

Validation of a Boost Circuit Model Using Acceptance Sampling

Jon Zumberge
AFRL

John Mersch
P.C. Krause & Associates

ABSTRACT

Cost and performance requirements are driving military and commercial systems to highly integrated, optimized systems which require more sophisticated, highly complex controls. To realize benefits and make confident decisions, the validation of both plant and control models becomes critical. To quickly develop controls for these systems, it is beneficial to develop models and determine the uncertainty of those models so as to predict performance and stability. A process of model validation for a boost circuit based on acceptance sampling is presented here. The validation process described in this paper includes the steps of defining requirements, performing a screening and exploration of the system, completing a system and parameter identification, and finally executing a validation test. To minimize the cost of experimentation and simulation, design of experiments is used extensively to limit the amount of data taken without losing information. One key contribution in this paper is the use of tolerance intervals as an estimation of model accuracy.

CITATION: Zumberge, J. and Mersch, J., "Validation of a Boost Circuit Model Using Acceptance Sampling," *SAE Int. J. Aerosp.* 7(1):2014, doi:10.4271/2014-01-2104.

INTRODUCTION

With the increasing complexity of systems fielded by both military and commercial entities, there is an ever growing need to prove these systems will perform to specifications in a stable manner. To address these issues, there is ongoing work to include model based engineering earlier in the development cycle. One such program is the Air Force Research Laboratory's (AFRL) Integrated Vehicle Energy Technology (INVENT) program [1], [2], and [3]. To realize the benefits of model based engineering, there is a need for validation of models, control algorithms, and software used in these complex systems at a reasonable cost [4], [5].

The goal of this paper is to describe a novel model validation process. The four steps of the model validation process are requirements gathering, screening and exploration, system and/or parameter identification, and experimental validation. The novel techniques used include integrating and applying tools from statistics and system identification and offering evidence of validity outside of the data taken for non-normal modeling errors.

A dc-dc boost converter will be used to demonstrate the model validation process. Power converters are used throughout aircraft to help regulate voltage levels and have the potential to

greatly impact power quality. With advances in electrical, mechanical, and structural design onboard more electric aircraft, the electrical systems have undergone considerably higher level of stresses, thus reducing power quality [6], [7]. More robust controls that can mitigate the disturbances presented on the aircraft can enable the ability to improve power quality resulting in components with longer life expectancy. To test those control methods, valid models are needed.

One of the keys attributes to any experimental or simulation test is the amount of testing performed. Taking too many data points will increase cost and time, while taking too few points means important information may not be found. Through experimental design, an optimized approach to testing is taken [8],[9].

This paper begins with a background on the various techniques for performing model validation. After the background, the approach taken to obtain a validated model is discussed. This section also includes the necessary setup information so this analysis can be reproduced. In the results section a discussion of the findings and demonstration of the process with a dc-dc boost converter model is presented. Then, future work and conclusions are discussed.

BACKGROUND

In this section, a number of ideas from statistics, system identification, and boost circuit modeling are presented, to allow for use of these items within the validation process. At the end of the section an overview of existing methodologies for model validation is given.

Risk

The two types of risks that are used within the field of statistics are risks associated with Type I error, or colloquially called producer risk, and Type II error, or colloquially called consumer risk. In the case of modeling, Type I error occurs when the modeler ("producer") determines that a good model is not a good model. In this instance, more time is spent on improving a model that already meets the specified requirement. Throughout this paper it will be referred to as α . The second risk that needs specified is that associated with Type II error, or otherwise known as "consumer" risk. A Type II error occurs when a "bad" model is said to be good. The end user ("consumer") then uses the model assuming it is supplying answers within a certain risk level when the model is NOT performing to the level specified. This risk is associated with β . More formal discussions on Type I and Type II errors can be found in [9] or [8].

Acceptance Sampling

Acceptance sampling will be the guiding principal to determine whether the model meets the required accuracy with an allowed amount of risk. The experimental validation testing consists of a number of confirmation tests that are randomly chosen (i.e., the operating conditions are randomly chosen) and evaluated against a pre-determined metric. For example, each confirmation test may measure the amount of error between the model and experimental data for a given output, and if that error exceeds some threshold then that individual confirmation test fails. If the number of confirmation tests that fail exceeds the critical number then the validation test fails. This critical number not only protects us from not identifying a good model (i.e., Type I error) but also identifying a bad model as good (i.e., Type II error).

The number of confirmation tests and the number of confirmation tests allowed to exceed the interval is determined from use of a single sampling technique that is used in statistical quality control [10], [11], and [12].

The acceptance sampling technique is described through

$$P_a(p_1) \geq 1 - \alpha, \quad (1)$$

$$P_a(p_2) \leq \beta, \quad (2)$$

where p_1 is called the acceptable quality level (AQL), and p_2 is the rejectable quality level (RQL) or lot tolerance percent defective (LTPD). In words, the sampling plan ensures the probability of acceptance at the AQL is higher than $1 - \alpha$, and lower than β at the RQL. The binomial distribution is used here as the population of possible computations (i.e., simulations) is large and thus samples can be pulled with replacement. The two probability equations above can be solved via the binomial distribution

$$P_a(p) = \sum_{d=0}^c \left\{ \left[\frac{n!}{d!(n-d)!} \right] p^d (1-p)^{n-d} \right\}.$$

The Eqs. (1 and 2) can then be simultaneously solved and optimized over a number of test runs n and a number of allowed bad points/rejects c (see [10] for more information on the optimization). The optimization goal is to minimize the number of tests runs, n .

Tolerance Intervals

A tolerance interval is one method to determine the likely model versus experimental error for the sampling plans pass/fail criteria. In this case the accuracy of the model is being measured. On the other hand, if one wishes to evaluate whether a model meets certain requirements then the tolerance interval is not needed and the pass/fail criteria is set via requirements. As will be demonstrated later, the model used here would not likely meet the requirements specified, and thus it was desired to measure the accuracy of the model through use of tolerance intervals.

Within the field of statistics the three intervals of interest are confidence intervals, prediction intervals, and tolerance intervals [8]. In this work, tolerance intervals are of greatest interest as tolerance intervals can be used to "predict" the likely behavior of the remaining population (i.e., how the model will behave in areas not tested). Prediction intervals typically look towards predicting the value of a single next test. Confidence intervals are used for specifying the accuracy of a prediction of a parameters of the population.

A tolerance interval indicates the percentage of the population that will exhibit a characteristic with a certain level of confidence. A single-sided tolerance interval for most distributions is specified as $\bar{y} + ks$ where s is the sampled population's standard deviation of the characteristic, $\bar{y}$ represents the sampled population's mean value, and k is a factor dependent on the confidence level, population percentage [13], and distribution type. A one-sided calculation of k for normally distributed data is

$$k = \frac{t_{\alpha, N-1, \phi}}{\sqrt{N}},$$

$$\phi = z_{p1} \sqrt{N},$$

where $t_{\gamma, N-1, \phi}$ is the inverse to the non-central T cumulative distribution function, N is the number of samples, and z_{p_1} is the inverse normal distribution with probability p_1 , α is the probability that p_1 of the population will exhibit the characteristic.

As mentioned, the k value used in the tolerance interval calculation in Eq. (3) assumes a normal distribution of the population, so this interval calculation should be used with care. A distribution-free tolerance interval calculation based on work from [14] and [15] will be examined for possible use in this problem. In the work presented here, a single sided tolerance interval will be used

$$P(F(U) \geq p_1) \geq \alpha,$$

where $F(U)$ represents the cumulative distribution function at limit U , and α is the probability that p_1 of the population will exhibit the characteristic. The non-parametric solution to the one-sided problem is to order the sequence of data to be examined such that $x(j) \leq x(j+1)$ for $j = 1 : N-1$ where N is the number of elements in the data set. An upper limit for the tolerance interval is found such that $U = x(s)$. The beta distribution is used to solve for s such that

$$1 - F_\beta(p_1, a, b) \geq \alpha \quad (4)$$

where F_β is the cumulative beta distribution, such that

$$F_\beta(x|a, b) = \frac{1}{B(a, b)} \int_0^x t^{a-1} (1-t)^{b-1} dt$$

where $B(a, b)$ is the beta function [14]. The goal is to find the index s of the ordered data series where a and b in equation (4) are

$$\begin{aligned} a &= s, \\ b &= N - s + 1. \end{aligned}$$

Figure 1 depicts the minimum number of points needed to meet levels of alpha for a given sample size. Unfortunately for some of the work in this paper, to obtain a 95% (i.e., $1 - \alpha$) level more data points are needed to obtain tolerance intervals at the confidence level desired. As a way around this problem, the tolerance levels will be based on a maximum of the distribution free tolerance interval and the tolerance interval based on the normality assumption. Some preliminary work with this formulation demonstrates that this methodology does improve the confidence of the tolerance calculation for most distributions at very low number of data points (e.g., 32 data points).

State Variable Filtering

A major issue with continuous system identification is the calculation of the state derivatives. In [16] an overview of several methods of handling the derivative of the states is

described. In the state variable filter approach [17], successive filters of the states are applied to the derivatives and all states. Consider a state equation

$$\frac{dx}{dt} = \theta_1 x(t) + \theta_2 \Delta(t), \quad (5)$$

where θ_1 and θ_2 are parameters and x is a state of the system, and $\Delta(t)$ is an unknown function. Then using the p operator, defined as $p^n x(t) = \frac{d^n x(t)}{dt^n}$, Eq. (5) becomes

$$px(t) = \theta_1 x(t) + \theta_2 \Delta(t),$$

then filtering both sides by a common filter,

$$\frac{p}{p + \lambda} x(t) = \theta_1 \frac{1}{p + \lambda} x(t) + \theta_2 \frac{1}{p + \lambda} \Delta(t). \quad (6)$$

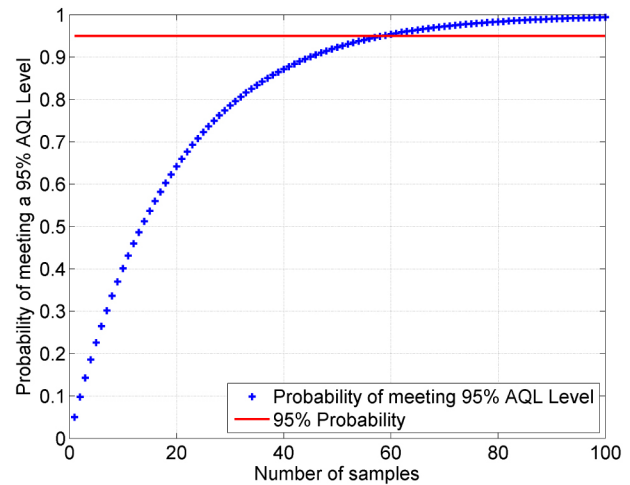


Figure 1. Non-Parametric Tolerance Interval (One-Sided) Number of Points Needed to Meet a Confidence Level.

Equation (6) can be used with discrete data, thus maintaining the ability to represent the parameters in the continuous time domain. Later, Eq. (6) will be used to estimate the magnitude of $\Delta(t)$. In this approach, information beyond the filter time constant is lost and the model is only good to the chosen bandwidth characterized by λ . Further, [17] states there are issues with state variable filter, specifically that the parameter estimates tend to be biased and inconsistent. Work done here does not disprove that statement, but does show this method works well and is relatively simple to implement. Further work should be done to examine other means to handle the state derivative issue to arrive at better parameter estimates.

DC-DC Boost Circuit Modeling

Boost converters efficiently convert a lower level unregulated voltage signal to a stabilized higher level voltage signal. The system used here has one active switch and a passive switch (diode) (see Fig. 2). The inductor in the circuit stores energy in a magnetic field while the active switch (n-MOSFET) is closed.

The passive switch (diode) will be closed and electrical energy stored in the capacitor supplies the output voltage and current for the load. When the active switch is open (passive switch will close), the inductor supplies the power to re-charge the capacitor and supplies power to the load. The additional sense resistors are present to measure the input and output currents. The n-MOSFET between R_Load1 and R_Load2 is used to switch in and out the second load.

Traditionally control of boost converters has been done with passive components in either a voltage only control or an inner current loop control with an outer loop voltage control [18] [19]. These controllers are historically developed around a linearization of the boost circuit. For voltage control, the duty cycle of the active switch is adjusted based on error in a reference signal and the output voltage. A longer duty cycle will tend to increase the voltage at the output. When the current loop is added, an inner current loop closes the active switch when the inductor current exceeds a reference current supplied by an outer loop voltage controller. This method tends to protect the components from over-current and improves the stability of the system. Instead of a linear control approach, a nonlinear controller is developed based on the model using backstepping control [20], [21]. The derivation of the control law can be found in [22].

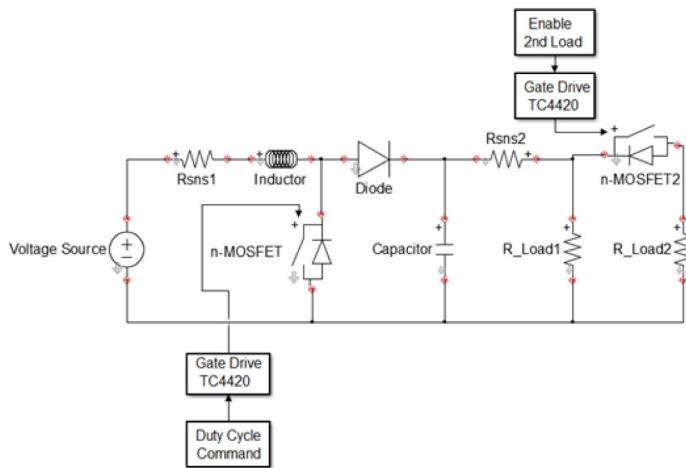


Figure 2. Physical Model of DC-DC Boost Converter

To develop the state equations for the boost converter, an average value synthesis is used for continuous conduction mode. For discussion on dc-dc converter average value models see [18]. The state equations used here are

$$\frac{dV_c}{dt} = -\frac{I_o + I_l (d_T - 1)}{C}, \quad (7)$$

$$\frac{dI_l}{dt} = \frac{V_i - I_l r - I_o r_{sns2} + V_F (d_T - 1) + V_o (d_T - 1)}{L}, \quad (8)$$

$$r = r_l + r_{sns1} + d_T r_{ds} - R_F (d_T - 1), \quad (9)$$

$$V_o = V_c - I_o r_{sns2} - r_c (I_o + I_l (d_T - 1)), \quad (10)$$

where V_o is the output voltage; d_T is the duty cycle command; C and L are the capacitance and inductance of the capacitor and inductor respectively; r is a combination of the series resistance of the inductor (r_l), the forward on resistance of the diode (R_F), the on resistance of the n-MOSFET (r_{ds}), and the inductor current's sense resistor; r_c is the series resistance of the capacitor; V_c is the voltage across the capacitance only in the capacitor; r_{sns2} is the sense resistor for the output current; I_o is the measured output current; V_i is the input voltage; V_F is the forward voltage drop across the diode; and I_l is the inductor current.

The above equations will be used as the model, but the comparison will be made as if V_o is the state variable as opposed to V_c . V_c is not a measurable quantity in the circuit as it is only the voltage across the capacitance in the capacitor and does not include the series resistance. One could transform the state equations above in a form for calculation of $\frac{dV_o}{dt}$ but is not shown here for brevity.

Model Validation

George P. Box wrote “essentially, all models are wrong, but some are useful” [23]. Which begs the question: How useful? Through validation one can begin to understand how useful a model is. Model validation defined by MIL-STD 3022 is “the process of determining the degree to which a model, simulation, or federation of models and simulations, and their associated data are accurate representations of the real world from the perspective of the intended use(s)” (i.e., “did you build the right model”) [24]. The term model validation is typically well defined, but processes and tools for model validation are actively being developed. There are a number of standards that have discussed how one might perform and/or document model validation [25, 26, 27, 28, 29] with some indication as to how one might measure validation within [27]. The following is a discussion on six approaches for model validation.

Feature Examination

There are two general categories of feature examination, qualitative and quantitative. Qualitative feature examination is simply comparing graphs between the simulation and the experimental data. Unfortunately this tends to be one of the most used methods of “model validation”, but lacks an unbiased evaluation and lacks a statistical measure of performance.

The second method, quantitative feature examination, includes methods such as signal norms, normalized square error, and others as documented in [30]. Also, in [30] the authors develop a time based metric for measuring “difference” between time histories. Specifically the metric developed measures magnitude differences, phase differences, and shape differences then weights the measurements to arrive at a single metric. These methods at a minimum offer an unbiased

evaluation of the experimental data when the method is chosen before data is obtained and a criteria for validation is pre-specified (i.e., the normalized square error should be less than a specified value). Unfortunately, these methods lack a statistical measure of performance and the pre-specified criteria often lack meaningful interpretations.

P-Box Type Techniques

In the book [31] and articles [32], [33] the author discusses the various types of uncertainty that should be considered in validation, how one might propagate uncertainties in a model, and how one might compare the output of a model and experimental results. The experimental comparison is done with a cumulative distribution area metric defined as:

$$d(F, S_n) = \int_{-\infty}^{\infty} |F(x) - S_n(x)| dx, \quad (11)$$

where $F(x)$ is the cumulative distribution function (CDF) from the simulation and $S_n(x)$ is the CDF from the experiment. In [27] the area metric is normalized by the mean of the experimental outcomes to derive a percentage difference in simulation versus experimentation. With this method of validation, there is a lack of statistical significance to the test as it is an inferential measure of the performance.

Mean Value Techniques

In [31] the author suggests another approach for model validation. This method, mean comparison, puts forth calculation of a confidence interval around the mean of the experimental data then sums the error from the simulation. The appropriate equation for the confidence interval is the following:

$$(y_m - \bar{y}_e - t_{\alpha/2, \nu} \frac{s}{\sqrt{n}}, y_m - \bar{y}_e + t_{\alpha/2, \nu} \frac{s}{\sqrt{n}}),$$

where y_m is the system response quantity (SRQ) from the model, $\bar{y}_e$ is the average SRQ from the experimental data from n number of replications, s is the sample standard deviation from the experimental SRQ data, and $t_{\alpha/2, \nu}$ is the student-t statistic based on risk, α , and the degrees of freedom ν . This technique is extended to multiple operating conditions and linear/nonlinear regression. The purpose of this validation technique is not to predict how a system will behave but to identify a parameter of interest in the experimental process (thus the use of confidence intervals as opposed to tolerance intervals). This technique offers both an unbiased validation metric and a statistical significance. Unfortunately for control theory, the goal is to find a maximum uncertainty as opposed to an average. The authors apply the mean value technique to find the maximum value of an error, but the method relies on replicated tests at the condition that causes the maximum condition. Under the approach presented here and use of tolerance intervals, those replicated tests are not needed.

Bayesian Hypothesis Testing (Interval Based)

Another technique of interest found in [34, 35, 36, 37, 38] deal with Bayesian hypothesis testing (interval based). In this test, a comparison is made between the null hypothesis $H_0: |\mu| \leq \epsilon$ and the alternative hypothesis $H_1: |\mu| > \epsilon$, where μ is the average system response quantity or vector of average system response quantities, and ϵ is the interval of interest (i.e., the average model error should be less than some fixed value). A

Bayes factor is calculated as $B_m = \frac{P(D|H_0)}{P(D|H_1)}$, where $P(D|H_0)$ is the probability of the experimental data given H_0 is true, and $P(D|H_1)$ is the probability of the experimental data given H_1 is true. If the Bayes factor is greater than 1.0, then it is more likely that the model should be accepted then rejected. This approach thus offers both an unbiased evaluation and a statistical significance to the test. Though one other author claims there is bias due to the ϵ selection [31]. The need for use of the ϵ in the formulation of the problem is acceptable as validation requires an intended use for the model and specifying ϵ is part of that specification. Note, in [37] the hypothesis are based on the error and not the average error.

Whiteness Tests

Typically done within the field of system identification for model validation or sometimes called model diagnostics [23] are tests involving calculation of the auto-correlation and cross-correlation of the residuals and inputs. As suggested by [23] the equations for autocorrelation estimation are

$$c_k = \frac{1}{N} \sum_{t=1}^{N-k} (z_t - \bar{z})(z_{t+k} - \bar{z}) \quad k = 0, 1, 2, \dots, K;$$

$$r_k = \frac{c_k}{c_0};$$

where z_t is the discretization of the signal and a value at index t , k is the lag under consideration, K is the maximum lag being examined, $\bar{z}$ is the mean value of the signal, and N are the number of data points. This formulation is slightly different than what is used in the field of signal processing where typically the mean values ($\bar{z}$) is not subtracted from the signal (z_t). Once the auto-correlation has been constructed it can be examined to determine if any one lag is significant that has not been modeled, or can be used as a whole to examine whether all lags are insignificant in affecting the model. Typically, to determine if all lags are insignificant, the autocorrelation is scaled and is determined whether it follows a chi-squared distribution with $(K - p)$ degrees of freedom

$$\tilde{Q} = N(N+2) \sum_{k=1}^K (N-k)^{-1} r_k^2.$$

Note, this formulation is modified from a form developed by Ljung-Box-Pierce [39]. Also note, it is recommended that the auto-correlation check be applied to the residuals of the model after an error model is constructed.