IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 26, 2018, accepted September 3, 2018, date of publication September 6, 2018, date of current version October 12, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2869043

Fast Linear Parameter Varying Model Predictive Control of Buck DC-DC Converters Based on FPGA

ZHEN LIU[1], LEI XIE[1], ALBERTO BEMPORAD[2], (Fellow, IEEE), AND SHAN LU[3]

[1]State Key Laboratory of Industrial Control Technology, Zhejiang University, Hangzhou, China
[2]IMT School for Advanced Studies, Lucca, Italy
[3]Institute of Intelligence Science and Engineering, Shenzhen Polytechnic, Shenzhen, China

Corresponding authors: Lei Xie (leix@iipc.zju.edu.cn) and Shan Lu (lushan@szpt.edu.cn)

This work was supported in part by the National Key R&D Program of China under Grant 2018YFF0214701, in part by the Natural Science Foundation of China under Grant 61673358 and Grant 61621002, and in part by the Natural Science Foundation of Zhejiang, China, under Grant LR17F030002.

ABSTRACT This paper introduces a novel fast model predictive control (MPC) methodology based on linear parameter-varying (LPV) systems. The proposed approach can deal with large-scale problems better than conventional fast MPC methods. First, the equality constraints given by the model equations are not eliminated to get a condensed quadratic programming (QP) problem, as the model of the LPV system changes and it will be time-consuming to reformulate the QP problem at each sampling time. Instead, the proposed approach constructs a sparse QP problem by keeping the equality constraints. Although the resulting QP problem has a larger dimension than the condensed one, it can be reformulated and solved as a system of piecewise affine equations given by the Karush–Kuhn–Tucker conditions of optimality. Finally, the problem will be solved through a Newton-method and an exact line search in a fast way. The performance is tested and compared with off-the-shelf QP solvers on the conventional buck dc–dc converter control problem both in simulations and the experiments on FPGA. The proposed methodology works well for the controller and is especially faster in comparison with some other conventional algorithms for large prediction horizons.

INDEX TERMS Buck dc-dc converter, linear parameter varying, model predictive control, non-condensed quadratic programming problem.

I. INTRODUCTION

Model predictive control (MPC) owes its advantages to the characteristic that can stabilize linear or nonlinear systems subject to hard input and state constraints [1]. For the simpler use of linear systems subject to linear constraints, the resulting optimal control problem can be reformulated as a ''condensed'' quadratic programming (QP) problem by eliminating the equality constraints reformed by the dynamic model of the system, and off-the-shelf QP solvers support the application of MPC for small-scale to medium-scale processes [2]. Meanwhile, when implementing such MPC methods to linear parameter varying (LPV) systems, such as to control nonlinear systems with real-time linearization it is necessary to reformulate the QP problem at each sampling time because the model is changing. However, the repeated multiplication of the coefficient matrices to get the condensed QP problem [3] may require substantial computation effort and meet the bottleneck of the methodology. Especially for high sampling rates and large-scale systems, the computation time for constructing the QP problem may become a limiting factor. The present work constructs a sparse non-condensed QP problem by keeping the equality constraints in the formulation [4] and apply a piecewise smooth Newton algorithm developed in [5] combined with Cholesky factorization and exact line search to maintain small computation time [6] regardless of the increasing dimension of the problem.

Our goal is to adopt MPC and its intrinsic capacity of handling constraints beyond slow dynamic systems, in particular to systems with high sampling rates such as power electronic converters. Experiments on controlling DC-DC converters by MPC are repeated in [7] showing that online MPC is possible for fast systems. During the last years many good algorithms have been developed for high speed MPC such as active-set methods [8], interior-point methods [9] and fast-gradient methods [10] and packages for QP are increasingly becoming the core of solving MPC problems fast enough to fulfill the real-time requirements. Recently online optimization provides several novel concepts of solving QP in a fast way. Reference [11] proposes a sparse MPC structure and uses the Cholesky factorization so that the Newton-method can be solved in low flops. And [12] provides the idea of fast solvers based on the KKT condition. The algorithm of this paper utilizes these ideas synthetically.

A switching converter system can be stable around the operating point, however, it may be unstable when the system meets parameters varying [13], [14]. In order to take parameter uncertainty into consideration, the study of how to set up the converter models is still a significant field of investigation [15]. This paper presents a linear design method based on a developed LPV representation of the buck DC-DC converter [16] in order to achieve robust stability and performance when model inaccuracies happen.

In recent works, [17]–[19] present a conventional MPC controller which contains the transition from MPC to condensed QP and an off-the-shelf QP solver. Meanwhile [20] summarizes that the recent literatures on MPC of power converters mainly concern the prediction model, the cost function and optimization algorithm. These works propose approaches for better prediction model of the converter, for more appropriate cost function selection that can match closely to the type of the power converter, for better design of the weighting factor that is rather important in MPC construction. Finally, about the optimization algorithm issues, computational cost reduction and long prediction horizon are concerned. To solve the problem online efficiently in such fast way limits the algorithms that the MPC strategy should be motivated for particular applications [21]. In our paper, the focus is cast on the MPC strategy, which converges fast and can be used for long prediction horizon so that more kinds of fast systems can benefit from the algorithm.

The main contribution of this paper includes, (i) A novel reformulation of MPC as a sparse non-condensed QP problem. By keeping the equality constraints, the sparse MPC-structure algorithm works well especially on LPV models. (ii) A piecewise Newton-method with exact line-search approach for solving the sparse non-condensed QP problem. A brief proof on the convergence of the algorithm is also provided.

To demonstrate the feasibility and effectiveness of the proposed design, an LPV-MPC controller is tested in the PLECS [22] simulation to get to know the characteristics of the buck DC-DC converter and in close-loop tests which are based on a FPGA platform with a wind turbine generator. It achieves good control performance and computes faster than some other algorithms mentioned before. It shows that the proposed methodology can be applied to a wide range of control applications with various constraints of large-scale by which robustness and fast-tracking are sought.

This paper is organized as follows. In Section II the details of MPC for constrained linear systems and the reformulation of the MPC problem in non-condensed QP form are discussed. Also in this section, a regularized piecewise Newton method with exact line search is presented to solve the QP problem. Section III introduces the LPV model of a buck DC-DC converter. The comparisons are then made between several state-of-the-art QP solvers and also with the condensed formulation of the MPC problem in Section IV. In addition, the experiments are made on FPGA for testing the proposed approach. Finally, Section V summarizes the key results of the paper.

II. MPC ALGORITHM WITH EQUALITY CONSTRAINTS FOR LPV MODELS

Let the LPV model be obtained by the difference equations of the time-invariant system under the assumption:

$$x_{k+1} = A(k)x_k + B(k)u_k, \tag{1}$$

$(A(k), B(k))$ are the model matrices at the current sampling time k, $A(k) \in \mathbb{R}^{n_x \times n_x}$ and $B(k) \in \mathbb{R}^{n_x \times n_u}$. As it is continuous when dealing with MPC of the LPV system, it is assumed that $A(k)$, $B(k)$ remain constant over the prediction horizon and in the following derivations constant matrices A, B will replace the symbols $A(k)$, $B(k)$. k is the prediction step, where $x_k \in \mathbb{R}^{n_x}$ is the state and $u_k \in \mathbb{R}^{n_u}$ is the control input with a box-constraints $u_{\min} \leq u_k \leq u_{\max}$, while $u_{\min} \leq u_{\max} \in \mathbb{R}^{n_u}$. The target of the optimization is to track the reference set-point (x_{ref}, u_{ref}) while minimizing the cost from the current state vector x_0,

$$\min \sum_{k=1}^{N} \left\| x_k - x_{ref} \right\|_Q^2 + \sum_{k=0}^{N-1} \left\| u_k - u_{ref} \right\|_R^2, \tag{2}$$

where N is the length of the prediction horizon, $Q \in \mathbb{R}^{n_x \times n_x}$ as well as $R \in \mathbb{R}^{n_u \times n_u}$ is the weighting positive diagonal matrix in the cost function.

In order to keep matrices of the problem sparse and easy to build up, the sequence

$$u_0, \ldots, u_{N-1}, x_1, \ldots, x_N$$

of predicted states and inputs is introduced, and let $u = \left[u_0' \cdots u_{N-1}'\right]'$ (Here "$'$" means the transposition of a matrix instead of the symbol "T"), $x = \left[x_1' \cdots x_N'\right]'$ and $z = \left[u', x'\right]' \in \mathbb{R}^m$, $m = Nn_u + Nn_x$. The goal is to find optimal value for z such that the finite-horizon cost is minimized over the prediction horizon while satisfying the bound constraints on u_k and to apply $u(t) = u_0$ to control the system. Contrarily to most conventional examples [23], a non-condensed optimal problem formulation is proposed:

$$\begin{aligned} &\min \frac{1}{2} z'Hz + c'z \\ &s.t. \; z_{min} \leq Gz \leq z_{max} \\ &\qquad A_{eq}z = b_{eq}, \end{aligned} \tag{3}$$

where $H \in \mathbb{R}^{m\times m}$ is the block diagonal and the positive-definite Hessian matrix, $G = \left[I_{Nn_u}\ 0_{Nn_u\times Nn_x}\right] \in \mathbb{R}^{Nn_u\times m}$ (Here I_{Nn_u} means the identity matrix of which the number of the diagonal elements is Nn_u), and $A_{eq} \in \mathbb{R}^{Nn_x\times m}$. H and A_{eq} are sparse matrices described as follows:

$$H = 2\begin{bmatrix} R & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & R & 0 & \cdots & 0 \\ 0 & \cdots & 0 & Q & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & Q \end{bmatrix}$$

$$A_{eq} = \begin{bmatrix} -B & 0 & \cdots & 0 & I_{n_x} & 0 & \cdots & 0 \\ 0 & -B & \cdots & 0 & -A & I_{n_x} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & -B & 0 & \cdots & -A & I_{n_x} \end{bmatrix}.$$

$c \in \mathbb{R}^m$ is a vector made up of the reference and the weights that presents

$$c = -2\left[(Ru_{ref})' \cdots (Ru_{ref})'\, (Qx_{ref})' \cdots (Qx_{ref})'\right]'. \qquad (4)$$

Constraint vectors $(z_{min}, z_{max}) \in \mathbb{R}^{Nn_u}$ are described by $[u_{min} \cdots u_{min}]'$ and $[u_{max} \cdots u_{max}]'$ respectively. To make the variable looking simple, lb and ub are denoted to replace z_{min} and z_{max} respectively. Vector

$$b_{eq} = \left[A', \underbrace{0_{n_x\times n_x} \cdots 0_{n_x\times n_x}}_{N-1}\right]' \cdot x_0 \in \mathbb{R}^{Nn_x}.$$

The KKT condition of optimality, which is also called as stationary function for the convex QP progress (3) is:

$$Hz + c + G'\gamma + A'_{eq}\lambda = 0, \qquad (5)$$

where $\lambda \in \mathbb{R}^{Nn_x}$ deals with the complementarity conditions but will be eliminated in the same way as z. And $\gamma \in \mathbb{R}^{Nn_u}$ is denoted as the Lagrange parameter for box-constraints, that is, the two-side constraint share the same variable through constructing the piecewise primal-dual condition in the following equations:

$$\begin{aligned} lb_i &= G_i z \Rightarrow \gamma_i \leq 0 \\ lb_i &< G_i z < ub_i \Rightarrow \gamma_i = 0 \\ G_i z &= ub_i \Rightarrow \gamma_i \geq 0. \end{aligned} \qquad (6)$$

Notice that it is possible to transform the complementarity conditions (6) into a piecewise affine (PWA) system by using the mid function in [24] as follows:

$$Gz - mid(lb, ub; Gz + \gamma) = 0. \qquad (7)$$

Here *mid* function means the middle value of the three variables. As $lb_i \leq ub_i$,

$$mid\ (lb, ub; Gz + \gamma) = \begin{cases} ub_i, & ub_i \leq G_i z + \gamma_i \\ lb_i, & G_i z + \gamma_i \leq lb_i \\ G_i z + \gamma_i, & lb_i \leq G_i z + \gamma_i \leq ub_i. \end{cases} \qquad (8)$$

Next (5) and the equality constraint in (3) will be used to eliminate z from (7) and get a new mid function as follows:

$$\Phi_{mid}(\gamma) \triangleq D\gamma + d + mid(lb, ub; \gamma - D\gamma - d) = 0, \qquad (9)$$

where
$D = GH^{-1}G' - GH^{-1}A'_{eq}(A_{eq}H^{-1}A'_{eq})^{-1}A_{eq}H^{-1}G'$ and
$d = GH^{-1}c - GH^{-1}A'_{eq}(A_{eq}H^{-1}A'_{eq})^{-1}(b_{eq} + A_{eq}H^{-1}c)$.

Here $\triangleq$ is a definition symbol to define the mid function. To implement this algorithm, D and d will be built in real time at each sampling. And it will take less time than the reformulation of the dense QP from MPC because all the matrices to be multiplied are diagonal leading to a vector-matrix multiplication rather than matrix-matrix one.

To solve the the piecewise function (9) by using the Newton-method and line-search, it is necessary to find a smooth and convex merit function. First to normalize the piecewise function, $[]_+$, meaning the bigger one between the function in [] and 0, is used as follows:

$$\begin{aligned} &\gamma_i - (D\gamma + d)_i \\ &\quad < lb_i \Rightarrow \left[(D\gamma + d + lb)_i - \gamma_i\right]_+ \\ &\quad = (D\gamma + d + lb)_i - \gamma_i, \left[\gamma_i - (D\gamma + d + ub)_i\right]_+ = 0; \\ &\gamma_i - (D\gamma + d)_i \\ &\quad > ub_i \Rightarrow \left[(D\gamma + d + lb)_i - \gamma_i\right]_+ = 0, \\ &\left[\gamma_i - (D\gamma + d + ub)_i\right]_+ \\ &\quad = \gamma_i - (D\gamma + d + ub)_i; \\ &lb_i \leq \gamma_i - (D\gamma + d)_i \leq ub_i \\ &\quad \Rightarrow \left[(D\gamma + d + lb)_i - \gamma_i\right]_+ = 0, \\ &\qquad \left[\gamma_i - (D\gamma + d + ub)_i\right]_+ = 0; \end{aligned} \qquad (10)$$

Substitute (10) into (9) to get a new function:

$$\begin{aligned} \Phi_{mid}(\gamma) \triangleq \gamma &+ [(D\gamma + d + lb) - \gamma]_+ \\ &- [\gamma - (D\gamma + d + ub)]_+ = 0. \end{aligned} \qquad (11)$$

A merit function Ψ is denoted here for the function (11) and the minimizer of it equals to the solutions of function (11). As $\nabla(\frac{1}{2}\left\|[\Phi]_+\right\|^2) = [\Phi]_+$, an obvious selection of the merit function is:

$$\begin{aligned} \Psi(\gamma) \triangleq \frac{1}{2}\gamma'(I - D)\gamma &- \frac{1}{2}\left\|[(D\gamma + d + lb) - \gamma]_+\right\|^2 \\ &- \frac{1}{2}\left\|[\gamma - (D\gamma + d + ub)]_+\right\|^2 \end{aligned} \qquad (12)$$

Thus the gradient for finding the minimizer of (12) is $(I - D)\Phi_{mid}(\gamma)$. However, in this function it is not guaranteed that the $I - D$ is positive-definite, which means that the merit function will not always be convex, continuously and differential and cannot be solved by means of the Newton-method. However, for any $\tau \in (0, \frac{1}{\|D\|}]$ the solution is not changed if one pre-multiplies $\Phi_{mid}(\gamma)$ by the positive definite matrix $I-\tau D$: $(I-\tau D)\Phi_{\tau,mid}(\gamma)=0$, where

$$\Phi_{\tau,mid}(\gamma) \triangleq \gamma + [\tau(D\gamma+d+lb)-\gamma]_+ \\ -[\gamma-\tau(D\gamma+d+ub)]_+ = 0. \quad (13)$$

Obviously, it can be easily inferred that $(I-\tau D)\Phi_{\tau,mid}(\gamma)$ is the gradient of the following quadratic function

$$\Psi_\tau(\gamma) \triangleq \frac{1}{2}\gamma'(I-\tau D)\gamma - \frac{1}{2}\left\|[\tau(D\gamma+d+lb)-\gamma]_+\right\|^2 \\ -\frac{1}{2}\left\|[\gamma-\tau(D\gamma+d+ub)]_+\right\|^2 \quad (14)$$

which is continuous, differentiable and convex.

Therefore, the KKT system can be solved by finding the dual Lagrange parameter γ that minimizes $\Psi_\tau(\gamma)$, where

$$\Phi_{\tau,mid}(\gamma) \triangleq \tau(D\gamma+d) \\ + mid(\tau lb, \tau ub; \gamma-\tau(D\gamma+d)) = 0 \quad (15)$$

and then retrieves the primal solution

$$z = -H^{-1}(c+G'\gamma - A_{eq}'(A_{eq}H^{-1}A_{eq}')^{-1}(A_{eq}H^{-1}c+ \\ b_{eq}+A_{eq}H^{-1}G'\gamma\Big). \quad (16)$$

In order to determine the Newton direction, the PWA function (15) is rewritten as,

$$\Phi_{\tau,mid}(\gamma) = \begin{cases} \tau(D\gamma+d)_i+\tau lb_i, \gamma_i-\tau(D\gamma+d)_i<\tau lb_i, \\ \quad i\in(1,m), \\ \tau(D\gamma+d)_i+\tau ub_i, \gamma_i-\tau(D\gamma+d)_i>\tau ub_i, \\ \quad i\in(1,m), \\ \gamma_i, \quad otherwise \end{cases} \quad (17)$$

and equivalently as

$$\Phi_{\tau,mid}(\gamma) = \begin{cases} 1\cdot\tau(D\gamma+d)_i+1\cdot\tau lb_i+0\cdot\tau ub_i+0\cdot\gamma_i, \\ \gamma_i-\tau(D\gamma+d)_i<\tau lb_i, i\in(1,m), \\ 1\cdot\tau(D\gamma+d)_i+0\cdot\tau lb_i+1\cdot\tau ub_i+0\cdot\gamma_i, \\ \gamma_i-\tau(D\gamma+d)_i>\tau ub_i, i\in(1,m), \\ 0\cdot\tau(D\gamma+d)_i+0\cdot\tau lb_i+0\cdot\tau ub_i+1\cdot\gamma_i \\ otherwise. \end{cases} \quad (18)$$

Obviously, the three results can be described as one form by using 0 and 1 to reform the mid-function. Thus the three functions in (18) can be replaced by $m\times m$ diagonal matrices $E_{\delta_1}, E_{\delta_2}, E_{\delta_3}, E_{\delta_4}$ with the ith diagonal element being equal to 1 or 0 and (18) will finally be described as

$$\Phi_{\tau,mid}(\gamma) = (E_{\delta_1}\tau D+E_{\delta_4})\gamma + \tau(E_{\delta_1}d+E_{\delta_2}lb+E_{\delta_3}ub). \quad (19)$$

The following Algorithm 1 explains the above reformulation to solve the QP (3).

Algorithm 1 Piecewise Smooth Newton Method With Exact Line-Search

```
1: Choose the initial guess $\gamma^0$.
2: Set $k = 0$.
3: If $\|\Phi_{\tau,mid}(\gamma^k)\| \le \varepsilon$ stop.
4: Solve the Newton direction $r^k$ through
   $\nabla\Phi_{\tau,mid}(\gamma^k)\cdot r = -\Phi_{\tau,mid}(\gamma^k)$.
5: Compute the step-size $t_k$ using exact line-search
   $t^k = \arg\min_{t>0} \Psi(\gamma^k + tr^k)$.
6: Set $\gamma^{k+1} = \gamma^k + t^k r^k$ and $k \leftarrow k+1$. Go to step 3.
```

In Step 4, several tricks can be used to reduce the computation due to the special structure and the simplicity of the elements coming from the multiplication by diagonal matrices. First q is denoted to substitute $E_{\delta_1}d+E_{\delta_2}lb+E_{\delta_3}ub$. Then it is obvious to find the diagonal elements among E_{δ_1} and E_{δ_4} are complementary, that is the 1-position in E_{δ_1} matches with 0-position in E_{δ_4} and vice versa. Meanwhile in q it can also be found that the 0-position is same as E_{δ_1}. Thus I and N are chosen as the new mark for the diagonal matrices E_{δ_i} to decompose the Newton function by judging the diagonal position 1 and 0 respectively. Consequently, $(E_I\tau D+E_N)r = -(E_I\tau D+E_N)\gamma - \tau q$ is obtained and the dimension of the Newton direction function is reduced effectively by the order I and N

$$r_N = -\gamma_N \quad (20a)$$

$$D_{II}r_I = -D_{II}\gamma_I - q_I. \quad (20b)$$

Computing (20b) is the only time-consuming computation to get

$$r_I = -\gamma_I - {D_{II}}^{-1}q_I. \quad (21)$$

Since D is positive semidefinite, D_{II} is also positive semidefinite as a principle submatrix of D. One can use the Cholesky factorization to gain reduced-order functions for linear function. Thus (21) will be easily solved corresponding to lower and upper triangular linear function through the Cholesky factor [24].

Remark 1: Consider the linear function (21), D_{II} is decomposed into LL' (L is the Cholesky factor) through the Cholesky factorization. Then solve $L'y = q_I$ for y by forward substitution and finally solve $-\gamma_I - L^{-1}y$ by back substitution to gain r_I in (21). It is easy to compute and can avoid the singularity problem.

In Step 5, the following line search Algorithm 2 is proposed.

Remark 2: Assume that $\gamma \in \mathbb{R}^{Nn_u}$ is a solution of $\Phi_{\tau,mid}(\gamma) = 0$ and suppose that any $\nabla\Phi_{\tau,mid}(\gamma)$ is nonsingular

1) *For any γ^0 sufficiently close to γ^*, the sequence $\{\gamma^k\}$ generated from Algorithm 1 has a quadratic convergence to γ^*.*
2) *As r^k is a descent direction, any minimizer of Ψ_τ solved in Algorithm 2 converges to γ^*.*

Algorithm 2 Exact Line-Search

```
1: Choose the initial guess of step-size $t$ (mostly start from $t = 1$), set $i = 1$.
2: Given $\gamma_0 = \gamma^k$, the descent direction $r^k$, $0 < \alpha < 1$ and $0 < \zeta < 1$.
3: Compute $\gamma_i = \gamma_{i-1} + tr^k$ and $\varphi = \nabla\Psi_\tau(t) = \Phi'_{\tau,mid}(\gamma_i)\, r^k$.
4: If $\|\varphi\| \leq \zeta$ stop.
5: Set $t = t - \alpha\varphi$ and $i \leftarrow i + 1$. Go to step 3.
```

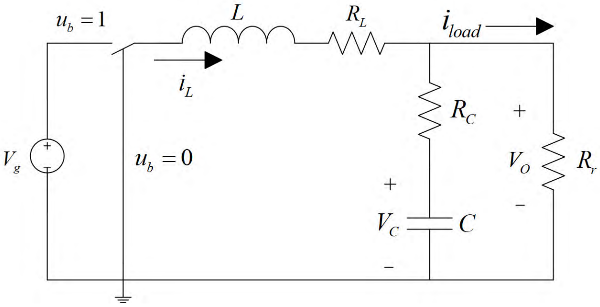


FIGURE 1. Schematic of the buck converter.

As the piecewise function reformed by (13)-(15) is continuous, differential and smooth, it is sufficient to use the Newton method to find the optimal solution. Thus the proof of the convergence in this algorithm follows the tradition line-search and Newton-method convergence theorems in [25] and [26].

III. LINEAR PARAMETER VARYING MODEL OF UNCERTAIN BUCK CONVERTER

The algorithm proposed in Section 2 is applied to the LPV system of buck converter. Figure 1 shows the schematic of a buck DC-DC converter in which V_o is the output voltage that must be kept at a certain reference value and V_g is the input voltage. *Rr* presents the converter load, meanwhile C and L represent capacitance of the capacitor and inductance of the inductor respectively with equivalent series resistance R_c and R_L. The measurable states are the capacitor voltage V_C and the inductor current i_L. The binary signal u_b shows that the switch turns on and off controlled by a fixed-frequency PWM (PWM stands for Pulse Width Modulation). The ratio of the switch-on time in a switching period T is defined as duty-cycle dc. To develop a state-space model of the converter, the circuit will be divided into two modes of operation which are obtained in relation to the switch position and conduction of the diode. It is assumed that the converter works in continuous conduction mode (CCM) and that the inductor current is not saturated.

Mode 1: when $u_b = 1$ which equals to the time zone $(0, dcT)$ of each switching period, it is easy to get the equations

$$\begin{cases} L\dfrac{di_L}{dt} = V_g - V_C - R_L \cdot i_L - R_C \cdot C \cdot \dfrac{dV_C}{dt} \\ V_C + R_C \cdot C \cdot \dfrac{dV_C}{dt} = R_r \cdot \left(i_L - C \cdot \dfrac{dV_C}{dt}\right). \end{cases} \tag{22}$$

Leading to state-space formulation:

$$\begin{bmatrix} \dot{i}_L(t) \\ \dot{V}_C(t) \end{bmatrix} = A_{ss} \begin{bmatrix} i_L(t) \\ V_C(t) \end{bmatrix} + \begin{bmatrix} \dfrac{1}{L} \\ 0 \end{bmatrix} V_g. \tag{23}$$

Here

$$A_{ss} = \begin{bmatrix} -\dfrac{1}{L} \cdot \dfrac{R_r \cdot R_C + R_r \cdot R_L + R_L \cdot R_C}{R_r + R_C} & -\dfrac{1}{L} \cdot \dfrac{R_r}{R_r + R_C} \\ \dfrac{1}{C} \cdot \dfrac{R_r}{R_r + R_C} & \dfrac{1}{C} \cdot \dfrac{1}{R_r + R_C} \end{bmatrix},$$

$\dot{i}_L(t) = \frac{di_L}{dt}$ and $\dot{V}_C(t) = \frac{dV_C}{dt}$, similarly hereinafter.

Mode 2: when $u_b = 0$ which equals to the time zone (dcT, T) of each switching period, the equations can also be set up as follows:

$$\begin{cases} 0 = L\dfrac{di_L}{dt} + R_L \cdot i_L + V_C + C \cdot R_C \cdot \dfrac{dV_C}{dt} \\ V_C + C \cdot R_C \cdot \dfrac{dV_C}{dt} = R_r \cdot (i_L - \dfrac{1}{C} \cdot \dfrac{dV_C}{dt}). \end{cases} \tag{24}$$

Leading to the state-space formulation:

$$\begin{bmatrix} \dot{i}_L(t) \\ \dot{V}_C(t) \end{bmatrix} = A_{ss} \begin{bmatrix} i_L(t) \\ V_C(t) \end{bmatrix}. \tag{25}$$

In fact, the equivalent series resistances of the capacitor and inductance are small enough to be neglected. Under this assumption, the matrices of the state-space functions in mode 1 (23) and mode 2 (25) can be simplified as follows:

$$A_1 = \begin{bmatrix} 0 & -\dfrac{1}{L} \\ \dfrac{1}{C} & -\dfrac{1}{R_r C} \end{bmatrix}, \quad B_1 = \begin{bmatrix} \dfrac{V_g}{L} \\ 0 \end{bmatrix} \tag{26}$$

$$A_2 = \begin{bmatrix} 0 & -\dfrac{1}{L} \\ \dfrac{1}{C} & -\dfrac{1}{R_r C} \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{27}$$

The following expression shows the state-space averaged model of a PWM converter [27]:

$$\dot{x}(t) = Ax(t) + B_u u(t), \tag{28}$$

where

$$A = \begin{bmatrix} 0 & -\dfrac{1}{L} \\ \dfrac{1}{C} & -\dfrac{1}{R_r C} \end{bmatrix}, \quad B_u = \begin{bmatrix} \dfrac{V_g}{L} \\ 0 \end{bmatrix}. \tag{29}$$

and $x(t) = \begin{bmatrix} i_L(t) \\ V_c(t) \end{bmatrix}$, $u(t)$ means the current duty-cycle dc as the PWM signal.

The matrices A, B_u in (29) may be uncertain and time varying. Especially B_u depends on input voltage V_g, which leads to the LPV model [28] and is described as follows:

$$\dot{x}(t) = A(t)x(t) + B_u(t)u(t), \tag{30}$$

where

$$A(t) = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{R_r(t)C} \end{bmatrix}, \quad B_u(t) = \begin{bmatrix} \frac{V_g(t)}{L} \\ 0 \end{bmatrix}. \qquad (31)$$

This LPV model will be used in Section II to design an MPC controller for the buck DC-DC converter.

IV. NUMERICAL SIMULATIONS AND EXPERIMENTS CERTIFICATE

Now the performance of the algorithm proposed in Section II will be tested for the problem described in Section III. As the tracking target is the output voltage but not the state, the first thing is to get the working point (x_{ref}, u_{ref}) mentioned in (2) by solving the linear system coming from the model (28)

$$\begin{cases} x_{ref} = Ax_{ref} + B_u u_{ref} & (32a) \\ y_{ref} = C_m x_{ref} & (32b) \end{cases}$$

at each sampling time, leading to the linear function

$$\begin{bmatrix} x_{ref} \\ u_{ref} \end{bmatrix} = \begin{bmatrix} I_{n_x} & -B \\ C_m & 0 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ y_{ref} \end{bmatrix}. \qquad (33)$$

The output voltage V_o (see Figure 1)

$$V_o = C_m \begin{bmatrix} i_L(t) \\ V_C(t) \end{bmatrix}, C_m = \begin{bmatrix} \frac{R_r \cdot R_C}{R_r + R_C} & \frac{R_r}{R_r + R_C} \end{bmatrix} \qquad (34)$$

equals to the output reference y_{ref}.

The numerical simulation study is carried out on a personal computer with the following configuration: Intel Core i7-2600 3.40GHz CPU, 4.00GB RAM, 64-bit Windows 10 Operating System. The model is implemented in PLECS and the experiments are based on NI CompactRIO platform using Xilinx FPGA.

A. NUMERICAL SIMULATIONS

The values of the converter parameters set are shown in Table 1. The nominal value of the supply voltage is 5 V. Sampling time $T_s = 0.01ms$ is set to form the discrete model (1): $x_k \in \mathbb{R}^2, u_k \in \mathbb{R}, A = \begin{bmatrix} 0.9672 & -0.2992 \\ 0.0224 & 0.9983 \end{bmatrix}, C_m = \begin{bmatrix} 0.0269 & 0.9980 \end{bmatrix}$. As the experiments are made on a fixed resistive load, $A(t)$ in (31) is a fixed one while B_u changes with the sawtooth input voltage at each sampling time. For MPC, the weights $Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, R = 800$ is a trial in the cost function. The constraints come from the real limit of the duty-cycle ranging from 0 to 1. The predictive horizon N is set to 10.

The professional software PLECS is used to model and simulate the circuit in Figure 2. Typical power electronics components such as semiconductors, inductors and capacitors are placed on the circuit diagram and simply connected by drawing wires [22]. The parameters come from the real converters shown in Table 1 and the simulations are done to test whether the closed loop using the MPC controller proposed in this paper is fit for wind turbine generator.

TABLE 1. Buck DC-DC converter parameters.

Parameter	Value
R_r	10 Ω
R_C	0.027 Ω
L	33 μ H
C	440 μ F
V_g	[7 , 12] V
$V_o(V_{ref})$	5 V
T_s	0.01 ms

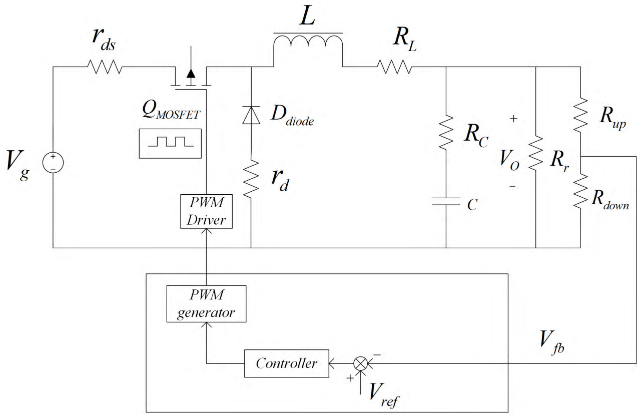


FIGURE 2. Control design of buck converter in PLECS.

From [29] and [30], it is known that in CCM mode, when the MOSFET works in high level which depends on the PWM signal, the diode will switch-off and when the MOSFET works in low level, the diode will be on. To model the converter easier, the two modes will be replaced by a switch just like the one in Figure 1. However, in Figure 2, there are some more parameters such as resistances r_{ds} and r_d which can be neglected when modelling because their values are far more smaller than the resistances R_L, R_C and R_r. Meanwhile, R_{up} and R_{down} are divider resistances used for protecting the circuit from high voltage and current. And these two resistances will not influence the modelling of the converter as in the digital control V_{fb} will be transferred to V_o by $V_{fb} = \frac{R_{down}}{R_{up}+R_{down}} V_o$ to compare with V_{ref}.

First before implementing the MPC controller to the real-time platform, a comparison is made between the PID controller and the MPC controller as DC-DC converter is an SISO LPV system with variable gain. Figure 3 shows a simulation between the PID controller and the MPC controller based on PLECS-Simulink. The output set-point of the Buck DC-DC is set as 5V and the load is 10Ω. Figure 4 depicts that MPC can provide a smaller overshoot and a faster converging time than the PID controller. What is more in the figure, if the inductor current is imited to $[0, 15A]$, in contrast to MPC, the PID controller cannot satisfy the state constraints since the PID controller only works on the input-output constraints [31].

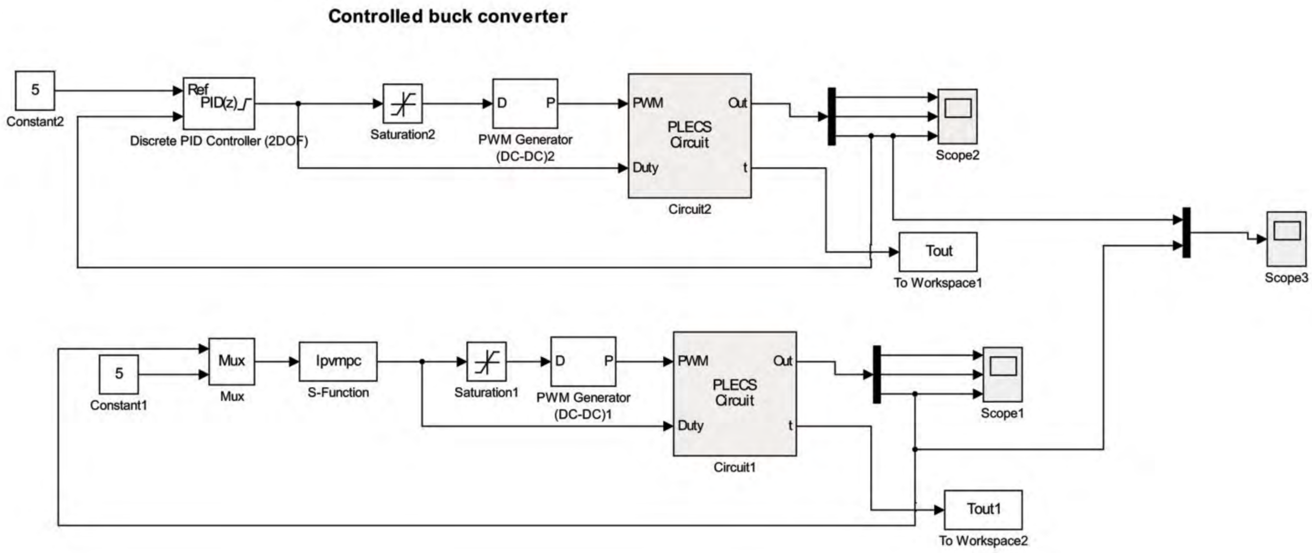


FIGURE 3. Simulation comparison between PID and MPC controllers through PLECS Simulation.

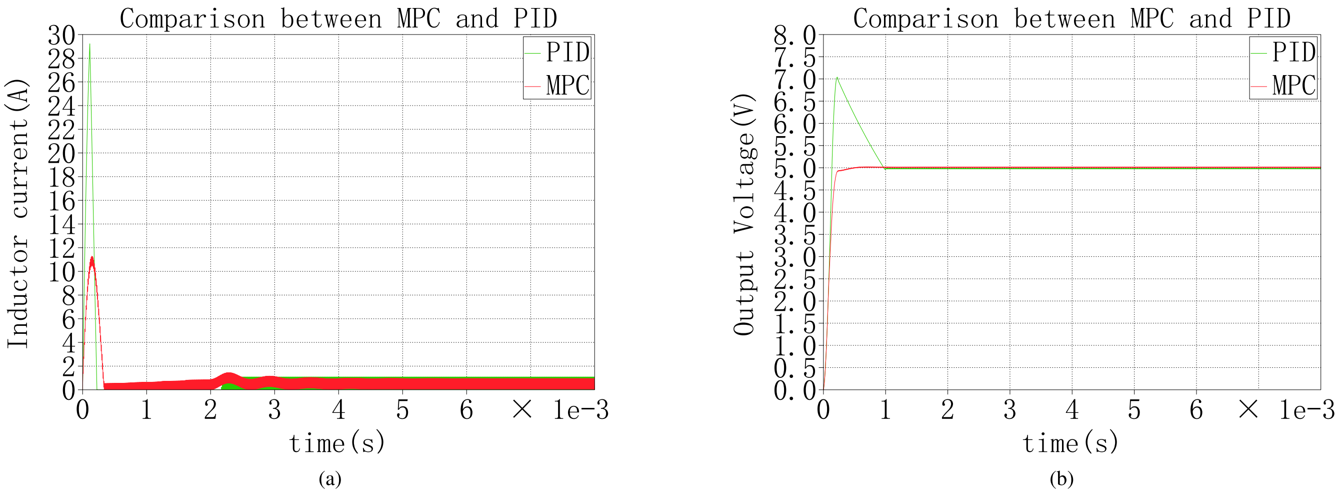


FIGURE 4. Simulation curves between PID and MPC controllers. (a) Inductor current of the process using MPC and PID controller. (b) Output voltage of the process using MPC and PID controller.

Figure 5 shows the simulation of the buck converter working in closed loop with sawtooth input voltages which is similar to the real process. The buck DC-DC converter acts as a second-order asymptotically stable system, although a ripple wave resists in steady-state which will be demonstrated in the next experiments. Considering the modelling of the converter from Section III, regardless the noise from the environment, the main disturbance comes from the input voltage and the load. Figure 5 shows that no matter how the input voltages change, the closed loop system using the proposed MPC algorithm can reach a desired output voltage and this controller will be implemented in real-time platform in the next section.

B. COMPARISON WITH EXISTING APPROACHES

The numerical simulation comparisons between the algorithm proposed in this paper and the other state-of-art QP solvers are running on PC: Intel Core i7-2600 3.40GHz CPU, 4.00GB RAM, 64-bit Windows 10 Operating System and MATLAB R2016a. All the solvers share the same model and its coefficient parameters containing discrete model parameter A, C_m, the weights Q, R, set-point and the constrains in the last section. Meanwhile the discrete matrix B_u changes with the sawtooth input voltage at each sampling time and all the simulations follow the same rhythm when the model updates mentioned in Section IV-A. To compare the optimization performances of these algorithms, different prediction

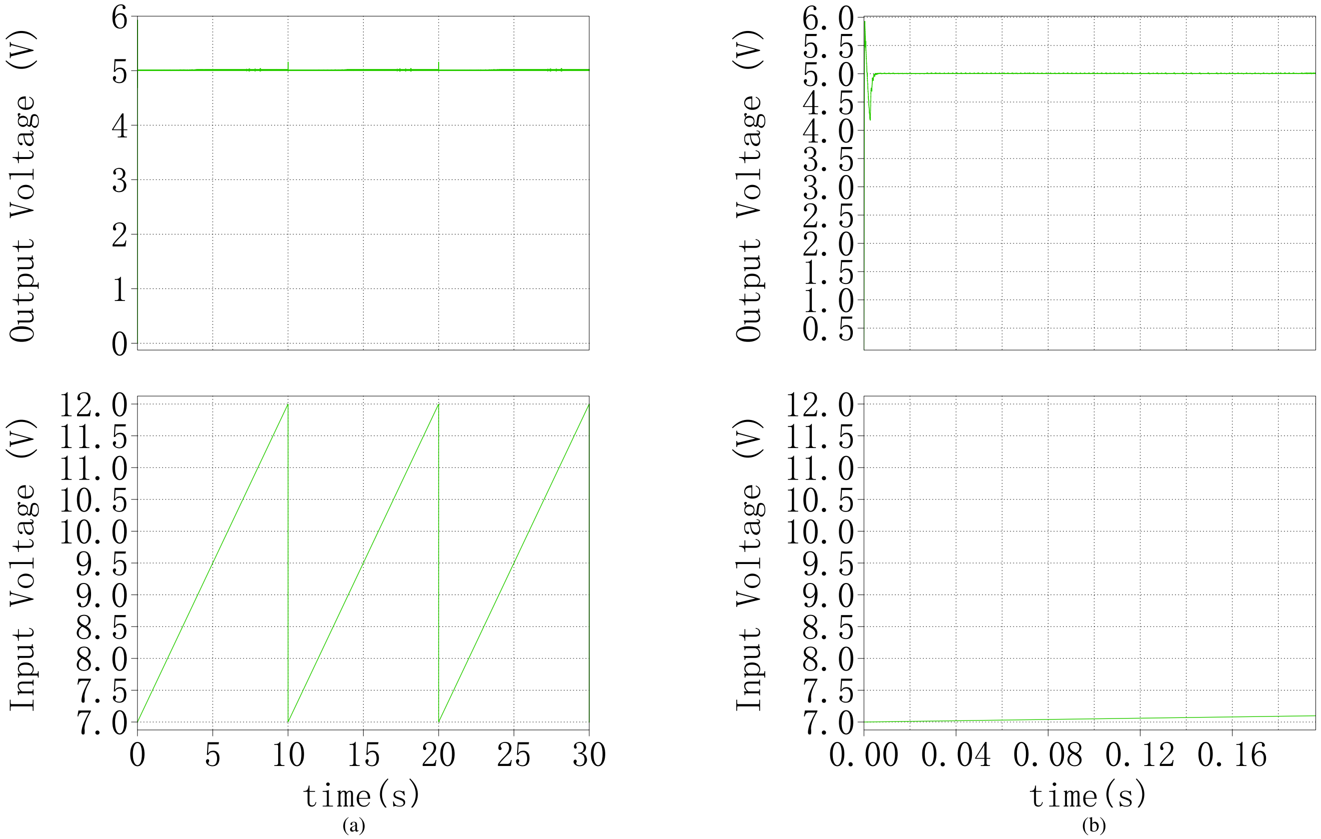


FIGURE 5. Closed loop with a sawtooth input voltage. (a) The whole process executed similar to the real process. (b) The starting curve in the first second of the process.

horizons have been used to form QP problems with different scales. Note that t he computational time of both formulating and solving QP problem is recorded. More precisely, in each control step, the MPC problem is converted to QP and solved 50 times, and the minimum is recorded. Finally the whole process is set in 100-control-step and the minimum time at each step is accumulated.

1) RUNTIME COMPARISON

The solver proposed in Section II is compared with other state-of-art QP solvers in a LPV-MPC problem of which the prediction horizons are between 10 and 100 with increment 10. To avoid the interrupts coming from other systems the solution will be executed at each sample step 50 times and the minimum time will be accepted for the particular simulation time. In Algorithm 1, as in [6] it shows that τ is chosen based on the examples and after some trials $\tau = \frac{1}{1.01\|D\|}$ leads to a faster convergence. In addition, $\alpha = 0.01$ and $\varepsilon = \zeta = 1e-9$ are the settings of the proposed algorithm. The QP solvers considered in the comparison are interior-point, that also involves both primal and dual variables but has favorable sparsity pattern for MPC problems [11], ADMM [32] and its OSQP variant [33], the online active-set solver qpOASES [34], GPAD [35] and Gurobi [36]. About the settings of these algorithm, max-iteration is set to 1000 and the terminal tolerance is set to $1e-9$.

Figure 6 depicts the non-condensed piecewise smooth Newton method with exact line search (non-condensed PWA fast MPC) proposed in this paper keeps in a low runtime especially in long prediction horizon. The time order here is mainly based on the CPU scale, thus the time-scale "seconds" does not mean the real time consuming in the embedded platform but can show the trend of each algorithm's cost. Several observations based on the results demonstrate the superiority of the proposed algorithm. Although it does not perform much better than qpOASES, OSQP, GPAD and interior, when the prediction horizon increases, the computation time of the other algorithms increases a lot while the proposed algorithm keeps in a low degree. What is more all through the figure, the algorithms using sparse structure performs well when computation dimension increases. Thus the special dealing with the equality function in the proposed MPC algorithms makes it much more scalable to the LPV problem size.

2) ITERATIONS COMPARISON

The prediction horizon will be set from 10 to 30 increased by 5 and the maximum as well as the minimum iterations of the benchmark solvers are recorded during each

TABLE 2. The maximum and minimum iterations of the algorithms at different predictive horizons.

Algorithms	N=10		N=15		N=20		N=25		N=30	
	iter.(min)	iter.(max)	iter.(min)	iter.(max)	iter.(min)	iter.(max)	iter.(min)	iter.(max)	iter.(min)	iter.(max)
non-condensed PWA fmpc	1	4	1	5	1	5	1	6	1	6
ADMM	3	1000	3	1000	3	1000	3	1000	3	1000
qpOASES	2	12	6	16	11	20	16	26	21	35
OSQP	25	75	50	100	75	100	75	100	100	150
GPAD	1	17	1	23	1	26	1	31	1	37
non-condensed GPAD	88	284	115	358	122	365	131	411	155	423
Gurobi	7	11	7	11	7	11	7	12	8	12
non-condensed Gurobi	9	13	9	13	11	14	11	14	12	14
Interior	3	7	3	8	3	9	3	9	3	9
non-condensed Interior	4	7	4	8	4	8	4	9	4	10

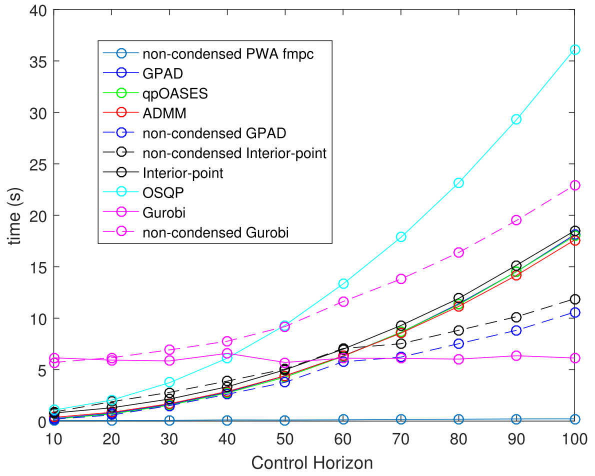


FIGURE 6. Runtime with respect to prediction horizon comparison with existing approaches.

sampling step. Table 2 depicts that the algorithm of this paper always keeps in a low number as the computation dimension augments while the others increase in some degree.

C. EXPERIMENTS

Buck DC-DC converter is a typical switch-mode system and has an LPV model. Although existing control approaches have been proved effective, such as PID, sliding mode control and so on, several challenges have not been fully addressed yet, such as ease of controller design and tuning as well as robustness to load parameter variants [37]. Moreover, PID control has its weakness in tuning and satisfying the state constraints compared with modern advanced control.

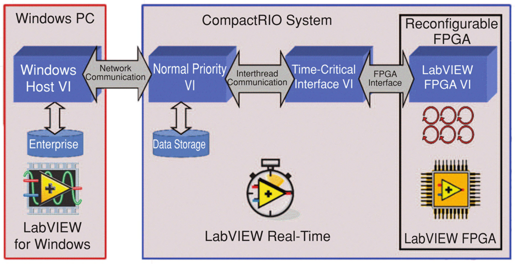


FIGURE 7. Software application architecture for NI CompactRIO platform.

Thus, the FPGA-experiments on Buck DC-DC converter are done on the platform National Instruments (NI) CompactRIO for verifying the algorithm whether efficient or not. The unique computational feature of this system is that it contains a real-time processor and an FPGA. Furthermore using the LABVIEW graphical development environment both devices are programmable. The CompactRIO platform uses cRIO-9082 with Xilinx FPGA (1.33 GHz, Dual Intel Core i7 CPU) as controller of which the time base is 40MHz and the precision reaches 100ppm ("ppm" stands for "parts per million". It is like percent which is really parts per hundred but based on million instead of hundred. Therefore, 100ppm=100/1000000=0.01%), cRIO-9223 as the 16-bit analog input ranging from −10V to 10V and maximum s ampling time 1M Samples/s and cRIO-9401 as the digital output (PWM) that has the feature of 8 channels, update rate 100ns and signal level 5V TTL (high level is 5V, low level is 0V). Figure 7 is a software application architecture of the platform including a windows PC (host, monitoring and data storage), a real-time program (analog input

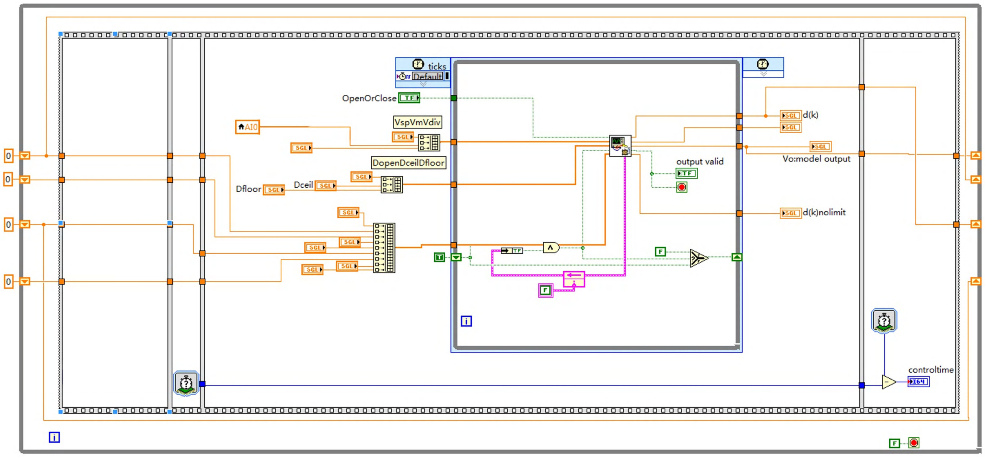


FIGURE 8. Implementing MPC using LABVIEW.

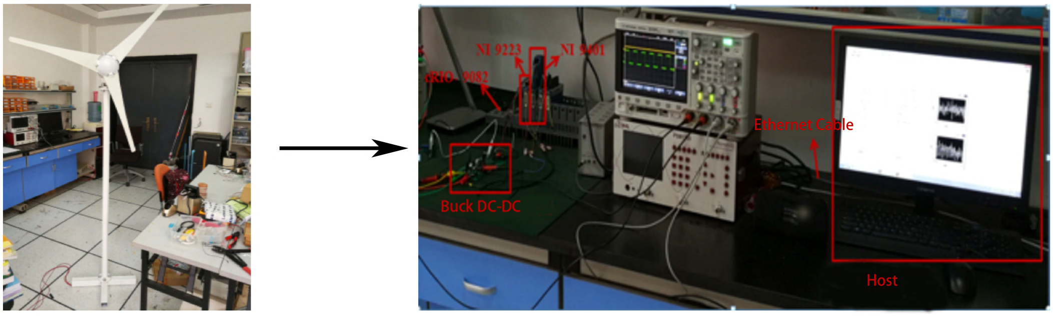
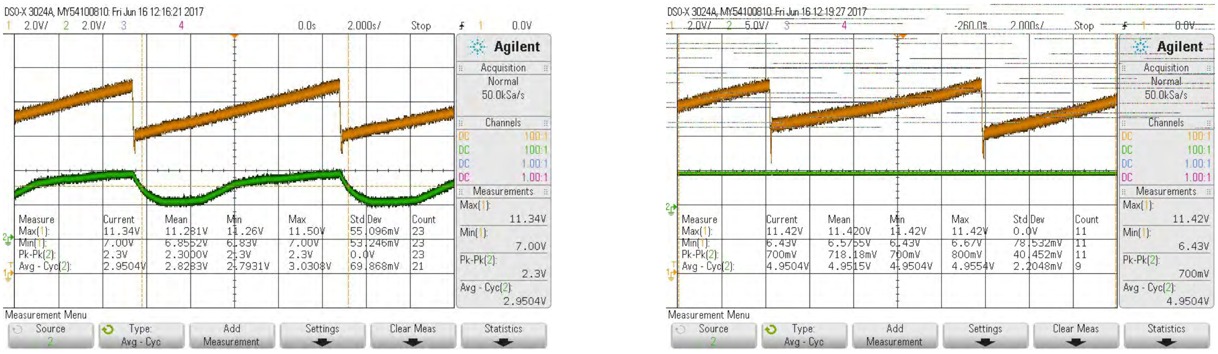


FIGURE 9. Experiment platform system setup.



Measure	Current	Mean	Min	Max	Std Dev	Count
Max(1):	11.34V	11.281V	11.26V	11.50V	55.096mV	23
Min(1):	7.00V	6.8552V	6.83V	7.00V	53.246mV	23
Pk-Pk(2):	2.3V	2.3000V	2.3V	2.3V	0.0V	23
Avg - Cyc(2):	2.9504V	2.8283V	2.7931V	3.0308V	69.868mV	21

(a)

Measure	Current	Mean	Min	Max	Std Dev	Count
Max(1):	11.42V	11.420V	11.42V	11.42V	0.0V	11
Min(1):	6.43V	6.5755V	6.43V	6.67V	78.532mV	11
Pk-Pk(2):	700mV	718.18mV	700mV	800mV	40.452mV	11
Avg - Cyc(2):	4.9504V	4.9515V	4.9504V	4.9554V	2.2048mV	9

(b)

FIGURE 10. Oscilloscope image of input-output in the circuit. (a) Open loop. (b) Closed loop.

sampling and digital output) running on the processor and an FPGA program (MPC implementation in IP builder), which contains several high-level blocks for control and signal processing that approximate floating-point implementation using the integer math available [38]. The IP builder (shown in Figure 8) mentioned before is to implement the MPC algorithm because it automatically optimizes the high level algorithm especially the matrix-vector multiplication, arrays and loops.

Although cRIO can run fast with high sampling frequency, it is still needed to prepare some divider resistances to satisfy the limitation of the I/O port. The Agilent DSO-X 3024A is chosen as the oscilloscope which has 4 channels with 200MHz width and the maximum sampling rate 4G Sample/s. To drive the converter, a 600W DC supply Agilent N6705B and a 500V/30A/750W DC load ITECH IT8813B are used as the resistance load. Finally a 100W wind generator NE-100S (which starts at a wind velocity 2m/s and stays

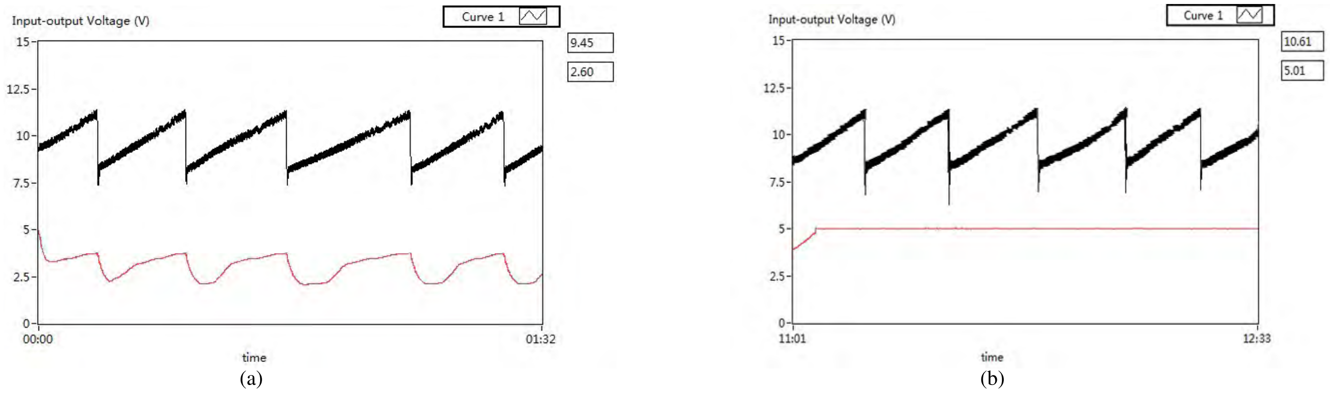


FIGURE 11. Labview images of input-output voltage. (a) Open loop. (b) Close loop.

stable at 10m/s) with a AC-DC converter is selected as the input of the system. When it works around 3m/s inside the lab, the output DC voltage ranging from 7V to 12V. The whole structure of the platform is depicted in Figure 9.

Both figures in Figure 10 show that the wind turbine running in low-velocity generates a kind of sawtooth wave raging from 7V to 12V because the speed of the wind cannot reach the stable working point. In Figure 10(a) it is shown that the system changes much around a certain output voltage with a large overshoot ±2.3V taking no account of the ripple wave both in the input and the output generated by the converter itself. The upper curve in the figure is the input voltage as well as the lower one is output voltage. Both amplitudes of the voltage can be read in the picture (number 1 and 2 means two channels, the first is the input voltage and the second is output voltage) and the time scale is 2 seconds per grid. In Figure 10(b) the overshoot is about ±0.7V and neglecting the effect of the ripple wave, the output voltage has an average value of 4.95V within 5% tolerable error. Thus the buck converter with the MPC controller can reach a desired output voltage and the overshoot in the process is less than that in the open loop system. Moreover, the MPC controller generates little overshoot when the input voltage changes in a sudden and it gets into the stable state faster than open loop.

As the scope cannot save all the images to describe the whole process, the LABVIEW host provided by the cRIO will be used to observe a period of the process instead of a transient one. Figure 11 presents the sawtooth wave input and the output with fixed duty-cycle in open loop and MPC controller in closed loop system respectively. The LABVIEW host figure 11 records a history data after the system runs in 15 minutes (we choose only the first minute of the two states to show and compare) and x-axis of each picture depicts a period of time with the unit "minute". The system first works in an open-loop state and the output voltage violates with some oscillations shown in Figure 11(a). Figure 11(b) shows after several seconds in the 11th minute of the process, the system changes into close-loop state (the switch between open-loop and close-loop is operated online in the FPGA-host) and the output voltage converges and stays at a certain value 5V.

V. CONCLUSION

An effective computational method for linear parameter varying MPC and its application to buck converters is proposed and successfully tested in numerical simulations and FPGA platform experiments in this paper. The method mainly bases on a piecewise smooth Newton method with an exact line search used to solve a non-condensed QP formulation. As shown in this paper, this non-condensed MPC problem can be reformed to a non-condensed QP problem easily with no matrix-multiplication and most coefficient matrices are sparse as well as block triangular or diagonal which make the inverse during the optimization cost cheap. Meanwhile, the PWA equations for solving the QP problem require a low number of iterations, resulting in low CPU runtime when implementing the Newton method with exact line search because of the convergence properties [39]. Moreover the performance of the methodology shows well not only in comparison with several other algorithms though numerical test but also in a professional software simulation. Finally it is worth pointing out that this algorithm is implemented in the FPGA device to control the low velocity wind generator in the lab and shows its good response for the voltage's sudden break. In future research this algorithm will be tried in some other large-scale systems such as fluid transmission process [40], [41] which needs to track the multi-variables, UAV (unmanned aerial vehicle) to solve the problem composed by pitch angle, altitude, elevator angle with constraints [42] and some other systems benefitting from the long horizon of MPC such as drivers with LC filters [43], grid-connected converters [44] and some other high power converters [45], because of its fast runtime property in long prediction horizons and the good performance.

REFERENCES

[1] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[2] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control Theory and Design*. San Francisco, CA, USA: Nob Hill, 2009.

[3] A. Richards, "Fast model predictive control with soft constraints," *Eur. J. Control*, vol. 25, pp. 51–59, Sep. 2015.

[4] M. Rubagotti, P. Patrinos, and A. Bemporad, "Stabilizing linear model predictive control under inexact numerical optimization," *IEEE Trans. Autom. Control*, vol. 59, no. 6, pp. 1660–1666, Jun. 2014.

[5] W. Li and J. Swetits, "A new algorithm for solving strictly convex quadratic programs," *SIAM J. Optim.*, vol. 7, no. 3, pp. 595–619, 1997.

[6] P. Patrinos, P. Sopasakis, and H. Sarimveis, "A global piecewise smooth Newton method for fast large-scale model predictive control," *Automatica*, vol. 47, no. 9, pp. 2016–2022, 2011.

[7] 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, Nov. 2008.

[8] H. J. Ferreau, H. G. Bock, and M. Diehl, "An online active set strategy to overcome the limitations of explicit MPC," *Int. J Robust Nonlinear Control*, vol. 18, no. 8, pp. 816–830, 2007.

[9] J. Mattingley and S. Boyd, "CVXGEN: A code generator for embedded convex optimization," *Optim. Eng.*, vol. 13, no. 1, pp. 1–27, 2012.

[10] S. Richter, C. N. Jones, and M. Morari, "Real-time input-constrained MPC using fast gradient methods," in *Proc. 48th IEEE Conf. Decis. Control, Held Jointly 28th Chin. Control Conf. (CDC/CCC)*, Dec. 2009, pp. 7387–7393.

[11] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Trans. Control Syst. Technol.*, vol. 18, no. 2, pp. 267–278, Mar. 2010.

[12] S. Richter, C. N. Jones, and M. Morari, "Computational complexity certification for real-time MPC with input constraints based on the fast gradient method," *IEEE Trans. Autom. Control*, vol. 57, no. 6, pp. 1391–1403, Jun. 2012.

[13] C. Vlad, P. Rodriguez-Ayerbe, E. Godoy, and P. Lefranc, "Advanced control laws of DC–DC converters based on piecewise affine modelling. Application to a stepdown converter," *IET Power Electron.*, vol. 7, no. 6, pp. 1482–1498, 2014.

[14] C. Olalla, R. Leyva, I. Queinnec, and D. Maksimović, "Robust gain-scheduled control of switched-mode DC–DC converters," *IEEE Trans. Power Electron.*, vol. 27, no. 6, pp. 3006–3019, Jun. 2012.

[15] C. Olalla, R. Leyva, A. E. Aroudi, and I. Queinnec, "Robust LQR control for PWM converters: An LMI approach," *IEEE Trans. Ind. Electron.*, vol. 56, no. 7, pp. 2548–2558, Jul. 2009.

[16] R. El Houda Thabet and H. Chafouk, "Set-membership methodology for multiple fault detection and isolation in DC-DC buck converters," in *Proc. IEEE 18th Medit. Electrotechn. Conf. (MELECON)*, Apr. 2016, pp. 1–6.

[17] A. Dehghanzadeh, G. Farahani, H. Vahedi, and K. Al-Haddad, "Model predictive control design for DC-DC converters applied to a photovoltaic system," *Int. J. Elect. Power Energy Syst.*, vol. 103, pp. 537–544, Dec. 2018.

[18] M. P. Kumar, P. Ponnambalam, S. Sreejith, J. BelwinEdward, and K. Krishnamurthy, "Comparison of fuzzy and MPC based buck converter," in *Proc. IEEE Int. Conf. Power Electron., Drives Energy Syst. (PEDES)*, Dec. 2014, pp. 1–6.

[19] K. Gaouzi, H. El Fadil, A. Rachid, F. Z. Belhaj, and F. Giri, "Constrained model predictive control for dc-dc buck power converters," in *Proc. Int. Conf. Elect. Inf. Technol. (ICEIT)*, Nov. 2017, pp. 1–5.

[20] S. Vazquez, J. Rodriguez, M. Rivera, L. G. Franquelo, and M. Norambuena, "Model predictive control for power converters and drives: Advances and trends," *IEEE Trans. Ind. Electron.*, vol. 64, no. 2, pp. 935–947, Feb. 2017.

[21] S. Vazquez *et al.*, "Model predictive control: A review of its applications in power electronics," *IEEE Ind. Electron. Mag.*, vol. 8, no. 1, pp. 16–31, Mar. 2014.

[22] J. Alimeling and W. P. Hammer, "PLECS-piece-wise linear electrical circuit simulation for Simulink," in *Proc. IEEE Int. Conf. Power Electron. Drive Syst. (PEDS)*, vol. 1, Jul. 1999, pp. 355–360.

[23] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice—A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[24] W. Li and J. Swetits, "Regularized Newton methods for minimization of convex quadratic splines with singular Hessians," in *Reformulation: Nonsmooth, Piecewise Smooth, Semismooth and Smoothing Methods*. Springer, 1998, pp. 235–257.

[25] W. J. Leong and B. San Goh, "Convergence and stability of line search methods for unconstrained optimization," *Acta Applicandae Mathematicae*, vol. 127, no. 1, pp. 155–167, 2013.

[26] F. Facchinei and J.-S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Springer, 2007.

[27] R. Leyva, A. Cid-Pastor, C. Alonso, I. Queinnec, S. Tarbouriech, and L. Martinez-Salamero, "Passivity-based integral control of a boost converter for large-signal stability," *IEE Proc.-Control Theory Appl.*, vol. 153, no. 2, pp. 139–146, Mar. 2006.

[28] U. Sadek, A. Sarjaš, A. Chowdhury, and R. Svečko, "FPGA-based optimal robust minimal-order controller structure of a DC–DC converter with Pareto front solution," *Control Eng. Pract.*, vol. 55, pp. 149–161, Oct. 2016.

[29] R. Middlebrook and S. Cuk, "A general unified approach to modelling switching-converter power stages," in *Proc. IEEE Power Electron. Spec. Conf.*, Jun. 1976, pp. 18–34.

[30] R. D. Middlebrook and S. Ćuk, "A general unified approach to modelling switching-converter power stages," *Int. J. Electron. Theor. Exp.*, vol. 42, no. 6, pp. 521–550, 1977.

[31] A. Afram and F. Janabi-Sharifi, "Theory and applications of HVAC control systems—A review of model predictive control (MPC)," *Building Environ.*, vol. 72, pp. 343–355, Feb. 2014.

[32] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[33] G. Banjac, P. Goulart, B. Stellato, and S. Boyd. (2017). *Infeasibility Detection in the Alternating Direction Method of Multipliers for Convex Optimization*. [Online]. Available: http://optimization-online.org

[34] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: A parametric active-set algorithm for quadratic programming," *Math. Program. Comput.*, vol. 6, no. 4, pp. 327–363, 2014.

[35] P. Patrinos and A. Bemporad, "An accelerated dual gradient-projection algorithm for embedded linear model predictive control," *IEEE Trans. Autom. Control*, vol. 59, no. 1, pp. 18–33, Jan. 2014.

[36] Z. Gu, E. Rothberg, and R. Bixby. *Gurobi Optimizer v7.0*. [Online]. Available: http://www.gurobi.com

[37] D. E. Quevedo, R. P. Aguilera, and T. Geyer, "Predictive control in power electronics and drives: Basic concepts, theory, and methods," in *Advanced and Intelligent Control in Power Electronics and Drives*. Springer, 2014, pp. 181–226.

[38] C. Dase, J. S. Falcon, and B. Maccleery, "Motorcycle control prototyping using an FPGA-based embedded control system," *IEEE Control Syst.*, vol. 26, no. 5, pp. 17–21, Oct. 2006.

[39] B. Chen and M. Pinar, "On Newton's method for Huber's robust M-estimation problems in linear regression," *BIT Numer. Math.*, vol. 38, no. 4, pp. 674–684, 1998.

[40] Z. Ren, C. Xu, Q. Lin, R. Loxton, and K. L. Teo, "Dynamic optimization of open-loop input signals for ramp-up current profiles in tokamak plasmas," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 32, pp. 31–48, Mar. 2016.

[41] Z. Ren, Z. Zhao, Z. Wu, and T. Chen, "Dynamic optimal control of a one-dimensional magnetohydrodynamic system with bilinear actuation," *IEEE Access*, vol. 6, pp. 24464–24474, 2018.

[42] K.-V. Ling, B. F. Wu, and J. Maciejowski, "Embedded model predictive control (MPC) using a FPGA," *IFAC Proc. Vol.*, vol. 41, no. 2, pp. 15250–15255, 2008.

[43] T. Geyer, P. Karamanakos, and R. Kennel, "On the benefit of long-horizon direct model predictive control for drives with LC filters," in *Proc. IEEE Energy Convers. Congr. Expo. (ECCE)*, Sep. 2014, pp. 3520–3527.

[44] C. S. Lim, S. S. Lee, X. Kong, and I. U. Nutkani, "Long horizon linear MPC of grid-connected VSIs: Regulation problems and a plug-in solution," in *Proc. IEEE Energy Convers. Congr. Expo. (ECCE)*, Oct. 2017, pp. 4950–4956.

[45] T. Geyer, "Model predictive control of high power converters and industrial drives," in *Proc. IEEE Energy Convers. Congr. Expo. (ECCE)*, Oct. 2017, pp. 1–260.

ZHEN LIU received the bachelor's degree in electrical engineering and automation from Zhejiang University, Hangzhou, China, in 2009, and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, in 2018. His research interests include model predictive control, numerical simulation, optimal control theory, and applications on embedded platform in power electronics.

LEI XIE received the B.S. and Ph.D. degrees from Zhejiang University, China, in 2000 and 2005, respectively. From 2005 to 2006, he was a Post-Doctoral Researcher with the Berlin University of Technology and an Assistant Professor from 2005 to 2008. He is currently a Professor with the College of Control Science and Engineering, Zhejiang University. His research activities culminated in over 40 articles that are published in internationally renowned journals and conferences, three book chapters, and a book in the area of applied multivariate statistics and modeling. His research interests focus on the interdisciplinary area of statistics and system control theory.

ALBERTO BEMPORAD (F'10) received the master's degree in electrical engineering and the Ph.D. degree in control engineering from the University of Florence, Italy, in 1993 and 1997, respectively. In 1996 and 1997, he was with the Center for Robotics and Automation, Department of Systems Science and Mathematics, Washington University, St. Louis. From 1997 to 1999, he held a post-doctoral position with the Automatic Control Laboratory, ETH Zurich, Switzerland, where he collaborated as a Senior Researcher until 2002. From 1999 to 2009, he was with the Department of Information Engineering, University of Siena, Italy, becoming an Associate Professor in 2005. From 2010 to 2011, he was with the Department of Mechanical and Structural Engineering, University of Trento, Italy. Since 2011, he has been a Full Professor with the IMT School for Advanced Studies Lucca, Italy, where he served as the Director of the institute from 2012 to 2015. He spent visiting periods at the University of Michigan, Zhejiang University, and Stanford University. In 2011, he cofounded ODYS S.r.l., a consulting and software development company specialized in advanced controls and embedded optimization algorithms. He has authored over 300 papers in the area of model predictive control, automotive control, hybrid systems, multiparametric optimization, computational geometry, robotics, and finance. He holds eight patents. He has authored or co-authored various MATLAB toolboxes for model predictive control design, including the Model Predictive Control Toolbox and the Hybrid Toolbox. He was a recipient of the IFAC High-Impact Paper Award for the 2011–2014 triennial. He was an Associate Editor of the IEEE Transactions on Automatic Control from 2001 to 2004 and the Chair of the Technical Committee on Hybrid Systems of the IEEE Control Systems Society from 2002 to 2010.

SHAN LU received the B.S. and Ph.D. degrees from Zhejiang University, China, in 2011 and 2016, respectively. He is currently an Assistant Professor with the Institute of Intelligence Science and Engineering, Shenzhen Polytechnic. His research interests include modeling and optimization of supply chain scheduling, machine learning, and process modeling.