Model Validation and Formal Verification of PWM DC-DC Converters

Omar Ali Beg, *Student Member, IEEE,* Houssam Abbas, *Member, IEEE,* Taylor T. Johnson, *Member, IEEE,* and Ali Davoudi, *Senior Member, IEEE*

***Abstract*—This paper presents hybrid automaton modeling, comparative model validation, and formal verification of stability through reachability analysis of PWM DC-DC converters. Conformance degree provides a measure of closeness between the proposed hybrid automaton models and experimental data. Nondeterminism due to variations in circuit parameters is modeled using interval matrices. In direct contrast to the unsound and computationally-intensive Monte Carlo simulation, reachability analysis are introduced to overapproximate the set of reachable states and ensure stable operation of PWM DC-DC converters. Using a 200 W experimental prototype of a buck converter, hybrid automaton models of open-loop and hysteresis-controlled converters are first validated against experimental data using their conformance degrees. Next, converter stability is formally verified through reachability analysis, and informally validated using Monte Carlo simulations and experimental results.**

***Index Terms*—DC-DC converter, formal verification, hybrid automaton, model validation, reachability analysis.**

I. Introduction

MODELING and control of PWM DC-DC converters require building an abstract model that reasonably matches the experimental data obtained from a prototype, and ensuring converter's proper operation despite parametric uncertainty. Conventional analysis techniques involve simulation-based Monte Carlo paradigms [1]–[5]. However, considering all possible parameter variations and initial conditions is computationally prohibitive. The boundaries of state trajectories can be found from average-value models [6]–[9]. We use rigorous model validation paradigms [10] by employing the *conformance degree* to quantify the closeness between the abstract model waveforms and experimental data [11]. Stable converter operation is *formally verified* using the reachability analysis. It overapproximates the set of all possible reachable states (i.e., the reach sets) from a given set of initial states and parameter values. One can then confidently ascertain stable converter operation if the reach sets remain within a desired region of the state space for a given time span.

General reachability analysis tools include, but are not limited to, HyTech [12], [13], PHAVer [14], UPPAAL [15], HSolver [16], d/dt [17], Flow* [18], and SpaceEx [19]–[21]. To effectively use such model checking tools, hybrid automaton models of DC-DC converters are required [22]. Hybrid automaton modeling of DC-DC converters is presented by the authors in [23]–[25], and others in [26]–[29]. However, [26]–[28] do not consider component losses/variations and the discontinuous conduction mode (DCM), and do not perform the reachability analysis. PHAVer in [30] computes the reach sets for an open-loop boost converter, but does not include DCM or component losses. MATLAB/Ellipsoidal Toolbox is used in [31] for the reachability analysis of DC-DC converters. However, Ellipsoidal-based set computations suffer from the curse of dimensionality. SpaceEx (the successor of PHAVer) scales quite efficiently, and is used as the reachability analysis tool in this paper. The main contributions of this paper are:

- The hybrid automaton models for DC-DC converters are automatically generated, validated against Simulink/Stateflow, PLECS simulations, and hardware measurements, and verified using reachability analysis in SpaceEx. These models include component nonidealities and different operational modes.
- The conformance degree of the hybrid automaton models validates these against the experimental data, by providing a proximity measure between executions/behaviors of these two in both time and space.
- Non-determinism due to parametric variations is modeled using interval matrices, which results in a set-valued additive input term in the system dynamics.
- The reachability analysis achieves a fixed point where there are no other reach sets (i.e., the model output will remain within reach sets as $t \to \infty$). It is impossible to get such success through Monte Carlo analysis.

The remainder of this paper is organized as follows: hybrid automaton modeling is discussed in SECTION II. Application of conformance degree for model validation is discussed in SECTION III. SECTION IV uses interval analysis to model the non-determinism caused by the parameter variation. SpaceEx-based reachability analysis is discussed in SECTION V. SECTION VI validates the developed models against a 200 W buck converter prototype using the conformance degree, formally verifies the model properties using reachability analysis, and presents comparison with the Monte Carlo simulation. SECTION VII concludes the paper.

This work was supported in part by the National Science Foundation under the award number ECCS-1137354, in part by the U.S. Office of Naval Research under grant N00014-14-1-0718, in part by the Air Force Research Laboratory under contract number FA 8750-13-2-0115, and in part by the Air Force Office of Scientific Research's Summer Faculty Fellowship Program. Omar A. Beg, T. Johnson, and A. Davoudi are with the University of Texas, Arlington, TX 76019, USA. H. Abbas is with University of Pennsylvania, Philadelphia, USA. (e-mail: omar.beg@mavs.uta.edu; habbas@seas.upenn.edu; taylor.johnson@uta.edu; davoudi@uta.edu).

Manuscript received Month –, 20–; revised Month –, ——.

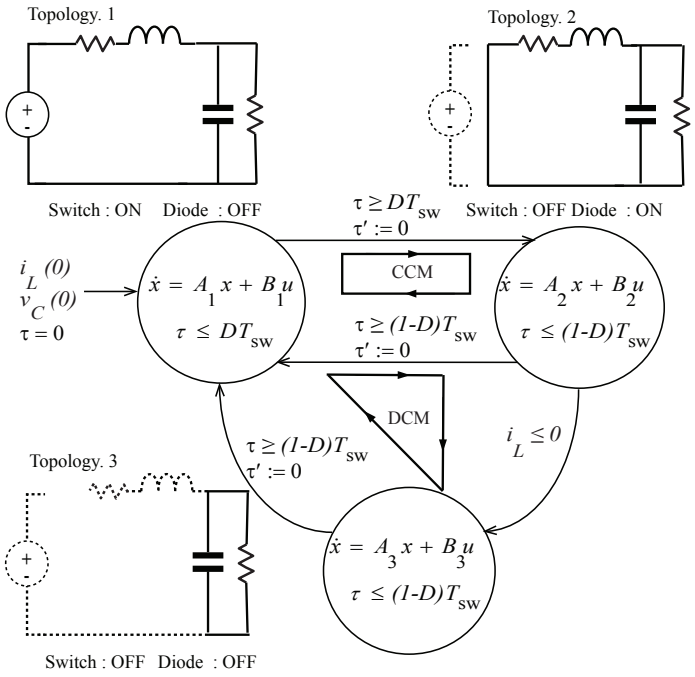

Fig. 1. Topologies, operational modes, and hybrid automaton modeling of a DC-DC buck converter.

II. Hybrid Automaton Modeling

A. Hybrid Automaton Model Syntax and Semantics

DC-DC converters exhibit both continuous and discrete behaviors due to the presence of passive elements and switching components, respectively. Hybrid automaton modeling [32], [33] integrates resulting differential equations and finite state machines in a single formalism. The state of a hybrid automaton model may change in two ways,i.e., through a continuous flow trajectory within a given topology, and through a discrete transition between two given topologies. A *topology* is defined as the circuit configuration in each switching sub-interval.

Definition 2.1: A *hybrid automaton model* is defined by a tuple $\mathcal{H} = \langle Q, X, init, U, inv, E, G, F, g, h \rangle$, where

- $Q = \{q_1, q_2, ..., q_N\}$ is a finite set of topologies.
- $X \subseteq \mathbb{R}^n$ represents continuous state variables, with x_i' being the value of the i^{th} state at the end of a transition.
- $init \subset Q_0 \times X_0$ is a set of initial conditions, such that $Q_0 \subseteq Q$ and $X_0 \subseteq X$.
- $U = \{u_1, u_2, ...u_N\}$ forms the input for each topology.
- $inv : Q \rightarrow 2^X$ is a mapping that assigns an invariant $inv\,(q_i) \subseteq X$ for each topology in Q. 2^X denotes the power set, i.e., the set of all subsets of X. An invariant $inv\,(q_i) \subseteq X$ is a property of the hybrid automaton model that must be satisfied by all reach sets for a given topology q_i. Once an invariant is violated, the real time τ is stopped, forcing the continuous state to stop evolving within a topology. Here, invariants are defined in the form of bounds for a continuous state variable (Fig. 2).
- $E \subset Q \times Q$ is a set of feasible discrete transitions allowed in the hybrid automaton model. It might not be possible to visit the entire set of topologies from one particular topology.

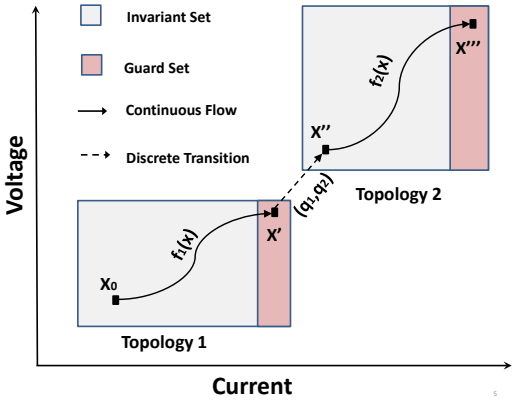

Fig. 2. Execution of the hybrid automaton model of DC-DC converters.

- G is a set of guard conditions for each element in E.
- The flow function $F : Q \times X \times U \rightarrow X \subseteq \mathbb{R}^n$ assigns a Lipschitz continuous vector space for the time derivative of x defined by the solution of the differential inclusion $\dot{x}_i \in F\,(q_i, x_i, u_i)$ in the i^{th} topology.
- $g : E \rightarrow G$ is called the guard function that maps each element in E to its corresponding guard in G. It ensures transitions to an appropriate topology, once the guard condition is reached, as shown in Fig. 2.
- $h : E \times X \rightarrow X$ resets the continuous state, i.e., if the transition from one topology to another takes place as defined by the set E with a final state in the set X, the continuous state has to be reset with a new value $x' = h(e, x(t))$ in X.

Definition 2.2: An *execution* of a hybrid automaton model $\mathcal{H}$ is an alternating sequence of continuous flow trajectories and discrete transitions, denoted by $X_0 \xrightarrow[\dot{x}=f_1(q_1,x_1,u_1)]{flow} X' \xrightarrow[(q_1,q_2)]{\rho} X'' \xrightarrow[\dot{x}=f_2(q_2,x_2,u_2)]{flow} X''' \ldots$, as shown in Fig. 2.

Definition 2.3: The *continuous flow trajectory* for a hybrid automaton model $\mathcal{H}$ is defined as: for a given $(q_i, x_i) \in init$ and $u_i \in U$, there is a flow function $\dot{x}_i = f(q_i, x_i, u_i)$ that results in a final continuous state x_i', whereas q_i remains unchanged, iff $inv(x_i)$ is true and the guard $g_i \in G$ is not violated, such that $x_i \xrightarrow[\dot{x}=f_1(q_1,x_1,u_1)]{flow} x'$.

Continuous flow trajectories evolve as the real time τ elapses. At each topology, converter dynamics can be modeled by ordinary differential equations (ODE); e.g., system matrices A_q and B_q describe the continuous flow trajectories in topology $q \in \{1, 2, 3\}$ of Fig. 1.

Definition 2.4: The *discrete transition* for a hybrid automaton model $\mathcal{H}$ is defined as: for a given $(q_i, x_i) \in init$ and $u_i \in U$, there is a function $\rho = h(e_{ij}, x_i)$ that resets the continuous state to x_i', and the topology to q_j, iff $inv(x_i)$ and the guard $g_i \in G$ are both true, and $\exists\ e_{ij} \in E$, such that the transition ρ is denoted by $x_i \xrightarrow[(q_i,q_j)]{\rho} x_i'$.

The switching instance can be determined either externally (e.g., by a duty cycle command for the MOSFET) or internally (e.g., by meeting appropriate threshold conditions for the

diode). The sequence of topologies, observed periodically in the steady state, defines an operational mode. Example of three topologies and two operational modes for a buck converter are shown in Fig. 1.

B. Model Instantiation for DC-DC Converters

We define D as the duty cycle, T_{sw} as the switching period, V_{in} as the DC input voltage, and V_{ref} as the reference voltage. We can represent the continuous dynamics for a given topology as a standard set of state-space equations

$$\frac{dx}{dt} = A_q x + B_q u \tag{1}$$

where, $x \in \mathbb{R}^n$ is a vector of continuous states, Q is a finite set of topologies, $u \subseteq U$ such that $U \subseteq \mathbb{R}^m$ is a set of input vectors, and $A_q \in \mathbb{R}^{n\times n}$ and $B_q \in \mathbb{R}^{n\times m}$ are system matrices. Such formation can be readily created for the buck converter in Fig. 1, as given in the APPENDIX. The instantiation of the hybrid automation model for an open-loop DC-DC converter, as per Definition 2.1 and Definition 2.2, is:

- Three topologies are denoted by $Q = \{q_1, q_2, q_3\}$.
- The continuous state vector is $x = [i_L \;\; v_C \;\; \tau]'$. τ represents real time such that $\frac{d\tau}{dt} = 1$.
- $U = \{[V_{in}, 0, 0]', [0, 0, 0]', [0, 0, 0]'\}$ forms the input vector set.
- $E = \{(q_1, q_2), (q_2, q_1), (q_2, q_3), (q_3, q_1)\}$ defines the feasible discrete transitions, e.g., (q_2, q_3) means a discrete transition from topology 2 to 3 is allowed.
- The continuous flow trajectory is computed using (1), with the corresponding state matrices for each topology. For topology 1, this can be denoted by $X_0 \xrightarrow[\dot{x}=f_1(q_1,x_1,u_1)]{flow} X'$, as shown in Fig. 2. X_0 is the initial and X' is the final set of states as the automaton continuously evolves with the continuous flow dynamics $f_1(q_1, x_1, u_1)$.
- Guard conditions, for elements of E, are defined by $G = \{(\tau \geq DT_{sw}), (\tau \geq (1-D)T_{sw}), (i_L \leq 0), (\tau \geq (1-D)T_{sw})\}$.
- The reset function h defines a new continuous state x'' for the new topology. For example, if a transition is to take place from topology 1 to topology 2 with some final state $x' \in X'$ in topology 1, h assigns the new state $x'' \in X''$ in topology 2. For topology 1 to topology 2, a transition ρ is denoted by $X' \xrightarrow[(q_1,q_2)]{\rho} X''$, as shown in Fig. 2.

The evolution of the hybrid automaton model starts with initial conditions from set $init$, e.g., $(q_1, x_0) \in init$ for a given input $u_1 = [V_{in}, 0, 0]'$ and, subsequently, the continuous state evolves according to the flow function. The discrete state (i.e., topology) remains constant; i.e., $q(t) = q_1$, as x_i evolves inside the invariant $inv(q_1)$. Once the continuous state trajectory reaches the guard $G(q_1, q_2)$ corresponding to the edge $E(q_1, q_2)$, the topology may transition from q_1 to q_2, and the continuous state is reset with a new value x'' in the new invariant set $inv(q_2) \subset X$.

This hybrid automaton model can be extended to closed-loop DC-DC converters, e.g., hysteresis-controlled converters. The tuple remains the same except that the guards shall be defined in terms of switching boundaries. The hysteresis band is formed by defining an upper switching boundary, $V_{ref} + \delta$, and a lower switching boundary, $V_{ref} - \delta$, where V_{ref} is the desired output voltage, and δ is the tolerance level. Thus, $G = \{(v_C \geq V_{ref} + \delta), (v_C \leq V_{ref} - \delta), (i_L \leq 0), (v_C \leq V_{ref} - \delta)\}$.

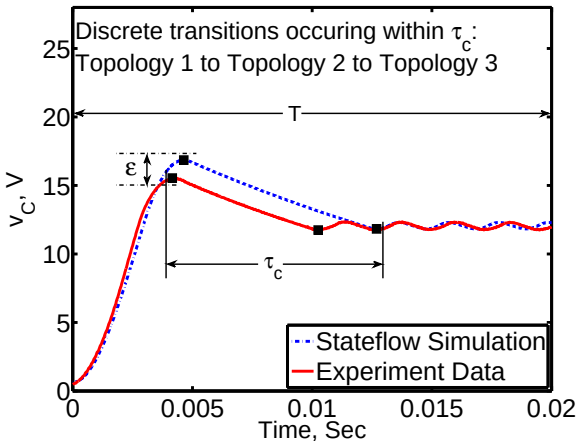

Fig. 3. Output trajectories of capacitor voltage for the closed-loop controlled buck converter - local mismatch for interval τ_c and corresponding ε.

It should be noted that time τ does not appear in the guard expressions. Therefore, we have developed two hybrid automaton models for the closed-loop buck DC-DC converter, i.e., one with variable τ (called the *time-dependent* hybrid automaton model), and another without variable τ (called the *time-independent* hybrid automaton model). For the time-independent hybrid automaton model, we perform the reachability analysis for an unbounded time, i.e., compute the reach sets as $t \to \infty$.

III. Validation through Conformance Degree

Model validation of DC-DC converters requires comparing output trajectories (or simulation traces) for a given model referred to as $\mathcal{M}$, and the measured data from an experimental prototype referred to as $\mathcal{I}$. The goal is to find an appropriate measure of distance for output trajectories of hybrid automata.

Definition 3.1: The *behavior* $\mathcal{B}_{\mathcal{H}}$ of the hybrid automaton model $\mathcal{H}$ with initial state (q_0, x_0) under the influence of the input u for the given time horizon T is defined by the *output trajectory* $y_{\mathcal{H}}((q_0, x_0), u, T)$, where, $q_0 \in Q_0$, $x_0 \in X_0$, and $u \in U$.

One can consider the output trajectories of the capacitor voltage (v_C) for a closed-loop buck converter shown in Fig. 3. The experimental data obtained from a prototype and output trajectory of the hybrid automaton model in Simulink/Stateflow are overlaid. Intuitively, the two output trajectories look similar, however, the sup norm would give a very large value to the distance between them. This is, partly, because $\mathcal{I}$ and $\mathcal{M}$ might transition among various topologies at slightly different moments in time. Therefore, our distance measure should allow some *wiggle room* in time; Rather than comparing only the states that are exactly time-aligned, it should allow comparison of states that are within some $\tau_c > 0$ time units of each other.

Moreover, it is not appropriate to compare outputs when two systems are in different topologies. Thus, our distance measure must only compare states after an equal number of discrete transitions between topologies of the two systems. Note that within the time window τ_c in Fig. 3, both the hardware prototype as well as the Stateflow model exhibit two discrete transitions between topologies. To this end, we introduce the parameter $j \in \mathbb{N}$, that counts the number of discrete transitions each system makes. It is reasonable to require that the transition times of the two systems be close to consider that the systems themselves are close: the value τ_c will also bound the difference in transition times. The distance measure will account for the distance between output trajectories, captured by the value $\varepsilon > 0$. Thus, we have a 2-value distance measure, with values τ_c and ε capturing the time and space distance between the two output trajectories. These are illustrated in Fig. 3.

The output trajectories of hybrid automaton models are parameterized with t and j. $t \in \mathbb{R}_+$ is the time spent in a given converter topology, and $j \in \mathbb{N}$ counts the number of discrete transitions between different topologies. We write $y_1(t; j)$ for the output trajectory at the hybrid time $(t; j) \in \mathbb{R}_+ \times \mathbb{N}$, i.e., at time t and after j transitions. Let $dom y_1 \subset \mathbb{R}_+ \times \mathbb{N}$ denote the domain of output trajectory y_1, i.e., the set of all $(t; j)$, so that $(T, J, \tau_c, \varepsilon)$-closeness [11] can be formally defined.

Definition 3.2: Take an output trajectory duration $T \in \mathbb{R}_+$, a maximum number of discrete transitions $J \in \mathbb{N}$, and parameters $\tau_c, \varepsilon > 0$. Two output trajectories y_1 and y_2 are $(T, J, \tau_c, \varepsilon)$-close, shown as $y_1 \approx_{(\tau_c,\varepsilon)} y_2$, if (a) for all $(t, j) \in \text{dom} y_1$ such that $t \leq T, j \leq J$, there exists $(s, j) \in \text{dom} y_2$ where $|t - s| \leq \tau_c$, and $\|y_1(t, j) - y_2(s, j)\| \leq \varepsilon$, and (b) for all $(s, j) \in \text{dom} y_2$ such that $s \leq T, j \leq J$, there exists $(t, j) \in \text{dom} y_1$ where $|t - s| \leq \tau_c$, and $\|y_2(s, j) - y_1(t, j)\| \leq \varepsilon$.

$(T, J, \tau_c, \varepsilon)$-closeness gives a proximity measure between the two output trajectories in both time and space. It shows that for every point $y_1(t, j)$, y_2 has a point ε-close to it, which may occur anywhere in the window $[t - \tau_c, t + \tau_c]$ (and vice versa). Allowing this wiggle room in time is important when comparing the output trajectories, because the discrete transitions could occur at different times. The two values T and J limit our testing horizon. $(T, J, \tau_c, \varepsilon)$-closeness can be lifted from output trajectories to systems. One can validate the model through the conformance degree between its output trajectory and measured data.

Definition 3.3: Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be two hybrid automata. The *conformance degree* of $\mathcal{H}_1$ to $\mathcal{H}_2$, given τ_c, is defined as the smallest ε such that for every trajectory y_1 of $\mathcal{H}_1$, there exists a trajectory y_2 of $\mathcal{H}_2$, where $y_1 \approx_{(\tau_c,\varepsilon)} y_2$. We denote this conformance degree by $\mathbf{CD}_\tau(\mathcal{H}_1, \mathcal{H}_2)$.

We will use this definition intuitively for model validation of DC-DC converters. We compute the conformance degree $\mathbf{CD}_\tau(\mathcal{H}_1, \mathcal{H}_2)$ for some $\tau_c > 0$ in different case studies of SECTION VI, and effectively say that some local mismatch is permissible within a window τ_c for the output trajectories of the models and the hardware prototype.

IV. Modeling Non-Determinism using Interval Analysis

The system matrices in the hybrid automaton models of DC-DC converters depend on component values. The variations due to manufacturing tolerance, aging, and temperature result in non-determinism of component values. We use the interval arithmetic [34] to incorporate the parameter variations within the reachability analysis framework. The range of component values are represented in terms of intervals. A real interval v is a set of real numbers given by

$$[\underline{v}, \overline{v}] = \{v \in \mathbb{R} : \underline{v} \leq v \leq \overline{v}\}, \tag{2}$$

where $\underline{v}$ is the infimum and $\overline{v}$ is the supremum. These intervals may also be defined by the midpoint-radius representation

$$mid(v) = \frac{1}{2}(\underline{v} + \overline{v}), \tag{3}$$

$$rad(v) = \frac{1}{2}(\overline{v} - \underline{v}). \tag{4}$$

The interval matrix for the system matrix is $\mathcal{A} = [\underline{A}, \overline{A}]$. System stability can be deferred by examining matrix extrema, i.e., $\underline{A}$ and $\overline{A}$ [35]. Therefore, it is sufficient to consider every combination of matrix extrema to overapproximate the reach set. The overapproximation of an interval matrix $\mathcal{A}$ is given by splitting it into two parts, i.e., a nominal part and a symmetric part [36]. For the i^{th} state variable, one has

$$\dot{x}_i = a_{i1}x_1 + a_{i2}x_2 + ... + a_{ij}x_j + ... + a_{in}x_n. \tag{5}$$

To incorporate parameter variation, one can replace the above coefficients with intervals

$$\begin{aligned}\dot{x}_i \in\ & [mid(a_{i1}) \pm rad(a_{i1})]x_1 + ...[mid(a_{ij}) \pm rad(a_{ij})]x_j \\ & + ... + [mid(a_{in}) \pm rad(a_{in})]x_n.\end{aligned} \tag{6}$$

The mid-points are constant terms, which can be separated

$$\dot{x}_i \in\ a_{i1}x_1 + r_{i1} + ...a_{ij}x_j + r_{ij}... + a_{in}x_{in} + r_{in}. \tag{7}$$

The radii $r_{i1}, r_{i2}, ..., r_{ij}, ..., r_{in}$ are given by

$$r_{ij} \in [-rad(a_{ij}), rad(a_{ij})]x_j, \tag{8}$$

which are used to define the invariants for the hybrid automaton model, i.e.,

$$-[-rad(a_{ij}), rad(a_{ij})]x_j \leq r_{ij} \leq [-rad(a_{ij}), rad(a_{ij})]x_j. \tag{9}$$

These invariants are defined for each topology of the DC-DC converter. As seen in (9), the state variable x_j is also included in the invariants.

V. Reachability Analysis for Hybrid Automata

Reachability analysis can be used for the formal verification of converter properties, e.g., stability in the sense of Lyapunov, i.e., $\dot{x} = f(x(t))$ is stable if $\forall\ \theta > 0\ , \exists\ \beta > 0$ such that if $\|x(0)\| \leq \beta \Rightarrow \|x(t)\| \leq \theta\ \forall\ t \geq 0$. We may define a bounded region and verify that the output of the hybrid automaton model eventually reaches, and always remains, in this stable region, as seen in Fig. 4. We define the stability

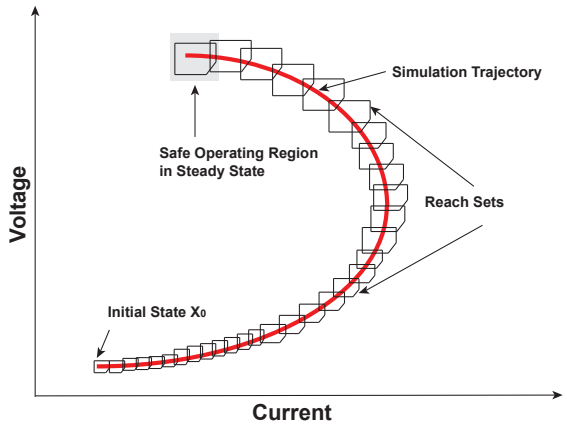

Fig. 4. Reachability analysis using reach sets for formal verification of a hybrid automaton model.

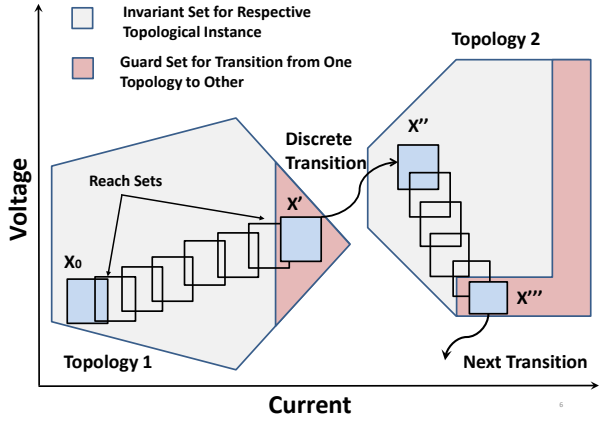

Fig. 5. Reach set in different topologies with transitions imposed by guards.

specification such that from the settling time t_s, the output voltage $V_C(t)$ should remain bounded within a tolerance γ of the reference voltage $V_{ref}(t)$, i.e., for $t \geq t_s \Rightarrow V_C(t) = V_{ref}(t) \pm \gamma$.

Definition 5.1: State x is *reachable* iff $\exists$ an execution α such that $x \in \alpha$.

The *set of reachable states* contains all the states that can be reached from a given set of initial conditions for a given time. Consider an example of an autonomous system $\dot{x} = Ax$. The set of states from initial time t_0 to final time t_f, reached from a given initial set X_0, is the union of the reachable states

$$\mathcal{R}_{t_0}^{t_f}(X_0) = \bigcup_{t \in [t_0, t_f]} e^{At} X_0. \tag{10}$$

However, (10) does not cater to the discrete transitions associated with the hybrid dynamical systems. Additionally, the exact set of all reachable states is undecidable. In practice, overapproximations of the reachable states are computed using geometrical data structures (e.g, boxes, polytopes, ellipsoids, or zonotopes [37]), called the overapproximated reach sets and denoted by $\overline{\mathcal{R}}$. For simplicity, we call these as the reach sets in this paper. This framework can be extended to hybrid dynamical systems by including invariants and guard sets (Fig. 5), and implemented in various reachability analysis tools by software research community as mentioned in SECTION I. The reach sets for continuous dynamics can be computed using continuous post-operators so long as the continuous dynamics of DC-DC converter are contained within the invariant set defined for the corresponding topology or do not enter the guard set. Once the guard condition is satisfied within an invariant, a transition takes place from topology 1 to topology 2 such that the next reach set is computed using discrete post-operator. This process goes on until either the final time in a local time horizon, or a fixed point, is reached. A *fixed point* signifies that the reachability algorithm cannot find any new reach set during the current iteration other than those computed in the previous iteration.

SpaceEx reachability tool computes the reach sets of a hybrid dynamical system. It is a classical fixed point algorithm based on computation of symbolic states [19], [20]. A *symbolic state* is defined as a pair (l, Ω), where l is a topological instance, and Ω is the corresponding convex continuous set. The reach set $\overline{\mathcal{R}}$ is obtained by computing the set of symbolic states. This reach set is the fixed point of the sequence $\overline{\mathcal{R}}_o = post_c(Init)$, and the successors are computed using

$$\overline{\mathcal{R}}_{k+1} := \overline{\mathcal{R}}_k \bigcup post_c\left(post_d\left(\overline{\mathcal{R}}_k\right)\right) \tag{11}$$

where, $post_d$ is the *discrete post-operator* that defines the reach sets by a discrete transition from $\overline{\mathcal{R}}$. This corresponds to the h function defined in Definition 2.1. $post_c$ is the *continuous post-operator* that defines the reach sets from $\overline{\mathcal{R}}$ after an arbitrary amount of time is elapsed. This corresponds to the flow function in Definition 2.1.

Computation of the reachability post-operators for Ω is challenging, so each Ω is represented by its corresponding support function to facilitate various set operations such as linear mapping, Minkowski sum, and convex hull. A *support function* is an exact representation of a given Ω. An approximated computation of Ω_k is given in [20] for the k^{th} time step. Hence, a sequence of convex continuous sets $\Omega_0, \Omega_1, ...\Omega_{N-1}$ is computed to form a *flowpipe* that covers the reach sets up to a pre-defined time such that N represents the number of time steps. This flowpipe is then used to compute the transition successors. Only those states can take the transition that satisfy the guard associated with the present topology and the invariant of the target topology. This process is continued until a fixed point is reached, i.e., if all the reach sets that are computed in the present iteration, are contained in reach sets computed in the previous iteration, i.e., $\overline{\mathcal{R}}_{k+1} \subseteq \overline{\mathcal{R}}_k$. This signifies that no new reach sets could be found and the computation process may be terminated. Interested readers may see [20] for further implementation detail.

SpaceEx is a development platform with various verification algorithms (called scenarios). Three scenarios are available in SpaceEx v0.9.8d; i.e., PHAVer (Polyhedral Hybrid Automaton Verifyer), LGG (Le Guernic-Girard) algorithm wherein the reach set is overapproximated by a set of polyhyedra, and STC algorithm (an enhancement of LGG with automatic clustering). The version of LGG implemented in SpaceEx uses outer polyhedral approximations to compute the image of discrete transitions, making it scalable. STC algorithm produces fewer

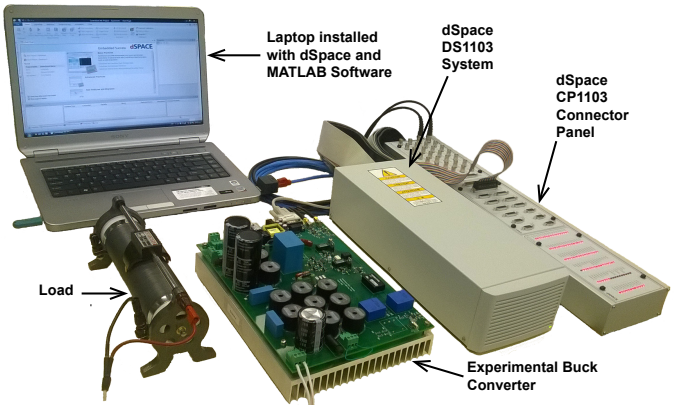

Fig. 6. Buck converter prototype controlled with a dSPACE DS1103 system.

convex continuous sets for a given accuracy, and computes more precise images of discrete transitions. Based on the LGG scenario, the flowpipes (i.e., the reach sets over time) are bounded with piecewise linear approximations of the support function over time. A comparison of both scenarios is given in [21].

VI. Case Studies

An experimental setup of a buck converter, controlled with a dSpace DS1103 unit, has been prototyped, as shown in Fig. 6. The experimental results are used for benchmarking purposes against MATLAB/PLECS [38], Simulink/Stateflow [39], and SpaceEx reachability analysis. Circuit parameters $L = 2.65$ mH, $C = 2.2$ mF, and $R = 10\ \Omega$ are used throughout this study. We have used the Hybrid Source Transformer (HyST) which is a source-to-source conversion tool for hybrid automaton models [40]. The hybrid automaton model is developed using the java interface in MATLAB, and transformed into a SpaceEx compatible model using HyST data structures. We have used the STC support function of SpaceEx v0.9.8d using an Intel Core i7 processor on a Windows 7 platform. We use the conformance degree to validate the hybrid automaton model against the experimental data. Then, the reachability analysis results are provided for an open-loop and a hysteresis-controlled buck converter.

A. Model Validation Using Conformance Degree Testing

We use notations $\mathcal{I}_O$ and $\mathcal{I}_C$ for hardware prototypes in open-loop and closed-loop configurations, respectively. PLECS and Stateflow models are denoted by $\mathcal{M}_{OP}$, $\mathcal{M}_{CP}$ and $\mathcal{M}_{OS}$, $\mathcal{M}_{CS}$, respectively, where subscript O denotes an open-loop and C denotes a closed-loop configuration. The computed ε values against τ_c (as defined in SECTION III) are tabulated in Table I for the corresponding output trajectories. It is evident from Table I that the ε values of $\mathcal{M}_{OP}$ and $\mathcal{M}_{OS}$ as well as $\mathcal{M}_{CP}$ and $\mathcal{M}_{CS}$ are close enough (also, as seen in Figs. 7, 9, and 10). We have also computed conformance degrees for the prototype buck converters, i.e., $\mathcal{I}_O$ and $\mathcal{I}_C$, in comparison with other models, i.e., $\mathcal{M}_{OP}$, $\mathcal{M}_{OS}$ and $\mathcal{M}_{CP}$, $\mathcal{M}_{CS}$. The values depicted in Table I provide enough wiggle room to validate that $\mathcal{M}_{OP}$ and $\mathcal{M}_{OS}$ are reasonable abstractions for $\mathcal{M}_O$, whereas $\mathcal{M}_{CP}$ and $\mathcal{M}_{CS}$ are reasonable abstractions for $\mathcal{I}_C$. Therefore, we have validated the hybrid automaton models against both the open-loop and the closed-loop converter prototypes.

TABLE I
Conformance Degree Analysis

Configuration	Type of Output Trajectories	τ_c Value	ε Value
Open loop	Current (i_L) - PLECS vs Experiment	2×10^{-3}	1.9117
	Current (i_L) - Stateflow vs Experiment	2×10^{-3}	1.9125
	Current (i_L) - Stateflow vs PLECS	2×10^{-3}	0.1785
	Voltage (v_C) - PLECS vs Experiment	8×10^{-3}	1.1231
	Voltage (v_C) - Stateflow vs Experiment	8×10^{-3}	1.1033
	Voltage (v_C) - Stateflow vs PLECS	8×10^{-3}	0.6666
Closed loop	Current (i_L) - PLECS vs Experiment	5×10^{-5}	3.0590
	Current (i_L) - Stateflow vs Experiment	5×10^{-5}	3.0590
	Current (i_L) - Stateflow vs PLECS	5×10^{-5}	0.0878
	Voltage (v_C) - PLECS vs Experiment	8×10^{-4}	1.3105
	Voltage (v_C) - Stateflow vs Experiment	8×10^{-4}	1.3105
	Voltage (v_C) - Stateflow vs PLECS	8×10^{-4}	0.0584

B. Formal Verification of the Open-loop Buck Converter

We consider the voltage stability specification to perform formal verification. For example, for $t_s = 0.025$ sec, and $V_{ref} = 48$ V, we define $\gamma = 7$ V. This results in an upper voltage bound of 55 V, and lower voltage bound of 41 V, as shown in Fig. 8(b) by dotted lines. The input parameters are $V_{in} = 100$ V, and $f_s = 60$ kHz. The output trajectories and phase-plane responses are considered for the startup transients of the open-loop buck converter. The converter models in PLECS, Simulink/Stateflow, and SpaceEx are verified, and an acceptable match is reported in Fig. 7. The parameters' variations have been modeled using interval analysis, and also included in the Monte Carlo simulation. The reachability analysis results, obtained using SpaceEx, are plotted in Fig. 8. It can be seen that the steady-state inductor current and capacitor voltage waveforms lie within the reachability analysis results, i.e., the simulations and measurement data are contained within the reach sets. Moreover, we verify that $v_C(t) \in [41, 55]$ for $t \geq t_s$ for Stateflow, PLECS, measurement data, Monte Carlo analysis, and SpaceEx analysis results.

C. Formal Verification of the Hysteresis-controlled Converter

We define the voltage stability specification for the closed-loop buck converter to perform formal verification. For $t_s = 0.012$ sec, and $V_{ref} = 12$ V, we define $\gamma = 3$ V. This leads to upper and lower voltage bounds of 15 and 9 V, respectively, as shown by dotted lines in Fig. 11(b). In this case study, the time-dependent and the time-independent models (as mentioned in SECTION II) are considered. First, SpaceEx reachability analysis is performed using both LGG and STC for the time-dependent model. The new parameters are $V_{in} = 24$ V, $V_{ref} = 12$ V, and $f_s = 50$ kHz. The trajectories are shown for

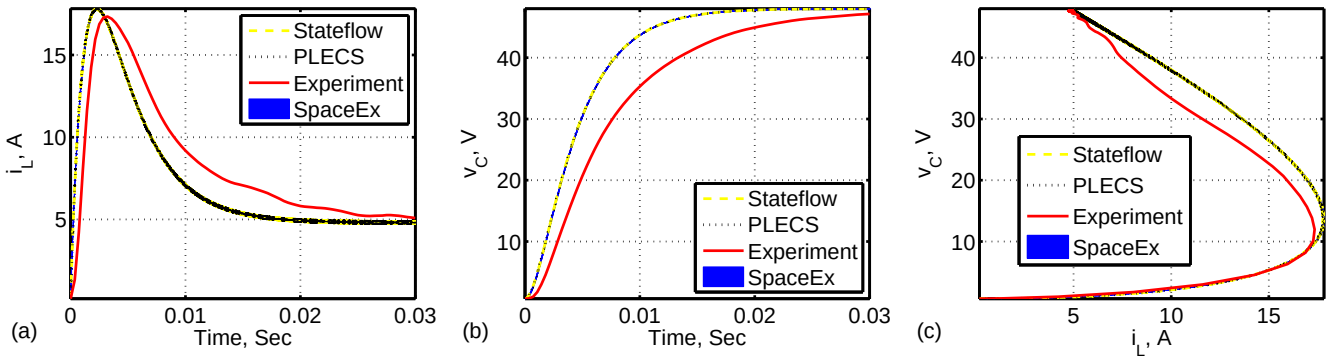

Fig. 7. Startup transients for an open-loop buck converter including Stateflow, PLECS, experiment, and SpaceEx; (a) current vs. time, (b) voltage vs. time, and (c) phase portrait.

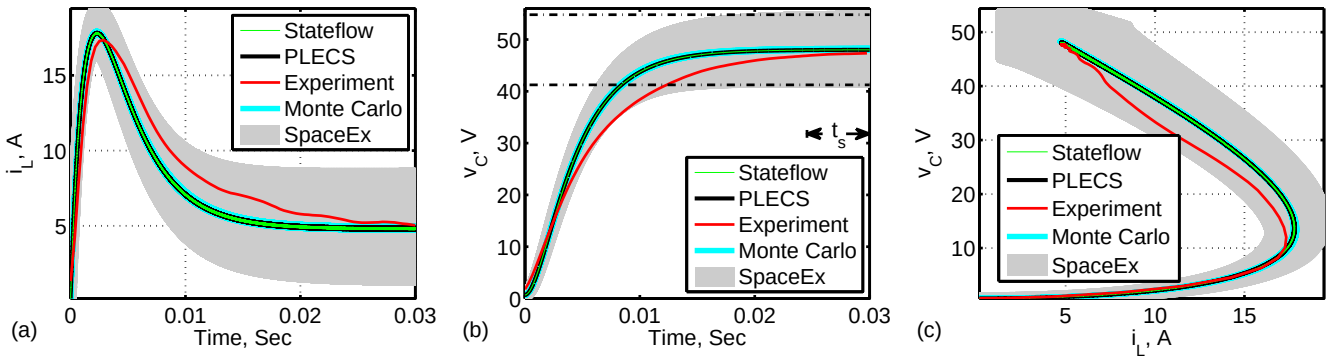

Fig. 8. Startup transients for an open-loop buck converter using interval matrices and the Monte Carlo simulation including Stateflow, PLECS, experiment, and SpaceEx; (a) current vs. time, (b) voltage vs. time, and (c) phase portrait.

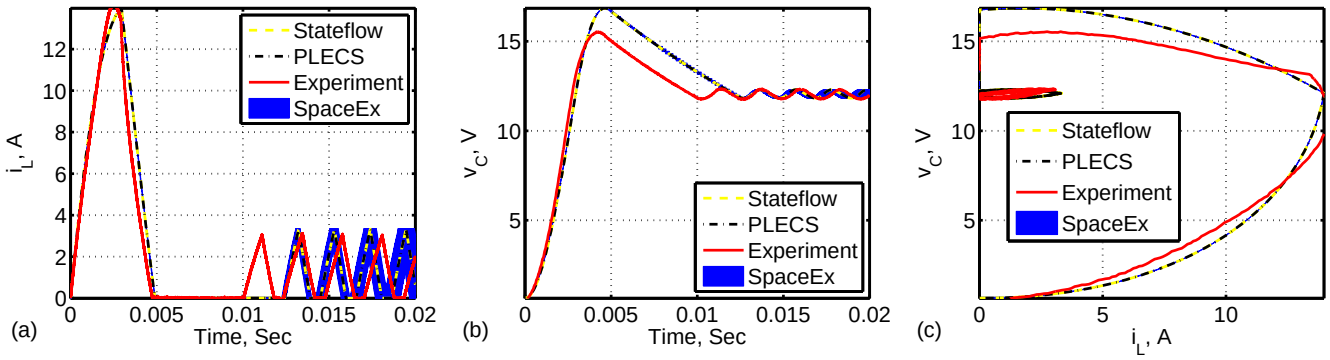

Fig. 9. Time-dependent hysteresis-controlled buck converter: Stateflow, PLECS, experiment, and SpaceEx LGG results using deterministic models; (a) current vs. time, (b) voltage vs. time, and (c) phase portrait.

Stateflow, PLECS, and experimental data along with reach sets computed using SpaceEx LGG and STC scenarios in Fig. 9 and Fig. 10, respectively. The Stateflow, PLECS, and SpaceEx results match right from the start until the steady state is reached. Experimental results match that of Stateflow, PLECS, and SpaceEx in the steady state. Next, the non-determinism due to the parameter variations is modeled using the interval matrices. It can be observed in Fig. 11 that Stateflow, PLECS, and measured results remain within the reach sets computed using SpaceEx, $v_C(t) \in [9, 15]$ for $t \geq t_s$.

We can formally verify the time-independent SpaceEx model for an unbounded time, i.e., $t \to \infty$, by excluding τ. This would not be possible through Monte Carlo analysis as, even for a limited time span, one has to take into account infinite number of possible combinations. We have successfully achieved a fixed point using SpaceEx LGG scenario, with unbounded time, and with all possible parameter variations. The phase-plane plots are given for the start-up transients in Fig. 12. As seen, all results remain within the computed reach sets as $t \to \infty$, verifying $v_C(t) \in [9, 15]$ as $t \to \infty$.

A comparison of Monte Carlo analysis and SpaceEx reachability analysis, in term of computation times, is shown in Table II. Both are run on a Windows 7 SP1 (64 bit) platform, with Intel (R) core i7-2600 CPU with 3.40 GHz,

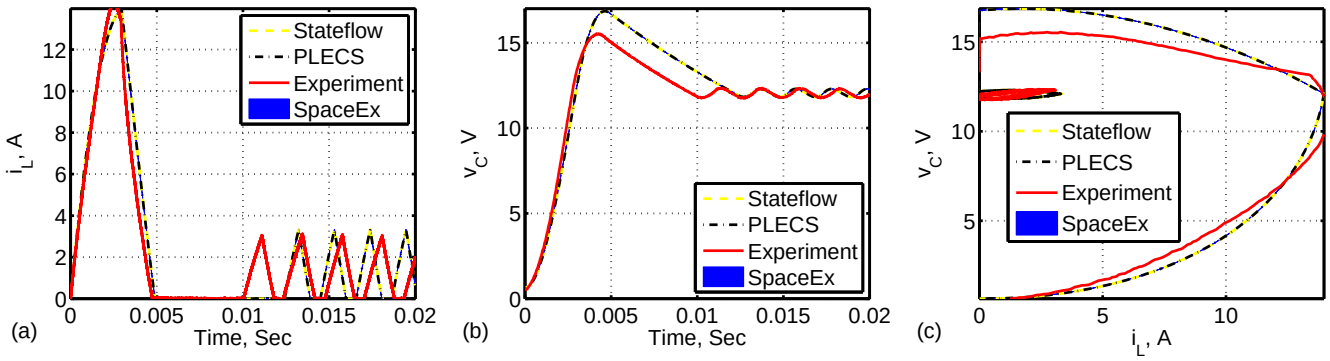

Fig. 10. Time-dependent hysteresis-controlled buck converter: Stateflow, PLECS, experiment, and SpaceEx STC results using deterministic models; (a) current vs. time, (b) voltage vs. time, and (c) phase portrait.

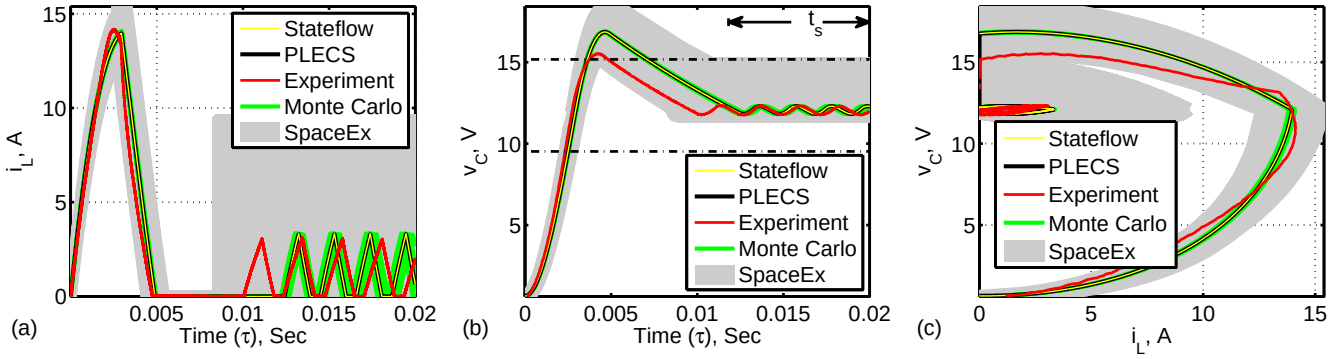

Fig. 11. Time-dependent hysteresis-controlled converter analysis using interval matrices including Stateflow, PLECS, experiment, Monte Carlo, and SpaceEx; (a) current vs. time, (b) voltage vs. time, and (c) phase portrait.

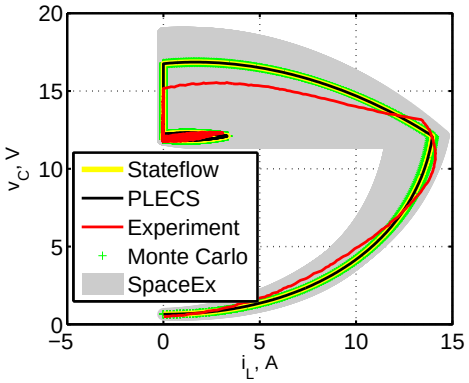

Fig. 12. Time-independent hysteresis-controlled converter analysis using interval matrices including stateflow, PLECS, experiment, Monte Carlo, and SpaceEx.

16.0 GB RAM processor, MATLAB version 8.5.0.197613 (R2015a), PLECS version 3.7.3, and SpaceEx version 0.9.8d. While infinite iterations are required to have full confidence in model validation through Monte Carlo analysis, we have only used finite (i.e., 2000) iterations as would be done in practice. Even then, it is evident that the SpaceEx reachability outperforms the Monte Carlo analysis in computation time, as seen in Table II.

VII. Conclusion

A hybrid automaton modeling approach for PWM DC-DC converters is developed. We have used the conformance testing for model validation when compared with a hardware prototype of DC-DC converters. The interval matrices analysis accommodates the model non-determinism caused by variations in component values. Reachability analysis frameworks are developed for formal verification of the resulting hybrid automaton models. It is shown that the proposed reachability analysis outperforms the brute force Monte Carlo analysis in computation time and confidence level.

Acknowledgment

The authors would like to thank Luan V. Nguyen and Vahidreza Nasirian for their help.

Appendix

The state-space matrices for circuit topology 1, 2, and 3 are:

$$A_1 = \begin{bmatrix} \frac{-(r_L + r_S)}{L} & \frac{-1}{L} & 0 \\ \frac{1}{C} & \frac{-1}{RC} & 0 \\ 0 & 0 & \frac{1}{\tau} \end{bmatrix}, B_1 = \begin{bmatrix} \frac{1}{L} \\ 0 \\ 0 \end{bmatrix}, \qquad (12)$$

TABLE II
COMPARISON OF MONTE CARLO AND SPACEEX ANALYSIS

System Configuration	Monte Carlo Iterations	Monte Carlo Time (sec)	SpaceEx Time (sec)	Times SpaceEx is Faster
Open Loop	2000	1.0151×10^4	1701.43	5.9662
Hysteresis control (time-dependent)	1000	315.43	137.65	2.2915
Hysteresis control (time-independent)	1000	315.43	230.57	1.3680
Hysteresis control, steady state (time-independent)	2000	1327	229.03	5.794

$$A_2 = \begin{bmatrix} \frac{-r_L}{L} & \frac{-1}{L} & 0 \\ \frac{1}{C} & \frac{-1}{RC} & 0 \\ 0 & 0 & \frac{1}{\tau} \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (13)$$

and

$$A_3 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{-1}{RC} & 0 \\ 0 & 0 & \frac{1}{\tau} \end{bmatrix}, B_3 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad (14)$$

respectively.

REFERENCES

[1] W. Huang *et al.*, "System accuracy analysis of the multiphase voltage regulator module," *IEEE Trans. Power Electron.*, vol. 22, no. 3, pp. 1019–1026, May 2007.

[2] M. Li *et al.*, "Design verification and testing of power supply system by using virtual prototype," *IEEE Trans. Power Electron.*, vol. 18, no. 3, pp. 733–739, May 2003.

[3] M. del Casale *et al.*, "Selection of optimal closed-loop controllers for dc-dc voltage regulators based on nominal and tolerance design," *IEEE Trans. Ind. Electron.*, vol. 51, no. 4, pp. 840–849, Aug 2004.

[4] T. Neugebauer and D. Perreault, "Computer-aided optimization of dc/dc converters for automotive applications," *IEEE Trans. Power Electron.*, vol. 18, no. 3, pp. 775–783, May 2003.

[5] D. Maksimovic *et al.*, "Modeling and simulation of power electronic converters," *Proc. IEEE*, vol. 89, no. 6, pp. 898–912, Jun. 2001.

[6] H. Behjati *et al.*, "Alternative time-invariant multi-frequency modeling of pwm dc-dc converters," *IEEE Trans. Circuits Syst. I: Regular Papers*, vol. 60, no. 11, pp. 3069–3079, Nov 2013.

[7] J. Kimball and P. Krein, "Singular perturbation theory for dc-dc converters and application to pfc converters," *IEEE Trans. Power Electron.*, vol. 23, no. 6, pp. 2970–2981, Nov 2008.

[8] Z. Mihajlovic *et al.*, "Output ripple analysis of switching dc-dc converters," *IEEE Trans. Circuits and Syst. I: Regular Papers*, vol. 51, no. 8, pp. 1596–1611, Aug 2004.

[9] B. Lehman and R. M. Bass, "Extensions of averaging theory for power electronic systems," *IEEE Trans. Power Electron.*, vol. 11, no. 4, pp. 542–553, Jul 1996.

[10] L. Ljung, *System Identification: Theory for the User*, 2nd ed. New Jersey, USA: Prentice-Hall, Inc., 1999.

[11] H. Abbas *et al.*, "Formal property verification in a conformance testing framework," in *Proc. ACM-IEEE 12th Int. Conf. on Formal Methods and Models for Syst. Design*, Lausanne, 2014, pp. 155–164.

[12] R. Alur *et al.*, "Automatic symbolic verification of embedded systems," *IEEE Trans. Software Eng.*, vol. 22, no. 3, pp. 181–201, Mar. 1996.

[13] T. Henzinger *et al.*, "HyTech: A model checker for hybrid systems," in *Computer Aided Verification*, O. Grumberg, Ed. Berlin Heidelberg: Springer, Mar. 1997, pp. 460–463.

[14] G. Frehse, "PHAVer: Algorithmic verification of hybrid systems past HyTech," vol. 10, no. 3, Jun. 2008, pp. 263–279.

[15] J. Bengtsson *et al.*, "UPPAAL a tool suite for automatic verification of real-time systems," in *Hybrid Systems III*. Berlin Heidelberg: Springer, Jun. 2005, pp. 232–243.

[16] S. Ratschan and Z. She, "Safety verification of hybrid systems by constraint propagation based abstraction refinement," in *Hybrid Systems: Computation and Control*, M. Morari and L. Thiele, Eds. Berlin Heidelberg: Springer, Mar. 2005, pp. 573–589.

[17] E. Asarin *et al.*, "The d/dt tool for verification of hybrid systems," in *Computer Aided Verification*, E. Brinksma and K. G. Larsen, Eds. Berlin Heidelberg: Springer, Sep. 2002, pp. 365–370.

[18] X. Chen *et al.*, "Flow*: An analyzer for non-linear hybrid systems," in *Computer Aided Verification*, ser. Lecture Notes in Computer Science, N. Sharygina and H. Veith, Eds. Springer Berlin Heidelberg, 2013, vol. 8044, pp. 258–263.

[19] G. Frehse, "An introduction to SpaceEx v0.8," http://spaceex.imag.fr/documentation/user-documentation/introduction-spaceex-27, dec 2010.

[20] G. Frehse *et al.*, "SpaceEx: Scalable verification of hybrid systems," in *Proc. 23rd Int. Conf. on Comput. Aided Verification*, Snowbird, UT, 2011, pp. 379–395.

[21] G. Frehse, "A brief experimental comparison of the STC and LGG analysis algorithms in SpaceEx," http://spaceex.imag.fr/documentation/user-documentation/brief-experimental-comparison-stc-and-lgg-analysis-algorithms-spaces, Nov 2012.

[22] M. Miranda and A. Lima, "Formal verification and controller redesign of power electronic converters," in *Proc. IEEE Int. Symp. Indust. Electron.*, Ajaccio, France, 2004, pp. 907–912.

[23] L. Nguyen and T. Johnson, "Benchmark: Dc-to-dc switched-mode power converters (buck converters, boost converters, and buck-boost converters)," in *Applied Verification for Continuous and Hybrid Systems Workshop*, Berlin, Germany, 2014, pp. 19–24.

[24] T. Johnson *et al.*, "Design verification methods for switching power converters," in *Proc. 3rd Power and Energy Conf. at Illinois*, Urbana, IL, 2012, pp. 1–6.

[25] S. Hossain *et al.*, "Reachability analysis of closed-loop switching power converters," in *Proc. 4th Power and Energy Conf. at Illinois*, Urbana, IL, 2013, pp. 130–134.

[26] M. Hongbo and F. Quanyuan, "Hybrid modeling and control for buck-boost switching converters," in *Proc. Int. Conf. Commun., Circuits and Syst.*, Milpitas, CA, 2009, pp. 678–682.

[27] M. Senesky *et al.*, "Hybrid modelling and control of power electronics," in *Proc. 6th Int. Workshop on Hybrid Systems: Computation and Control*, Prague, Czech Republic, 2003, pp. 450–465.

[28] C. Sreekumar and V. Agarwal, "A hybrid control algorithm for voltage regulation in dcdc boost converter," *IEEE Trans. Ind. Electron.*, vol. 55, no. 6, pp. 2530 – 2538, Jun. 2008.

[29] Y. Quan *et al.*, "Simultaneous ccm and dcm operations of boost converter by a pwm hybrid control strategy," in *Proc. IEEE 39th Annual Conf. Ind. Electron. Society*, Vienna,Austria, 2013, pp. 1260–1265.

[30] U. Kuhne, "Analysis of a boost converter circuit using linear hybrid automata," ENS Cachan, Cedex, France, Tech. Rep., 2010.

[31] E. Hope *et al.*, "A reachability-based method for large-signal behavior verification of dc-dc converters," *IEEE Trans. Circuits Syst. I*, vol. 58, no. 12, pp. 2944–2955, Dec. 2011.

[32] O. Stursberg and B. Krogh, "Efficient representation and computation of reachable sets for hybrid systems," in *Hybrid Systems: Computation and Control*, O. Maler and A. Pnueli, Eds. Springer Berlin Heidelberg, 2003, vol. 2623, pp. 482–497.

[33] T. Henzinger, "The theory of hybrid automata," in *Proc. IEEE Symp. on Logic in Comput. Science*, New Brunswick, NJ, 1996, pp. 278–292.

[34] R. Moore *et al.*, *Introduction To Interval Analysis*. Cambridge Uni Press, 2009.

[35] J. Rohn, "Stability of interval matrices: the real eigenvalue case," *IEEE Trans. Autom. Control*, vol. 37, no. 10, pp. 1604–1605, Oct. 1992.

[36] M. Althoff *et al.*, "Analyzing reachability of linear dynamic systems with parametric uncertainties," in *Modeling, Design, and Simulation of Systems with Uncertainties*, A. Rauh and E. Auer, Eds. Berlin Heidelberg: Springer, May 2011, pp. 69–94.

[37] P. Hnsch *et al.*, "Reachability analysis of linear systems with stepwise constant inputs," *Electronic Notes in Theoretical Computer Science*, vol. 297, no. 0, pp. 61–74, Dec. 2013.
[38] *PLECS Manual Version 3.7*, Plexim Inc., Cambridge, MA, USA, 2015.
[39] *MATLAB Stateflow User's Guide*, Mathworks, MA, USA, 2015.
[40] S. Bak *et al.*, "HyST: A source transformation and translation tool for hybrid automaton models," in *Proc. ACM 18th Int. Conf. on Hybrid Syst.: Computation and Control*, Seattle, WA, 2015, pp. 128–133.

Omar Ali Beg (S'14) received the B.E. and M.S. degrees in electrical engineering from National University of Sciences and Technology, Pakistan. He is presently working toward his PhD degree at University of Texas at Arlington, TX, USA. He is recipient of the US Air Force Research Laboratory summer research fellowship 2015. His research interests include the modeling, reachability analysis and formal verification of software-controlled power electronics devices.

Houssam Abbas is a postdoctoral fellow in the Department of Electrical and Systems Engineering at the University of Pennsylvania with Professor Rahul Mangharam. Houssam holds a PhD in Electrical Engineering from Arizona State University. His research interests are in the verification, control and conformance testing of Cyber-Physical Systems, in particular hybrid systems. Current research includes the verification of medical devices, verification and control of autonomous vehicles, and anytime computation and control.

Taylor T Johnson is an Assistant Professor of Computer Science and Engineering at the University of Texas at Arlington. Dr. Johnson completed his PhD and MSc in Electrical and Computer Engineering at the University of Illinois at Urbana-Champaign in 2013 and 2010, respectively. He was a visiting graduate research assistant at the Air Force Research Laboratory (AFRL)s Space Vehicles Directorate at Kirtland Air Force Base in 2011, and was a visiting faculty researcher at the AFRLs Information Directorate in 2014. Dr. Johnson worked in industry for Schlumberger at various times between 2005 and 2010 helping develop new downhole embedded control systems. Dr. Johnsons research focus is developing algorithmic techniques and software tools to improve the reliability of cyber-physical systems. Dr. Johnson has published over twenty papers on these methods and their applications in areas like power and energy systems, aerospace, and robotics, two of which were recognized with best paper awards, from the IEEE and IFIP, respectively.

Ali Davoudi (S'04-M'11-SM'15) received his Ph.D. in Electrical and Computer Engineering from the University of Illinois, Urbana-Champaign, IL, USA, in 2010. He is currently an Assistant Professor in the Electrical Engineering Department, University of Texas, Arlington, TX, USA. He was with Solar Bridge Technologies, Champaign, IL; Texas Instruments Inc., Rochester, MN; and Royal Philips Electronics Rosemont, IL. His research interests include various aspects of modeling and control of power electronics and finite-inertia power systems. Dr. Davoudi is an Associate Editor for IEEE Transactions on Transportation Electrification and IEEE Transactions on Energy Conversion. He has received 2014 Ralph H. Lee Prize paper award from IEEE Transactions on Industry Applications, best paper award from 2015 IEEE International Symposium on Resilient Control Systems, and 2014-2015 best paper award from IEEE Transactions on Energy Conversion.