Machine Learning Accelerated Real-Time Model Predictive Control for Power Systems

Ramij Raja Hossain, *Graduate Student Member, IEEE*, and Ratnesh Kumar, *Fellow, IEEE*

***Abstract*—This paper presents a machine-learning-based speed-up strategy for *real-time* implementation of model-predictive-control (MPC) in emergency voltage stabilization of power systems. Despite success in various applications, *real-time* implementation of MPC in power systems has not been successful due to the online control computation time required for large-sized complex systems, and in power systems, the computation time exceeds the available decision time used in practice by a large extent. This long-standing problem is addressed here by developing a novel MPC-based framework that i) computes an optimal strategy for nominal loads in an offline setting and adapts it for real-time scenarios by successive online control corrections at each control instant utilizing the latest measurements, and ii) employs a machine-learning based approach for the prediction of voltage trajectory and its sensitivity to control inputs, thereby accelerating the overall control computation by multiple times. Additionally, a realistic control coordination scheme among static var compensators (SVC), load-shedding (LS), and load tap-changers (LTC) is presented that incorporates the practical *delayed actions of the LTCs*. The performance of the proposed scheme is validated for IEEE 9-bus and 39-bus systems, with ±20% variations in nominal loading conditions together with contingencies. We show that our proposed methodology speeds up the online computation by 20-fold, bringing it down to a practically feasible value (fraction of a second), making the MPC real-time and feasible for power system control for the first time.**

***Index Terms*—Machine learning, model predictive control (MPC), neural network, perturbation control, voltage stabilization.**

I. Introduction

REAL-TIME control in power systems is of utmost importance for the resilience and security of the bulk power system with the increasing integration of renewable energy sources and dynamic loads. Although most power utilities are equipped with a fast, robust, and reliable protective relaying scheme, severe disturbances in power systems such as system faults, loss of generation, or circuit contingencies can cause *large-disturbance voltage instability* resulting in a significant decline in bus voltages even after several seconds of fault clearance [1]. This is termed an emergency voltage condition in [2]–[4], necessitating special protection systems (SPS) or remedial action schemes (RAS) to exercise control actions to stop the evolution of an unstable scenario before its conclusion into a voltage collapse. Standard practices generally include an empirical rule-based approach [5], but these approaches are not adaptable and, therefore, are not suitable for modern power systems with uncertain load and generation profiles. To this end, MPC is a promising alternative for traditional SPS-based control in power systems. The existing rich *theoretical* study on MPC in power system applications [6], [7] points to the possibility for this kind of control scheme, but its real-time implementation has evaded feasible demonstration because of the computational time of online optimization. In practice, each control action needs to be computed within a fraction of a second, which has not been feasible in the case of MPC for a practical-sized system. To understand the time required for each MPC iteration, one can note that it includes measuring the current state, predicting future trajectory, and solving optimization to compute the required control adjustment. Due to the nonlinearity in power systems dynamics, trajectory sensitivity [8] computation is one of the well-established approaches to predict the change in future trajectory due to a change in controls. However, the traditional computation of trajectory sensitivity needs the full-blown time-domain simulation of system dynamics, which is time-consuming for power systems, making the realization of MPC impractical in a real-time setting. To date, the MPC-based control for even small-scale power systems using existing state-of-the-art algorithms [9] needs several seconds of computation time for each control instant, prohibiting its practical use. This paper contributes to the issue of real-time implementation of MPC in power systems by proposing a novel method, which i) leverages machine learning in accelerating model prediction and trajectory-sensitivity estimation, and ii) computes successive online refinement of a nominal optimal policy. The proposed approach accelerates MPC by 20-fold, making its implementation real-time feasible for the first time.

A. Related Work

1) Standard MPC-Based Approaches: Predictive control has been studied for decades in power systems considering various aspects and applications. Among notable early works, [6] (2006, Glavic and Van Cutsem) provides a reflection on early MPC-based approaches in power systems and also proposed a quasi-steady-state (QSS) model and sensitivity-based MPC-scheme combining a static and dynamic optimization. Trajec-

Manuscript received August 2, 2022; revised September 27, 2022; accepted October 24, 2022. This work was supported in part by the National Science Foundation (NSF-CSSI-2004766, NSF-PFI-2141084). Recommended by Associate Editor Qinglai Wei. *(Corresponding author: Ramij Raja Hossain.)*

Citation: R. R. Hossain and R. Kumar, "Machine learning accelerated real-time model predictive control for power systems," *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 4, pp. 916–930, Apr. 2023.

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: rhossain@iastate.edu; rkumar@iastate.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2023.123135

tory sensitivity based MPC formulation for emergency voltage control can be found in [9] (2010, Jin *et al*.) utilizing shunt capacitors minimizing an objective function involving the weighted sum of voltage trajectory deviation and control cost. An improvement in computation time of trajectory sensitivity is found in [10] (2012, Hou and Vittal); this work adopted a so-called "very dishonest newton (VDHN)" method to update Jacobians required for sensitivity computation. The use of VDHN introduced approximation, and their implementation remains unscalable beyond a short prediction horizon and hence not suitable for generic real-world applications with a longer prediction horizon. Among recent works, [11] (2018, Zhang *et al*.), proposed an adaptive horizon based MPC scheme in voltage control utilizing the idea of trajectory sensitivity. This work mainly focuses on finding the optimal setting of horizon parameters using a predefined evaluation index and measurements. A comprehensive survey of different predictive control strategies in connection to the wind turbine system can be found in [12] (2019, Mahoumad and Oyedeji). This work also stressed the importance of adaptive MPC in uncertain systems. MPC-based active frequency response for bulk power system considering linear power system model is proposed in [13] (2019, Jin *et al*.) This work incorporates online sensitivity-based updates on the predetermined control actions to match the rapidity of real-time applications. Reference [14] (2019, Liu *et al*.) implemented MPC-based load frequency control methods with wind and thermal power generation using a linear state-space model. Reference [15] (2021, Oshnoei *et al*.) presented a two-step robust MPC for load frequency control with a linear state-space model of power systems. MPC-based control with optimal power allocation of energy storage devices considering linear dynamics is presented in [16] (2021, Subroto *et al*.). There is also growing interest in MPC-based control for microgrid operations [17] (2022, Zhang *et al*.), [18] (2022, Kamal and Chowdhury).

While prior research has been carried out to develop MPC-based solutions in power systems, there exist certain limitations as identified here:

i) An MPC-based framework for emergency control of power systems considering nonlinear DAE model involves the idea of trajectory sensitivities to approximate nonlinear dynamics. The control computation time for existing trajectory sensitivity-based formulations exceeds the available decision-making time used in practice (fraction of second), making the existing MPC approach inapplicable for a real-world practical system.

ii) In recent publications, the advantages of MPC-based frameworks are leveraged by utilizing the linear state-space model of power systems. But, those do not address MPC with a nonlinear DAE model that is needed for practical systems.

2) Deep Reinforcement Learning-Based Approaches: Deep reinforcement learning (DRL)-based grid control [19], [20] has recently been explored for power systems. Among DRL-based designs, [20] utilized a deep Q-network (DQN)-based algorithm for load shedding based voltage control. The deep deterministic policy gradient (DDPG) and distributional soft actor critic (SAC) methods were applied to the frequency regulation problems in [21] and [22], respectively. The DRL-based methods have certain disadvantages pertaining to the real-world application for power systems [23]:

i) DRL predominantly trains the policy in a model-free manner exploring the control action and optimizing the expected value of a cost function. With the increase of state and action space, exploration/training becomes un-scalable.

ii) DRL agents cannot be trained by direct interaction with real-world power networks, and those trained in a simulated environment might face catastrophic failure in a real-world application without a proven robust performance and safety guarantee [23].

B. Motivation and Our Approach

In contrast to DRL that has above mentioned limitations, MPC is a well-accepted control technique in many control applications. However as discussed earlier, MPC can become practically meaningful for power systems only if the computational bottleneck is addressed. We also found that explicit-MPC (eMPC), a computationally viable MPC variant is not scalable for large complex systems such as power systems. All these motivated us to address this gap by proposing an efficient approach to reduce the online computational time of MPC in power system applications. Our method comprises an offline (Phase-I) and an online phase (Phase-II), as detailed below.

1) Phase-I: For nominal load conditions, an optimal control sequence is generated offline, solving a standard MPC formulation.

2) Phase-II: The offline solution is updated successively to account for actual scenarios, using the real-time system measurements (as in neighboring optimal control or perturbation control [24]). The online updates, achieved by solving the predictive optimization problem of the perturbed systems, rely on trained NNs to predict the trajectory and corresponding sensitivities, thereby accelerating the computation. Also, the training is limited to variations around the nominal trajectory, thereby cutting down on the size of the training samples.

C. Key Differences With Other MPC Approaches

The key novelties of our approach over the standard MPC approaches are:

1) Standard formulations do not have the online phase that embeds machine-learning for trajectory prediction and sensitivity computation.

2) By performing refinement around the nominal control (computed offline), our approach reduces the training space.

D. Contributions

In summary, the main contributions of this paper are:

1) A novel machine learning accelerated perturbation control-based MPC method is presented for power systems. The proposed approach computes the control actions in a fraction of a second, achieving a 20-fold speed-up and making the MPC implementation feasible for real-world power systems.

2) A control scheme involving realistic coordination among SVC, LS, and LTC is proposed, where the slow-acting LTCs are formulated to have a delayed control effect to appropriately reflect their real-world behavior. Further, an LTC deci-

sion action is not issued until the delayed effect is realized.

3) A comprehensive and accurate framework for computation of trajectory sensitivities with respect to control inputs of SVC, LTC, and LS is developed and implemented in the PSAT/MATLAB platform.

4) The performance of our proposed method is demonstrated using IEEE 9-bus and 39-bus power system models. The robustness of the framework is validated under load variations of ±20% around the nominal load and different contingencies to achieve voltage stabilization satisfying network constraints, and the time constraint for real-time control computation.

II. Generic MPC Formulation for Power System

The dynamic model of a power system is a combination of the dynamic description of its different components, including generators, AVRs (automatic voltage regulators), TGs (turbine governors), loads, and the power flow equations resulting in a nonlinear differential-algebraic equation (DAE) [25] of the form

$$\dot{x} = f(x, y, u);\ \ 0 = g(x, y, u) \tag{1}$$

where we have $x :=$ state variables associated with respective dynamic components, $y :=$ algebraic variables represent bus voltages magnitudes and phase angles, and $u :=$ control inputs, e.g., the capacitance value of SVCs, tap-position of LTCs, loads that are switched off. After any disturbances, the behavior of the system is obtained by solving (1) numerically from the given equilibrium point $(x(t_0), y(t_0), u(t_0))$. In general, the effect of any small disturbances can be studied by linearizing (1) at the current equilibrium point. But, following any large disturbances (e.g., line fault, generator outage), which can cause a shift in the existing equilibrium, it is necessary to consider the complete DAE model for studying post-disturbance system behavior (not just its linearization around the pre-fault equilibrium). Accordingly, designing optimal control to improve system performance following large disturbances requires the inclusion of (1) as a constraint in the optimization process. Thus one can define a general nonlinear optimal control problem [24] for power systems as follows:

$$\min_{u(\cdot)} \int_{t_0}^{t_0+T} l(x(\tau), y(\tau), u(\tau))d\tau \tag{2a}$$

s.t.

$$\dot{x} = f(x, y, u),\quad 0 = g(x, y, u) \tag{2b}$$

$$x(t_0) = x_0\ ,\ y(t_0) = y_0 \tag{2c}$$

$$x \in \mathcal{X},\ \ y \in \mathcal{Y},\ \ u \in \mathcal{U} \tag{2d}$$

where $t_0 :=$ starting or initial time of the optimization, $T :=$ horizon of optimization, $l(\cdot,\cdot,\cdot) :=$ the performance measure, and $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{U}$ define the constraint set of x, y, and u, respectively. The problem defined in (2) follows the general structure of a nonlinear optimal control problem, and its solution can be obtained using dynamic programming (DP), which deals with the exact information about the future of the optimal trajectories [26]. This, in general, makes DP a very hard problem to solve. In power systems, where dimensions of f and g are large, the problem becomes impractical.

To this end, MPC offers a practical way of solving (2) by approximating the system model (2b). However, owing to the complex dynamics of power systems, it is difficult to approximate (2b) by its step-response or impulse response model, as is extensively used in dynamic matrix control (DMC) and model algorithmic control (MAC), two basic formulations of MPC [27]. This leads to trajectory sensitivity-based approximations that are commonly used in power systems [9], providing a reasonable approximation of the complex system model given by (1). The idea behind trajectory sensitivity [28] is time-dependent linear approximation to quantify the impact of the control variations on the nominal trajectories of system variables $x(t)$ and $y(t)$. Accordingly, the sensitivities of the trajectories to the control changes are expressed as

$$S^x(t) := x_u(t) = \frac{\partial x(t)}{\partial u(t)} \quad \text{and} \quad S^y(t) := y_u(t) = \frac{\partial y(t)}{\partial u(t)}$$

and the dynamics of $x_u(t)$ and $y_u(t)$ is obtained by differentiating (1) with respect to control input $u(t)$, providing

$$\dot{x_u}(t) = f_x(t)x_u(t) + f_y(t)y_u(t) + f_u(t) \tag{3a}$$

$$0 = g_x(t)x_u(t) + g_y(t)y_u(t) + g_u(t) \tag{3b}$$

where $f_x(t) = \frac{\partial f}{\partial x}(t)$, $f_y(t) = \frac{\partial f}{\partial y}(t)$, $g_x(t) = \frac{\partial g}{\partial x}(t)$, $g_y(t) = \frac{\partial g}{\partial y}(t)$, $f_u(t) = \frac{\partial f}{\partial u}(t)$, and $g_u(t) = \frac{\partial g}{\partial u}(t)$ are all time-varying Jacobians.

The knowledge of the trajectory sensitivities $[x_u(t), y_u(t)]$ is then used to estimate the predicated trajectories $[\hat{x}(t), \hat{y}(t)]$ when a small control correction $u(t)$ is introduced to the nominal system, that has the nominal trajectory $[\bar{x}(t), \bar{y}(t)]$

$$\hat{x}(t) \approx \bar{x}(t) + S^x(t)u(t), \quad \hat{y}(t) \approx \bar{y}(t) + S^y(t)u(t). \tag{4}$$

It is important to note that in the process of time domain simulation of (1), that provides $[\bar{x}(t), \bar{y}(t)]$, trajectory sensitivities $[x_u(t), y_u(t)]$ can also be obtained by solving (3a) and (3b). Later, in Section V, this computation procedure will be discussed in detail.

To further formulate the problem in a practical setting, where computations occur at discrete points in time, separated by sampling duration, denoted T_s, one replaces the integration/differentiation by numerical version. Also, the manipulated input $u(t)$ is held constant over control interval, denoted here as T_c, with $T_c = MT_s$ where $M \geq 1$ is the number of sample instances between any two control instants (for example if sample period is 0.1 s and control decision is taken every 3 s, then $M = 30$). We provide below a discretized MPC formulation where at any discrete control instant $k \geq 0$, MPC can evaluate $N_k \geq 0$ control decisions (N_k decrease by 1 each time k increases by 1, i.e., the case of receding control horizon), while optimizing the system behavior for a prediction horizon of N, where $N > N_k, \forall k$. At a control instant $k \geq 0$, the controller computation solves the following optimization with respect to the sequence of length N_k of control variables, $u_{k,\text{seq}} := u_k, u_{k+1}, \ldots, u_{k+N_k-1}$.

Since sampling occurs at a faster time scale than the control decision occurs ($M = \frac{T_c}{T_s}$ times faster), for any variable r and control time index $j \geq 0$, we introduce a "range" notation $r_{j:j+1}$ to represent the set of sample values the variable r takes between the control decision instants j and $j+1$, namely,

$$r_{j:j+1} := \begin{bmatrix} r_j = r(jT_c) \\ r(jT_c + T_s) \\ \vdots \\ r(jT_c + (M-1)T_s) = r((j+1)T_c - T_s) \end{bmatrix}.$$

Using this notation, we have the following discrete-time version of the optimization at each control instant $k \geq 0$:

$$\min_{u_k,\ldots,u_{k+N_k-1}} \sum_{i=0}^{N-1} l(\hat{x}_{k+i:k+i+1}, \hat{y}_{k+i:k+i+1}) \tag{5a}$$

s.t.

$$\hat{x}_{k+i:k+i+1} = \bar{x}_{k+i:k+i+1} + S^x_{k+i:k+i+1} \sum_{j=0}^{\min(i,N_{k-1})} u_{k+j} \tag{5b}$$

$$\hat{y}_{k+i:k+i+1} = \bar{y}_{k+i:k+i+1} + S^y_{k+i-1:k+i} \sum_{j=0}^{\min(i,N_{k-1})} u_{k+j} \tag{5c}$$

$$\hat{x}_{k+i:k+i+1} \in X, \ \ \hat{y}_{k+i:k+i+1} \in Y, \quad \forall i \in [0, N-1] \tag{5d}$$

$$u_{k+j} \in U, \quad \forall j \in [0, N_k - 1]. \tag{5e}$$

Note that the variables $\bar{x}, \bar{y}$ represent the nominal state and algebraic variables under nominal control, whereas the variables $\hat{x}, \hat{y}$ represent the state and algebraic variables under the indicated controls added sequentially, so at instant $k+i, i \in [0, N_k - 1]$, the cumulative added control is $\sum_{j=0}^{\min(i,N_{k-1})} u_{k+j}$. Also, as it is customary with MPC, at each control instant $k \geq 0$, the very first move u_k of the computed control sequence is implemented, and the controller continues to repeat the same above type of optimization with the updated measurements of the system variables at each next control instant. This iterative procedure constitutes implicit feedback (where the most recent measurements are used to adapt the control decisions) and is of utmost importance in reducing the effect of modeling and measurement imperfections.

III. Theoretical Framework for MPC-Based Coordinated Voltage Control

This section provides the theoretical formulation of the MPC-based coordinated voltage stabilization problem following a large disturbance, e.g., a line fault in transmission networks. It is often observed that even after the clearance of fault, voltage trajectories may diverge from stable values due to the effect of load transients and other inherent dynamics of the power system. Hence, to stabilize the voltage trajectories, coordinated management of various controllable devices (static VAR compensators (SVC), under voltage load shedding (UVLS), load tap changers (LTC)) are required. However, as noted in the introduction, the problem of appropriate coordination among the controllers of *different time-scales* in the MPC setting has not been provided: For example, LTC, which are commonly used with the transformers at the boundary of the transmission and distribution networks, are slow-time scale devices compared to the fast-acting SVC or UVLS relays. The effect of the disturbance in a transmission network can propagate to the distribution side, which may prompt a tap setting change in LTC, commanded by its local automatic voltage control (AVC) system. The mechanical time delay T_{mech} (typically ~ 5 s) is required for the LTC to move the taps by one position. For $T_c = 3$ s, $[T_{\text{mech}}/T_c] \sim 2$, hence the effect of the LTC comes after a delay of 2 control instants (as also illustrated in Fig. 1) and this delay must be accounted. We present a practical MPC framework for coordinated voltage control involving shunt compensation and load-shedding while also considering the impact of the delayed actions of LTC.

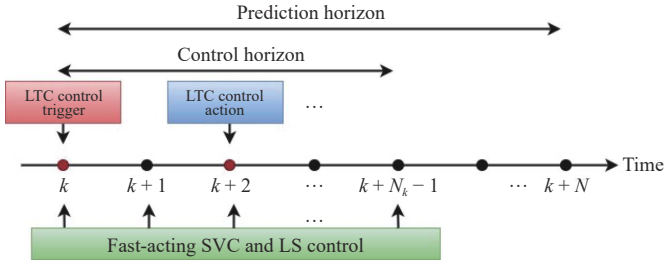

Fig. 1. Illustrating principle of control coordination.

Without loss of generality, let us consider that the distribution systems are modeled as lump loads connected with the transmission side through transformers equipped with LTCs. In the proposed setting, the controls of SVC and LS on the transmission side are achieved by the voltage trajectory sensitivity-based MPC, whereas LTC control is achieved through local AVCs depending on the secondary side or low-voltage (LV) side bus voltages of respective transformers. We employ the decision logic of LTC control, with an improvement to account for the T_{mech} delay in LTC action: LTC performs a tap-change by a unit step at a control decision instant $k \geq 0$ if the predicted voltage of the low-voltage side controlled bus, $\hat{V}^{\text{lv}}$, ends up violating a predefined dead-band threshold, V^{db}, with respect to a reference voltage V^{r} over the period starting from the current control instant kT_c and extending T_{mech} units later. In practical settings, $[T_{\text{mech}}/T_c] = 2$, AVC needs to check the dead-band constraint for the predicted voltage over the control instants from k to $k+2$ (i.e., for the variable $\hat{V}^{\text{lv}}_{k:k+2}$ in our "range" notation) to adjust the tap setting

$$\Delta \mathbf{N}_k = \begin{cases} +1, & \text{if } \hat{V}^{\text{lv}}_{k:k+2} \leq V^{\text{r}} - \dfrac{V^{\text{db}}}{2} \\ -1, & \text{if } \hat{V}^{\text{lv}}_{k:k+2} \geq V^{\text{r}} + \dfrac{V^{\text{db}}}{2} \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

The following observations are made about the LTC control:

- As discussed above and illustrated in Fig. 1, if a LTC change decision is taken at control instant k according to (6), it gets implemented two control instants later, at instant $k+2$.
- Accordingly, the impact of the LTC decision at instant k is captured in MPC optimization at instant $k+2$ onwards.
- The above also implies that if a LTC change decision is taken at instant k, then the earliest next instant an LTC change decision can be taken is at instant $k+2$ (i.e., no LTC decision at instant $k+1$). We incorporate this feature in our formulation (see (7b)).

Next, we formalize the MPC optimization problem at a control instant $k \geq 0$ for a power system voltage control comprising $\mathbf{N}_b$ no. of buses. The formulation is derived from the general one given in (5a)−(5d) considering only the trajectory of vector of voltage variable, $V := [V^1, \ldots, V^{\mathbf{N}_b}]^T$, where in (1) $V \in y$ is among the algebraic variables. For notational simplicity, we define the control-sequences, $u^{\text{SVC}}_{k,\text{seq}} := u^{\text{SVC}}_k, \ldots, u^{\text{SVC}}_{k+N_k-1}$ and $u^{\text{LS}}_{k,\text{seq}} := u^{\text{LS}}_k, \ldots, u^{\text{LS}}_{k+N_k-1}$.

$$\min_{u^{\text{SVC}}_{k,\text{seq}},\, u^{\text{LS}}_{k,\text{seq}}} \quad \sum_{i=0}^{N-1} (\hat{V}_{k+i:k+i+1} - V_{\text{ref}})^T \mathbf{R} (\hat{V}_{k+i:k+i+1} - V_{\text{ref}}) + W^T_{\text{SVC}} u^{\text{SVC}}_{k,\text{seq}} + W^T_{\text{LS}} u^{\text{LS}}_{k,\text{seq}} \tag{7a}$$

s.t.

$$\forall i \in [0, N-1] : \hat{V}_{k+i:k+i+1} = \bar{V}_{k+i:k+i+1} + S^{\text{SVC}}_{k+i:k+i+1} \times \sum_{j=0}^{\min(i, N_k-1)} u^{\text{SVC}}_{k+j} + S^{\text{LS}}_{k+i:k+i+1} \times \sum_{j=1}^{\min(i, N_k-1)} u^{\text{LS}}_{k+j} + \mathbf{I} \times S^{\text{LTC}}_{k+i:k+i+1} \times u^{\text{LTC}}_k, \text{ where}$$

$$\mathbf{I} = \begin{cases} 0, & \text{if } k \text{ is not a LTC decision point} \\ 0, & \text{if } k \text{ is a LTC decision point and } i < 2 \\ 1, & \text{otherwise} \end{cases} \tag{7b}$$

$$u^{\text{SVC}}_{\min} \leq u^{\text{SVC}}_{k+j} \leq u^{\text{SVC}}_{\max}, \quad \forall j \in [0, N_k - 1] \tag{7c}$$

$$u^{\text{LS}}_{\min} \leq u^{\text{LS}}_{k+j} \leq u^{\text{LS}}_{\max}, \quad \forall j \in [0, N_k - 1] \tag{7d}$$

$$V_{\min} \leq \hat{V}_{k+N} \leq V_{\max}. \tag{7e}$$

• V_{ref} := reference voltage, $\mathbf{R}$:= weight matrix for bus voltage deviation, W_{SVC} and W_{LS} : = weight vectors for SVC and LS control inputs, respectively.

• $S^{\text{SVC}}, S^{\text{LS}}$, and S^{LTC} := Voltage trajectory sensitivity matrices wrt. SVC, LS and LTC control input, respectively,

• $u^{\text{LTC}}_k := \Delta\mathbf{N}_k \times \Delta V_{\text{tap}}$, where $\Delta\mathbf{N}_k$:= LTC control decision made at k, and ΔV_{tap} := p.u. voltage change per tap operation.

• $[u^{\text{SVC}}_{\min}, u^{\text{SVC}}_{\max}]$, $[u^{\text{LS}}_{\min}, u^{\text{LS}}_{\max}]$, $[V_{\min}, V_{\max}]$:= Lower and upper bounds for changes in SVC, LS control inputs, and the voltage values at the end of the prediction horizon, respectively.

The optimization problem defined in (7a)−(7e) has a quadratic objective/cost function with linear inequality constraints, making it quadratic programming (QP), which can be efficiently solved for the nominal system. A positive-definite matrix choice of $\mathbf{R} > 0$ makes the optimization problem convex, ensuring a globally optimal solution. It is important to note that LTC action is taken based on the dead-band requirement of (6), and the impact of this action on the voltage trajectory computation is incorporated within the MPC optimization. This approach (of LTC action) is supported by the work in [29], which analyzes MPC-based voltage regulation in the distribution network with distributed generation (DG) and energy storage systems (ESS). This LTC action approach also lends computational benefits since optimization over discrete control variable of LTC is not required, rather implemented per (6).

IV. Online Adaptive Control: Control Corrections to Nominal Offline Control

In Sections II and III, we provided a theoretical framework for MPC-based voltage control in power systems. To translate the theory into real-time implementation, the following tasks need to be accomplished in runtime at each control instant $k \geq 0$:

1) Real-time measurement of system variables;

2) Model-based prediction of future trajectory and its sensitivity computation using the measured values;

3) Solving optimization problem (7a)−(7e);

For any practical power system, step 2), model-based trajectory prediction and trajectory sensitivity computation are computationally expensive for a practical prediction horizon (20–50 s). Hence, it is often infeasible to perform MPC optimization in the desired time frame of control computation. To this end, one possible direction is to explore artificial intelligence, particularly the advances in machine learning to reduce the online computation time. Research has indicated that for applications with high dimensional input-output and large data sets, neural networks (NN), particularly deep neural networks (DNN), outperform other machine learning-based methods, e.g., support vector regression (SVR) and decision trees (DT) in nonlinear regression problems [30]–[32]. In our case, while handling the trajectory prediction and corresponding sensitivity estimation, we also dealt with input-output data having dimensions greater than 1000 (for 39 bus systems), which is why we employed neural networks (NN)—We did try other machine learning approaches such as SVR and DT, but either the learning was not efficient, or the algorithms did not converge. Next, the main challenges lie in creating the training set required for offline training of the desired NNs for trajectory prediction and sensitivity computation. A simple analysis shows that in an MPC of N_c control steps with m-dimensional controls, each having q levels of quantization, the number of possible control combinations is $q^{m \times N_c}$, and if there are p number of load/contingency scenarios, then total combinations to explore for training is $p \times q^{m \times N_c}$, which is prohibitive. To overcome the prohibitive size of the training space, we introduce a novel approach involving successive control correction at each control instant, limiting the training space to the neighborhood of a nominal-case optimal trajectory for training the NNs, which brings down the training space size to order p (since the control choice is already fixed at the optimal values for nominal system found in the offline MPC optimization). This reduction in training space helps make it practically feasible.

In the proposed approach, one views the real system trajectory to be a corrected version of the nominal system trajectory, that is obtained by an offline MPC-based optimization of the given nominal system model. For example, the real-time operation load levels vary within ±20% of the nominal values of load. Fig. 2 presents an illustration. The **yellow** curve and the **blue** curve are the post fault trajectories of the nominal system versus the real system without any control. The offline

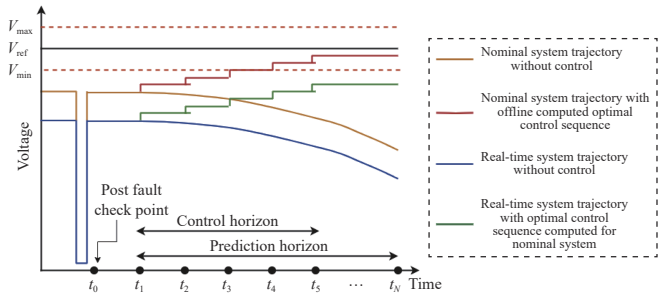

Fig. 2. Illustrating online control correction computation.

computed optimal control sequence $u^*_{\text{nom,seq}} := u^*_{1,\text{nom}}, \ldots, u^*_{N_c,\text{nom}}$ (with N_c control steps) can stabilize voltages of the nominal system within the desired range (see **red** curve), but the same is not true for the real system (see **green** curve). Hence a correction in the nominal control is required. Suppose $u^*_{\text{nom,seq}} := u^*_{1,\text{nom}}, \ldots, u^*_{N_c,\text{nom}}$ is the offline computed MPC-based optimal control sequence for the nominal system following a particular contingency. Then the optimal control sequence for the same contingency for the actual real system $u^*_{\text{real,seq}}$, which differs from the nominal system model, can be written as a correction Δu^*_{seq} to $u^*_{\text{nom,seq}}$

$$u^*_{\text{real,seq}} = u^*_{\text{nom,seq}} + \Delta u^*_{\text{seq}} . \tag{8}$$

In our approach, the sequence of control corrections $\Delta u^*_{\text{seq}} := \Delta u^*_1, \ldots, \Delta u^*_{N_c}$ in (8) is estimated online utilizing the measurements of the real system while meeting the computational time constraints imposed by the real-time operation of a power system. Next, we outline our approach for the computation of the corrections in control iteratively at the control instants, where to compute the desired control correction term at control instant $k \geq 0$, we propose the following steps:

a) First, predict the voltage trajectory of the real system under the influence of $u^*_{k,\text{nom}}$ over the time range kT_c to $(k+1)T_c - T_s$ (see description below).

b) Next, compute the sensitivities of the voltage trajectory obtained in a) with respect to controls (see description below).

c) Using the information obtained in a) and b), solve a 1-step MPC optimization to compute Δu^*_k.

d) Repeat the above steps at the next control instant, until the final control instant is reached.

To facilitate the trajectory prediction and sensitivity computation in steps a) and b) in a real-time fashion, we introduce two categories of trained NNs (see Fig. 3). This eliminates the necessity of time-consuming model-based online time-domain simulation for trajectory prediction and sensitivity estimation, making the online computation of control corrections viable for real-time MPC application.

• **The Prediction-NN, $f_{\text{NN}-1}(\cdot,\cdot)$:** At each control instant $k \geq 0$, it receives as inputs i) the measured voltage trajectory of the real system $V_{k-1:k}$ between the last two control instants (from $(k-1)T_c$ to $kT_c - T_s$), and ii) the nominal optimal control at kT_c, $u^*_{k,\text{nom}}$, while it outputs the predicted trajectory $\bar{V}^{\text{nom}}_{k:k+1}$ over kT_c to $(k+1)T_c - T_s$. The idea here is that the measured voltage trajectory $V_{k-1:k}$ over the period $(k-1)T_c$ to $kT_c - T_s$ serves as the proxy to the state at kT_c, which, when

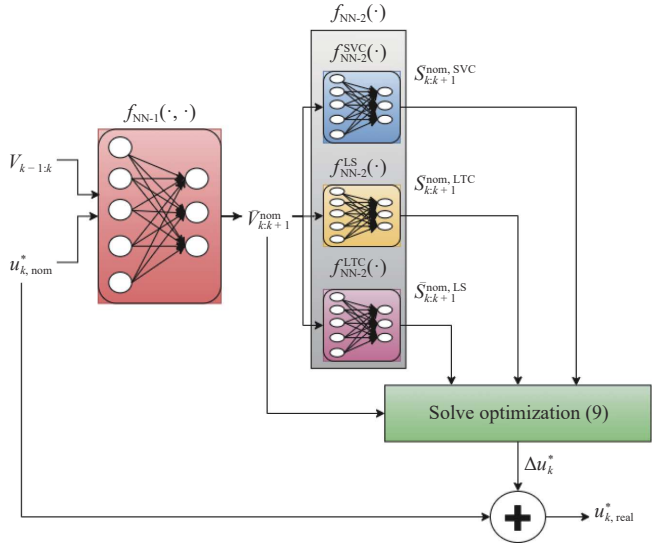

Fig. 3. Iterative online computation of control refinement.

combined with the current control information $u^*_{k,\text{nom}}$ at kT_c, allows the prediction of the future state proxy, namely, the voltage trajectory $\bar{V}^{\text{nom}}_{k:k+1}$ over the period kT_c to $(k+1)T_s - T_s$ under the nominal control.

• **The Sensitivity-NNs, $f^{SVC}_{\text{NN}-2}(\cdot), f^{LS}_{\text{NN}-2}(\cdot), f^{LTC}_{\text{NN}-2}(\cdot)$:** Corresponding to each control input category, we have respective Sensitivity-NNs. They receive the predicted trajectory, i.e., the output $\bar{V}^{\text{nom}}_{k:k+1}$ of $f_{\text{NN-1}}(\cdot,\cdot)$ as input, and produces the sensitivity matrices $\bar{S}^{\text{nom,SVC}}_{k:k+1}, \bar{S}^{\text{nom,LS}}_{k:k+1}, \bar{S}^{\text{nom,LTC}}_{k:k+1}$ as the respective outputs.

Next, we introduce the single step MPC optimization that we propose to solve at each control instant $k \geq 0$ towards the step c) above.

$$\min_{\Delta u_k} \ (\hat{V}_{k:k+1} - V_{\text{ref}})^T \mathbf{R}_\Delta (\hat{V}_{k:k+1} - V_{\text{ref}}) + w^T_\Delta \Delta u_k \tag{9a}$$

s.t.

$$\hat{V}_{k:k+1} = \bar{V}^{\text{nom}}_{k:k+1} + \bar{S}^{\text{nom}}_{k:k+1} \Delta u_k \tag{9b}$$

$$= \bar{V}^{\text{nom}}_{k:k+1} + \bar{S}^{\text{nom, SVC}}_{k:k+1} \Delta u^{\text{SVC}}_k + \bar{S}^{\text{nom, LS}}_{k:k+1} \Delta u^{\text{LS}}_k + \bar{S}^{\text{nom,LTC}}_{k:k+1} \Delta u^{\text{LTC}}_k \tag{9c}$$

$$\bar{V}^{\text{nom}}_{k:k+1} = f_{\text{NN-1}}(V_{k-1:k}, u^*_{k,\text{nom}}) \tag{9d}$$

$$\bar{S}^{\text{nom,SVC}}_{k:k+1} = f^{SVC}_{\text{NN-2}}(\bar{V}^{\text{nom}}_{k:k+1}) \tag{9e}$$

$$\bar{S}^{\text{nom,LS}}_{k:k+1} = f^{LS}_{\text{NN-2}}(\bar{V}^{\text{nom}}_{k:k+1}) \tag{9f}$$

$$\bar{S}^{\text{nom, LTC}}_{k:k+1} = f^{LTC}_{\text{NN-2}}(\bar{V}^{\text{nom}}_{k:k+1}) \tag{9g}$$

$$u_{\min} \leq u^*_{k,\text{nom}} + \Delta u_k \leq u_{\max} \tag{9h}$$

$$V_{k,\min} \leq \hat{V}_{k+1} \leq V_{k,\max} \tag{9i}$$

where $\mathbf{R}_\Delta$ and w_Δ are the appropriately truncated portions of $\mathbf{R}$ and $[W^T_{\text{SVC}}\ W^T_{\text{LS}}]$ respectively to account for the single step costs, $\bar{V}^{\text{nom}}_{k:k+1}$ is the predicted voltage trajectory over kT_c to $(k+1)T_c - T_s$ under nominal optimal control using $f_{\text{NN-1}}$, and

$\bar{S}^{\text{nom}}_{k:k+1} = \left[\bar{S}^{\text{nom,SVC}}_{k:k+1}, \bar{S}^{\text{nom,LS}}_{k:k+1}, \bar{S}^{\text{nom,LTC}}_{k:k+1}\right]$ is the predicted sensitivity over kT_c to $(k+1)T_c - T_s$ under nominal optimal control using $f_{\text{NN-2}} = \left[f^{SVC}_{\text{NN-2}}, f^{LS}_{\text{NN-2}}, f^{LTC}_{\text{NN-2}}\right]$.

Note that the control input u of our setting is of 3 different types corresponding to SVC, LS and LTC, and accordingly, (9b) is expanded to show the contribution of the individual controls in (9c), in which Δu^{SVC}_k, Δu^{LS}_k are computed by the above single step MPC optimization, whereas Δu^{LTC}_k is known from beforehand (at $k-2$): At a control decision point $k \geq 0$, the local AVC controllers predict the LV side voltage trajectories of designated buses over the control instants k to $k+2$ (from kT_c to $(k+2)T_c - T_s$) and issue control decisions according to (6). These decisions are implemented at $k+2$ to account for the delayed action of LTC. For a real-time prediction of future LV voltage, we use another NN, an **AVC-NN,** $f_{\text{NN}-3}(\cdot)$, which receives the LV side bus voltage information over $(k-1)T_c$ to $kT_c - T_s$ and outputs the predicted LV side voltage from kT_c to $(k+2)T_c - T_s$ without assuming the influence of any future control action at $k+1$.

The output of the optimization problem (9a)–(9i) gives Δu^*_k, which is then added to $u^*_{k,\text{nom}}$ to get $u^*_{k,\text{real}}$ as in (8), and finally implemented at k. The block diagram of the perturbation control computation is shown in Fig. 3, and the flow chart of the complete Offline/Online phases of implementation in given in Fig. 4.

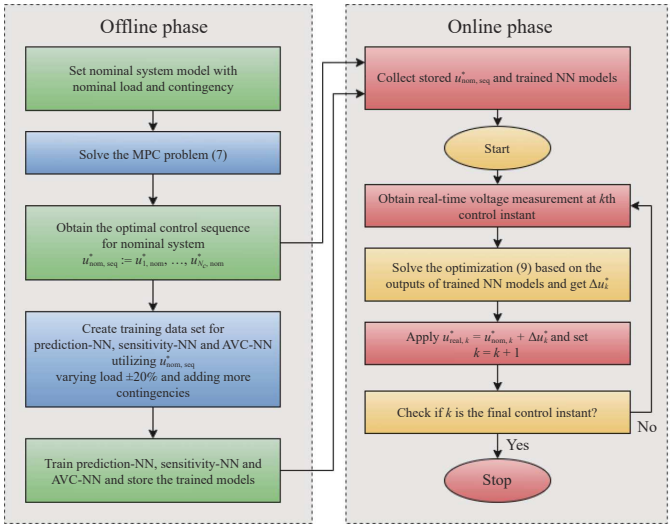

Fig. 4. Flow chart of the Offline/Online implementation phases.

V. Sensitivity Computations for Offline MPC

This section presents the trajectory sensitivity computation required to i) solve the MPC problem offline for the nominal system and ii) generate the training data set for sensitivity-NNs. We used the power system simulator PSAT [33] for our studies and found that the trajectory sensitivity computation with respect to control is not supported in it. There exist some earlier works [9], [10] that mention the use of PSAT for trajectory sensitivity-based MPC, but the details for the trajectory sensitivity computation are not provided. So one needs to essentially introduce certain extensions to the source code of PSAT to achieve the trajectory sensitivity computation [34].

As discussed earlier, sensitivity can be obtained solving (3a) and (3b) for which we need the Jacobians f_x, f_y, g_x, g_y, f_u, g_u at each time steps of the simulation. The issue is that PSAT does not support the computation of f_u and g_u, and also does not store the values of the other Jacobians. For the latter, we introduced additional coding in PSAT's time domain integration subroutine to store the Jacobians for each step of time domain simulation. On the other hand, to overcome the lack of support for the computation of Jacobians f_u and g_u, we utilized the fact that PSAT supports the computation of Jacobians with respect to state variables. Hence, we treated the control u as a state variable, but having zero-dynamics ($\dot{u} = 0$) (recall that for computing sensitivity with respect to control, the control is held constant at its nominal value, i.e., is rate of change is indeed zero). With this augmentation of states to include zero-dynamics input, we computed the Jacobians f_u and g_u as follows. Denoting the control-augmented state variables as, $\bar{x} = \begin{bmatrix} x \\ u \end{bmatrix}$, algebraic variables as y, and combining, we have

$$\dot{\bar{x}} = \begin{bmatrix} \dot{x} \\ \dot{u} \end{bmatrix} = \begin{bmatrix} f(x,y,u) \\ 0 \end{bmatrix} =: \bar{f}(\bar{x},y); \quad 0 = g(\bar{x},y). \tag{10}$$

Upon differentiation of (10) with respect to control input u, we get

$$\dot{\bar{x}}_u(t) = \bar{f}_{\bar{x}}\bar{x}_u(t) + \bar{f}_y y_u(t); \quad 0 = g_{\bar{x}}\bar{x}_u(t) + g_y y_u(t). \tag{11}$$

Note $\bar{f}_{\bar{x}} = \begin{bmatrix} f_x & f_u \\ 0 & 0 \end{bmatrix}$, $\bar{f}_y = \begin{bmatrix} f_y \\ 0 \end{bmatrix}$, and $g_{\bar{x}} = \begin{bmatrix} g_x & g_u \end{bmatrix}$. Now with the above control-augmented states, PSAT can compute as well as store the Jacobians $\bar{f}_{\bar{x}}$, $\bar{f}_y$, $g_{\bar{x}}$, and g_y in course of the time domain simulation, from which we extract the desired f_u and g_u. Having all the Jacobians $f_x, f_y, g_x, g_y, f_u, g_u$ available, (11) are solved numerically to obtain the required trajectory sensitivities x_u and y_u.

The default PSAT dynamic models of SVC and LTC are given by (12a) and (12b), respectively,

$$\dot{b}_{SVC} = \frac{K_r(V_{ref} - V) - b_{SVC}}{T_r} \tag{12a}$$

$$\dot{m} = -K_d m + K_i(V - V_{ref}). \tag{12b}$$

Where the susceptance b_{SVC} and tap-ratio m is the control inputs, respectively, to make these models suitable for sensitivity computation, we chose the certain parameters of existing SVC and LTC blocks appropriately: In SVC, to zero the b_{SVC} dynamics, we set the time-constant T_r to be very high, and set the gain K_r to be very low. Similarly, by assigning very low values to parameters K_d and K_i, we also set the dynamics of m in the LTC block to close to zero. Certain other adjustments in the PSAT code were performed to account for these blocks also having anti-windup limiters.

The computation of load-shedding sensitivity is little more involved. In this study, we used exponential recovery load for which the active power dynamics is given by

$$\dot{x}_P = -x_P/T_P + P_0(V/V_0)^{\alpha_s} - P_0(V/V_0)^{\alpha_t} \tag{13a}$$

$$P = x_P/T_P + P_0(V/V_0)^{\alpha_t}. \tag{13b}$$

Here, P_0 is the base-load value that needs to be altered to

exercise load-shedding. So we introduced an additional equation $\dot{P}_0 = 0$ to augment P_0 as another zero dynamics state-variable. Similar state-variable augmentation was done for base reactive power Q_0. To introduce the new state variables, we made certain modifications in the corresponding sub-routine of PSAT for the exponential recovery load.

VI. Implementation, Test Cases, and Results

Our proposed methodology is implemented in PSAT and applied for voltage stabilization in IEEE-9 and IEEE-39 bus systems as proof of validation. We modified these test systems to accommodate different control inputs and distribution-side loads through LTC.

A. Test System 1: IEEE 9-Bus System

We utilized the standard IEEE 9-bus system with slight modification (see IEEE 9-bus example in Fig. 5) to address the voltage stability problem following a 3-phase fault at bus-5 with a fault lasting 0.10 sec, that got cleared by tripping the line between bus 4 and 5. In order to include the distribution-side loads, we added 2 additional buses 10 and 11, connected through LTC to bus 6 and 8, respectively, or the original 9-bus system. Their local AVCs control these 2 LTCs. The other control inputs include 3 SVCs connected at buses 5, 7, and 8, varying from 0 to 0.2 p.u. per step, and load-shedding of up to 0.1 p.u. (approx. 10% of the bus load) at buses 10 and 11. Sustained under-voltages are observed following the fault without any control actions. Such voltage behavior is deemed as an emergency condition [3] considering, i) progressive voltage decline reaching unacceptable values, much lower than the desired level, and ii) potential to voltage collapse (see the left plot of Fig. 6 for IEEE-9, bus-5 voltage decline in the blue curve).

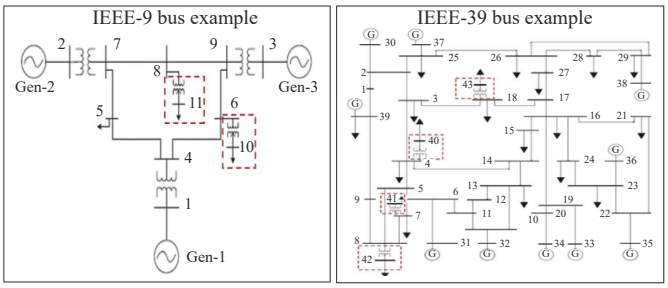

Fig. 5. IEEE 9-bus and 39-bus systems.

B. Test System 2: IEEE 39-Bus System

For the IEEE-39 bus system (see IEEE 39-bus example in Fig. 5), we consider the 3-phase fault at bus-15, which is cleared by tripping the transmission line in-between buses 15 and 16 within 0.10 sec of the fault occurrence. Here, the added 4 buses, 40, 41, 42, and 43, represent loads of the distribution side, connected through 4 LTCs to the buses 4, 7, 8, and 18, respectively, of the original network. In this power system, the SVCs are located at buses 4, 5, 7, 8, 15, 17, 18, and 25, ranging from 0 to 0.20 p.u. per step, whereas the load-shedding can be exercised at the buses 15, 40, 41, 42, and 43 by up to 0.3 p.u. per step (approx. 10% of the bus load). Like the IEEE 9-bus example, following the fault, voltage drops below the desirable level almost immediately, constituting an emergency voltage condition. This must be stabilized to avoid an impending voltage collapse around 40 sec as seen in the right plot of Fig. 6 for IEEE-39, bus-15 voltage decline (blue curve).

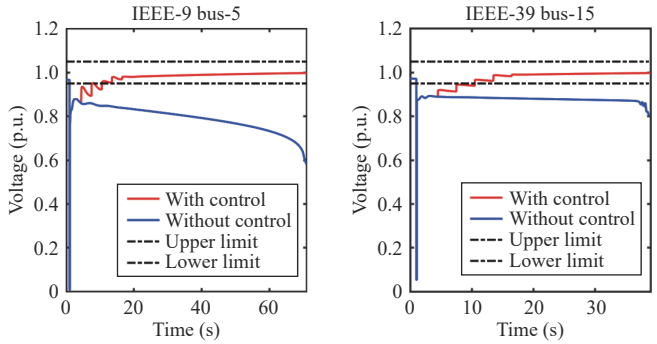

Fig. 6. Voltage profile without control (blue) and with MPC control (red) for nominal models of IEEE 9-bus and 39-bus systems, respectively.

C. Offline MPC Computed Controls for Nominal Models

We utilized nominal models of the above 2 systems to compute their MPC control sequence $u^*_{\text{nom,seq}}$ offline with V_{ref} = 1.00, and store those values. In this computation, sampling interval $T_s = 0.1$ s, whereas the control intervals T_c are 3 s. The computed optimal control sequences for the respective nominal models are listed in Tables I and II , which are corrected online during runtime operation. The voltage trajectories with MPC control (**red**) and without MPC control (**blue**) MPC for bus-5 of the 9-bus system and bus-15 of the 39-bus system are shown in Fig. 6 , which clearly show the need of voltage stabilizing control in order to maintain voltages within the safe limits, usually [0.95, 1.05] p.u.

TABLE I
IEEE 9-Bus: Optimal Sequence for Nominal Model

Time instant	4.5 sec	7.5 sec	10.5 sec	13.5 sec	16.5 sec
SVC-5 (in p.u.)	0.2	0.2	0.2	0.1378	0.0534
SVC-7 (in p.u.)	0.2	0.1791	0.0	0.0	0.0
SVC-8 (in p.u.)	0.2	0.0	0.0	0.0	0.0
L/S 10 (in p.u.)	0.0619	0.0024	0.0167	0.0	0.0
L/S 11 (in p.u.)	0.0353	0.0	0.0	0.0	0.0
LTC b/w 10–6	0	0	0	0	0
LTC b/w 11–8	0	0	+1	0	0

D. NNs for Online MPC-Based Adaptive Control Correction

The real-time application of our methodology relies on 3 categories of NNs, as mentioned in Section IV, namely Prediction-NN, Sensitivity-NN, and AVC-NN. The training of these NNs is an important factor in the success of the proposed methodology.

1) Structure of the NNs: We used fully-connected neural networks (FCNN) with 2 hidden layers to build the NNs of 3 categories: Prediction-NN, Sensitivity-NN, and AVC-NN. For the IEEE-9 Bus system, the number of neurons in each hid-

TABLE II
IEEE 39-BUS: OPTIMAL SEQUENCE FOR NOMINAL MODEL

Time instant	4.5 sec	7.5 sec	10.5 sec	13.5 sec	16.5 sec
SVC-4 (in p.u.)	0.2	0.2	0.2	0.2	0.0
SVC-5 (in p.u.)	0.2	0.2	0.2	0.2	0.0
SVC-7 (in p.u.)	0.2	0.2	0.2	0.2	0.134
SVC-8 (in p.u.)	0.2	0.2	0.2	0.2	0.2
SVC-15 (in p.u.)	0.2	0.2	0.2	0.2	0.2
SVC-17 (in p.u.)	0.0	0.0	0.0	0.0	0.0
SVC-18 (in p.u.)	0.2	0.2	0.0	0.0	0.0
SVC-25 (in p.u.)	0.0	0.0	0.0	0.0	0.0
L/S 15 (in p.u.)	0.3	0.3	0.3	0.3	0
L/S 40 (in p.u.)	0.3	0.3	0.3	0.3	0
L/S 41 (in p.u.)	0.3	0.3	0.3	0.3	0
L/S 42 (in p.u.)	0.3	0.3	0	0	0
L/S 43 (in p.u.)	0.3	0.3	0.3	0	0
LTC b/w 40–4	0	0	+1	0	+1
LTC b/w 41–7	0	0	+1	0	0
LTC b/w 42–8	0	0	+1	0	+1
LTC b/w 43–18	0	0	+1	0	0

den layer is 64, while in the IEEE-39 Bus system, it is chosen as 256. We used tanh(·) as the nonlinear activation function for all categories of NN. The choice of NN hyper-parameters is based on experiments, where we relied on the recent developments, e.g., those utilized in [20], [35] involving NNs for power system setting (while the cited works are for reinforcement learning, their complexity is also dictated by the underlying power systems as is the case with our work).

i) Prediction-NN: As shown in Fig. 3, the input of the Prediction-NN, $f_{\text{NN-1}}(\cdot,\cdot)$ is $\{V_{k-1:k}, u_{k,\text{nom}}\}$, and the output is $\bar{V}^{\text{nom}}_{k:k+1}$. It maps the observed voltage trajectory (since the last control action) together with the current nominal control action to the predicted voltage trajectory until the next control instant. $V_{k-1:k}$ and $\bar{V}^{\text{nom}}_{k:k+1}$ are vectors of dimension $N_b \times M = 11 \times 30 = 330$ (for IEEE-9) and $= 43 \times 30 = 1290$ (for IEEE-39), where $N_b :=$ No. of Buses, and $M :=$ No. of samples from $(k-1)$ th to kth instant. $u_{k,\text{nom}}$ is only for control instant k, and its dimension is 7 (for IEEE-9) and 17 (for IEEE-39).

ii) Sensitivity-NN: The input of Sensitivity-NN, $f_{\text{NN-2}}(\cdot)$ is $\bar{V}^{\text{nom}}_{k:k+1}$, while the output is $\bar{S}^{\text{nom}}_{k:k+1}$. Since our implementation considered the coordination of 3 different category of control inputs: static var compensators (SVC), load-shedding (LS), and load tap-changers (LTC), we have Sensitivity-NNs $f^{\text{SVC}}_{\text{NN-2}}(\cdot)$, $f^{\text{LS}}_{\text{NN-2}}(\cdot)$, and $f^{\text{LTC}}_{\text{NN-2}}(\cdot)$ in Fig. 3 responsible for estimating the sensitivities for SVC, LS, and LTC, respectively. Dimension of $\bar{V}^{\text{nom}}_{k:k+1}$ (input) is $N_b \times M = 11 \times 30 = 330$ (for IEEE-9) and $= 43 \times 30 = 1290$, while $\bar{S}^{\text{nom, SVC}}_{k:k+1}$, $\bar{S}^{\text{nom, LS}}_{k:k+1}$, and $\bar{S}^{\text{nom, LTC}}_{k:k+1}$ (target) are vectors of dimension ($N_b \times M \times$ No. of SVC/LS/LTC) respectively.

iii) AVC-NN: The AVC-NN is used separately from the framework shown in Fig. 3, and helps the local AVC at buses 10, and 11 (for IEEE-9) and 40, 41, 42, and 43 (for IEEE-39) to predict the $V^{\text{lv}}_{k:k+2}$ (target) using the currently measured voltage $V^{\text{lv}}_{k-1:k}$ (input). Hence, we collected data tuple $\{V^{\text{lv}}_{k-1:k}, V^{\text{lv}}_{k:k+2}\}$ where, the dimension of $V^{\text{lv}}_{k-1:k}$ is $2 \times 30 = 60$ (for IEEE-9) and $= 4 \times 30 = 120$ (for IEEE-39), while the dimension of $V^{\text{lv}}_{k:k+2}$ is $2 \times 30 \times 2 = 120$ (for IEEE-9) and $= 4 \times 30 \times 2 = 240$ (for IEEE-39). The details of LTC logic and AVC-NN are mentioned in Sections III and IV.

2) Training Data: For both IEEE 9-bus and 39-bus cases, we created a large pool of training data by simulating the respective systems under the offline computed optimal control sequence for nominal loads with changing load within ±20% around the nominal loads and considering more contingencies for each load conditions. As mentioned earlier, power systems follow DAE dynamics given by (1). Bus voltages belong to y in (1), and the voltage trajectories are the manifestation of the dynamics defined in (1). But in general the power system dynamics also depends on other factors like operational conditions (e.g., initial load levels and contingencies) and system parameters (e.g., transmission line admittances). To capture these dependencies, we utilized a parameter θ extending the DAE into: $\dot{x} = f(x, y, u, \theta); 0 = g(x, y, u, \theta)$, so the nonstationary behavior of power systems is captured by variations in θ (see for example [36] for a similar approach). It is practical to assume no parametric change in the system (and hence in θ) over a shorter period of voltage stabilizing control, and that a change in θ is seen only over a longer time scale. While creating the training data, we introduced changes in θ by i) load variations of ±20% around the nominal values, and ii) adding more contingencies for each load condition. For notational convenience, we denote the set of operating conditions as Θ (to differentiate it from parameter θ). We generate voltage trajectories corresponding to a particular $\Theta_j \in \Theta$ for a time period of $T_f \approx 20$ s, and then picked the next Θ_j value and repeated the process (as shown in Fig. 7 for *Prediction-NN*). All these data are then pooled together for training, thereby capturing the nonstationarity due to variations in load and contingencies. This is tantamount to a continuous dynamical evolution of a power system over time with different randomly selected levels of Θ (i.e., loads and contingencies), capturing the nonstationary behavior in the generated trajectory data. It should be noted that generally most prediction mechanisms assume stationarity in the underlying data/process, and NN-based methods are no exception. While it is found that NN models are effective in dealing with non-stationarity up to a certain level [37], considering many possibilities of power systems configuration and operation, it is not rational to assume that a single trained model can handle all possible scenarios. The important distinction of our proposed method is

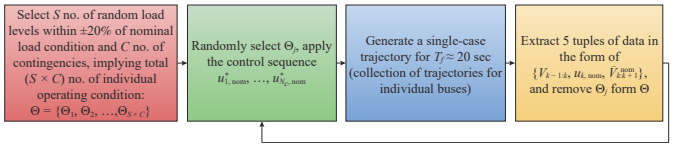

Fig. 7. Flow chart of the training data generation for Prediction-NN.

that we only need to explore the variations in the vicinity of the nominal optimal trajectory. This allows us to deal with the non-stationarity using a single trained NN for each NN category.

For training of the NNs, we selected 2500 different initial load conditions for both IEEE-9 and 39 bus systems, plus faults at bus-8, 15, and 26 for IEEE-39 bus systems and a fault at bus-5 for IEEE-9 bus systems (as it is a small system).

3) Optimizer and Training Method: We used standard mini-batch supervised learning, minimizing the error between the ground truth versus the estimated value. The optimizer chosen for the training is ADAM, with gradient momentum $\beta_1 = 0.9/0.95$ and RMS momentum $\beta_2 = 0.999/0.95$. The loss function, batch size, and learning rate used are: mean squared error (MSE), 32, and 10^{-3}, respectively. Standard techniques to avoid over-fitting and facilitate fast learning were practiced: i) adding drop-out layers and ii) normalizing the inputs and outputs of the NNs in the range $[0, 1]$.

4) Training and Testing Data Ratio: We divided the respective data sets into a $70:30$ ratio to create the training versus the testing data for all three categories of NNs.

5) Training and Testing Results: In Figs. 8–10, the training performance is shown in terms of MSE, and the test performance is determined by measuring the coefficient of determination or $R^2 \in [0, 1]$ score of the NN predicted value and the respective actual values over the test data sets (a R^2 value of 1 indicates an exact fit). Figs. 8–10 confirm the prediction accuracy of more than 95% of the trained NN models, establishing that they offer a good fit for the online adaptive control scheme proposed in this article. Given that NN-1 and NN-2 are in cascade (see Fig. 3), the effective accuracy for overall estimation step is $0.95 \times 0.95 = 0.90$. However, since new optimization is solved at each new control instant using new measurements, the errors do not propagate/accumulate over time. Thus, this level of accuracy represents a reasonable trade-off in optimality for up to 20% load fluctuations versus the resulting reduction in computation-time (a 20-fold speed-up, which then makes the MPC real-time).

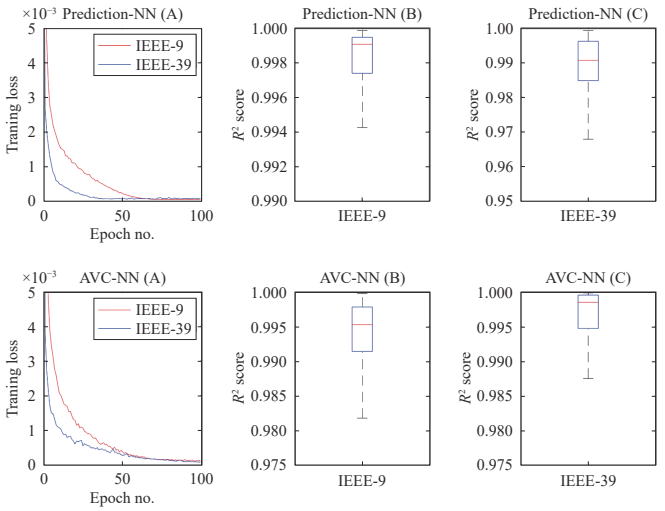

Fig. 8. Performance of Prediction-NNs and AVC-NNs.

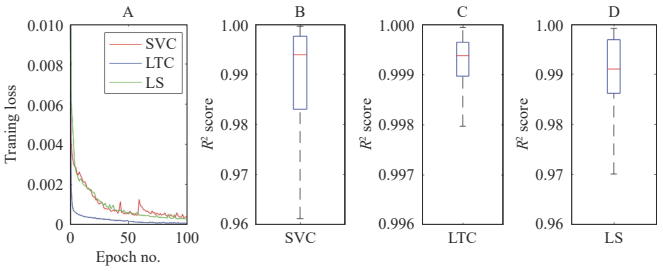

Fig. 9. Performance of Sensitivity-NNs for IEEE 9-Bus systems.

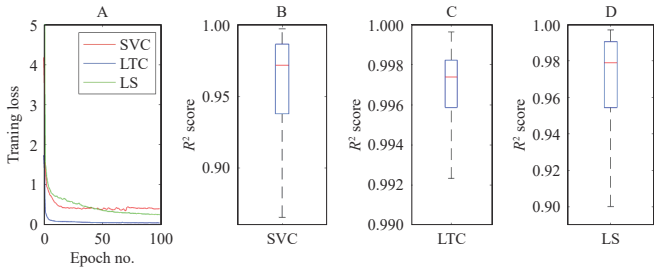

Fig. 10. Performance of Sensitivity-NNs for IEEE 39-Bus systems.

E. Discussion of Results—Online MPC and its Robustness

The real-time performance of the proposed scheme is evaluated for both IEEE 9-bus and 39-bus systems under different load and fault conditions. The real-time control corrections are computed based on current measurements and the offline computed respective optimal control sequences for the nominal load models. We consider 4 different load levels, 80%, 90%, 110%, 120% of the nominal load for showing the performance and robustness of the methodology. In addition to fault at bus-15, we consider faults at bus-4, bus-7 and bus-21 for IEEE 39-bus system. These faults are cleared by tripping the transmission lines in between bus-3 to bus-4, bus-7 to bus-8, and bus-21 to bus-22, respectively. In IEEE 9-bus system, we consider faults at bus-5 and bus-7. The voltage profiles for each of the above cases are shown in Figs. 11 and 12, validating that the proposed scheme is successful in restoring the desired voltage levels under different operating conditions (and so effectiveness and robustness of the proposed approach). To confirm the control input values, we computed the total SVC and LS actions at each control instants and plotted the respective cumulative actions in Figs. 13–18. The trend suggests that the amount of controls introduced increased with the increase of load, which is as expected.

For further testing, we created more unknown operating conditions by considering i) random load levels (within ±20% of nominal level), ii) topological variations (as shown in Table III), and iii) variations in contingency and tested our methodology to show that the proposed method can tackle the non-stationarity of power system due to changes in its operational conditions. The results are shown in Figs. 19–23.

It should be noted that in case the underlying power system differs greatly from its nominal model (because of major operational changes/modifications), the nominal model itself shall be first updated, following which the above proposed Offline-Online phases shall be executed.

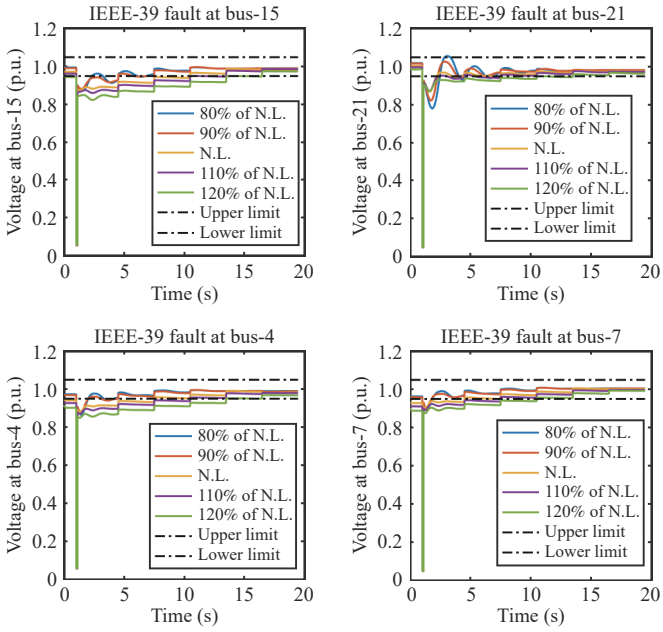

Fig. 11. Voltage profile with online MPC control for IEEE 39-bus systems.

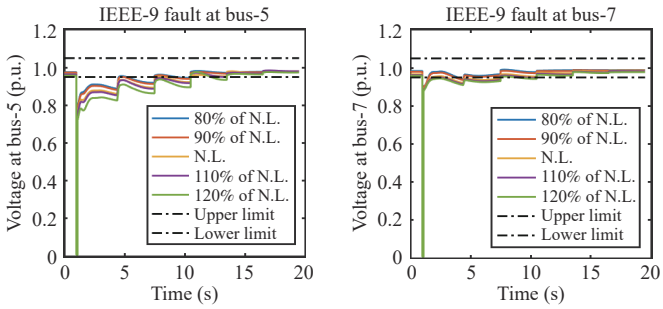

Fig. 12. Voltage profile with online MPC control for IEEE 9-bus systems.

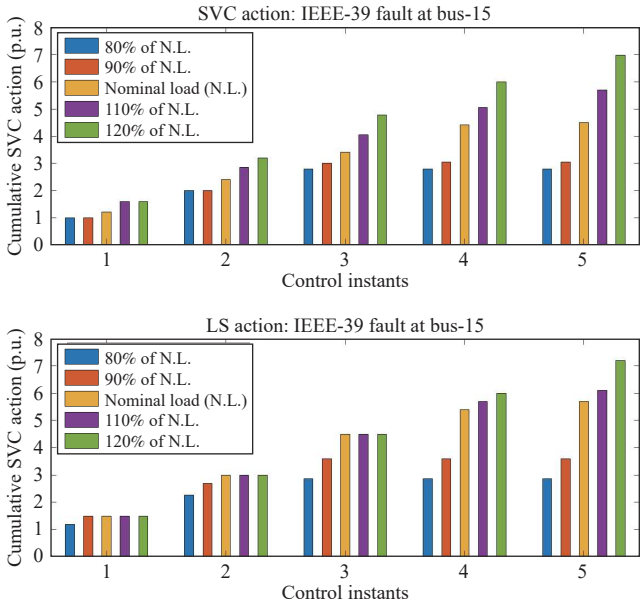

Fig. 13. SVC and LS controls for IEEE 39-bus system for fault at bus-15.

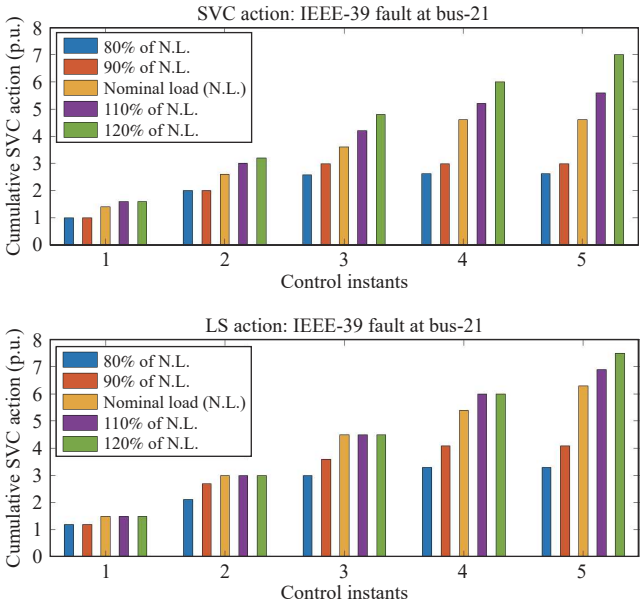

Fig. 14. SVC and LS controls for IEEE 39-bus system for fault at bus-21.

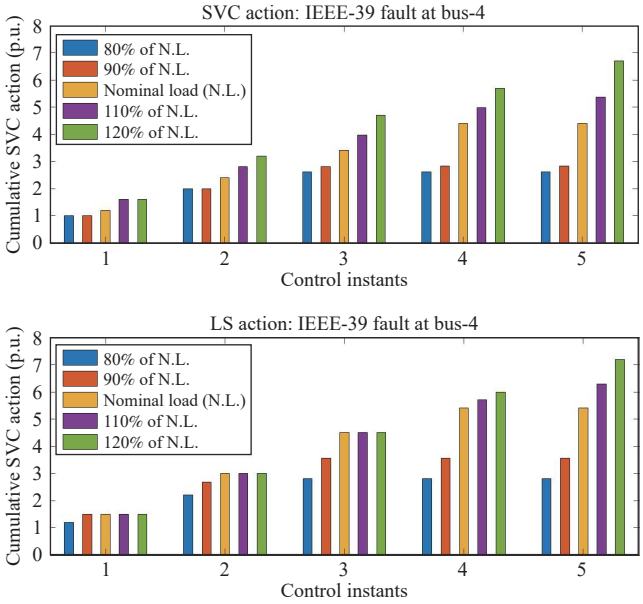

Fig. 15. SVC and LS controls for IEEE 39-bus system for fault at bus-4.

F. Comparison of Results

Finally, and importantly, the average online computation times of the traditional MPC [9] and the proposed online scheme are compared in Table IV, which demonstrates that the proposed scheme is ~20-fold faster than the original offline computed MPC implementation and takes under 0.3 s to compute a control at each online decision instant, which is comparable to the one used in practice, making MPC real-time and practical for power systems. It is important to note that even the traditional controllers, e.g., UVLS relaying scheme, generally needs ~ 0.5 s to decide a control action [38]. For our implementation and computation, we used an Intel(R) Core(TM) i7-4790 CPU @ 3.60GHz processor with 16 GB RAM. In addition to the time performance, we compared the

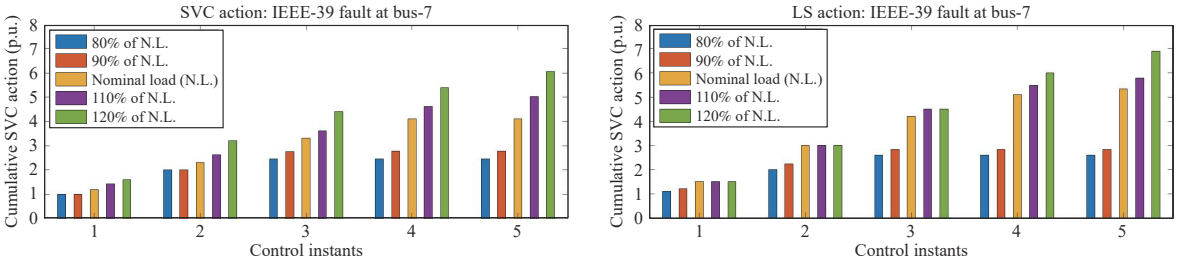

Fig. 16. SVC and LS controls for IEEE 39-bus system for fault at bus-7.

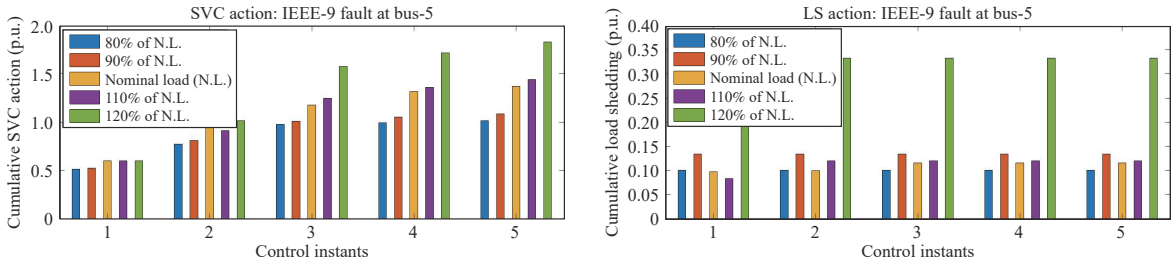

Fig. 17. SVC and LS controls for IEEE 9-bus system for fault at bus-5.

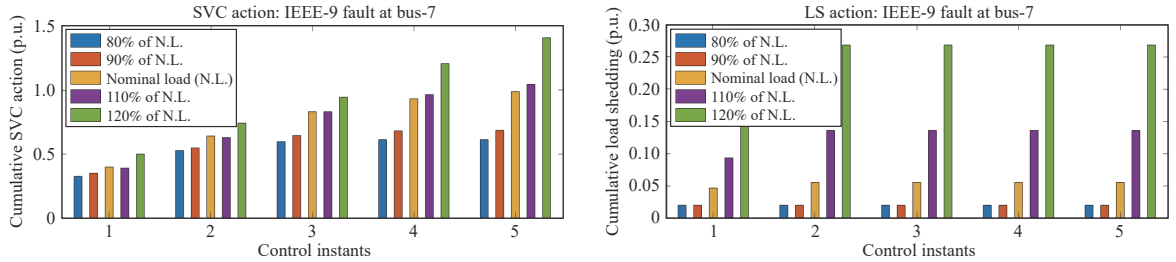

Fig. 18. SVC and LS controls for IEEE 9-bus system for fault at bus-7.

TABLE III
TOPOLOGICAL INFORMATION

Fault at bus-15		Fault at bus-21	
Topology-1	Line 3-18 removed	Topology-4	Line 4-14 removed
Topology-2	Line 10-13 removed	Topology-5	Line 8-9 removed
Topology-3	Line 26-29 removed	Topology-6	Line 26-27 removed
Fault at bus-4		**Fault at bus-7**	
Topology-7	Line 10-11 removed	Topology-10	Line 6-5 removed
Topology-8	Line 14-15 removed	Topology-11	Line 22-23 removed
Topology-9	Line 28-29 removed	Topology-12	Line 17-18 removed

voltage and control performance of our proposed method and the traditional MPC [9]. For this, we define a *performance measure* $\mathcal{J}$ as an aggregation of the squared sum of the voltage trajectory deviations of all buses with respect to $V_{\text{ref}} = 1.00$ p.u., and total applied control input with respective weights (like (7a)). Tables V and VI show that the performance measure $\mathcal{J}$ is almost same (maximum percentage error $\approx 0.9\%$) for the traditional MPC [9] and the proposed online scheme for different fault scenarios under randomly selected load condition. A representative voltage plot to show the similarity is given in Fig. 24. Thus while a speed up of 20-fold is obtained through the proposed MPC acceleration method, at the same time, there is no loss of control performance.

VII. CONCLUSIONS

The paper proposed a framework for *real-time* implementation of MPC in power systems for the first time. A combination of offline MPC-based control optimization for *nominal system*, and an iterative online control correction based on measurements of *real system* is proposed, where the online step is further sped through the introduction of trained NNs for voltage trajectory prediction and its sensitivity estimation. By exploring the space in the neighborhood of the nominal trajectory of offline computed control, the search space for NN training was drastically reduced to make it practical. The

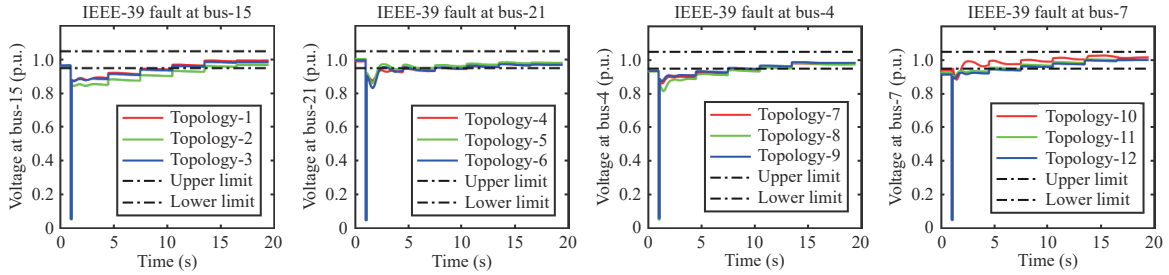

Fig. 19. IEEE 39-bus system: Voltage profile with online MPC control.

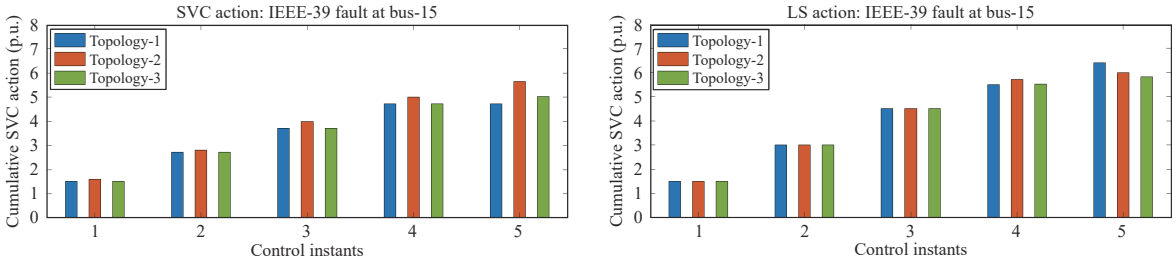

Fig. 20. IEEE 39-bus system: SVC and LS controls for fault at bus-15.

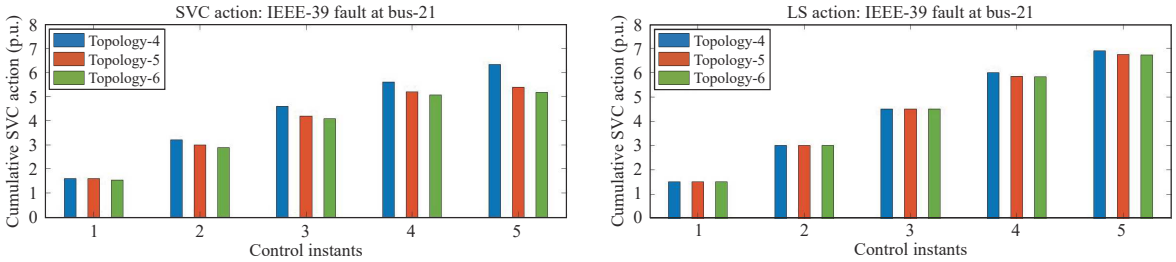

Fig. 21. IEEE 39-bus system: SVC and LS controls for fault at bus-21.

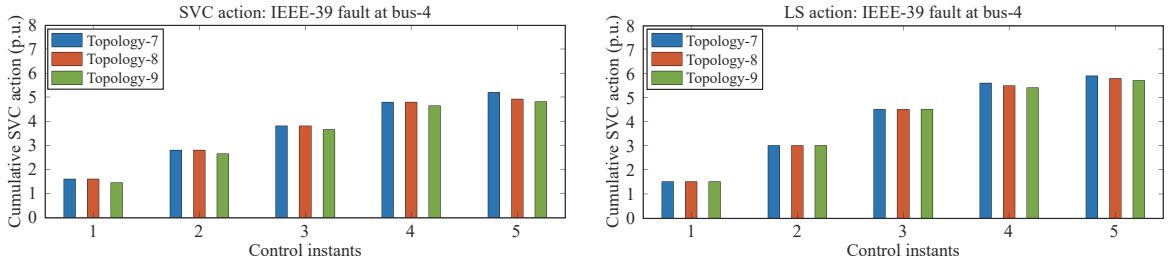

Fig. 22. IEEE 39-bus system: SVC and LS controls for fault at bus-4.

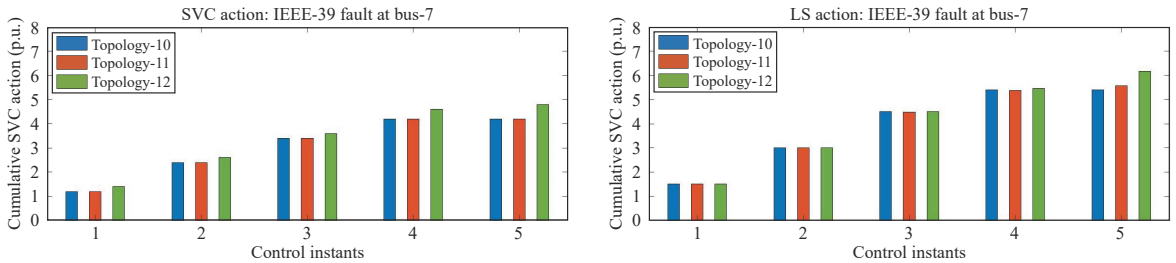

Fig. 23. IEEE 39-bus system: SVC and LS controls for fault at bus-7.

test results applied to IEEE 9-bus and 39-bus systems show the remarkable performance of the newly proposed scheme in terms of efficacy, robustness with respect to load variations, and online computation time, which has been reduced to a level comparable to traditional control computations (fraction of second), making the real-time implementation of the MPC practical. Future research directions can include quantification of resilience indices [39] of the MPC-controlled system.

TABLE IV
COMPARISON OF COMPUTATION TIME

Method	Average time	
	IEEE 9-bus	IEEE 39-bus
Traditional MPC [9]	4.50 sec/step	7.00 sec/step
Proposed method	0.27 sec/step	0.29 sec/step

TABLE V
COMPARISON OF PERFORMANCE FOR IEEE-39 BUS SYSTEM

Scenarios	Performance measure ($\mathcal{J}$)	
	Traditional MPC [9]	Proposed method
Fault at bus-15	52.6062	52.1633
Fault at bus-21	36.1988	36.4880
Fault at bus-4	38.8621	39.2018
Fault at bus-7	42.5220	42.2615

TABLE VI
COMPARISON OF PERFORMANCE FOR IEEE-9 BUS SYSTEM

Scenarios	Performance measure ($\mathcal{J}$)	
	Traditional MPC [9]	Proposed method
Fault at bus-5	17.8349	17.9748
Fault at bus-7	12.8355	12.9326

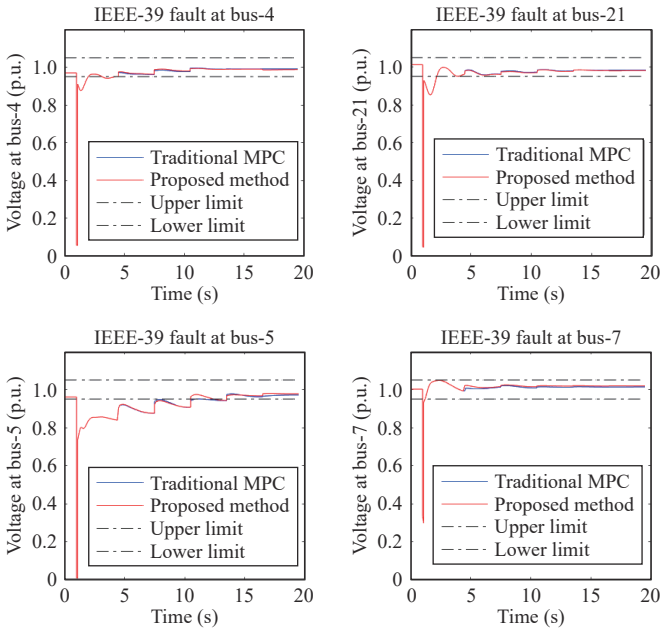

Fig. 24. Comparison voltage plots between traditional MPC and proposed method.

REFERENCES

[1] N. Hatziargyriou, J. Milanovic, C. Rahmann, V. Ajjarapu, C. Canizares, I. Erlich, D. Hill, I. Hiskens, I. Kamwa, B. Pal, P. Pourbeik, J. Sanchez-Gasca, A. Stankovic, T. Van Cutsem, V. Vittal, and C. Vournas, "Definition and classification of power system stability-revisited & extended," *IEEE Trans. Power Syst.*, vol. 36, no. 4, pp. 3271–3281, Jul. 2021.

[2] L. Robitzky, T. Weckesser, U. Häger, C. Rehtanz, and T. Van Cutsem, "Agent-based identification and control of voltage emergency situations," *IET Gener., Transm. Distrib.*, vol. 12, no. 6, pp. 1446–1454, Mar. 2018.

[3] C. Vournas and M. Karystianos, "Load tap changers in emergency and preventive voltage stability control," *IEEE Trans. Power Syst.*, vol. 19, no. 1, pp. 492–498, Feb. 2004.

[4] California ISO, "Notification of revised RC west operating procedure RC0410 - System Emergencies." 2021. [Online]. Available: https://www.caiso.com/Documents/Notification-RevisedRCWestOperatingProceduresRC0210-RC0410-RC0660.html.

[5] D. Pal, B. Mallikarjuna, P. Gopakumar, M. J. B. Reddy, B. K. Panigrahi, and D. K. Mohanta, "Probabilistic study of undervoltage load shedding scheme to mitigate the impact of protection system hidden failures," *IEEE Syst. J.*, vol. 14, no. 1, pp. 862–869, Mar. 2020.

[6] M. Glavic and T. Van Cutsem, "Some reflections on model predictive control of transmission voltages," in *Proc. 38th North American Power Symp.*, Carbondale, USA, 2006, pp. 625−632.

[7] I. A. Hiskens and M. Vrakopoulou, "Model predictive control for power networks," in *Encyclopedia of Systems and Control*, J. Baillieul and T. Samad, Eds. Cham, Switzerland: Springer, 2021, pp. 1234−1239.

[8] S. J. Geng and I. A. Hiskens, "Second-order trajectory sensitivity analysis of hybrid systems," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 66, no. 5, pp. 1922–1934, May 2019.

[9] L. C. Jin, R. Kumar, and N. Elia, "Model predictive control-based real-time power system protection schemes," *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 988–998, May 2010.

[10] G. J. Hou and V. Vittal, "Trajectory sensitivity based preventive control of voltage instability considering load uncertainties," *IEEE Trans. Power Syst.*, vol. 27, no. 4, pp. 2280–2288, Nov. 2012.

[11] Y. Zhang, M. Liu, W. Zhang, W. C. Sun, X. W. Hu, and G. Kong, "Power system voltage correction scheme based on adaptive horizon model predictive control," *Appl. Sci.*, vol. 8, no. 4, p. 641, Apr. 2018.

[12] M. S. Mahmoud and M. O. Oyedeji, "Adaptive and predictive control strategies for wind turbine systems: A survey," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 2, pp. 364–378, Mar. 2019.

[13] C. C. Jin, W. D. Li, J. K. Shen, P. Li, L. Liu, and K. R. Wen, "Active frequency response based on model predictive control for bulk power system," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3002–3013, Jul. 2019.

[14] J. Z. Liu, Q. Yao, and Y. Hu, "Model predictive control for load frequency of hybrid power system with wind power and thermal power," *Energy*, vol. 172, pp. 555–565, Apr. 2019.

[15] A. Oshnoei, M. Kheradmandi, R. Khezri, and A. Mahmoudi, "Robust model predictive control of gate-controlled series capacitor for LFC of power systems," *IEEE Trans. Ind. Inf.*, vol. 17, no. 7, pp. 4766–4776, Jul. 2021.

[16] R. K. Subroto, K. L. Lian, C. C. Chu, and C. J. Liao, "A fast frequency control based on model predictive control taking into account of optimal allocation of power from the energy storage system," *IEEE Trans. Power Delivery*, vol. 36, no. 4, pp. 2467–2478, Aug. 2021.

[17] Z. B. Zhang, O. Babayomi, T. Dragicevic, R. Heydari, C. Garcia, J. Rodriguez, and R. Kennel, "Advances and opportunities in the model predictive control of microgrids: Part I–primary layer," *Int. J. Electr. Power Energy Syst.*, vol. 134, p. 107411, Jan. 2022.

[18] F. Kamal and B. Chowdhury, "Model predictive control and optimization of networked microgrids," *Int. J. Electr. Power Energy Syst.*, vol. 138, p. 107804, Jun. 2022.

[19] Z. D. Zhang, D. X. Zhang, and R. C. Qiu, "Deep reinforcement learning for power system applications: An overview," *CSEE J. Power Energy Syst.*, vol. 6, no. 1, pp. 213–225, Mar. 2020.

[20] Q. H. Huang, R. K. Huang, W. T. Hao, J. Tan, R. Fan, and Z. Y. Huang, "Adaptive power system emergency control using deep reinforcement learning," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1171–1182, Mar. 2020.

[21] C. Y. Chen, M. J. Cui, F. X. Li, S. F. Yin, and X. N. Wang, "Model-free emergency frequency control based on reinforcement learning," *IEEE Trans. Ind. Inf.*, vol. 17, no. 4, pp. 2336–2346, Apr. 2021.

[22] J. Xie and W. Sun, "Distributional deep reinforcement learning-based emergency frequency control," *IEEE Trans. Power Syst.*, vol. 37, no. 4, pp. 2720–2730, Jul. 2022.

[23] X. Chen, G. N. Qu, Y. J. Tang, S. Low, and N. Li, "Reinforcement learning for selective key applications in power systems: Recent advances and future challenges," *IEEE Trans. Smart Grid*, vol. 13, no. 4, pp. 2935–2958, Jul. 2022.

[24] K. L. Teo, B. Li, C. J. Yu, V. Rehbock, *Applied and Computational Optimal Control: A Control Parametrization Approach*. Cham, Switzerland: Springer, 2021.

[25] J. Machowski, Z. Lubosny, J. W. Bialek, and J. R. Bumby, *Power System Dynamics: Stability and Control*. 3rd ed. John Wiley & Sons, 2020.

[26] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. vol. 1, Athena Scientific, 2012.

[27] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, " Review on model predictive control: An engineering perspective," *Int. J. Adv. Manuf. Technol.*, vol. 117, no. 5, pp. 1327–1349, Aug. 2021.

[28] I. A. Hiskens and M. A. Pai, "Trajectory sensitivity analysis of hybrid systems," *IEEE Trans. Circuits Syst. I: Fundam. Theory Appl.*, vol. 47, no. 2, pp. 204–220, Feb. 2000.

[29] Y. F. Guo, Q. W. Wu, H. L. Gao, X. Y. Chen, J. Østergaard, and H. H. Xin, " MPC-based coordinated voltage regulation for distribution networks with distributed generation and energy storage system," *IEEE Trans. Sustainable Energy*, vol. 10, no. 4, pp. 1731–1739, Oct. 2019.

[30] J. Lago, F. De Ridder, and B. De Schutter, "Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms," *Appl. Energy*, vol. 221, pp. 386–405, Jul. 2018.

[31] M. Chammas, A. Makhoul, and J. Demerjian, "An efficient data model for energy prediction using wireless sensors," *Comput. Electr. Eng.*, vol. 76, pp. 249–257, Jun. 2019.

[32] M. A. F. B. Lima, L. M. F. Ramírez, P. C. M. Carvalho, J. G. Batista, and D. M. Freitas, "A comparison between deep learning and support vector regression techniques applied to solar forecast in Spain," *J. Sol. Energy Eng.*, vol. 144, no. 1, p. 010802, Feb. 2022.

[33] F. Milano, " An open source power system analysis toolbox," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1199–1206, Aug. 2005.

[34] R. R. Hossain and R. Kumar., "Computation of trajectory sensitivities with respect to control and implementation in PSAT," in *Proc. 16th Int. Conf. Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, Prague, Czech, 2019, pp. 752−759.

[35] K. Zhang, J. Zhang, P. D. Xu, T. L. Gao, and D. W. Gao, "Explainable AI in deep reinforcement learning models for power system emergency control," *IEEE Trans. Comput. Soc. Syst.*, vol. 9, no. 2, pp. 419–427, Apr. 2022.

[36] T. Schreiber, " Interdisciplinary application of nonlinear time series methods," *Phys. Rep.*, vol. 308, no. 1, pp. 1–64, Jan. 1999.

[37] C. Q. Cheng, A. Sa-Ngasoongsong, O. Beyca, T. Le, H. Yang, Z. Y. Kong, and S. T. S. Bukkapatnam, "Time series forecasting for nonlinear and non-stationary processes: A review and comparative study," *IIE Trans.*, vol. 47, no. 10, pp. 1053–1071, Apr. 2015.

[38] L. Ye, J. L. Sun, T. B. Zhou, J. Zhang, W. Z. Sun, and H. L. Xi, "A practical under-voltage load shedding strategy for regional power grid considering multiple operating modes," *Energy Rep.*, vol. 7, no. S1, pp. 175–182, Apr. 2021.

[39] S. Talukder, M. Ibrahim, and R. Kumar, " Resilience indices for power/cyberphysical systems," *IEEE Trans. Syst.*, *Man*, *Cybern.: Syst.*, vol. 51, no. 4, pp. 2159–2172, Apr. 2021.

Ramij Raja Hossain (Graduate Student Member, IEEE) received the bachelor degree in electrical engineering from Jadavpur University, India in 2013. He is currently pursuing the Ph.D. degree in electrical engineering with the Department of Electrical and Computer Engineering, Iowa State University, USA. He served as a Senior Engineer, Distribution with CESC Limited, India from 2013 to 2018. His current research interests include data-driven control, learning accelerated MPC, distributed optimization, and artificial intelligence (AI) based approaches for security, stability, and control of power systems.

Ratnesh Kumar (Fellow, IEEE) is a Palmer Professor at the Department of Electrical and Computer Engineering, Iowa State University, USA, where he directs the ESSeNCE (Embedded Software, Sensors, Networks, Cyberphysical, and Energy) Lab. Previously, he held faculty position at the University of Kentucky, and various visiting positions with the University of Maryland (College Park), the Applied Research Laboratory at the Pennsylvania State University (State College), the NASA Ames, the Idaho National Laboratory, the United Technologies Research Center, and the Air Force Research Laboratory. He received the B. Tech. degree in electrical engineering from IIT Kanpur, India, in 1987 and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Texas at Austin in 1989 and 1991 respectively. Ratnesh is a Fellow of IEEE, also a Fellow of AAAS, and was a Distinguished Lecturer of the IEEE Control Systems Society. He is a recipient of *D. R. Boylan Eminent Faculty Award for Research* and *Award for Outstanding Achievement in Research* from Iowa State University, and also the *Distinguished Alumni Award* from IIT Kanpur. Ratnesh received Gold Medals for the Best EE Undergrad, the Best All Rounder, and the Best EE Project from IIT Kanpur, and the Best Dissertation Award from UT Austin, the Best Paper Award from the IEEE Transactions on Automation Science and Engineering, and has been Keynote Speaker and paper award recipient from multiple conferences. He is or has been an Editor of several journals (including of IEEE, SIAM, ACM, Springer, IET, MDPI).