

Real-time Model Predictive Control of a DC-DC Buck Converter

Kamlesh Sawant*, Ryan Caverly†, Jason Poon‡, Sairaj Dhople*

**Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, United States*
sawan035@umn.edu, sdhople@umn.edu

†*Aerospace Engineering and Mechanics, University of Minnesota, Minneapolis, MN, United States*
rcaverly@umn.edu

‡*Electrical Engineering, Cal Poly, San Luis Obispo, CA, United States*
jasonp@calpoly.edu

Abstract—We propose a digitally assisted analog computing circuit for real-time model predictive control (MPC) of a DC-DC buck converter. The computing circuit comprises analog elements for speed and digital components for programmability. It implements gradient-flow dynamics for a penalty-based reformulation of the quadratic optimization problem corresponding to the original MPC formulation. The MPC problem is set up to regulate output voltage while explicitly enforcing constraints on the buck converter's inductor current and duty cycle. Simulation results in a closed-loop configuration demonstrate superior dynamic response with lower settling time and overshoot compared to the Type-III controller and linear quadratic regulator (LQR). The proposed approach achieves accuracy similar to the numerically computed optimal solution from the interior-point-convex algorithm of MATLAB's `quadprog` solver while offering real-time implementation capability.

Index Terms—Analog circuits, DC-DC power converters, digital circuits, gradient methods, optimization, predictive control.

I. INTRODUCTION

The drive to reduce the size of passive components has increased switching frequencies for power converters across applications. This reduces the passive components' ability to store energy, leading to faster dynamics. These rapid dynamics necessitate that companion control and optimization routines be executed faster. However, bottlenecks due to delays in digital circuits (frequently leveraged for control and optimization) are unavoidable. As a result, it is common to resort to offline routines, such as look-up tables, rather than real-time optimization [1]–[4]. This compromise often results in sub-optimal converter performance and robustness.

This paper presents a digitally assisted analog computing circuit (DAACC) that implements real-time model predictive control (MPC) of a DC-DC buck converter. The optimization problem is formulated to regulate output voltage while enforcing bounds on inductor current and the duty cycle. The DAACC is built with operational amplifiers (op-amps), diodes, and passive circuit elements judiciously assisted by a digital circuit. The proposed solution preserves desirable attributes of digital computing (programmability customized to the application) and analog circuits (speed of response). From

a mathematical and control-theoretic point of view, the states of the DAACC realize gradient-flow dynamics for a penalty-based reformulation of the original MPC problem. Equilibria of the DAACC are engineered to coincide with the first-order Karush-Kuhn-Tucker (KKT) conditions for optimality [5].

Several works have proposed synthesizing analog circuits to simulate and optimize a wide range of physical systems, including power converters [3], [6]–[16]. Similarly, there has been significant interest in MPC for power electronics [17]–[27]. A comprehensive review of MPC formulations in power electronics, including computational complexity and stability, can be found in [2]. Most solutions for power electronics applications have been implemented in the digital domain, with the exception being [11]. Our implementation involving penalty-based gradient-flow dynamics offers a different solution strategy compared to [11]. While not directly pursued in this work, it can also encompass nonlinear objective (cost) functions [3], [6], [28]. We demonstrate via switching-level simulations that the proposed DAACC facilitates real-time implementation of MPC.

The remainder of this paper is structured as follows. Section II introduces the formulation of the MPC problem. The reformulation of the MPC problem using the penalty method and the circuit-based realization are described in Section III. Section IV presents simulation results for the MPC problem and a comparative analysis with conventional control methods. Finally, Section V concludes the paper.

II. MODEL PREDICTIVE CONTROL (MPC) PROBLEM FORMULATION

This section presents the Model Predictive Control (MPC) problem formulation for a DC-DC buck converter. We begin by introducing the state-space model of the buck converter, followed by the MPC problem formulation.

A. Buck Converter State-space Model

We develop an MPC solution for a buck converter to step down input voltage, V_{in} , to output voltage, v_C , across a (variable) load modeled by resistance R_{load} . (See Fig. 1.) The DAACC regulates v_C to a desired reference voltage, v_{ref} ,

This work was supported in part by the National Science Foundation (NSF) through awards 2305431 and 2305432.

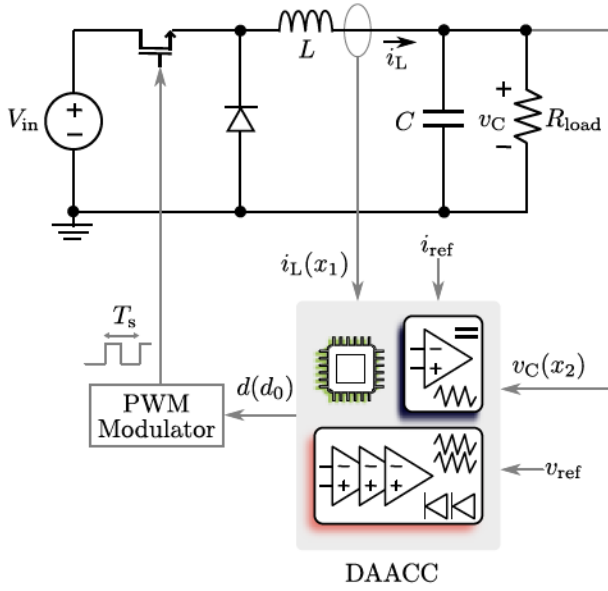


Fig. 1: A DC-DC buck converter with a DAACC implementing MPC. The goal is to enforce constraints $i_L \leq I_{\max}$ (to avoid inductor saturation and limit switch current) and $d_{\min} \leq d \leq d_{\max}$ while regulating v_C to v_{ref} .

by computing a control input duty cycle, d . The continuous-time dynamics of the buck converter are governed by state-space equations given in (1) [29]. The state variables, inductor current and capacitor voltage, are represented as $x = [x_1 \ x_2]^T = [i_L \ v_C]^T$. The control input $u = d$ is used to regulate the converter output voltage. System dynamics can be expressed in state-space form

$$\dot{x} = A_c x + B_c u, \quad y = C_c x, \quad (1)$$

where:

$$A_c = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{R_{\text{load}}C} \end{bmatrix}, \quad B_c = \begin{bmatrix} \frac{V_{\text{in}}}{L} \\ 0 \end{bmatrix}, \quad C_c = [0 \ 1].$$

B. MPC Problem Formulation

We consider an MPC formulation that minimizes a quadratic objective function f over a finite prediction horizon, N , with constraints on states and control inputs: [30], [31]

$$\min_{\substack{d_0, d_1, \dots, d_{N-1} \\ x_1, x_2, \dots, x_{N-1}}} f = \sum_{k=1}^{N-1} (x_{\text{ref}} - x_k)^T Q (x_{\text{ref}} - x_k) + \sum_{k=0}^{N-1} d_k^T R d_k, \quad (2a)$$

$$\text{s.t. } x_{k+1} = A_d x_k + B_d d_k, \quad \forall k = 0, \dots, N-1, \quad (2b)$$

$$x_{\min} \leq x_k \leq x_{\max}, \quad \forall k = 1, \dots, N-1, \quad (2c)$$

$$d_{\min} \leq d_k \leq d_{\max}, \quad \forall k = 0, \dots, N-1, \quad (2d)$$

where $x_{\text{ref}} = [i_{\text{ref}} \ v_{\text{ref}}]^T$ is the reference state, and the duty cycles over the horizon, d_k 's are the optimization variables. The state of the system at the $k+1$ -st instant, x_{k+1} , is predicted using the current state x_k , and the control input

d_k by discretizing the dynamics (1) using a zeroth-order hold. Symmetric weighting matrices $Q \succeq 0$ and $R \succ 0$ are tuned for desired closed-loop performance. The limits on state variable values are given by $x_{\min}$ and $x_{\max}$, while the bounds on control input d_k are given by $d_{\min}$ and $d_{\max}$. The problem is solved in each sampling period (which is equal to the switching period in our case), T_s , with initial state measurement, x_0 , and the duty cycle, d , is set to be the optimizer d_0^* . Matrices A_d, B_d are obtained through the discretization of the continuous system dynamics (1) over T_s :

$$A_d = e^{A_c T_s}, \quad B_d = \left(\int_{\tau=0}^{T_s} e^{A_c \tau} d\tau \right) B_c.$$

C. MPC with Terminal Cost for Stability

Unlike the stability properties of infinite-horizon linear quadratic regulator (LQR), in finite-horizon MPC, ensuring stability—convergence of the solution to the optimization problem—requires careful design of the objective function, prediction horizon, and constraints [30], [31]. To promote the stability of the MPC formulation, a terminal cost term, $(x_{\text{ref}} - x_N)^T P (x_{\text{ref}} - x_N)$, can be added to the objective function (2a) of the optimization problem in (2) to approximate the infinite-horizon LQR behavior. The modified objective function becomes: [30], [31]

$$\min_{\substack{d_0, d_1, \dots, d_{N-1} \\ x_1, x_2, \dots, x_N}} f = \sum_{k=1}^{N-1} (x_{\text{ref}} - x_k)^T Q (x_{\text{ref}} - x_k) + \sum_{k=0}^{N-1} d_k^T R d_k + (x_{\text{ref}} - x_N)^T P (x_{\text{ref}} - x_N). \quad (3)$$

The constraints remain the same as in (2b), (2c), and (2d). The matrix P is determined from the discrete algebraic Riccati equation: [31]

$$P = Q + A_d^T P A_d - (A_d^T P B_d) (R + B_d^T P B_d)^{-1} (B_d^T P A_d). \quad (4)$$

While we focus on a quadratic objective function, it is worth noting that simpler objective functions (e.g., $\|\cdot\|_1$ norm) are used to reduce the computational burden in fast-dynamic power electronic systems [2]. These alternatives often trade off closed-loop performance for faster computation.

III. SYNTHESIS OF THE DIGITALLY ASSISTED ANALOG COMPUTING CIRCUIT (DAACC) FOR MPC

This section presents the design of a Digitally Assisted Analog Computing Circuit (DAACC) implementing the MPC problem from Section II. We transform the MPC problem in terms of control variables, reformulate it using the penalty method to handle constraints, and describe the DAACC implementation using gradient-flow dynamics for a real-time solution.

A. Transformation of MPC Problem

To implement the MPC problem using a DAACC, the problem is first transformed so that the objective function only features the control variables, d_k [30]. This transformation eliminates the need for explicit state prediction in the circuit implementation. It is achieved by using the discrete-time state-space solution to the equality constraint (2b), which expresses the current state, x_k , in terms of the initial measured state, x_0 , and the sequence of control inputs, $d_k, \forall k = 1, \dots, N$:

$$x_k = A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j. \quad (5)$$

Substituting (5) into the objective function f from (2a) and constraints from (2c), we obtain a transformed MPC problem that depends only on the control variables d_k :

$$\min_{d_0, d_1, \dots, d_{N-1}} \tilde{f} = \sum_{k=1}^{N-1} (x_{\text{ref}} - (A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j))^T Q (x_{\text{ref}} - (A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j)) + \sum_{k=0}^{N-1} d_k^T R d_k, \quad (6a)$$

$$\text{s.t. } x_{\min} \leq A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j \leq x_{\max}, \quad \forall k = 1, \dots, N, \quad (6b)$$

$$d_{\min} \leq d_k \leq d_{\max}, \quad \forall k = 0, \dots, N-1. \quad (6c)$$

Notice that the equality constraint (2b) is eliminated from the original MPC problem (2), as the system dynamics are now directly represented in the objective function and inequality constraints on the state.

B. Penalty-based Reformulation of the MPC Problem

To solve the constrained optimization problem in (6) using a DAACC, we further reformulate it into an unconstrained problem using the penalty method [5]. This is accomplished by penalizing inequality constraint violations via a penalty function $\phi: \mathbb{R} \rightarrow \mathbb{R}$. The problem then boils down to:

$$\min_{d \in \mathbb{R}^N} \tilde{f} + \frac{\rho}{2} \phi, \quad (7)$$

where

$$\begin{aligned} \phi = & \sum_{k=1}^{N-1} \left(\left(\min(0, A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j - x_{\min}) \right)^2 \right. \\ & \left. + \left(\min(0, x_{\max} - (A_d^k x_0 + \sum_{j=0}^{k-1} A_d^{k-j-1} B_d d_j)) \right)^2 \right) \\ & + \sum_{k=0}^{N-1} \left((\min(0, d_k - d_{\min}))^2 + (\min(0, d_{\max} - d_k))^2 \right). \end{aligned}$$

The inequality constraints are implicitly handled through the penalty function, ϕ , which, in our implementation, is a sum of quadratic forms $(\min(\cdot, \cdot))^2$. The quadratic form ensures that the penalty is zero when the constraint is satisfied and

increases quadratically with the magnitude of the violation. A penalty parameter $\rho \in \mathbb{R}_{>0}$ determines the extent of the constraint violation applied and is selected such that $\frac{\rho}{2} \phi \gg 0$. Sufficiently large values ensure that the solution to (7) closely approximates the solution to the original constrained problem in (2). This penalty-based reformulation allows us to solve the MPC problem using gradient-based methods, which can be implemented using DAACC. The following subsection will detail the DAACC implementation based on gradient-flow dynamics to solve this reformulated problem in real-time.

C. DAACC Based on Gradient-flow Dynamics

The reformulated unconstrained problem is solved by applying the following continuous-time gradient-flow dynamics for all $d_k, k = 0, \dots, N-1$:

$$\frac{dd_k}{dt} = -\gamma \left(\frac{\partial \tilde{f}}{\partial d_k} + \frac{\rho}{2} \frac{\partial \phi}{\partial d_k} \right), \quad (8)$$

where $\gamma \in \mathbb{R}_{>0}$ is a constant, $\frac{\partial \tilde{f}}{\partial d_k}$ is the gradient of the function $\tilde{f}(\cdot)$, and $\frac{\partial \phi}{\partial d_k}$ is the gradient of the penalty function $\phi(\cdot)$ given by:

$$\begin{aligned} \frac{\partial \tilde{f}}{\partial d_k} = & 2R d_k - 2 \sum_{\ell=k+1}^{N-1} (x_{\text{ref}} - x_{\ell})^T Q A_d^{\ell-k-1} B_d \\ & - 2 (x_{\text{ref}} - x_N)^T P A_d^{N-k-1} B_d, \end{aligned} \quad (9)$$

$$\begin{aligned} \frac{\partial \phi}{\partial d_k} = & 2B_d (\min(0, x_{k+1} - x_{\min}) - \min(0, x_{\max} - x_{k+1})) \\ & + 2(\min(0, d_k - d_{\min}) - \min(0, d_{\max} - d_k)). \end{aligned} \quad (10)$$

The circuits in Fig. 2 show the proposed DAACC implementing dynamics (8) for prediction horizons $N = 1$ with terminal cost term ($P \neq 0$), $N = 2$, and $N = 3$ without terminal cost term ($P = 0$) respectively. To keep the circuit size low, we enforce the inductor current constraint only at the first prediction step. The circuit realization relates to the dynamics (8) via the parametric relationship:

$$\gamma = \frac{1}{R_{\gamma} C_{\gamma}}, \quad \rho = \frac{R_{\rho}}{R_o}.$$

Analog Subsystem: The terms in $\frac{\rho}{2} \frac{\partial \phi}{\partial d_k}$ are implemented via an analog circuit consisting of op-amps, diodes, and resistors. Inputs to the analog circuit stage are either constants (such as $I_{\max}$, $d_{\max}$, $d_{\min}$) or derived from inverting op-amp stages (not shown in the Fig. 2) that linearly manipulate voltages such as $A_{d11} i_L$, $A_{d12} v_C$, $B_{d11} d_0$, where the subscripts denote the respective elements of matrices A_d and B_d (e.g., A_{d11} is the (1,1) element of matrix A_d). High-gain inverting amplifiers with a gain (R_{ρ}/R_o) , then, impose significant penalties (ρ) to constrain d and i_L within bounds. The $\min(\cdot, \cdot)$ function is realized using an op-amp paired with a diode, where R_{lim} restricts current flowing through the op-amp output terminal and associated diode. Unity-gain buffer follows the $\min(\cdot, \cdot)$ function stage to provide sufficient buffering. Finally, we incorporate an inverting amplifier, which provides a gain $(R_{\ell,k}/R_o)$ equivalent to scalar term in $\frac{\partial \phi}{\partial d_k}$ (such as B_d),

where ℓ denotes the specific constraint and k the prediction step.

Digital Subsystem: Terms in $\frac{\partial \tilde{f}}{\partial d_k}$ can be generated by a digital circuit, such as a Field-Programmable Gate Array (FPGA) or digital microcontroller.

The analog integrators, each composed of a capacitor C_γ and resistor R_γ , sum the output from the digital circuit generating $\frac{\partial \tilde{f}}{\partial d_k}$ and the analog circuit implementing $\frac{R_p}{R_o} \frac{\partial \phi}{\partial d_k}$ to produce the sequence of optimal duty cycles d_k^* of which, d_0^* is applied as the duty cycle for the subsequent sampling (switching) period.

IV. PERFORMANCE VALIDATION

This section presents simulation results for the proposed DAACC implementing MPC for a buck converter, followed by a comparative analysis with conventional control methods and numerical optimization techniques.

A. DAACC Simulation Results

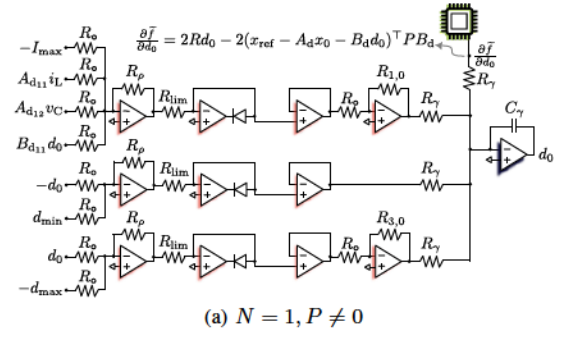
The proposed DAACC circuit implementing MPC for a buck converter, shown in Fig. 2, is simulated using the LTspice circuit simulator. The buck converter parameters used in our simulations are listed in Table I. The digital circuit is implemented using behavioral sources with its outputs limited to ± 5 V. In these LTspice simulations, we have assumed instantaneous output from the digital circuit without delay. However, it is important to note that digital delays can introduce instability depending on time delay values [15], [28], [32]. The analog circuit implementing $\frac{R_p}{R_o} \frac{\partial \phi}{\partial d_k}$ terms and integrators, utilizes AD8039 op-amps chosen for their high speed (350 MHz bandwidth) and high slew rate (425 V/ μ s). The analog circuit uses a high gain ($R_p/R_o = 100$) to implement large penalties (ρ) for constraint handling, with its outputs limited to ± 12 V.

The MPC formulation uses weighting matrices $Q \succeq 0$ and $R \succ 0$ selected as follows:

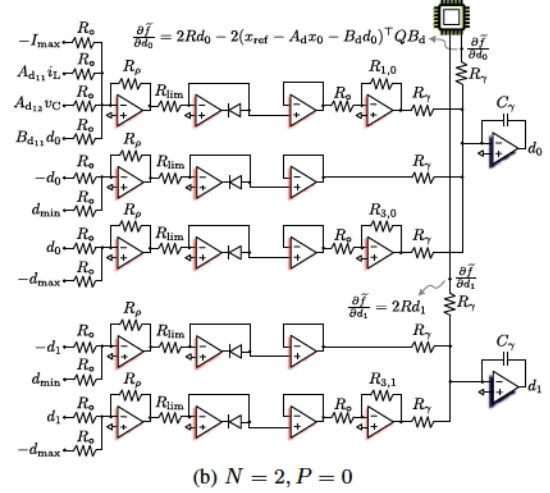
$$Q = \begin{bmatrix} 0 & 0 \\ 0 & 10^3 \end{bmatrix}, \quad R = 1. \quad (11)$$

These values are chosen to prioritize output voltage regulation while moderating control efforts. Additionally, the zero elements in Q reduce computational complexity by simplifying the calculation of terms in $\frac{\partial \tilde{f}}{\partial d_k}$.

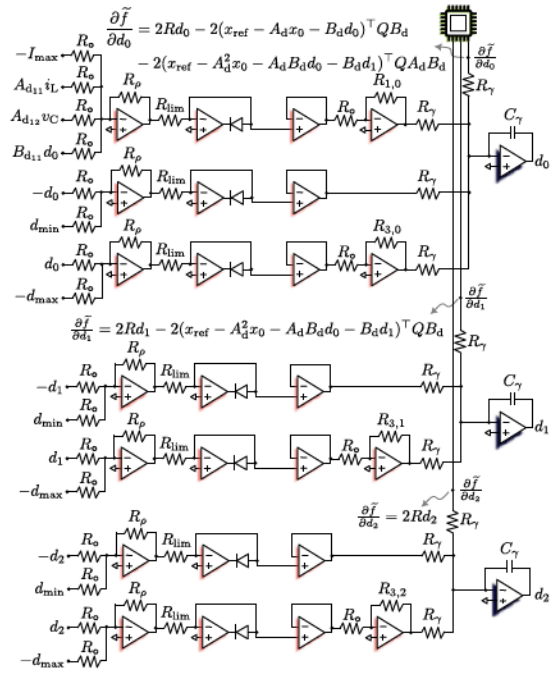
All results are reported in Figs. 3a, 3b, and 3c and described subsequently. The output (capacitor) voltage v_C aligns with the reference voltage v_{ref} of 5 V at start-up. Subsequently, v_{ref} is step changed to 10 V at time $t = 0.2$ ms, and v_C



(a) $N = 1, P \neq 0$



(b) $N = 2, P = 0$



(c) $N = 3, P = 0$

Fig. 2: Circuit implementation of the DAACC solving the MPC problem. The circuit enforces constraints $i_L \leq I_{\text{max}}$ and $d_{\text{min}} \leq d \leq d_{\text{max}}$. The circuit component values are $R_o = R_{\text{lim}} = R_{3,0} = R_{3,1} = R_{3,2} = R_\gamma = 10$ k Ω , $R_{1,0} = B_{d11} R_o$, $R_p = 1$ M Ω , and $C_\gamma = 1$ nF.

TABLE I: Buck Converter Parameters

Parameter	Value
Input voltage, V_{in}	48 V
Inductor, L	30 μ H
Capacitor, C	10 μ F
Load resistance, R_{load}	10 Ω
Switching period, T_s	1 μ s

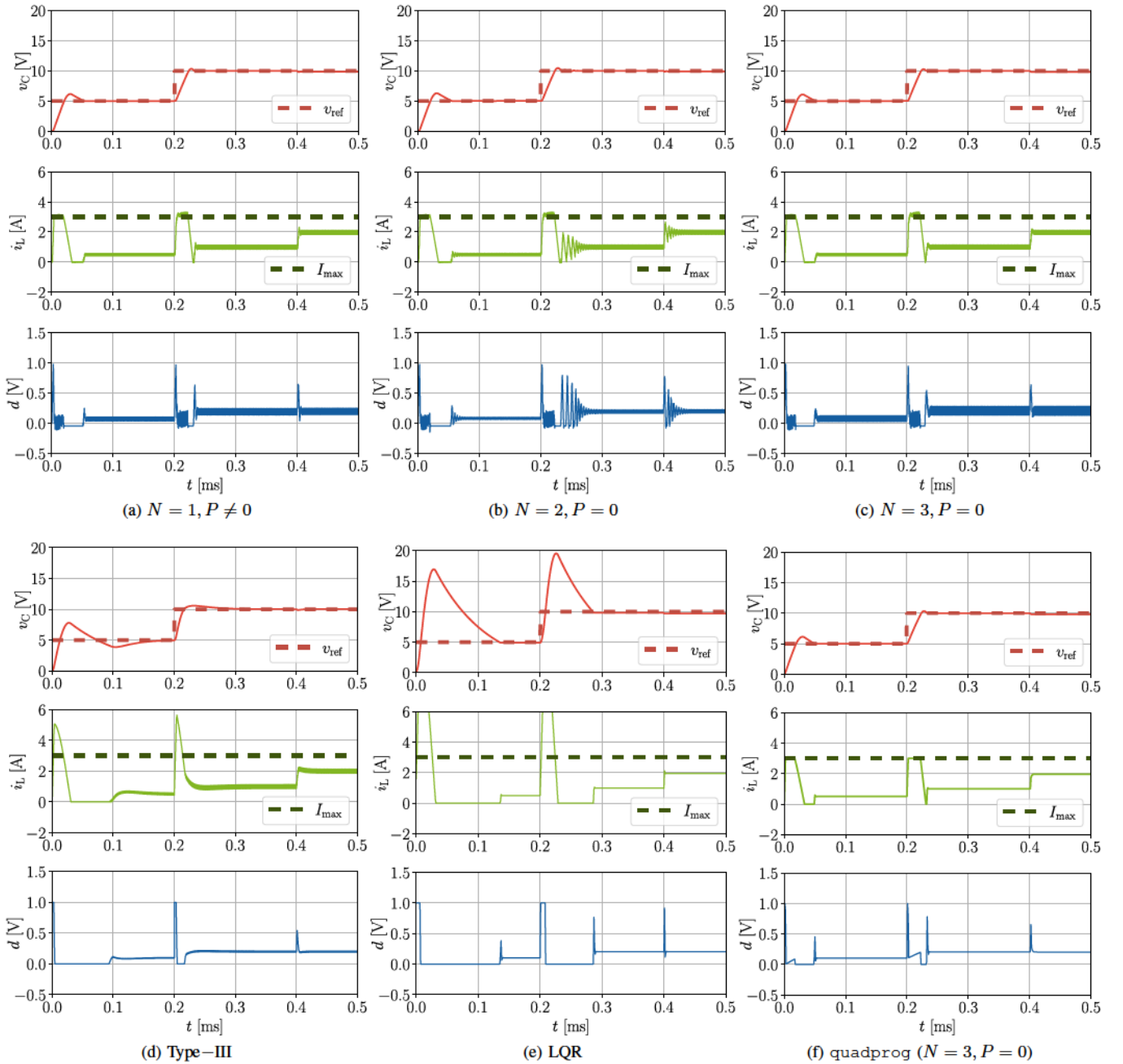


Fig. 3: Simulation results for the buck converter controlled by: (a),(b),(c) the proposed DAACC using MPC with different prediction horizons (switched-model simulation), compared with (d) Type-III controller (switched-model simulation), (e) LQR (averaged-model simulation), and (f) MATLAB's quadprog solver (averaged-model simulation).

settles within approximately $50 \mu\text{s}$. The DAACC maintains the state constraints to limit the inductor current $i_L \leq 3 \text{ A}$, which significantly reduces overshoot in v_C . The v_C regulation for a step change in R_{load} from 10Ω to 5Ω carried out at $t = 0.4 \text{ ms}$ is also depicted in Figs. 3a, 3b, and 3c.

The dynamic response of the buck converter improves with an increasing prediction horizon $N = 2$ to $N = 3$ without terminal cost term ($P = 0$), as seen in Figs. 3b and 3c, although

with the trade-off of an increased circuit size. Furthermore, the effect of including a terminal cost is demonstrated in Fig. 3a, which shows the case where $N = 1, P \neq 0$, corresponding to the objective function in (3), which includes the terminal cost term. Here, P is computed using the discrete algebraic Riccati equation (4). This achieves a better dynamic response compared to the higher prediction horizon case of $N = 2, P = 0$ shown in Fig. 3b, which corresponds to the

objective function in (2a) without the terminal cost term.

As N increases, although dynamic response improves, the number of analog components grows linearly. Assuming that the bounds on i_L are implemented at the start of the prediction horizon, there is an increase of approximately 8 op-amps per prediction horizon. Moreover, the computation in the digital circuit increases, resulting in a longer computational delay.

B. Comparative Analysis

We compare the DAACC-based MPC with a third-order Type-III controller (Fig. 3d) and discrete LQR (Fig. 3e). Type-III controller can be viewed as a proportional-integral-derivative (PID) controller with a second-order lag [33]. The transfer function $G(s)$ of a Type-III controller is:

$$G(s) = G_0 \frac{(1 + s/\omega_{z1})(1 + s/\omega_{z2})}{s(1 + s/\omega_{p1})(1 + s/\omega_{p2})},$$

where $G_0 = 1.2 \times 10^4$ is the controller gain, $\omega_{z1} = 5.1 \times 10^4$ rad/s and $\omega_{z2} = 5.7 \times 10^4$ rad/s are the angular frequencies of the zeros, and $\omega_{p1} = 3.3 \times 10^6$ rad/s and $\omega_{p2} = 3.6 \times 10^6$ rad/s are the angular frequencies of the poles. These parameters are designed using MATLAB's *sisotool* to provide a gain margin of 20 dB and a phase margin of 60° at 102 kHz. This s-domain transfer function is simulated in a closed loop with the buck converter using Plexim's PLECS simulator. Note that simulating the Type-III controller with op-amp can introduce additional limitations that further degrade performance compared to the s-domain simulation.

The LQR control minimizes a quadratic objective function J over an infinite horizon, given as: [31]

$$J = \sum_{k=0}^{\infty} \left((x_{\text{ref}} - x_k)^T Q (x_{\text{ref}} - x_k) + d_k^T R d_k \right),$$

where $d_k = K(x_{\text{ref}} - x_k)$ is the control input that minimizes J . The gain matrix K is computed in MATLAB as:

$$K = (R + B_d^T P B_d)^{-1} (B_d^T P A_d) = [0.96 \quad 7.28]$$

where P satisfies the algebraic Riccati equation given by (4). The matrices Q and R for discrete LQR control are the same as that for MPC, as given by (11). This discrete LQR controller is simulated using MATLAB in a closed loop with the buck converter.

As shown in Fig. 3, the DAACC-based MPC achieves faster settling time with a lower overshoot in the output voltage v_C while limiting i_L . In contrast, both the Type-III controller and LQR lack inherent structure for constraining i_L (which leads to overshoots in v_C). Furthermore, while DAACC-based MPC directly incorporates control input d constraints within its formulation, the conventional Type-III controller and LQR require external limiting for d . However, these traditional controllers have the advantage of lower computational complexity.

To validate the optimality of our proposed DAACC approach, we compare the DAACC case of $N = 3, P = 0$ in Fig. 3c with an equivalent MPC implementation using MATLAB's *quadprog* solver in Fig. 3f, which employs

an interior-point-convex algorithm. The *quadprog* solver returns the global optimal solution of the convex quadratic problem and serves as a benchmark for our DAACC implementation. The dynamic response obtained from MATLAB's *quadprog* solver closely aligns with that of the DAACC implementing penalty-based gradient-flow dynamics (Fig. 3c). This similarity in response indicates that both methods converge to comparable optimal solutions. However, while MATLAB's *quadprog* solver faces real-time implementation challenges, the DAACC-based MPC is designed for real-time application in fast dynamic power converters.

V. CONCLUSIONS

This paper presents a digitally assisted analog computing circuit (DAACC) for real-time MPC applied to a DC-DC buck converter. The proposed DAACC yields a fast dynamic response with minimal overshoots in the output voltage while maintaining limits on the inductor current and duty cycle. The DAACC outperforms conventional Type-III and LQR controllers in dynamic response and constraint handling. In addition, while achieving performance comparable to MATLAB's *quadprog* solver, the DAACC-based MPC enables real-time implementation for power converters with rapid dynamics.

REFERENCES

- [1] S. Kouro, M. A. Perez, J. Rodriguez, A. M. Llor, and H. A. Young, "Model predictive control: MPC's role in the evolution of power electronics," *IEEE Ind. Electron. Mag.*, vol. 9, no. 4, pp. 8–21, 2015.
- [2] P. Karamanakos, E. Liegmann, T. Geyer, and R. Kennel, "Model predictive control of power electronic systems: Methods, results, and challenges," *IEEE Open J. Ind. Appl.*, vol. 1, pp. 95–114, 2020.
- [3] J. Poon, M. Sinha, S. V. Dhople, and J. Rivas-Davila, "Real-time selective harmonic minimization using a hybrid analog/digital computing method," *IEEE Trans. Power Electron.*, vol. 37, no. 5, pp. 5078–5088, 2022.
- [4] C. Wang, Q. Zhang, W. Yu, and K. Yang, "A comprehensive review of solving selective harmonic elimination problem with algebraic algorithms," *IEEE Trans. Power Electron.*, vol. 39, no. 1, pp. 850–868, 2024.
- [5] J. R. R. A. Martins and A. Ning, *Engineering Design Optimization*. Cambridge: Cambridge University Press, 2021.
- [6] M. Kennedy and L. Chua, "Neural networks for nonlinear programming," *IEEE Trans. Circuits Syst.*, vol. 35, no. 5, pp. 554–562, 1988.
- [7] J. M. Quero, E. F. Camacho, and L. G. Franquelo, "Neural network for constrained predictive control," *IEEE Trans. Circuits Syst. I*, vol. 40, no. 9, pp. 621–626, 1993.
- [8] Y. Xia and G. Feng, "An improved neural network for convex quadratic optimization with application to real-time beamforming," *Neurocomputing*, vol. 64, pp. 359–374, 2005.
- [9] P. Mitros, "Constraint satisfaction modules: A methodology for analog circuit design," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, USA, 2007.
- [10] S. Vichik and F. Borrelli, "Solving linear and quadratic programs with an analog circuit," *Comput. Chem. Eng.*, vol. 70, pp. 160–171, 2014.
- [11] S. Vichik, "Quadratic and linear optimization with analog circuits," Ph.D. dissertation, Univ. of California, Berkeley, CA, USA, 2015.
- [12] R. M. Levenson and A. A. Adegbege, "Analog circuit for real-time optimization of constrained control," in *Proc. Amer. Control Conf.*, 2016, pp. 6947–6952.
- [13] T. Skibik and A. A. Adegbege, "Rapid prototyping of constrained linear quadratic optimal control with field programmable analog array," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 1864–1869.

- [14] J. N. Bruno, F. D. Moran, H. I. Khajanchi, and A. A. Adegbege, "Analog solver for embedded model predictive control with application to quadruple tank system," in *Proc. Amer. Control Conf.*, 2021, pp. 4680–4685.
- [15] H. Shah, A. Kowli, and M. Chandorkar, "Analog acceleration in digital and real-time simulation for power engineering applications," in *Proc. 49th Annu. Conf. IEEE Ind. Electron. Soc.*, 2023, pp. 1–6.
- [16] J. Wu, X. He, Y. Niu, T. Huang, and J. Yu, "Circuit implementation of proximal projection neural networks for composite optimization problems," *IEEE Trans. Ind. Electron.*, vol. 71, no. 2, pp. 1948–1957, 2024.
- [17] T. Geyer, G. Papafotiou, and M. Morari, "Hybrid model predictive control of the step-down dc-dc converter," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1112–1124, 2008.
- [18] S. Mariethoz, M. Hecceg, and M. Kvasnica, "Model predictive control of buck dc-dc converter with nonlinear inductor," in *Proc. 11th Workshop Control Model. Power Electron.*, 2008, pp. 1–8.
- [19] S. Mariethoz, S. Almer, M. Baja, A. G. Beccuti, D. Patino, A. Wernrud, J. Buisson, H. Cormerais, T. Geyer, H. Fujioka, U. T. Jonsson, C.-Y. Kao, M. Morari, G. Papafotiou, A. Rantzer, and P. Riedinger, "Comparison of hybrid control techniques for buck and boost dc-dc converters," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 5, pp. 1126–1145, 2010.
- [20] K. Gaouzi, H. E. Fadil, A. Rachid, F. Z. Belhaj, and F. Giri, "Constrained model predictive control for dc-dc buck power converters," in *Proc. Int. Conf. Electr. Inf. Technol.*, 2017, pp. 1–5.
- [21] Z. Liu, L. Xie, A. Bemporad, and S. Lu, "Fast linear parameter varying model predictive control of buck dc-dc converters based on FPGA," *IEEE Access*, vol. 6, pp. 52 434–52 446, 2018.
- [22] J. Chen, Y. Chen, L. Tong, L. Peng, and Y. Kang, "A backpropagation neural network-based explicit model predictive control for dc-dc converters with high switching frequency," *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 8, no. 3, pp. 2124–2142, 2020.
- [23] M. E. Albira and M. A. Zohdy, "Adaptive model predictive control for dc-dc power converters with parameters' uncertainties," *IEEE Access*, vol. 9, pp. 135 121–135 131, 2021.
- [24] P. V. Harisyam and K. Basu, "Stability analysis for model predictive voltage control of buck converter," in *Proc. IEEE Int. Conf. Power Electron., Drives and Energy Syst.*, 2022, pp. 1–6.
- [25] P. V. Harisyam, A. Nandanwar, and K. Basu, "Stability analysis of single prediction horizon continuous-control-set model-predictive-control of buck converter with voltage and current control," in *Proc. 11th Nat. Power Electron. Conf.*, 2023, pp. 1–5.
- [26] M. Guerreiro, P. D. Santos, and S. Liu, "On the design of a model predictive controller for a dc-dc buck converter," in *Proc. 25th Eur. Conf. Power Electron. Appl.*, 2023, pp. 1–8.
- [27] A. Garcés-Ruiz, S. Rizzo, C. González-Castaño, and C. Restrepo, "Model predictive control with stability guarantee for second-order dc/dc converters," *IEEE Trans. Ind. Electron.*, vol. 71, no. 5, pp. 5157–5165, 2024.
- [28] K. Sawant, D. Nguyen, A. Liu, J. Poon, and S. Dhople, "A hybrid-computing solution to nonlinear optimization problems," *IEEE Trans. Circuits Syst. I Reg. Papers*, 2024, in press.
- [29] R. H. G. Tan and L. Y. H. Hoo, "DC-DC converter modeling and simulation using state space approach," in *Proc. IEEE Conf. Energy Convers.*, 2015, pp. 42–47.
- [30] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Madison, WI, USA: Nob Hill Publishing, 2017.
- [31] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*, 1st ed. Cambridge: Cambridge University Press, 2017.
- [32] R. Vichnevetsky, "Stability contours for the analysis of analog/digital hybrid simulation loops," in *Proc. AFIPS*, Boston, MA, USA, 1969, pp. 859–866.
- [33] C. Basso, *Designing Control Loops for Linear and Switching Power Supplies: A Tutorial Guide*, 1st ed. Norwood, MA, USA: Artech House, 2012.