International Journal of Computer Applications (0975 – 8887)
Volume 11– No.3, December 2010

# Model Predictive Controller of Boost Converter with RLE Load

N. Murali
Nizwa College of Technology
Ministry of Manpower
Sultanate of Oman

K.V.Shriram
Vellore Institute of Technology University
Vellore

S.Muthukumar
Nizwa College of Technology
Ministry of Manpower
Sultanate of Oman

## ABSTRACT
This paper discuss about the new approach of control mechanism for dc to dc converters. Using this technique fast response and steady state output can be achieved. Model predictive controller will predict the future output by suitable training of the present and past occurrences. Conventional PI controllers will get the output at lesser response and vast deviation from the output. In this algorithm single input and single output is developed to describe the boost converter. The control objectives of voltage tracking are used by weighted cost function. The proposed algorithm of Model Predictive Controller is done by simulation using MATLAB. Comparative analysis and results of MPC and PI control is observed.

## Keywords
Model Predictive Controller, Boost converter, PI control.

## 1. INTRODUCTION
A DC-to-DC converter is a device that accepts a DC input voltage and produces a DC output voltage. Typically the output produced is at a different voltage level than the input. In addition, DC-to-DC converters are used to provide noise isolation, power bus regulation.

DC to DC converters are important in portable electronic devices such as cellular phones and laptop computers, which are supplied with power from batteries. Such electronic devices often contain several sub-circuits with each sub-circuit requiring a unique voltage level different than that supplied by the battery (sometimes higher or lower than the battery voltage, and possibly even negative voltage). Additionally, the battery voltage declines as its stored power is drained. DC to DC converters offer a method of generating multiple controlled voltages from a single variable battery voltage, thereby saving space instead of using multiple batteries to supply different parts of the device.

They are extensively used in power supplies for electronic equipment to control the energy flow between two DC systems (e.g. well-regulated DC-to-DC power converters are critical for mission success on space platforms). Control of a DC-DC converter power circuit is based, explicitly or implicitly, on a model that describes how control actions and disturbances are expected to affect the future behavior of the system. Usually, the control problem consists in defining the desired nominal operating condition, and then regulating the circuit so that it stays close to the nominal, when the system is subject to disturbances and modeling errors that cause its operation to deviate from the nominal.

A Proportional-Integral-Derivative controller is a generic control loop feedback mechanism widely used in industrial control systems. A PI controller attempts to correct the error between a measured process variable and a desired set point by calculating and then outputting a corrective action that can adjust the process accordingly.

The PI controller calculation (algorithm) involves three separate parameters; the Proportional, the Integral.The Proportional value determines the reaction to the current error, the Integral determines the reaction based on the sum of recent errors and the Derivative determines the reaction to the rate at which the error has been changing. The weighted sum of these three actions is used to adjust the process via a control element such as the position of a control valve or the power supply of a heating element. By tuning the three constants in the PI controller algorithm the PI can provide control action designed for specific process requirements.

The response of the controller can be described in terms of the responsiveness of the controller to an error, the degree to which the controller overshoots the set point and the degree of system oscillation. Note that the use of the PI algorithm for control does not guarantee optimal control of the system.
Some applications may require using only one or two modes to provide the appropriate system control. This is achieved by setting the gain of undesired control outputs to zero.

Typical control system configurations for power circuits include open-loop as well as closed-loop control strategies. In both cases, PI controllers are utilized. The controller must keep the DC-DC converter within a certain percentage of the specified nominal operating point in the presence of disturbances and modeling errors. Unfortunately, PI control does not always fulfill the above mentioned control specifications, especially when disturbance rejection and transient response time requirements are concerned, due to the highly non-linear characteristics of the DC-DC converters. As a result of this fact, there is much interest in developing more intelligent and robust control structures.
The schematic in the below illustrated

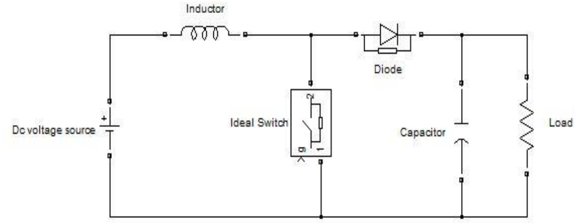


Fig.1Schematic diagram of a Dc-Dc Converter

## 2. PROBLEM DEFINITION

The input to the system is unregulated Dc Voltage, and the output is regulated Dc Voltage. The main problems faced by the usage of the converter are:

1) The system may be non-linear.
2) The switch positions are discrete valued. For a non linear system discontinuous switching may not be efficient to control the output voltage.
3) Maintaining constraints of input states and output states is difficult.

The objective of the paper is to highlight the better performance of the Model Predictive Controller over the classical control methods (PI Controller).This is achieved by the comparative analysis of the both control strategies. Design of PI and Model Predictive Controller is done through the Matlab /Simulink.

## 3. MODEL PREDICTIVE CONTROLLER

The essence of MPC is to optimize, over the manipulate inputs, forecasts of process behavior. The forecasting is accomplished with a process model, and therefore the model is the essential element of an MPC controller. As discussed subsequently, models are not perfect forecasters, and feedback can overcome some effects of poor models, but starting with a poor process model is akin to driving a car at night without headlights; the feedback may be a bit late to be truly effective.

The advantages of MPC are handles multivariable control problems naturally, It can take account of actuator limitations, It allows operation closer to constraints, hence increased profit, It has plenty of time for on-line computations, It can handle non-minimal phase and unstable processes, It is an easy to tune method and It handles structural changes.

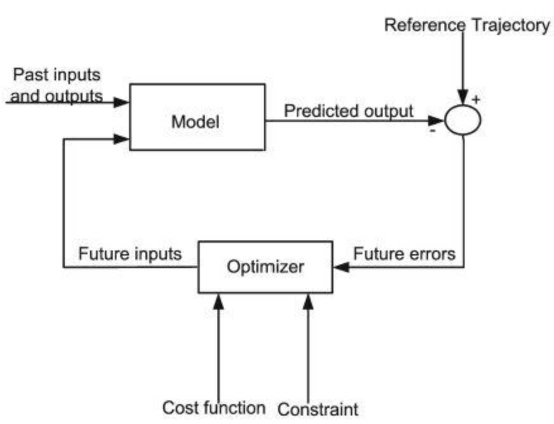


Fig.2Schematic diagram of Model Predictive Controller

There exist many methods, including classical frequency-domain techniques, for designing stabilizing control laws for time-invariant linear systems. In contrast, there exist relatively few methods for time-varying linear systems, and fewer still for non-linear systems. The major difficulty in the design of feedback control laws for non-linear systems arises from the necessity to explore the whole state space. The problem of the design of the feedback controls for nonlinear systems has found a general solution only in the case of systems which are feedback equivalent to linear systems.

The working of a Model Predictive Control system can be explained with the illustrated diagram.

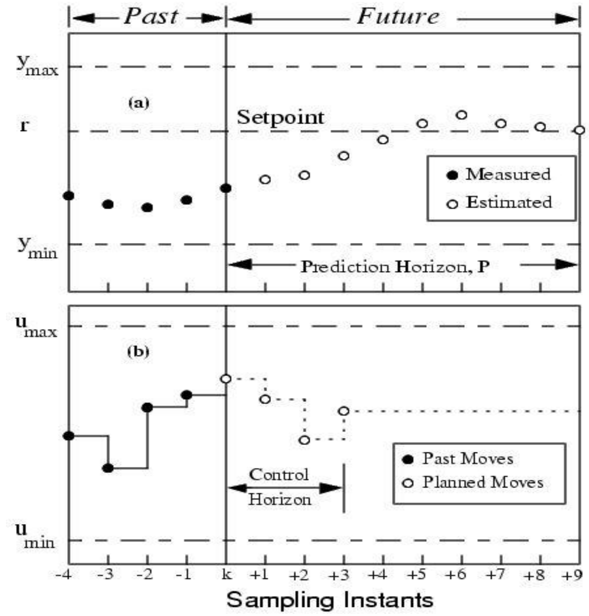


Fig.3 Working of Model Predictive Controller

The part 3(a) shows the state of a hypothetical SISO MPC system that has been operating for many sampling instants. Integer k represents the current instant. The latest measured output, $y_k$, and previous measurements, $y_{k-1}$, $y_{k-2}$... are known and are the filled circles in Figure 3(a). If there is a measured disturbance, its current and past values would be known.

The part 3(b) shows the controller's previous moves, $u_{k-41}$... $u_{k-1}$, as filled circles. As is usually the case, a zero-order hold receives each move from the controller and holds it until the next sampling instant, causing the step-wise variations.

To calculate its next move, $u_k$ the controller operates in two phases are Estimation and Optimization. In order to make an intelligent move, the controller needs to know the current state. This includes the true value of the controlled variable, $\hat{y}_k$, and any internal variables that influence the future trend, $\hat{y}_{k+1}$... $\hat{y}_{k+P}$ . To accomplish this, the controller uses all past and current measurements and the models, $u\rightarrow\hat{y}$, $d\rightarrow\hat{y}$, $w\rightarrow\hat{y}$ and $z\rightarrow\hat{y}$.This process is an estimated process.

In Optimization the values of set points, measured disturbances, and constraints are specified over a finite horizon of future sampling instants, k+1, k+2... k+P , where P (a finite integer ≥1) is the prediction horizon. The controller computes M moves $u_k$, $u_{k+1}$... $u_{k+M-1}$, where M (≥1, ≤P) is the control horizon. In the hypothetical example shown in the figure, P = 9 and M = 4. The moves are the solution of a constrained optimization problem.

The optimal moves are the four open circles in Figure 3(b), and the controller predicts that the resulting output values will be the nine open circles in Figure 3(a). It also maintains both the values within their constraints, $u_{min} \leq u_{k+j} \leq u_{max}$ and $y_{min} \leq y_{k+I} \leq y_{max}$..When it's finished calculating, the controller sends move $u_k$ to the system. The system operates with this constant input until the next sampling instant, Δt time units later. The controller then obtains new measurements and totally revises its plan. This cycle repeats indefinitely. Reformulation at each sampling instant is essential for good control. The predictions made during the optimization stage are imperfect. Periodic measurement feedback allows the controller to correct for this error and for unexpected disturbances.

At the time point $t_{k-1}$ the optimum of the [quadratic objective function] $Z_k$ is sought. The resulting control [input] vector $U^*_{(k)}$ depends on $x_{(k-1)}$ and contains all control [input] vectors $u^*_k$, $u^*_{k+1}$… $u^*_N$ which control the process optimally over the interval [$t_{k-1}$, $T$]. Of these control [input] vectors, one implements the vector $u^*_k$ (which depends on $x_{(k-1)}$) as input vector for the next interval [$t_{k-1}$, $t_k$]. At the next time point $t_k$ a new input vector $u^*_{k+1}$ is determined. This is calculated from the objective function $Z_{k+1}$ and is dependent on $x_{(k)}$. Therefore, the vector $u_k$, which is implemented in the interval $t_k$, is dependent on the state vector $x_{(k-1)}$. Hence, the sought feedback law consists of the solution of a convex optimization problem at each time point $t_{k-1}$ ($k$ = 1, 2… $N$).

When the input of the system for the instant $t_k$ is decided, the controller repeats the optimization procedure for the prediction horizon length starting from $t_k$. So the time period is being advanced to the next p time units, i.e. till $t_{k+P+1}$. And this procedure is carried in the fashion of an iterative process over the time period. Hence, the process control is sometimes referred to as the receding horizon prediction control.

A typical model of system in state space is given by

$$X(k+1) = AX(k) + Bu(k) \quad -(1)$$

$$Y(k) = C_y X(k) \quad -(2)$$

$$Z(k) = C_z X(k) \quad -(3)$$

$$where\ X \in R^n, u \in R^l, Y \in R^{my}, Z \in R^{mz}$$

The basic formulation of the cost function can be done as follows

$$V(k) = \sum_{i=H_w}^{H_p} \|\hat{z}(k+i\backslash k) - r(k+i\backslash k)\|^2_{Q(i)} + \sum_{i=0}^{H_u-1} \|\Delta\hat{u}(k+i\backslash k)\|^2_{R(i)}$$

## 4. MPC TOOLBOX IN MATLAB SIMULINK

The Model Predictive Control Toolbox is a collection of software that helps you design, analyze, and implement an advanced industrial automation algorithm. Like other MATLAB® tools, it provides a convenient graphical user interface (GUI) as well as a flexible command syntax that supports customization.

A Model Predictive Control Toolbox controller automates a target system (the system) by combining a prediction and a control strategy. An approximate system model provides the prediction. The control strategy compares predicted system signals to a set of objectives, and then adjusts available actuators to achieve the objectives while respecting the system's constraints.

The controller's constraint-tolerance differentiates it from other optimal control strategies (e.g., the Linear-Quadratic-Gaussian approach supported in the Control System Toolbox). The impetus for this is industrial experience suggesting that the drive for profitability often pushes the system to one or more constraints. The Model Predictive Control Toolbox controller considers such factors explicitly, allowing it to allocate the available system resources intelligently as the system evolves over time.

The Model Predictive Control Toolbox uses the same powerful linear dynamic modeling tools found in the Control System Toolbox and System Identification Toolbox. You can employ transfer functions, state-space matrices, or a combination. You can also include delays, which are a common feature of industrial systems.

When using the Model Predictive Control Toolbox the following steps have to be followed in order to ensure a proper working of the strategy define your system using the Control System Toolbox modeling tools (LTI transfer function and state space models), derive a linear system model from a nonlinear Simulink representation. Design Model Predictive Control for your system using *mpctool*, the graphical user interface (GUI) and Simulate Model Predictive Control performance using *mpctool*, Simulink, or commands.

## 5. DESIGN PROCEDURE

The design of Dc-Dc converter is made as basic as possible for further calculations to be hassle free. The circuit is as shown in Fig.3.1.

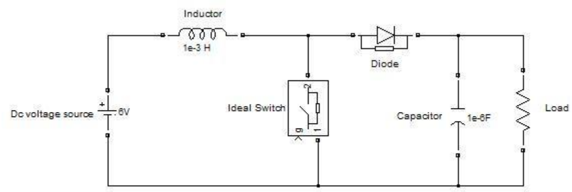


Fig.4 Dc-Dc Converter Circuit

The values of the components are as follows:
Inductor (L) = 1e-3 Henry
Capacitor (C) = 1e-6 Farad
Source voltage ($V_{in}$) = 6 Volts

The circuit is designed for the convenience that for prescribed input of 6V, the output voltage obtained across the load is 12V. The duty cycle for the obtained operation is 0.5. Hence the system is compensated and these values of $K_p$ and $K_i$ are substituted in the simulation and the result is obtained.

$$k_p = \frac{\cos\theta}{A} = 1.0125$$

$$k_i = -\frac{\omega \sin\theta}{A} = 9.05e-5$$

## 6. SIMULATION RESULTS

The results obtained after simulating the circuit of boost converter with PI Controller are as follows. The graph is the result of simulating the circuit with an input voltage of 12 volts and the reference voltage level set to 12 volts. Upon effectively carrying out the simulation, the output voltage level was observed to be 12.12 volts and the settling time of the circuit as clearly seen in the graph is 5.5 seconds.

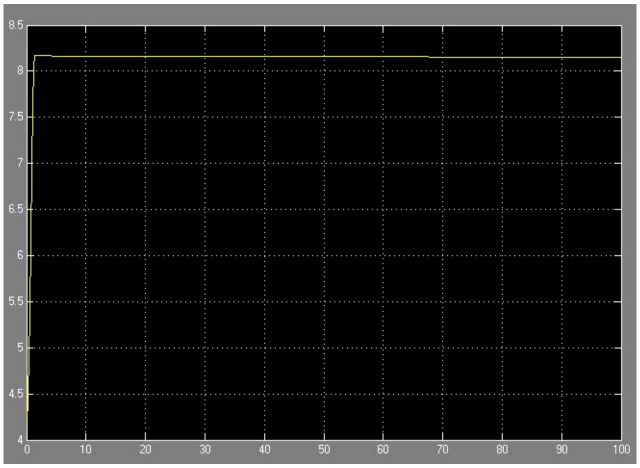


Fig 5 Simulation result for RLE load (PI)

**Table 1Simulation results with RLE load**

| S.No | Input voltage | Output Voltage |
|---|---|---|
| 1 | 5 | 6.96 |
| 2 | 5.5 | 7.83 |
| 3 | 5.6 | 8.01 |
| 4 | 5.7 | 8.18 |
| 5 | 5.8 | 8.36 |
| 6 | 5.9 | 8.53 |
| 7 | 6.0 | 8.71 |
| 8 | 6.1 | 8.88 |
| 9 | 6.2 | 9.06 |
| 10 | 6.3 | 9.23 |
| 11 | 6.4 | 9.41 |
| 12 | 6.5 | 9.58 |
| 13 | 7 | 10.46 |

**Table 2 Variance tabulation of PI controller with RLE load**

| **Input voltage** | **Output voltage** | **Reference voltage** | **Standard deviation** | **Variance** |
|---|---|---|---|---|
| 5 | 6.96 | 8 | 1.04 | 1.0816 |
| 5.5 | 7.83 | 8 | 0.17 | 0.0289 |
| 5.6 | 8.01 | 8 | -0.01 | 1E-04 |
| 5.7 | 8.18 | 8 | -0.18 | 0.0324 |
| 5.8 | 8.36 | 8 | -0.36 | 0.1296 |
| 5.9 | 8.53 | 8 | -0.53 | 0.2809 |
| 6 | 8.71 | 8 | -0.71 | 0.5041 |
| 6.1 | 8.88 | 8 | -0.88 | 0.7744 |
| 6.2 | 9.06 | 8 | -1.06 | 1.1236 |
| 6.3 | 9.23 | 8 | -1.23 | 1.5129 |
| 6.4 | 9.41 | 8 | -1.41 | 1.9881 |
| 6.5 | 9.58 | 8 | -1.58 | 2.4964 |
| 7 | 10.46 | 8 | -2.46 | 6.0516 |
| Mean of Variance = 1.2311 | | | | |

The below graph is the result of simulating the circuit with an input voltage of 12 volts and the reference voltage level set to 12 volts. Upon effectively carrying out the simulation, the output voltage level was observed to be 11.09 volts and the settling time of the circuit as clearly seen in the graph is 1.4 seconds.

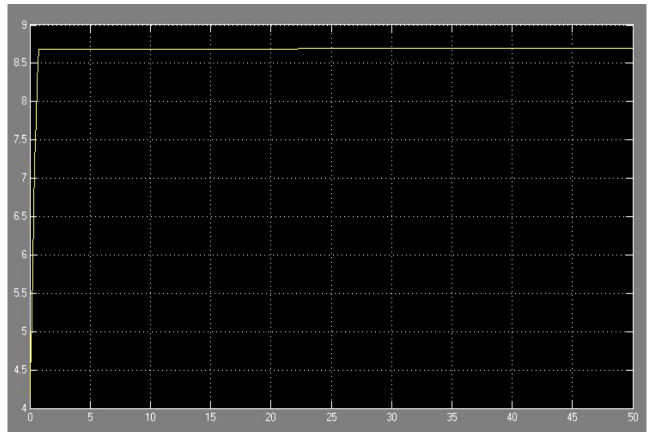


Fig 6 Simulation result for RLE load (MPC)

**Table 3 Simulation results of RLE load with MPC control**

| S.No | Input voltage | Output Voltage |
|---|---|---|
| 1 | 5 | 7.13 |
| 2 | 5.5 | 7.78 |
| 3 | 5.6 | 7.87 |
| 4 | 5.7 | 7.95 |
| 5 | 5.8 | 8.02 |
| 6 | 5.9 | 8.08 |
| 7 | 6.0 | 8.13 |
| 8 | 6.1 | 8.18 |
| 9 | 6.2 | 8.24 |
| 10 | 6.3 | 8.31 |
| 11 | 6.4 | 8.39 |
| 12 | 6.5 | 8.48 |
| 13 | 7 | 9.08 |

**Table 4 Variance tabulation of MPC controller with RLE load**

| **Input voltage** | **Output voltage** | **Reference voltage** | **Standard deviation** | **Variance** |
|---|---|---|---|---|
| 5 | 7.13 | 8 | 0.87 | 0.7569 |
| 5.5 | 7.78 | 8 | 0.22 | 0.0484 |
| 5.6 | 7.87 | 8 | 0.13 | 0.0169 |
| 5.7 | 7.95 | 8 | 0.05 | 0.0025 |
| 5.8 | 8.02 | 8 | -0.02 | 0.0004 |
| 5.9 | 8.08 | 8 | -0.08 | 0.0064 |
| 6 | 8.13 | 8 | -0.13 | 0.0169 |
| 6.1 | 8.18 | 8 | -0.18 | 0.0324 |
| 6.2 | 8.24 | 8 | -0.24 | 0.0576 |
| 6.3 | 8.31 | 8 | -0.31 | 0.0961 |
| 6.4 | 8.39 | 8 | -0.39 | 0.1521 |
| 6.5 | 8.48 | 8 | -0.48 | 0.2304 |
| 7 | 9.08 | 8 | -1.08 | 1.1664 |
| Mean of Variance = 0.19872 | | | | |

# 7. CONCLUSIONS

It is observed from the results that the Model Predictive Controller has a faster settling time compared to that of the PI controller. This is because; the Model Predictive Controller has the acute ability to predict the input state so that the output of the system stays as close as to the true value or the set point value. Hence, for every passing moment, the Model Predictive Controller modifies the input such that the output of the system reaches the given set point. The mean of variance can be observed to be less for the Model

Predictive Controller. This indicates a better performance for the Model Predictive Controller for the RLE load in the boost Converter. MPC algorithm can be applied in real time applications using microcontroller and FPGA for controlling the Converters and Drives

## 8. ACKNOWLEDGEMENT

The authors like to acknowledge the management of Vellore Institute of Technology, for the support during the process of making in to a paper also like to thank everyone in Department of Electrical and Electronics division, for the variable commands assigned and useful discussion while preparing this paper.

## 9. REFERENCES

[1]. James B. Rawlings, "Model Predictive Control Technology" Proceedings of the American Control Conference San Diego, California June 1999.

[2]. F. Borrelli, A. Bemporad, M. Fodor, and D. Hrovat, "An MPC/hybrid system approach to traction control," *IEEE Trans. Contr. Systems Technology*, vol. 14, no.3, pp. 541–552, May 2006.

[3]. S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, pp.733–764, 2003.

[4]. Pannocchia and J. B. Rawlings, "Disturbance models for offset-free MPC control," *AIChE J.*, vol. 49, no. 2, pp. 426–437, Feb. 2003.

[5]. A. Bemporad, N. Ricker, and J. Owen, "Model predictive control –New tools for design and evaluation,"in *American Control Conference*, Boston, MA, 2004, pp. 5622–5627.

[6]. D. W. Clarke, C. Mohtadi, P. S. Tuffs, Generalized Predictive Control – Part I.The Basis Algorithm, *Automatica*, Vol. 23, No. 2, pp. 137-148, 1987.

[7]. Bordons, C., Camacho, E.F.: A generalized predictive controller for a wide class of industrial processes, IEEE Transactions on control systems technology, Vol.26, No. 3,May 1998.

[8]. D. W. Clarke, C. Mohtadi, "Properties of GeneralizedPredictive Control", *Automatica*, Vol. 25, No. 6, pp. 859-875,1989.