

An Optimal Control Application in Power Electronics Using Numerical Algebraic Geometry

Daniel J. Bates[†], A. G. Beccuti*, Ioannis A. Fotiou* and Manfred Morari*

Abstract—We present an optimal control application in power electronics using the homotopy continuation method for solving systems of polynomial equations. The proposed approach breaks the computations associated with the optimal control problem into two parts, an off-line and an on-line. In the off-line part, the approach solves a generic polynomial system by means of a linear homotopy and stores its solution. Then, the on-line part uses this solution and, given the initial state value, it calculates by means of a coefficient parameter homotopy the optimal control input of the problem. The approach exhibits a probability-one guarantee of finding the global optimal solution to the problem at hand.

I. INTRODUCTION

Optimal control is a very active research area with broad attention from industry [1]. It is among the few control methodologies providing a systematic way to perform nonlinear control synthesis while handling at the same time system constraints. To a great extent, it is thanks to this capability of handling constraints on system states, inputs and outputs that model predictive control (MPC) has proven to be very successful in practice [2].

Model predictive control uses on-line optimization to obtain the solution of the optimal control problem in real time. For the sake of simplicity and because of implementation issues on a digital control unit, the optimal control problem is typically cast into a discrete time mathematical program, whose solution yields a sequence of optimal control moves. To ensure feedback, i.e. to deal with model uncertainties and disturbances, only the first move of the sequence is actually applied to the plant, and the optimization is repeated with new data measurements. This scheme is known as receding horizon control (RHC).

Technology and cost factors, however, make the implementation of receding horizon control difficult, if not impossible. In certain cases, we can circumvent these issues by computing the solution of the optimal control problem off-line, i.e. by solving the corresponding mathematical program parametrically. That is, the explicit formula giving the solution (the control inputs) of the program as a closed-form expression of the problem parameters (the measured state) is computed off-line. The solution then is efficiently implemented on-line as a lookup table. This approach is also known as the “explicit” MPC solution technique [3].

Motivated by this framework, in the present work we apply the approach recently proposed in [4] to the optimal control

of a power electronics circuit, the buck dc-dc converter. The converter is first modelled as a nonlinear polynomial system, i.e. a system whose state update equation is given by a polynomial vector field. The homotopy continuation method for solving systems of polynomial equations [5] is then used to obtain the solution of a generic polynomial system. Subsequently, given the state measurement, an on-line algorithm uses the precomputed generic system solution to efficiently solve the actual optimal control problem and retrieve the optimal control input to the dc-dc converter.

In contrast to other parametric optimization approaches based on continuation suggested mainly in the nineties (see [6] for a more recent survey), the approach followed here [4] is based on polynomial homotopy continuation over the field of *complex* numbers. This bears the advantage of having a deep and rich supporting theory which can be used, among other things, to avoid singularities.

II. OPTIMAL CONTROL

Let $z \in \mathbb{R}^r$ be the decision-variable vector and $x_0 \in \mathbb{R}^n$ be the vector of parameters. The constrained optimal control problem discussed in this paper can be formulated as a polynomial optimization problem of the general form

$$\min_z J(z, x_0) \quad \text{s.t.} \quad \begin{cases} q(z, x_0) \leq 0 \\ h(z, x_0) = 0 \end{cases}, \quad (1)$$

where $J(z, x_0) \in \mathbb{R}[z, x_0]$ is the objective function and $q \in \mathbb{R}[z, x_0]^M$, $h \in \mathbb{R}[z, x_0]^A$ are vector polynomial functions representing the inequality and equality constraints. Control systems lending themselves to such formulations are those whose vector fields are polynomial or, in some cases like the one studied in this paper, *rational* functions of the state and control input.

A. Parametric optimization

In the optimal control context, we want to minimize the function $J(z, x_0)$ with respect to z for any given value of the parameter $x_0 \in \mathcal{X} \subseteq \mathbb{R}^n$, where \mathcal{X} is the set of admissible parameter values. This is what we call parametric optimization. In other words, the polynomial parametric optimization problem consists of finding an efficient computational procedure for evaluating the maps

$$\begin{array}{ccc} z^*(x_0) : \mathbb{R}^n & \longrightarrow & \mathbb{R}^r \\ x_0 & \longmapsto & z^* \end{array} \quad \begin{array}{ccc} J^*(x_0) : \mathbb{R}^n & \longrightarrow & \mathbb{R} \\ x_0 & \longmapsto & J^* \end{array}, \quad (2)$$

where

$$z^* = \arg \min_z J(z, x_0) \quad \text{and} \quad J^* = \min_z J(z, x_0). \quad (3)$$

[†]Institute for Mathematics and Its Applications, University of Minnesota, MN 55455-0436, USA

*Automatic Control Laboratory, ETH Zurich, 8092 Zurich, Switzerland
 Corresponding author: fotiou@control.ee.ethz.ch

B. Posing the problem

We assume throughout the paper that the feasible set defined by $q(z, x_0)$ and $h(z, x_0)$ is compact and hence the minimum is attained. Furthermore we assume certain regularity conditions (so-called constraint qualifications [7]) such that the optimizer is attained at a Karush-Kuhn-Tucker (KKT) point [8], which is a point in the solution set of system

$$\begin{aligned} \nabla_z J(z, x_0) + \sum_{i=1}^{\mathcal{M}} \mu_i \nabla_z q_i(z, x_0) \\ + \sum_{i=1}^{\Lambda} \lambda_i \nabla_z h_i(z, x_0) &= 0 \\ h(z, x_0) &= 0 \\ \mu_i q_i(z, x_0) &= 0 \\ \mu_i &\geq 0 \\ q(z, x_0) &\leq 0, \end{aligned} \quad (4)$$

where $\mu_i, i = 1, \dots, \mathcal{M}$ and $\lambda_i, i = 1, \dots, \Lambda$ are the Lagrange multipliers (the λ_i are free), associated with the inequality and equality constraints, respectively.

The assumptions made above are nontrivial. Nonetheless, they are satisfied in many practical situations and are generally assumed in parametric optimization. Note that for the class of problems we consider, the first three relations of the KKT conditions (4) form a *square system of polynomial equations* in z, μ and λ . We assume that the solution of KKT system (4) is a set of isolated points.

III. HOMOTOPY CONTINUATION

Homotopy continuation is a numerical technique for finding the *isolated* complex solutions of square polynomial systems, i.e., those not on positive-dimensional solution components (such as curves or surfaces). This section contains a brief overview of homotopy continuation for polynomial systems. As this method was employed in a similar setting in [4], further details may be found there. Full details are available in [5].

The first step in applying homotopy continuation to a square system $f(z) : \mathbb{C}^r \rightarrow \mathbb{C}^r$ of polynomial equations is to cast $f(z)$ as a member of a parameterized family of polynomial systems \mathcal{F}_σ , where σ is the vector of parameters. Almost all polynomial systems in any such family will have the same number of solutions, M . Those corresponding to a certain measure zero set χ of the parameter space could have fewer than M solutions, but no system in the family can have more. The key property when choosing a family \mathcal{F}_σ for $f(z)$ is that \mathcal{F}_σ must contain a polynomial system $g(z) : \mathbb{C}^r \rightarrow \mathbb{C}^r$ having M *known* solutions. Such families often arise naturally, based on the natural parameters of the problem, although there are numerous ways of producing such families artificially if necessary. Methods for constructing artificial families are discussed in detail in Chapter 8 of [5]. Both types of families are used in the optimal control approach followed in this paper: naturally parameterized families are employed in the on-line part of the algorithm, while artificial families are used in the off-line part.

Once the family \mathcal{F}_σ and the *start system* $g(z)$ have been chosen, the next step is to form a homotopy $H(z, t) : \mathbb{C}^r \times \mathbb{R} \rightarrow \mathbb{C}^r$ equal to $g(z)$ at $t = 1$ and $f(z)$ at $t = 0$. The most

common choice is the linear homotopy $H(z, t) = f(z) \cdot (1 - t) + \gamma \cdot g(z) \cdot t$, where $\gamma \in \mathbb{C}$. By choosing $g(z)$ to have M solutions and by choosing γ randomly, there is a *probability one* guarantee that every system H between $t = 1$ and $t = 0$ will have exactly M solutions, except possibly at $t = 0$, thus forming M solution paths connecting the solutions at $t = 1$ to those at $t = 0$.

Beginning with the known solutions of $g(z)$ at $t = 1$, standard predictor-corrector techniques may be employed to follow the solution paths to $t = 0$. In particular, an ODE method such as Euler's method may be used to predict to a smaller value of t from a known point on the path, after which Newton's method may be used to bring the predicted point closer to the solution path for a fixed value of t . Specialized methods, discussed in [5], are used to handle the presence of singularities.

All told, homotopy continuation provides a means for finding approximations to all complex isolated solutions of a polynomial system. It should be noted that these approximations may be found as accurately as desired. Also, the real isolated solutions can be extracted from the set of all (complex) solutions by considering the size of the imaginary part.

IV. THE OPTIMAL CONTROL ALGORITHM

The proposed algorithm consists of two parts: the off-line part, where a multihomogeneous start system associated with the KKT system (4) is used to solve a generic polynomial system $\mathcal{F}_{\tilde{x}_0}$, where $\mathcal{F}_{\tilde{x}_0}$ denotes the polynomial system in family \mathcal{F}_σ that we obtain by specializing the parameter vector σ to the value \tilde{x}_0 ; and the on-line part, where given the (state measurement) value \hat{x}_0 of the (natural) parameter x_0 , the precomputed solution points are tracked from $\mathcal{F}_{\tilde{x}_0}$ to the desired $\mathcal{F}_{\hat{x}_0}$, in order to efficiently arrive at the optimal solution of the original optimization problem (1), i.e., the optimal control input to the problem at hand.

A. Off-line Part

Assume that we want to solve problem (4) for all possible values of the parameter x_0 . As a first step we ignore all inequality constraints to get a naturally parametrized family $\mathcal{F}_{x_0}^{\text{nat}}$ of square systems of polynomial equations parametrized by $x_0 \in \mathcal{X}$:

$$\mathcal{F}_{x_0}^{\text{nat}}(w) : \begin{cases} \nabla_z J(z, x_0) + \sum_{i=1}^{\mathcal{M}} \mu_i \nabla_z q_i(z, x_0) \\ + \sum_{i=1}^{\Lambda} \lambda_i \nabla_z h_i(z, x_0) = 0 \\ \mu_i q_i(z, x_0) = 0 \\ h(z, x_0) = 0, \end{cases} \quad (5)$$

where $w \in \mathbb{C}^{r+\mathcal{M}+\Lambda}$ denotes the compound vector consisting of all z, μ_i and λ_i appearing in (5). We now want a generic member of the family of naturally parametrized polynomials $\mathcal{F}_{x_0}^{\text{nat}}$. Therefore, we fix x_0 to a random complex value $\tilde{x}_0 \in \mathbb{C}^n$ to obtain instance $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$. System $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$ is generic in the family of naturally parametrized polynomial systems $\mathcal{F}_{x_0}^{\text{nat}}$ because we have chosen a generic (i.e., random) parameter value \tilde{x}_0 .

To solve $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$, we embed it in a multihomogeneous family $\mathcal{F}_{\sigma}^{\text{mhom}}$ and, by choosing an appropriate start system $g(w) \in \mathcal{F}_{\sigma}^{\text{mhom}}$, we construct the homotopy

$$H(w, t) = (1 - t)\mathcal{F}_{\hat{x}_0}^{\text{nat}}(w) + \gamma t g(w) \quad (6)$$

to trace the solutions of $g(w)$ to those of $\mathcal{F}_{\hat{x}_0}^{\text{nat}}(w)$. Please refer to §8.4 of [5] regarding the construction of $g(w)$.

The off-line part of the algorithm then solves the generic system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ by means of homotopy (6) and stores the finite solution points, denoted with $\tilde{w}^{(j)} \in \mathbb{C}^{r+\mathcal{M}+\Lambda}$, $j = 1, \dots, \tilde{n}_w$, as well as the corresponding \tilde{x}_0 . A summary of the off-line algorithm appears in Algorithm 1.

Algorithm 1 *Off-line Part: Compute the solution of a generic instance $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ of the family $\mathcal{F}_{x_0}^{\text{nat}}$ of polynomial systems*

Input: Parametrized family of systems $\mathcal{F}_{x_0}^{\text{nat}}$
Output: $\tilde{x}_0 \in \mathbb{C}^n$ and $\tilde{w}^{(j)} \in \mathbb{C}^{r+\mathcal{M}+\Lambda}$, $j = 1, \dots, \tilde{n}_w$.
 1: Choose a random $\tilde{x}_0 \in \mathbb{C}^n$ and form $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$.
 2: Construct an appropriate start system $g(w) \in \mathcal{F}_{\sigma}^{\text{mhom}}$.
 3: Find solutions to $g(w)$, denoted with $w_g^{(j)}$, $j = 1, \dots, n_g (\geq \tilde{n}_w)$.
 4: Track solution paths from $w_g^{(j)}$ to obtain solutions $\tilde{w}^{(j)}$, $j = 1, \dots, \tilde{n}_w$ of system $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$.

B. On-line Part

Given the solutions $\tilde{w}^{(j)}$, $j = 1, \dots, \tilde{n}_w$ of system $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$, the on-line part of the approach takes the measurement value \hat{x}_0 of the parameter x_0 and computes the optimum $J^*(\hat{x}_0)$ and an optimizer $z^*(\hat{x}_0)$. The three main steps of the on-line part are:

- 1) Solve system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$, using coefficient parameter homotopy with $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$ as the start system;
- 2) Remove non-real and infeasible (with respect to the inequalities in (4)) solutions; and
- 3) Among the remaining feasible, real solutions, select the one that induces the minimum cost in the objective function.

The on-line part of the algorithm considers system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ as the start system and, given the on-line state measurement $x_0 = \hat{x}_0$, it constructs a coefficient parameter homotopy from \tilde{x}_0 to \hat{x}_0 in order to solve $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ via continuation from the solutions of $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$. This is possible, because we have already computed the solutions to $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$ in the off-line part of the algorithm.

After system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ has been solved, we check all its solutions $w^{(j)} \in \mathbb{C}^{r+\mathcal{M}+\Lambda}$, $j = 1, \dots, n_w$ and keep the ones that are both real and feasible with respect to the inequalities in (4).

Let the set of real feasible solutions w with respect to the constraints in (4) be S . Furthermore, let the set of all $z^{(j)}$ contained in the compound vectors $w^{(j)} \in S$ form the set S_z . The optimizer $z^*(\hat{x}_0)$ for parametric optimization problem (1) can then be determined by selecting that $z \in S_z$,

which induces the minimum cost in the objective function, i.e.,

$$z^*(\hat{x}_0) = \arg \min_{z \in S_z} J(z, \hat{x}_0),$$

where the optimal cost is

$$J^*(\hat{x}_0) = \min_{z \in S_z} J(z, \hat{x}_0).$$

A summary of the on-line part of the proposed approach can be seen in Algorithm 2.

Algorithm 2 *On-line part: Requires system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ and its solutions $\tilde{w}^{(j)}$, $j = 1, \dots, \tilde{n}_w$.*

Input: Parameter value \hat{x}_0 .
Output: Optimal cost $J^*(\hat{x}_0)$ and an optimizer $z^*(\hat{x}_0)$.
 1: Construct the coefficient parameter homotopy from $\mathcal{F}_{\tilde{x}_0}^{\text{nat}}$ to $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$ and solve it to obtain solutions $w^{(j)}$, $j = 1, \dots, n_w$ of $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$.
 2: Keep the real feasible solutions respecting the inequalities of the optimality conditions and store them in set S .
 3: Form the discrete set S_z of all candidate optimizers.
 4: Compare values $J(z, \hat{x}_0)$ for all $z \in S_z$ and select optimizer $z^*(\hat{x}_0)$ and corresponding optimal cost $J^*(x_0)$.

Remark 1: One important feature of the presented approach lies in the fact that it scales well with the dimension of the parameter vector x_0 , in contrast to similar existing methods [9]. This is due to the fact that parameter x_0 is not treated symbolically - therefore its dimension is relatively irrelevant regarding the computational complexity of the approach - because a specific value (\tilde{x}_0 or \hat{x}_0) is always assigned to it.

V. MODELLING OF THE DC-DC CONVERTER

In this paper the controller synthesis problem of the fixed-frequency buck dc-dc converter is considered (Figure 1). The semiconductor switches are operated by a pulse sequence with constant switching frequency f_s (resp. period T_s) so that it is then possible to regulate the dc component of the output voltage v_o (lower in value than the available dc voltage source v_s , hence the name *buck*) by appropriately commutating the switches.

More specifically, as a consequence of its switched topology the buck converter features two operation modes with two different affine dynamics. At the beginning of each period k , switch S_1 is conducting, S_2 is not, and the controller selects the control input, the duty cycle $d(k) = \frac{t_1(k)}{T_s}$ with $t_1(k) \leq T_s$, determining when the switch from the first mode to the second takes place. During the time interval $kT_s \leq t < (k + d(k))T_s$, power is transferred from the voltage source to the load through S_1 . At the end of this interval S_1 is switched off and S_2 is switched on so that v_s is now disconnected from the circuit. Subsequently, at the end of the period, switch S_1 (S_2) is switched on (off) and the procedure is periodically repeated. On the basis of this

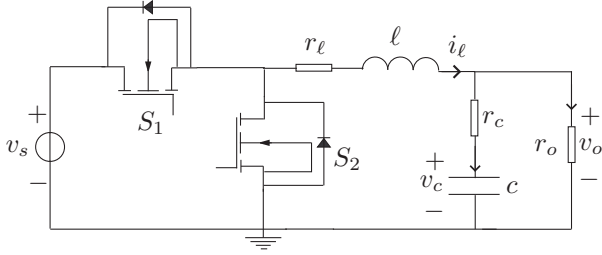


Fig. 1. Topology of the buck converter. r_o denotes the output load resistance, r_c and r_l are the parasitic resistances of the capacitor and the inductor and c and l represent the capacitance and inductance of the converter.

principle of operation, $d(k)$ is physically constrained to lie in the interval $[0, 1]$.

For the two possible modes of operation the evolution of the system can be respectively described in terms of the following set of differential equations:

$$\dot{x} = Ax + B, \quad S_1 \text{ is on} \quad (7)$$

$$\dot{x} = Ax, \quad S_2 \text{ is on} \quad (8)$$

where matrix A and vector B are obtainable through elementary circuit theory and the chosen state is $x(t) = [i'_l(t) \ v'_o(t)] = [\frac{i_l(t)}{v_s} \ \frac{v_o(t)}{v_s}]^T$, i.e., the inductor current and the output voltage normalized over the voltage source.

A. Nonlinear control model

The principle of operation of the circuit naturally lends itself to formulating a discrete time model of the system with sampling interval T_s . In particular, through a forward Euler approximation over T_s of (7), (8) one has the expressions

$$x(k+1)_{\text{mode } 1} = x(k) + (Ax(k) + B)T_s \quad (9)$$

$$x(k+1)_{\text{mode } 2} = x(k) + (Ax(k))T_s, \quad (10)$$

respectively valid for the system update if the duty cycle were 1 (circuit always with S_1 in conduction) or 0 (circuit always with S_2 in conduction). The control model dynamics then assume the form

$$\begin{aligned} x(k+1) &= x(k+1)_{m1}d(k) + x(k+1)_{m2}(1-d(k)) \\ &= \Phi(r_o)x(k) + \Psi(r_o)d(k) \end{aligned} \quad (11a)$$

$$0 \leq d(k) \leq 1, \quad (11b)$$

wherein the two discrete time modes are averaged over the period according to their effective duration and where it can be easily verified that matrix $\Phi(r_o)$ and vector $\Psi(r_o)$ depend on a nonlinear (rational) function of the load r_o .

Remark 2: To deal with the rational expressions appearing in relations (11) and make them conform to the polynomial framework presented in Section II, we clear denominators from their right-hand side. This does not pose any limitations, since the quantities appearing in the denominators are nonzero physical quantities. We then obtain polynomial equations, representing the state-update rational relation (11a), which are in turn used as equality constraints in the associated optimal control problem formulation (1).

B. Control scheme

1) *Control objectives:* The main control objective for the boost dc-dc converter is to regulate the (average) dc component of the output voltage v_o to its reference $v_{o,ref}$, with an accuracy of $\pm 1\%$. This regulation has to be achieved in the presence of the hard constraints on the manipulated variable (the duty cycle) which is bounded between 0 and 1, and on the inductor current, which must not exceed its maximum value $i_{l,max}$, and must be maintained in the face of measurable variations in v_s and the load r_o ¹. Moreover, the controller must render a steady state operation point with constant duty cycle, thus avoiding the occurrence of fast-scale instabilities (subharmonic oscillations).

2) *Constrained finite time optimal control:* The control model (11a) and (11b) must be augmented to account for the constraint on the inductor current. Through simple considerations [10] it can be seen that this constraint can be directly captured by simply adding the (nonlinear) inequality

$$i'_l(k) + d(k) \frac{1 - v'_o}{l} \leq i'_{l,max}, \quad (12)$$

where $i'_{l,max} = \frac{i_{l,max}}{v_s}$ is the maximum current scaled over the source voltage and l is the coil inductance, as can also be seen in Figure 1.

The control objectives are to regulate the output voltage to its reference as fast and with as little overshoot as possible, or equivalently, to minimize the quadratic scaled output voltage error $v'_{o,err}(k) = (v'_o(k) - v'_{o,ref})^2$. Let $\Delta d(k) = (d(k) - d(k-1))^2$ indicate the quadratic value of the difference between two consecutive duty cycles. This term is introduced in order to reduce the presence of unwanted chattering in the input when the system has almost reached stationary conditions and entails adding $d(k-1)$ to the parameter variables (initial state variables). The reference $v_{o,ref}$ may then change during operation and since all circuit values are represented as scaled over v_s , also $i'_{l,max}$ will change whenever v_s does; lastly, r_o will also be subjected to step variations. Summing up, $v'_{o,ref}$, $i'_{l,max}$ and r_o must thus likewise be featured among the parameter variables.

Define the penalty matrix $Q = \text{diag}(q_1, q_2)$ with $q_1, q_2 \in \mathbb{R}^+$ and the vector $\varepsilon(k) = [v'_{o,err}(k) \ \Delta d(k)]^T$. Consider the objective function

$$J(z, x(k), d(k-1), v'_{o,ref}, i'_{l,max}, r_o) = \sum_{j=0}^{L-1} Q \varepsilon(k+j|k), \quad (13)$$

which penalizes the predicted evolution of $\varepsilon(k+j|k)$ from time-instant k on over the finite horizon L , where $z = [X_1^L, D(k)]$, with $D(k) := [d(k), \dots, d(k+L-1)]$ and $X_1^L := [x(k+1), \dots, x(k+L)]$. Vector $z \in \mathbb{R}^{(m+n)L}$ is the optimization vector containing all the decision variables (present and future states and control inputs) and $x(k) \equiv x_0$ is the initial state (on-line measurement) vector of the discrete-time system.

¹The assumption that the load r_o is effectively measurable is not trivial in effective industrial practice and would for example require the implementation of an extended Kalman filter.

The control input at time-instant k is then obtained by minimizing the objective function (13) over the sequence of control moves $D(k) = [d(k), \dots, d(k+L-1)]^T$ (in practice over the space of the decision vector z) subject to the related system equations and constraints for the model (11a), (11b) and (12); the resulting nonlinear optimization program is referred to as the constrained finite time optimal control (CFTOC) problem and is solved by means of the algorithm presented in Section IV; specifically, the augmented state vector $[x(k)^T, d(k-1), v'_{o,ref}, i'_{l,max}, r_o]^T$ represents the parameter vector x_0 and constraints (11b), (12) are subsumed by $q(z, x_0)$, as defined in (1).

Remark 3: It should be mentioned that a similar optimal control scheme, based on a piecewise affine approximation of the converter dynamics (11a) was presented in previous work [10], [11], [12]. Therein however the particular modelling framework does not allow for the inclusion of r_o as a parametric state, so that load variations have to be dealt with through an external estimation loop appropriately adjusting the controller reference to compensate for prediction errors. The present approach on the other hand *inherently comprises such variations within the controller itself by directly modelling the intrinsically nonlinear expression (11a), thereby forgoing the need for an external loop.*

VI. NUMERICAL RESULTS

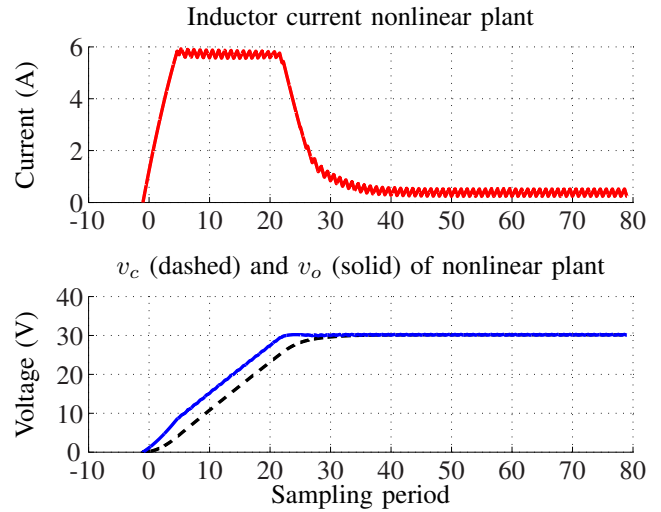
In this section, we present simulation results demonstrating the performance of the proposed control methodology. The circuit parameters are given by $x_c = 234 \mu F$, $x_l = 75 \mu H$, $r_c = 0.15 \Omega$, $r_l = 0.3 \Omega$ and $r_o = 15 \Omega$. If not otherwise stated, the voltage source is $v_s = 50 V$; the output voltage reference is set to $v_{o,ref} = 30 V$ and $i_{l,max} = 6 A$, and a switching frequency of $100 kHz$ is assumed so that the switching period is $10 \mu s$. For the cost function, the penalty matrix is chosen to be $Q = \text{diag}(20, 1)$ and the prediction horizon is $L = 2$. Longer horizons are also possible but for the considered case study they actually tend to de-tune the controller, thereby reducing its performance. The simulation cases considered represent different scenarios typically of interest in practical applications.

A. Startup

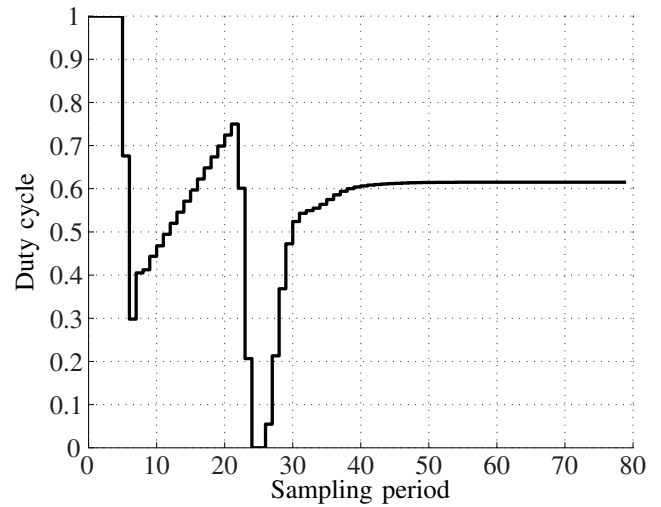
The first case to be analyzed is that of the transient behavior during startup. Figure 2(a) and Figure 2(b) depict the step responses of the different schemes during start-up, i.e. when $x(0) = [0, 0]^T$. The proposed optimal control scheme yields an output voltage that reaches its desired stationary value within 25 switching periods. The hard constraint on the inductor current is respected, in that the controller succeeds in keeping the current always below the value of $6 A$.

B. Voltage source and load disturbance

For the second case, results stemming from a 50% increase in the voltage source v_s during the previously attained steady state operation and from a 90% decrease in the output load r_o are shown in Figures 3(a)-3(b); more specifically, the voltage source increase occurs at the 90th switching period and on



(a) Inductor current (above) and output/capacitor voltage (below)



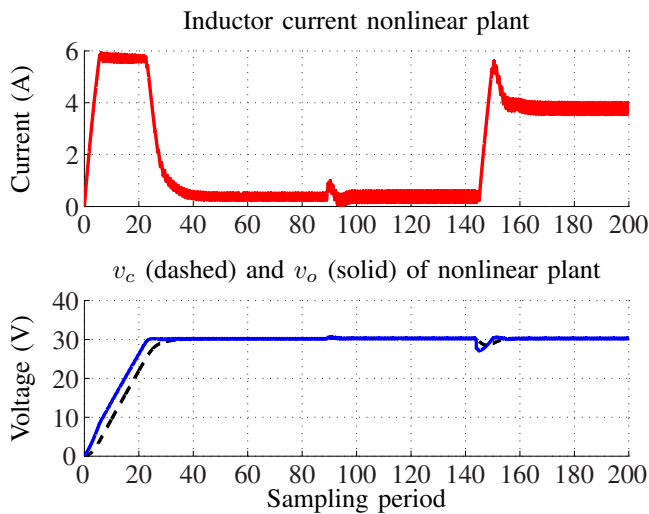
(b) Duty cycle during the startup phase

Fig. 2. Simulation results for the startup scenario

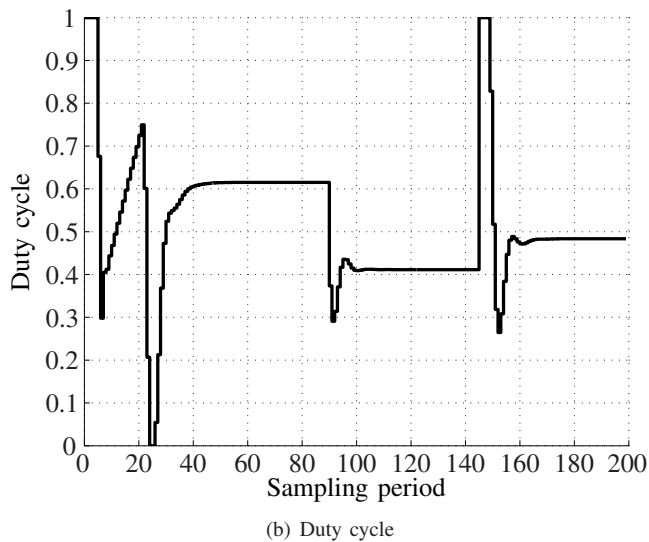
top of that, at the 145th switching period, the load decrease kicks in. In the simulation diagrams we can see that the controller can cope with both disturbances, guaranteeing a constant regulated output voltage of $30 V$.

C. Implementation details

The method described in this paper has been implemented as a Matlab routine which makes calls to the Bertini software package [13] for the solution of polynomial systems. For the off-line part of the algorithm, the Matlab routine writes a Bertini input file containing the generic system to be solved, which Bertini solves via a multihomogeneous start system with the standard linear homotopy. The Matlab routine then reads in the list of finite solutions. The finite solutions, along with the appropriate coefficient parameter homotopy, are written to Bertini-style input files for each step of the on-line part of the algorithm. Matlab reads in the real, finite solutions after each on-line step and interprets the results, as discussed in Section IV.



(a) Inductor current (above) and output/capacitor voltage (below)



(b) Duty cycle

Fig. 3. Simulation results for the scenario featuring a 50% increase of v_s and a 90% decrease in the output load r_o .

For the off-line part of the application discussed in this paper, there were 1392 paths to be tracked, leading to 19 finite, multiplicity one solutions. The tracking for the off-line part of the algorithm took 88 seconds on a 3.2 GHz Pentium 4 machine. The on-line part of the algorithm involved tracking the 19 finite solutions from the off-line part through a coefficient parameter homotopy. The tracking for each iteration took 0.2 seconds on the same machine. All 19 finite paths stayed finite, although in every case, two paths landed at complex solutions while the other 17 were real.

It should be noted that these timings are highly dependent upon the settings of various tracking-related tolerances (which are too numerous to detail) and also include the file read-write overhead. Therefore, they are meant to merely convey the order of magnitude of time needed to solve the example.

VII. CONCLUSIONS

In this paper, we have applied a new approach to nonlinear constrained optimal control to a power electronics case study, namely the control of the buck dc-dc converter. The approach uses the homotopy continuation method for polynomial systems to precompute off-line the solution set of a generic start system $\mathcal{F}_{\hat{x}_0}^{\text{nat}}$, associated with the optimal control problem at hand. Then, given the value \hat{x}_0 (measured state) of the parameter x_0 , the on-line stage of the algorithm utilizes coefficient parameter homotopy to retrieve the optimal control input of the optimal control problem.

The approach extends to the general setting of nonlinear parametric optimization and it exhibits a probability-one guarantee of finding the global optimum, without relying on approximations or convexity assumptions.

The illustrated application of the dc-dc converter is tackled successfully by means of the proposed approach, one big advantage of which is that the dimension of the parameter vector no longer poses a restrictive factor - not at least to the extent that similar methods do.

REFERENCES

- [1] S. Qin and T. Badgwell, "A Survey of Industrial Model Predictive Control Technology," *Control Engineering Practice*, vol. 11, pp. 733–764, 2003.
- [2] M. Morari and J. Lee, "Model Predictive Control: Past, Present and Future," *Computers and Chemical Engineering*, vol. 23, pp. 667–682, 1999.
- [3] A. Bemporad, M. Morari, V. Dua, and E. Pistikopoulos, "The Explicit Linear Quadratic Regulator for Constrained Systems," *Automatica*, vol. 38, pp. 3–20, 2002.
- [4] D. Bates, I. Fiotou, and P. Rostalski, "A Numerical Algebraic Geometry Approach to Nonlinear Constrained Optimal Control," in *IEEE Conference on Decision and Control*, New Orleans, LA, December 2007, pp. 6256–6261.
- [5] A. Sommese and C. Wampler, *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. Hackensack, NJ: World Scientific Publishing, 2005.
- [6] E. Allgower and K. Georg, "Continuation and Path Following," in *Acta Numerica*. Cambridge, UK: Cambridge University Press, 1993, pp. 1–64.
- [7] H. Kuhn and A. Tucker, "Nonlinear Programming," in *2nd Berkeley Symposium*, Berkeley, CA, 1951, pp. 481–492.
- [8] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, UK: Cambridge University Press, 2004.
- [9] I. Fiotou, P. Rostalski, B. Sturmfels, and M. Morari, "An Algebraic Geometry Approach to Nonlinear Parametric Optimization in Control," in *American Control Conference*, Minneapolis, MN, June 2006, pp. 3618–3623.
- [10] A. Beccuti, G. Papafotiou, and M. Morari, "Optimal Control of the Buck Dc-Dc Converter Operating Both in the Continuous and Discontinuous Conduction Modes," in *IEEE Conference on Decision and Control*, San Diego, CA, December 2006.
- [11] —, "Explicit Model Predictive Control of the Boost Converter," in *IFAC Conference on Analysis and Design of Hybrid Systems*, Alghero, Italy, June 2006.
- [12] A. Beccuti, G. Papafotiou, R. Frasca, and M. Morari, "Explicit Hybrid Model Predictive Control of the Dc-Dc Boost Converter," in *IEEE Power Electronics Specialists Conference*, Orlando, FL, June 2007.
- [13] D. Bates, J. Hauenstein, A. Sommese, and C. Wampler, "Bertini: Software for Numerical Algebraic Geometry," Available at <http://www.nd.edu/~sommese/bertini>.