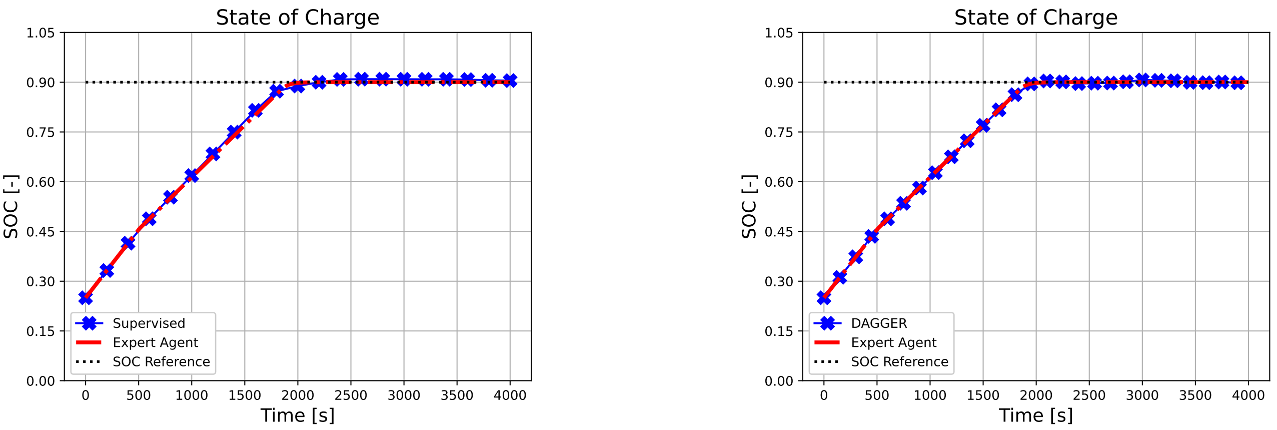
Subsection IV-A elaborates on the process of synthetic data generation and the training phase of the deep learning model employed within the DAGGER framework. Moreover, it provides insights into the training of an algorithm based on the behavioral cloning approach proposed in [29], which serves as a benchmark in this study. Subsequently, Subsection IV-B highlights the limitations of the benchmark approach by focusing on a particular scenario, and presents how effectively the DAGGER methodology addresses these issues. In Subsection IV-C, a comprehensive comparison is made between the performance of the DAGGER approach and the supervised benchmark across a range of scenarios. This comparative analysis primarily focuses on the algorithms' abilities to emulate the expert agent's behavior under various state conditions and their capability to adhere to voltage and temperature constraints.

The experimental results underscore that the implementation of a DAGGER-based algorithm for imitation learning presents a significant improvement over the existing battery charging protocols, particularly in scenarios characterized by uncertainties in states and parameters.

A. DATASET GENERATION AND TRAINING PHASE

In the following, the framework to generate the synthetic dataset required for the DAGGER training is discussed in detail.

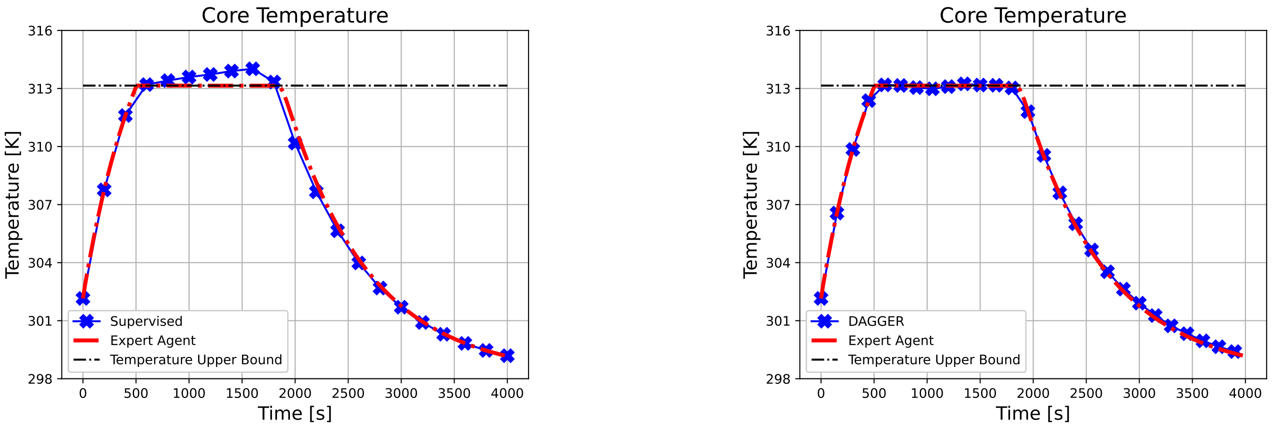
Firstly, it is fundamental to describe the generation of the initial dataset $\mathcal{D}_0$ which appears to be a necessary requirement as stated in Algorithm 1. Entries within this initial dataset



(a) Behavioral Cloning.

(b) DAGGER.

FIGURE 1. This figure showcases the progression of the battery's state of charge during the charging process, as dictated by the expert agent, and imitated by the behavioral cloning method and the DAGGER approach. The state of charge evolution dictated by the expert agent is represented by the dash-dotted red line, while the crossed blue lines depict the trajectories followed by the two imitating agents. As the figure clearly shows, both the behavioral cloning approach (left) and the DAGGER approach (right) manage to closely follow the expert agent's state of charge trajectory, demonstrating the efficiency of both methods in tracking the desired state of charge. However, as seen in other figures, it is essential to consider this result in conjunction with the ability to respect other operational constraints, where significant differences between the two imitating methodologies emerge.

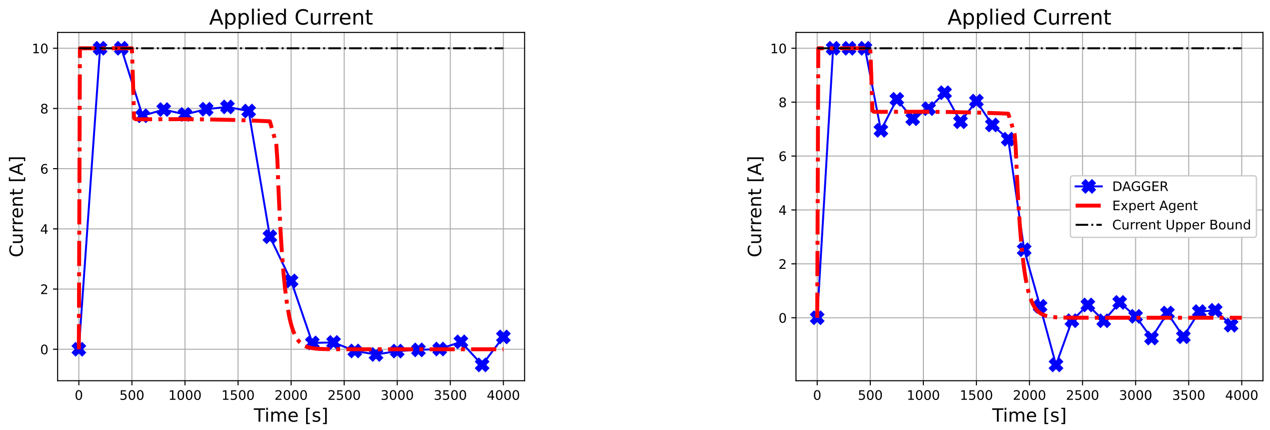


(a) Behavioral Cloning.

(b) DAGGER.

FIGURE 2. This figure portrays the evolution of the battery core temperature during the charging process, under the guidance of the expert agent and the two imitating agents, namely the behavioral cloning method and the DAGGER approach. The dash-dotted red line represents the expert agent's temperature profile, while the crossed blue lines illustrate the imitating agents' temperature profiles. On the left, the behavioral cloning approach fails to maintain the temperature constraint adequately due to the distributional shift phenomenon, indicating a significant deviation from the expert agent's actions when operating outside the learned trajectory. On the other hand, the DAGGER approach, presented on the right, exhibits a remarkable degree of adherence to the temperature constraint, deviating only minimally, which underscores its robustness in maintaining operational safety of the battery. This figure provides a comparative representation of the temperature management strategies employed by the two imitation learning methods, emphasizing their distinct capabilities and limitations in emulating the expert agent's control actions."

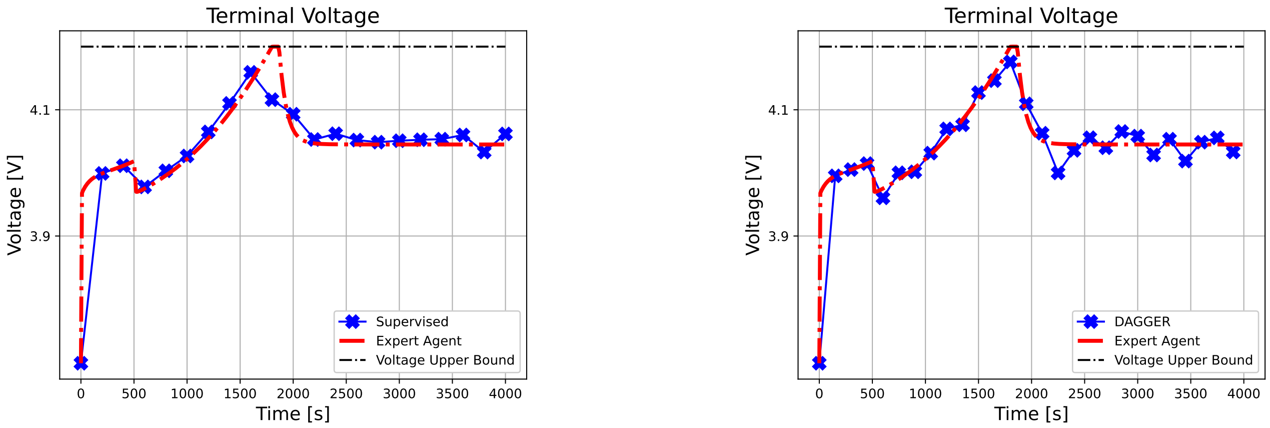
represent measurements collected when the expert agent interacts with the battery simulator. The expert agent is represented by an MPC controller, fulfilling the control task as specified in Subsection II-C. It's crucial to recall that under this setup, both the battery parameters and battery states are presumed to be directly accessible to the expert agent. This represents an omniscient scenario where the expert agent is privy to complete and accurate knowledge of both the states and parameters of the battery. Such a presumption is reasonable in this context given that the data are synthetically generated using the battery simulator. The initial dataset $\mathcal{D}_0$ is generated through the execution of the expert agent over a span of 500 episodes. Each episode consists of 200 time-steps, with a sample time of 10 seconds. For every time-step t_k, the available measurements – including voltage, surface temperature, applied current – along with the optimal current value computed by the expert agent, and the reference for the state of charge for the j-th episode, are collected. To incorporate a window of historical measurements of size n_w, the database $\mathcal{D}_0$ is structured in such a way that the



(a) Behavioral Cloning.

(b) DAGGER.

FIGURE 3. This figure illustrates the current applied over time by the expert agent and the two imitating agents, namely, the behavioral cloning method and the DAGGER approach. The expert agent's profile is signified by a dash-dotted red line, while the imitating agents' profiles are denoted by crossed blue lines. The behavioral cloning method is demonstrated on the left side of the figure, with the DAGGER approach displayed on the right. A noteworthy aspect of the DAGGER agent's current profile is its characteristic "chattering", a series of rapid fluctuations that allows the agent to respond quickly and flexibly to the state variations. This layout offers a clear side-by-side comparison of the current profiles generated by these two imitation learning strategies, underscoring their respective endeavors to emulate the expert agent's control decisions.



(a) Behavioral Cloning.

(b) DAGGER.

FIGURE 4. This figure depicts the voltage profiles of the battery over time, as controlled by the expert agent and the two imitating agents, namely the behavioral cloning method and the DAGGER approach. The expert agent's profile is represented by a dash-dotted red line, while the profiles of the imitating agents are marked by crossed blue lines. The behavioral cloning method is showcased on the left, while the DAGGER approach is presented on the right. This configuration facilitates a clear comparison of how these two imitation learning approaches manage to maintain voltage within the safety constraints and replicate the behavior of the expert agent.

k-th row r_k represents the current measurements as well as the historical ones up to the window size n_w. In other words, the data is reshaped as follows:

$$\begin{aligned} r_k = [&V(t_{k-n_w}),\ T(t_{k-n_w}),\ I(t_{k-n_w}), \\ &\ldots,\ V(t_k),\ T(t_k),\ I(t_k),\ \mathrm{soc}_{\mathrm{ref},j},\ I^{\star}(t_k)]. \end{aligned} \tag{13}$$

where $I^{\star}(t_k)$ is the target variable.

This structure provides an extensive historical context for each time-step, equipping the DAGGER algorithm with the necessary information to make effective decisions based on past observations. This is particularly crucial as, in accordance with the discussion in Subsection III-C, the DAGGER algorithm does not have direct access to the internal states of the battery.

The synthetic nature of the data, generated using a battery simulator, allows for the random sampling of batteries with different parameters for each episode, thus enriching the diversity of the dataset. This varied dataset, representing a broad spectrum of battery conditions, enables the DAGGER agent to discern patterns related to the battery parameters from within the window of historical measurements. Consequently, it facilitates a more comprehensive and generalized understanding of assorted battery conditions, thereby augmenting the efficacy of the learned policy. Specifically, for each episode j, the initial battery conditions are drawn from a uniform distribution: $\mathrm{soc}(t_0) \sim \mathcal{U}(0,\ 1)$ and $T_s(t_0),\ T_c(t_0) \sim \mathcal{U}(298.15\,K,\ 313.15\,K)$. Likewise, the reference state of charge for the j-th episode is randomly sampled as $\mathrm{soc}_{\mathrm{ref},j} \sim \mathcal{U}(0.7,\ 1)$. Lastly, to ensure the algorithm is resilient to changes in parameters during the battery aging process, also the battery capacity and the SEI resistance are sampled from a uniform distribution, leading to a unique parameter vector $\mathbf{p}_j$ for each simulation: $C_j \sim U(5.5\,Ah,\ 8\,Ah)$ and $R_{sei,j} \sim U(14\,m\Omega,\ 19\,m\Omega)$. It's important to note that the pair $C = 8\,Ah$ and $R_{sei} = 14\,m\Omega$ represents a newly manufactured battery, whereas a pair $C = 5.5\,Ah$ and $R_{sei} = 19\,m\Omega$ symbolizes a battery at the end of its life.

Building upon the methodology outlined, each episode is augmented by the inclusion of an additional 30 steps at the beginning, during which no current is applied. This approach enables the prediction of optimal current even at the onset, when the battery initiates from a resting state. It effectively obviates the need for an initial estimate of the first n_w control actions, a limitation inherent in the algorithm proposed in [29]. In this earlier approach, a guess of control actions was necessitated due to the predictor's requirement for a complete window of n_w measurements before producing accurate predictions. Conversely, the proposed algorithm in this paper is capable of generating accurate predictions from the outset, irrespective of the battery's initial state. This improvement is facilitated by the inclusion of state-action pairs corresponding to a resting battery within the training dataset. Consequently, the learning algorithm is equipped to recognize and effectively handle such scenarios, thereby enhancing its predictive capabilities from the commencement of the control process. As a result, this expanded dataset, representative of a broader array of battery states, contributes to an increase in the robustness and generalization capability of the learned policy.

The initial dataset, $\mathcal{D}_0$, serves as the training input for the policy π_{θ_0}, which is then employed in tandem with the expert agent's policy during the initial iteration of the DAGGER algorithm (refer to Algorithm 1). For any generic i-th iteration, the policy π_i is enacted across 100 episodes, each episode initialized as previously outlined, to compile the dataset $\mathcal{D}_i$. The process is repeated for $n_D = 15$ iterations, ultimately yielding an aggregate dataset containing 2000 episodes and the final policy $\pi_{\theta_{15}}$. The initial decay ratio β_0 is taken equal to 1, and it is decreased exponentially according to the following rule $\beta_i = 0.5\beta_{i-1}$.

The machine learning model employed for policy parameterization is consistent with the one utilized in [29]. This model is a Recurrent Neural Network (RNN), selected for its inherent ability to process sequential data, a characteristic integral to the dataset samples in this study. The architecture of the chosen model comprises four Long Short-Term Memory (LSTM) hidden layers, each layer containing 128, 64, 32, and 16 neurons, respectively. Additionally, the model includes four fully connected hidden layers, two of which contain 100 neurons each, one with 50 neurons, and the final layer with 10 neurons. The window size for historical measurements, denoted as n_w, is set to 20. The selection of this model and its configuration is premised on its proven efficacy in handling the task at hand, whilst maintaining computational efficiency. It is noteworthy that the model under consideration employs a Rectified Linear Unit (ReLU) activation function in the hidden layers and a hyperbolic tangent (tanh) activation function in the output layer. The tanh activation function in the final layer serves to constrain the network's output within a specific interval. Within the context of battery charging, this corresponds to maintaining the optimally applied current within its operating range. To further improve the learning performance of the model, a preprocessing pipeline, encompassing both scaling and standardization of the dataset's features, is utilized.

The deep learning model training process employed in this study utilizes the stochastic gradient descent method, specifically adopting the Adam optimizer. The mean squared error serves as the chosen loss function, with a learning rate configured to 5×10^{-4}. It's important to note that Gaussian noise is introduced to the features in the training set for two primary reasons: to deter overfitting and to increase the model's robustness in real-world scenarios where measurement disturbances can be introduced due to faulty or imprecise sensors. In this context, Gaussian noise with a standard deviation of 20 mV for voltage and 1 K for temperature was deemed suitable.

1) TRAINING OF THE BENCHMARK

To demonstrate the improvements offered by the DAGGER algorithm, a direct comparison is conducted with a traditional supervised learning approach. Specifically, the machine learning model utilized within the DAGGER framework is also trained following the methodology outlined in [29]. The rationale for choosing this particular supervised method as a benchmark is that the RNN with the specified layer configuration has been demonstrated in [29] to be among the best supervised learning approaches tried for similar tasks. For this comparison, a distinct dataset, composed of 2000 episodes, is generated solely through the implementation of the expert agent's policy. This comprehensive dataset encompasses a wide range of scenarios, capturing the expert agent's optimal behavior across diverse conditions. The comparison between the performance of the traditional supervised learning method, here serving as a benchmark, and the proposed DAGGER-based approach is detailed in Subsections IV-B and IV-C. This comparative analysis aims to highlight the strengths of the DAGGER algorithm in handling uncertainties and variations in battery conditions effectively, thereby underscoring its robustness and practicality.

B. DAGGER'S PERFORMANCE

The ensuing discussion is devoted to examining a specific simulation, with the intent to underscore potential issues associated with the inherent occurrence of a distributional shift during the implementation of the supervised learning approach. This focus allows scientists to explore the challenges and limitations of behavioral cloning when confronted with distributional changes, a key aspect of real-world operations. Further, it offers an opportunity to demonstrate how the DAGGER methodology effectively addresses these issues.

Consider, as an example, the task of charging a battery from an initial state of charge of 25% to a target one of 90%. The battery begins with an initial temperature of 302.5K, both at the core and on the surface, and is in a resting condition at the onset of the charging process. With respect to the aging level of the battery, it is characterized by a solid electrolyte interface resistance of 0.0165 Ω and a capacity of 6.75 Ah. It's required that the constraints outlined in Table 1 are complied with throughout the entire charging procedure. The process is deemed complete after a duration of 4000 seconds.

In Figures 1, 2, 3, and 4, the evolution of the main battery variables is displayed under both the behavioral cloning and DAGGER methodologies. These are represented by a crossed blue line (on the left and right respectively), juxtaposed against the performance of the expert agent, which is indicated by a dash-dotted red line. This visual comparison provides a clear illustration of the performance differences between the methods, and the expert agent.

Turning attention initially to the outcomes produced by the supervised learning approach, depicted in the left portion of the aforementioned figures, successful tracking of the desired state of charge is observed, as illustrated in Figure 1a. However, a breach of the temperature constraint is noticeable in Figure 2a. The principal observation to emphasize is not merely the constraint violation itself-which, if substantial, could compromise battery safety-but rather, the fact that the behavioral cloning methodology fails to respond to this violation by adjusting the trajectory of the applied current (see Figure 3a). This lack of adaptability is a clear manifestation of the distributional shift: having only been trained on the expert agent's trajectories, the behavioral cloning algorithm has not learned how to adjust its behavior outside of the safety region since the expert agent never breaches these constraints during training.

The DAGGER approach is specifically devised to mitigate such issues. In particular, examining the right sections of the above-mentioned figures, where the same battery charging simulation is carried out using the DAGGER-based algorithm, it is apparent that the temperature constraint, as seen in Figure 2b, is nearly fully adhered to. Exceptions are minimal violations, which are unavoidable in such a realistic scenario where both battery states and parameters are presumed unavailable.

In its operation, the DAGGER-based agent utilizes a chattering current profile (refer to Figure 3b), sustaining an average value similar to that of the expert agent. This chattering behavior can be construed as the agent's countermeasure to minor deviations in the temperature constraint or state of charge, and is integral for precise tracking of both. Essentially, this can be traced back to two main factors: firstly, the expert agent itself adopts an assertive control approach, willing to significantly vary the applied input to achieve rapid battery charging whilst upholding safety parameters; secondly, the introduction of Gaussian noise in the window of historical measurements could potentially distort the agent's interpretation of the prevailing battery states, thereby causing swift oscillations between conservative and aggressive actions.

In conclusion, it is noteworthy that both approaches – the supervised learning and the DAGGER algorithm – manifest similar efficiency when handling voltage constraints. This is evident in Figure 4, where the pronounced chattering of the DAGGER algorithm's profile is primarily attributable to the high-frequency components present in the applied current, as previously discussed. Importantly, this chattering does not impart any significant adverse impact on the overall battery charging process.

Not only does this comparative study help validate the effectiveness of the DAGGER algorithm, but it also provides invaluable insights into the extent of improvements it offers in the context of imitation learning for optimal battery charging.

C. STATISTICAL VALIDATION OF DAGGER ALGORITHM

In order to statistically validate the performance of the methodology proposed in this paper, both the DAGGER and the behavioral cloning approaches are executed over a series of 100 different simulations. The varying simulation conditions consist of randomly selected battery parameters and initial states, chosen according to the same distribution that was utilized for the generation of the synthetic training dataset for the deep learning models. This rigorous and comprehensive testing seeks to underscore the superiority of the DAGGER-based approach in emulating the expert agent across an expansive range of battery conditions, particularly when contrasted against the performance of the traditional supervised learning procedure.

In a primary component of the conducted analysis, the disparity between the actions executed by the two imitation agents (the DAGGER and the behavioral cloning agents) and those actions that the expert agent would employ under identical conditions (battery states) is examined. It is noted that, in this context, the expert agent is assumed to have full knowledge of both the states and parameters of the battery, while the two imitation approaches operate under the assumption of agnosticism with regard to these variables.

The distribution of the discrepancies in the imitation of the expert agent's applied current, as demonstrated by both methodologies under consideration, is illustrated in Figure 5. From the figure, it can be seen that a well-shaped, Gaussian-like distribution is exhibited by the DAGGER framework, with a mean close to zero (precisely $-0.03A$) and a standard deviation of $2.66A$. In contrast, a bimodal distribution is observed for the traditional behavioral cloning approach, with a mean of $1.76A$ and a variance of $4.76A$. The presence of one peak centered around zero is noted, while the other peak around 7.5 A is identified as a

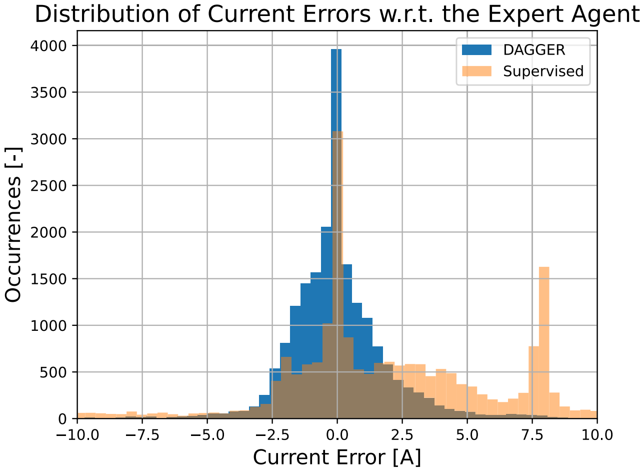


FIGURE 5. Comparison of current error distributions with respect to the expert agent between the DAGGER approach (depicted in blue) and the behavioral cloning methodology (shown in orange). The DAGGER approach exhibits a well-shaped Gaussian-like distribution around near-zero mean, indicating its superior ability to mimic the expert agent's actions across a variety of battery states. In contrast, the behavioral cloning method presents a bimodal distribution, reflective of its struggles with the distributional shift problem. This figure underscores the benefits of employing the DAGGER methodology over behavioral cloning for imitation learning in the context of battery charging.

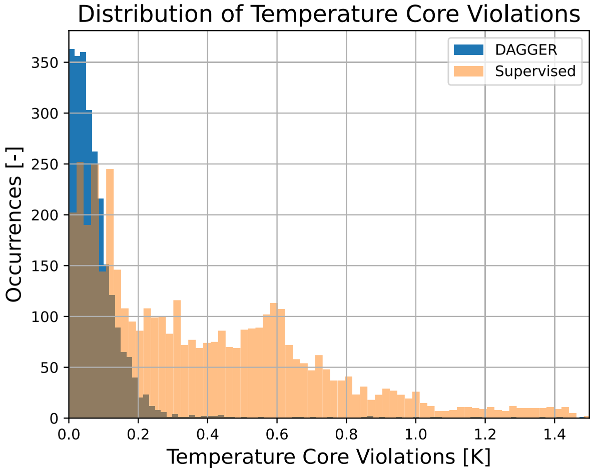


FIGURE 6. The graph illustrates the distributions of temperature constraint violations for both the DAGGER approach (shown in blue) and the behavioral cloning method (depicted in orange). Each violation represents an instance where the core temperature of the battery exceeded the specified safe limit. The distributions are derived from the outcomes of 100 distinct simulations. Notably, the DAGGER approach displays a tighter distribution, centered around a lower mean violation, indicating a superior ability to maintain the battery temperature within the designated safety limits.

key contributor to the distributional shift. Specifically, this corresponds to the errors committed by the behavioral cloning agent when the battery is outside the safety limits. In these conditions, while the expert agent would apply zero current, the supervised methodology continues to implement what it incorrectly perceives as the most probable current necessary to track the boundary of the feasible temperature range.

Additionally, the distribution of temperature violations, as displayed in Figure 6, offers noteworthy insights. Specifically, instants within the 100 simulations where the core temperature of the battery exceeds the specified boundary are considered. Out of a total of 20,000 time-steps, the DAGGER-based approach resulted in over-limit instances for 2,500 time-steps, while the supervised approach did so for 4,340 time steps. More intriguing, however, is the average violation: for the DAGGER approach, the mean violation is $0.08\,K$ with a standard deviation of $0.11\,K$, while for the supervised approach, the mean violation is significantly higher at $0.4\,K$ with a standard deviation of $0.35\,K$. This is further illustrated in Figure 6, where it can be observed that the distribution of the behavioral cloning approach has a significantly longer tail compared to that of the DAGGER approach.

Moreover, in terms of violations of the voltage limit, both algorithms exhibit similar behavior. Specifically, for the DAGGER approach, the number of instances of violation stands at 975, with a mean violation of $16.5\,mV$ and a standard deviation of $30.7\,mV$. For the behavioral cloning approach, the number of instances of violation is slightly higher at 1089, with a mean violation of $11.4\,mV$ and a standard deviation of $25.8\,mV$. Despite the slight differences, it's important to highlight that both algorithms, to a substantial degree, effectively manage the constraints on voltage, showcasing their capacity to handle this aspect of battery charging proficiently.

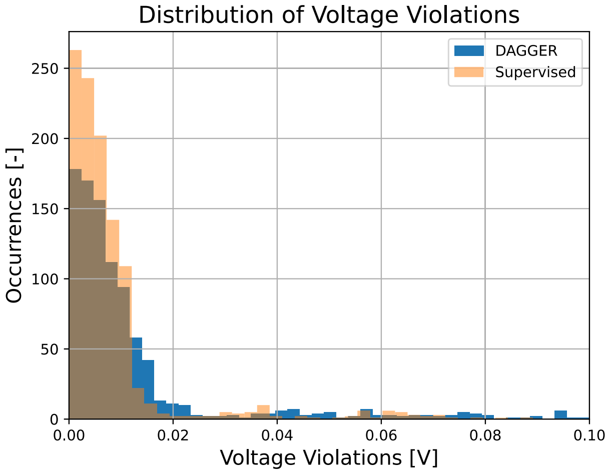


FIGURE 7. This plot represents the distributions of voltage constraint violations observed in the DAGGER approach (denoted in blue) and the behavioral cloning method (highlighted in orange). Each data point signifies an occasion when the battery voltage transgressed the pre-set safety bounds. The data are compiled from 100 separate simulations. Remarkably, both methodologies exhibit comparable distributions, suggesting similar effectiveness in adhering to the voltage safety limits.

Lastly, both the DAGGER and the behavioral cloning methodologies showcase comparable and noteworthy capability in adhering to the reference state of charge, demonstrating their potential and efficacy in maintaining the desired state of charge trajectory.

V. CONCLUSION

This work has presented a methodology for optimal battery charging, leveraging the Dataset Aggregation (DAGGER) algorithm within an imitation learning paradigm. A modified version of DAGGER has been proposed to adapt it to the charging problem where states and battery parameters are not directly available, introducing a window of historical measurements for effectively capturing the system dynamics and acknowledging the stochastic nature of the available observations.

The proposed methodology has been compared with a benchmark method based on supervised learning (behavioral cloning) in a rigorous simulation environment, which mimics realistic battery operations. The results demonstrate the superiority of the DAGGER-based approach in imitating the expert agent across diverse battery conditions, especially when contrasted with the traditional supervised learning methodology. Particularly, the DAGGER algorithm demonstrates improved constraint handling and exhibited superior robustness against uncertainties, thus proving to be a reliable and safer charging strategy. The supervised learning approach, although proficient in tracking the desired state of charge, suffers from the distributional shift issue, as it fails to ensure adherence to safety constraints under different state conditions. The DAGGER algorithm, on the other hand, mitigates this issue effectively, maintaining a balance between performance and safety across various scenarios. These findings illustrate the potential of applying advanced imitation learning techniques, such as DAGGER, for optimal control problems in situations characterized by unmeasurable states and uncertain parameters, providing a promising direction for future research in the field of battery management systems.

Future work will be dedicated to further improving the proposed methodology, either by combining it with uncertainty quantification techniques to provide a robust measure of the performance, considering the variability of the battery parameters and states, or by exploring real-time adaptations of the DAGGER algorithm to enhance its applicability in dynamic environments.

REFERENCES

[1] E. J. Cairns and P. Albertus, "Batteries for electric and hybrid-electric vehicles," *Annu. Rev. Chem. Biomol. Eng.*, vol. 1, pp. 299–320, Jul. 2010.

[2] L. Lu, X. Han, J. Li, J. Hua, and M. Ouyang, "A review on the key issues for lithium-ion battery management in electric vehicles," *J. Power Sources*, vol. 226, pp. 272–288, Mar. 2013.

[3] N. A. Chaturvedi, R. Klein, J. Christensen, J. Ahmed, and A. Kojic, "Algorithms for advanced battery-management systems," *IEEE Control Syst. Mag.*, vol. 30, no. 3, pp. 49–68, Jun. 2010.

[4] E. F. Camacho and C. B. Alba, *Model Predictive Control*. Berlin, Germany: Springer, 2013.

[5] R. Klein, N. A. Chaturvedi, J. Christensen, J. Ahmed, R. Findeisen, and A. Kojic, "Optimal charging strategies in lithium-ion battery," in *Proc. Amer. Control Conf. (ACC)*, 2015, pp. 382–387.

[6] M. Torchio, N. A. Wolff, D. M. Raimondo, L. Magni, U. Krewer, R. B. Gopaluni, J. A. Paulson, and R. D. Braatz, "Real-time model predictive control for the optimal charging of a lithium-ion battery," in *Proc. Amer. Control Conf. (ACC)*, 2015, pp. 4536–4541.

[7] C. Zou, C. Manzie, and D. Nešić, "Model predictive control for lithium-ion battery optimal charging," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 947–957, Apr. 2018.

[8] A. Pozzi, M. Torchio, and D. M. Raimondo, "Assessing the performance of model-based energy saving charging strategies in Li-ion cells," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, Aug. 2018, pp. 806–811.

[9] C. Speltino, D. Di Domenico, G. Fiengo, and A. Stefanopoulou, "Comparison of reduced order lithium-ion battery models for control applications," in *Proc. 48th IEEE Conf. Decis. Control (CDC) Held Jointly 28th Chin. Control Conf.*, Dec. 2009, pp. 3276–3281.

[10] Z. Chu, G. L. Plett, M. S. Trimboli, and M. Ouyang, "A control-oriented electrochemical model for lithium-ion battery, Part I: Lumped-parameter reduced-order model with constant phase element," *J. Energy Storage*, vol. 25, Oct. 2019, Art. no. 100828.

[11] G. Galuppini, M. D. Berliner, H. Lian, D. Zhuang, M. Z. Bazant, and R. D. Braatz, "Efficient computation of safe, fast charging protocols for multiphase lithium-ion batteries: A lithium iron phosphate case study," *J. Power Sources*, vol. 580, Oct. 2023, Art. no. 233272.

[12] A. Alessio and A. Bemporad, "A survey on explicit model predictive control," in *Nonlinear Model Predictive Control*. Cham, Switzerland: Springer, 2009, pp. 345–369.

[13] B. Karg and S. Lucia, "Efficient representation and approximation of model predictive control laws via deep learning," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3866–3878, Sep. 2020.

[14] T. Parisini and R. Zoppoli, "A receding-horizon regulator for nonlinear systems and a neural approximation," *Automatica*, vol. 31, no. 10, pp. 1443–1451, Oct. 1995.

[15] A. Bemporad, A. Oliveri, T. Poggi, and M. Storace, "Ultra-fast stabilizing model predictive control via canonical piecewise affine approximations," *IEEE Trans. Autom. Control*, vol. 56, no. 12, pp. 2883–2897, Dec. 2011.

[16] L. H. Cseko, M. Kvasnica, and B. Lantos, "Explicit MPC-based RBF neural network controller design with discrete-time actual Kalman filter for semiactive suspension," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 5, pp. 1736–1753, Sep. 2015.

[17] S. Chen, K. Saulnier, N. Atanasov, D. D. Lee, V. Kumar, G. J. Pappas, and M. Morari, "Approximating explicit model predictive control using constrained neural networks," in *Proc. Annu. Amer. Control Conf. (ACC)*, 2018, pp. 1520–1527.

[18] M. Hertneck, J. Köhler, S. Trimpe, and F. Allgöwer, "Learning an approximate model predictive controller with guarantees," *IEEE Control Syst. Lett.*, vol. 2, no. 3, pp. 543–548, Jul. 2018.

[19] A. Pozzi, S. Moura, and D. Toti, "Deep learning-based predictive control for the optimal charging of a lithium-ion battery with electrochemical dynamics," in *Proc. IEEE Conf. Control Technol. Appl. (CCTA)*, Aug. 2022, pp. 785–790.

[20] S. Park, D. Kato, Z. Gima, R. Klein, and S. Moura, "Optimal experimental design for parameterization of an electrochemical lithium-ion battery model," *J. Electrochemical Soc.*, vol. 165, no. 7, pp. A1309–A1323, 2018.

[21] A. Pozzi, G. Ciaramella, K. Gopalakrishnan, S. Volkwein, and D. M. Raimondo, "Optimal design of experiment for parameter estimation of a single particle model for lithium-ion batteries," in *Proc. 51st IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 6482–6487.

[22] A. Pozzi, X. Xie, D. M. Raimondo, and R. Schenkendorf, "Global sensitivity methods for design of experiments in lithium-ion battery context," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 7248–7255, 2020.

[23] G. Galuppini, M. D. Berliner, D. A. Cogswell, D. Zhuang, M. Z. Bazant, and R. D. Braatz, "Nonlinear identifiability analysis of multiphase porous electrode theory-based battery models: A lithium iron phosphate case study," *J. Power Sources*, vol. 573, Jul. 2023, Art. no. 233009.

[24] W. Waag, C. Fleischer, and D. U. Sauer, "Critical review of the methods for monitoring of lithium-ion batteries in electric and hybrid vehicles," *J. Power Sources*, vol. 258, pp. 321–339, Jul. 2014.

[25] A. Pozzi and D. M. Raimondo, "Stochastic model predictive control for optimal charging of electric vehicles battery packs," *J. Energy Storage*, vol. 55, Nov. 2022, Art. no. 105332.

[26] S. J. Moura, J. L. Stein, and H. K. Fathy, "Battery-health conscious power management in plug-in hybrid electric vehicles via electrochemical modeling and stochastic control," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 3, pp. 679–694, May 2013.

[27] Q. Zhang, W. Deng, and G. Li, "Stochastic control of predictive power management for battery/supercapacitor hybrid energy storage systems of electric vehicles," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 3023–3030, Jul. 2018.

[28] A. Pozzi, S. Moura, and D. Toti, "A deep learning-based predictive controller for the optimal charging of a lithium-ion cell with non-measurable states," *Comput. Chem. Eng.*, vol. 173, May 2023, Art. no. 108222.

[29] A. Pozzi, E. Barbierato, and D. Toti, "Optimizing battery charging using neural networks in the presence of unknown states and parameters," *Sensors*, vol. 23, no. 9, p. 4404, Apr. 2023.

[30] A. Hussein, M. M. Gaber, E. Elyan, and C. Jayne, "Imitation learning: A survey of learning methods," *ACM Comput. Surveys*, vol. 50, no. 2, pp. 1–35, Mar. 2018.

[31] S. Ross and D. Bagnell, "Efficient reductions for imitation learning," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 661–668.

[32] J. Togelius, R. De Nardi, and S. M. Lucas, "Towards automatic personalised content creation for racing games," in *Proc. IEEE Symp. Comput. Intell. Games*, Jul. 2007, pp. 252–259.

[33] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 627–635.

[34] S. Santhanagopalan, Q. Guo, P. Ramadass, and R. E. White, "Review of models for predicting the cycling performance of lithium ion batteries," *J. Power Sources*, vol. 156, no. 2, pp. 620–628, Jun. 2006.

[35] H. E. Perez, X. Hu, and S. J. Moura, "Optimal charging of batteries via a single particle model with electrolyte and thermal dynamics," in *Proc. Amer. Control Conf. (ACC)*, 2016, pp. 4000–4005.

[36] A. Pozzi and D. Toti, "Lexicographic model predictive control strategy in ageing-aware optimal charging procedure for lithium-ion batteries," *Comput. Chem. Eng.*, vol. 163, Jul. 2022, Art. no. 107847.

[37] M. Doyle, T. F. Fuller, and J. Newman, "Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell," *J. Electrochemical Soc.*, vol. 140, no. 6, pp. 1526–1533, Jun. 1993.

[38] S. J. Moura, F. B. Argomedo, R. Klein, A. Mirtabatabaei, and M. Krstic, "Battery state estimation for a single particle model with electrolyte dynamics," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 2, pp. 453–468, Mar. 2017.

[39] H. E. Perez, S. Dey, X. Hu, and S. J. Moura, "Optimal charging of Li-ion batteries via a single particle model with electrolyte and thermal dynamics," *J. Electrochemical Soc.*, vol. 164, no. 7, pp. A1679–A1687, 2017.

[40] M. Ecker, T. K. D. Tran, P. Dechent, S. Käbitz, A. Warnecke, and D. U. Sauer, "Parameterization of a physico-chemical model of a lithium-ion battery: I. Determination of parameters," *J. Electrochemical Soc.*, vol. 162, no. 9, pp. A1836–A1848, 2015.

[41] M. Ecker, S. Käbitz, I. Laresgoiti, and D. U. Sauer, "Parameterization of a physico-chemical model of a lithium-ion battery: II. Model validation," *J. Electrochemical Soc.*, vol. 162, no. 9, pp. A1849–A1857, 2015.

[42] A. Bemporad and M. Morari, "Robust model predictive control: A survey," in *Robustness in Identification and Control*. Cham, Switzerland: Springer, 1999, pp. 207–226.

[43] M. T. Spaan, "Partially observable Markov decision processes," in *Reinforcement Learning: State-of-the-Art*. Germany: Springer, 2012, pp. 387–414.

ANDREA POZZI (Member, IEEE) received the bachelor's degree in industrial engineering and the master's degree in electrical engineering and the Ph.D. degree in electronics, informatics, and electrical engineering from the University of Pavia, in 2015, 2017, and 2021, respectively. He was a Visiting Scholar with TU Braunschweig, in 2016, and UC Berkeley, in 2019. After serving as a Postdoctoral Researcher with the University of Pavia, he joined as an Assistant Professor of machine learning with the Faculty of Mathematical, Physical, and Natural Sciences, Catholic University of Sacred Heart, Brescia, Italy, in January 2022. His current research interests include reinforcement learning, imitation learning, machine learning, approximate dynamic programming, and advanced control theory.

DANIELE TOTI (Member, IEEE) received the bachelor's and master's degrees in computer engineering and the Ph.D. degree in computer science and engineering from Roma Tre University, in 2005, 2008, and 2012, respectively. Since 2020, he has been a member of the Faculty of Mathematical, Physical and Natural Sciences, Catholic University of the Sacred Heart, Brescia, Italy, where he is currently an Associate Professor of computer engineering. His career spans both worlds of academia and industry, with appointments in consultancy companies, research centers, universities, and software houses. He also acted as a scientific consultant for several international IT companies and research organizations. He has participated in several EU-funded international and national projects, including ARISTOTLE and LASIE (FP7); eJRM, MODERN, SIRET, and Wheesbee (PON); and SURVANT, DANTE, VIMMP, MarketPlace, OntoTrans, OntoCommons, OYSTER, OpenModel, and NanoMECommons (H2020). His current research interests include information extraction, natural language processing, and semantic knowledge discovery and modeling. Recently, he has been exploring limitation learning techniques for the optimal control of dynamic systems. He is an Oracle and Sun Certified Professional with seven industry-level certifications.