# ELEC 5564M

## POWER GENERATION BY RENEWABLE SOURCES

**Report on**

# MODELLING A BOOST CONVERTER -WITH AND WITHOUT PV CONNECTED

**Dagogo Godwin Orifama**

**201177661**

**MSc Electronics and Electrical Engineering**

**University of Leeds**

**November 2017**

## Introduction

The aim of this laboratory exercise is to investigate and validate the mathematical formulae that govern the operations of a boost converter with and without a PV connection. The procedure uses Runge-Kutta 4$^{th}$ order differential equation in solving the state space equations generated when analysing the boost converter circuit to develop a MATLAB program to simulate it. Furthermore, the session also covers designing new inductors and capacitor values and to analyse the effect of a change in the value of capacitance and inductance on the rippling behaviour of the state variable.

The boost (step-up) converter circuit is shown in figure 1

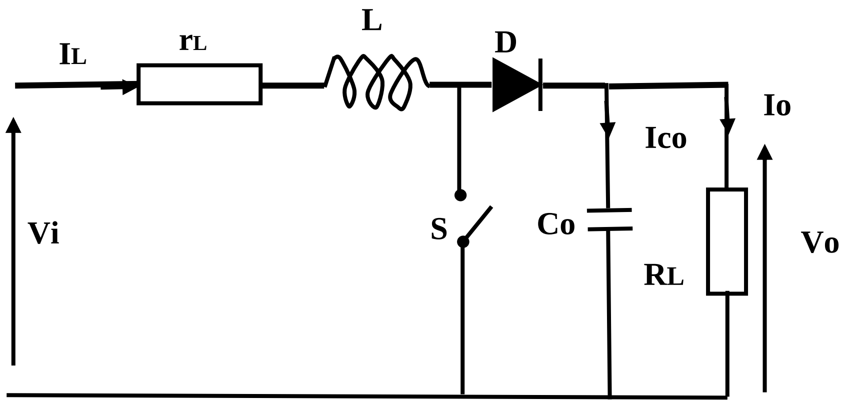


Figure 1: Step-up (Boost) converter

Where,

Vi is the input dc voltage volts

IL is the inductor current in ampere

rL is the input resistance in ohms

L is the inductance (mH)

D is the diode

S is the switch

Co is the output capacitance (uF)

RL is the load resistance (ohms)

Vo is the capacitor current(A)

Io is the load current (A)

From figure 1 above when switch is closed, the equivalent circuit is shown below

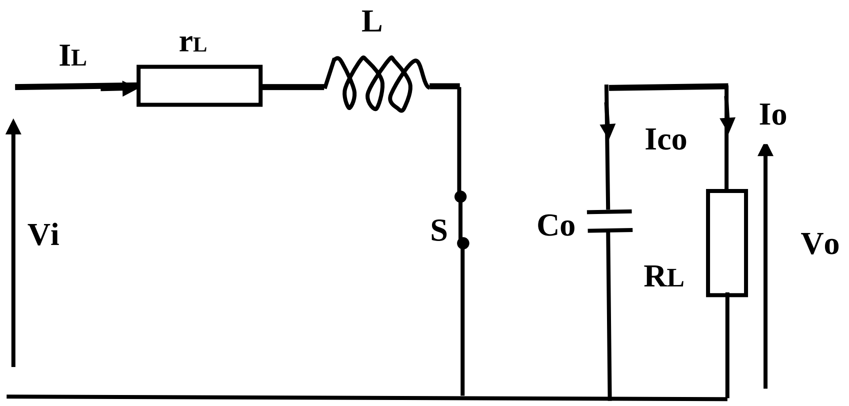


Figure 2: Step-up converter during “ON” state

The evaluation of the state variables from figure 2 is show below

From the input section

$V_i = I_L r_L + L\frac{\partial I_L}{\partial t}$ re-arranging the equation gives $\frac{\partial I_L}{\partial t} = -\frac{r_L}{L}I_L + \frac{1}{L}V_i$ (1)

Likewise, considering the output part of the circuit, since the capacitor discharges through the load, the evaluation of the load current is shown below.

From KCL $I_{co} = I_o$ substituting gives

$-C_o\frac{\partial V_o}{\partial t} = \frac{V_o}{R_L}$ re-arranging the equation gives $\frac{\partial V_o}{\partial t} = -\frac{1}{C_o R_L}V_o$ (2)

Using $\dot{I}_L = \frac{\partial I_L}{\partial t}$ and $\dot{V}_o = \frac{\partial V_o}{\partial t}$

Equation (1) and (2) can be represented using space state representation as shown below

$$\begin{bmatrix} \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} -\frac{r_L}{L} & 0 \\ 0 & -\frac{1}{C_o R_L} \end{bmatrix} \begin{bmatrix} I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_i \qquad (3)$$

Also, from figure 1, during the “OFF” state the equivalent circuit is shown below

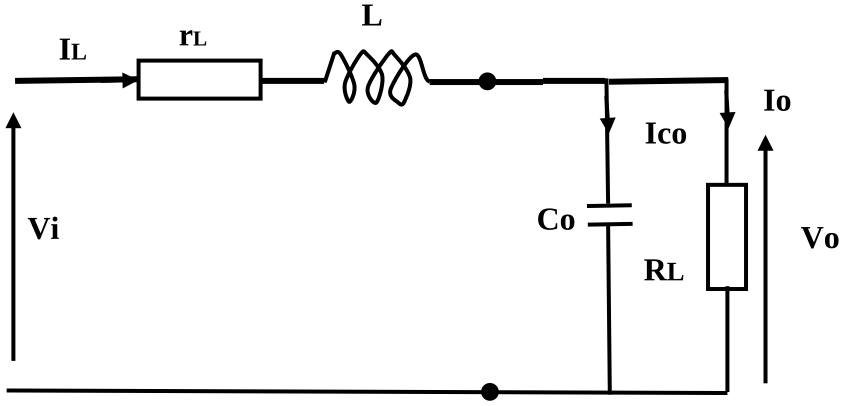

Figure 3: Step-up (Boost) converter in "OFF" state

From the input section we have

$$V_i = I_L r_L + L\frac{\partial I_L}{\partial t} + V_o \text{ re-arranging the circuit gives } \frac{\partial I_L}{\partial t} = -\frac{r_L}{L}I_L - \frac{1}{L}V_o + \frac{1}{L}V_i \quad (4)$$

Considering KCL on the output section $I_L = I_{co} + I_o$ substituting the corresponding values yields

$$I_L = C_o\frac{\partial V_o}{\partial t} + \frac{V_o}{R_L} \text{ re-arranging the equation gives } \frac{\partial V_o}{\partial t} = \frac{1}{C_o}I_L - \frac{1}{C_o R_L}V_o \quad (5)$$

Similarly, equation (4) and (5) can be represented using space state equation as shown below

$$\begin{bmatrix} \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} -\frac{r_L}{L} & -\frac{1}{L} \\ -\frac{1}{C_o} & -\frac{1}{C_o R_L} \end{bmatrix} \begin{bmatrix} I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_i \quad (6)$$

Equation 3 and 6 are combined to yield the state space representation of figure 1 during "ON" and "OFF" state

$$\begin{bmatrix} \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} -\frac{r_L}{L} & -\frac{(1-s)}{L} \\ -\frac{(1-s)}{C_o} & -\frac{1}{C_o R_L} \end{bmatrix} \begin{bmatrix} I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_i \quad (7)$$

Where s = 1 for "ON" state and s = 0 for "OFF" state

Runge-Kutta method was employed to provide a solution of $\dot{I}_L$ and $\dot{V}_o$. For each time increment $\Delta t$, four evaluations are generated for voltage and current respectively as shown below [1]

$$k1_{(iorv)} = f(I(t-1)orV(t-1)) \quad (8)$$

$$k2_{(iorv)} = f(I(t-1) + K1_i \times \frac{\Delta t}{2} orV(t-1) + K1_v \times \frac{\Delta t}{2}) \quad (9)$$

$$k3_{(iorv)} = f(I(t-1) + K2_i \times \frac{\Delta t}{2} orV(t-1) + K2_v \times \frac{\Delta t}{2})$$

(10)

$$k4_{(iorv)} = f(I(t-1) + K3_i \times \Delta t orV(t-1) + K3_v \times \Delta t)$$

(11)

The value of f is gotten as shown

$$f = -\frac{r_L}{L}I_L - \frac{(1-s)}{L}V_o + \frac{1}{L}V_i \text{ for current}$$

$$f = \frac{(1-s)}{C_o} I_L - \frac{1}{C_o R_L} V_o \text{ for voltage}$$

The values of k1, k2, k3 and k4 are estimations of the slope during the iteration. K1 is the approximate slope at the start of the iteration, K2 and K3 are approximated slopes of the midpoint between K1 and K4, K4 is the estimated slop at end of iteration.

Next, new values of current and voltage is evaluated as shown below

$$I(t) = I(t-1) + \frac{\Delta t}{6}(k1_i + 2 \times k2_i + 2 \times k3_i + k4_i) \quad (12)$$

$$V(t) = V(t-1) + \frac{\Delta t}{6}(k1_v + 2 \times k2_v + 2 \times k3_v + k4_v) \quad (13)$$

**Program Flowchart and Algorithm**

The program flowchart used to solve the state space equations by applying Runge-Kutta method is summarized in table 4

| |
|---|
| **Start** |
| Initialize variables |
| Initialize arrays VCSAVE, ILSAVE, sim_time |
| Increment counter by sampling time dt |
| Check |
| Is tDiff greater than the period |
| If yes, turn switch ON, i.e s = 1 |
| If NO, turn OFF switch, i.e, s = 0 |
| End check |
| Evaluate values for k1, k2, k3 and k4 |
| Update values of x |
| Assign component of x to current, IL and voltage, V0 |
| Update VCSAVE, ILSAVE AND sim_time arrays |
| Plot sim_time versus VCSAVE and sim_time versus ILSAVE |
| **End** |

Table 1: Algorithm for boost converter MATLAB program

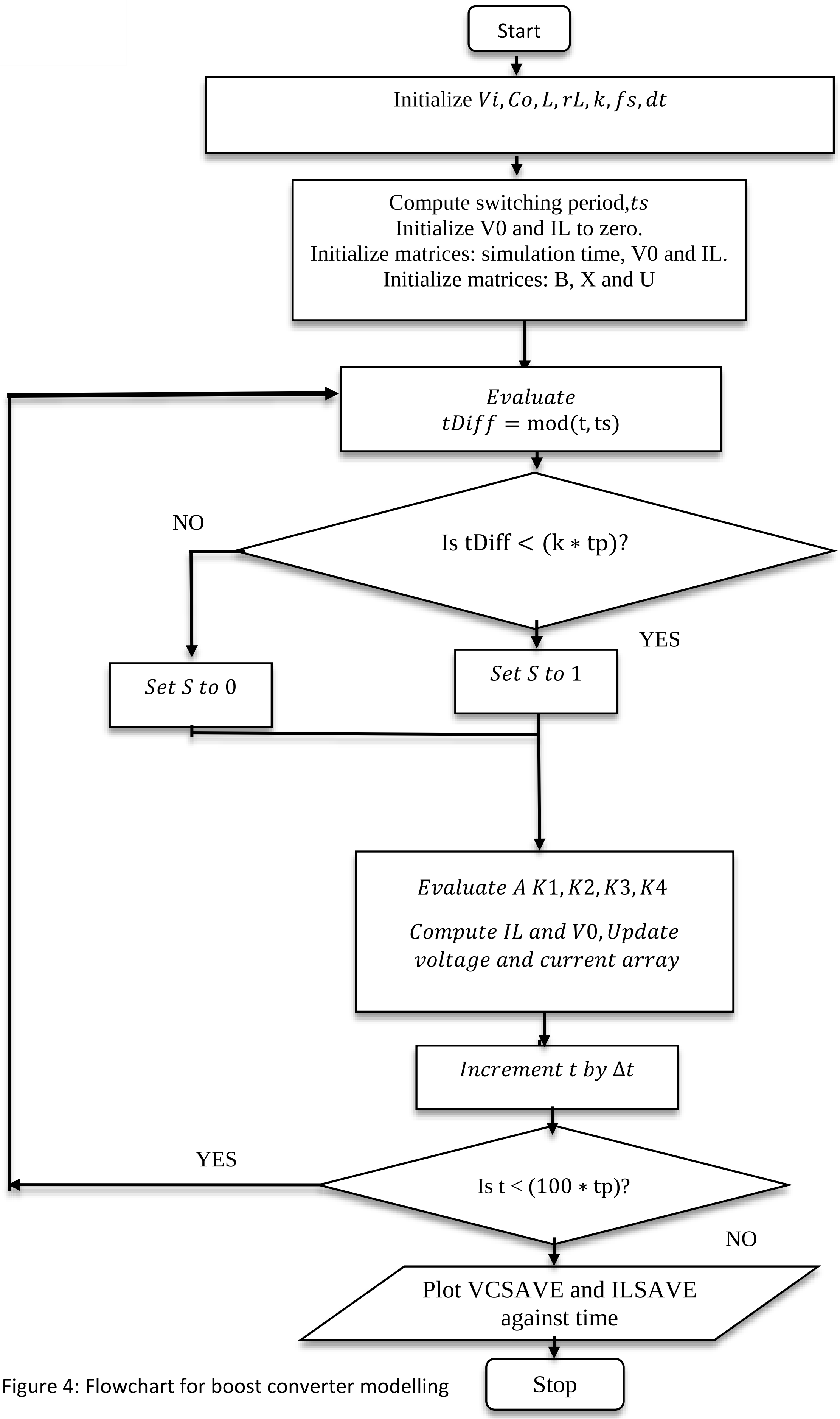


Figure 4: Flowchart for boost converter modelling

Figure 5 and 6 show the simulation result for the boost converter plot for inductor current versus simulation time and capacitor voltage versus simulation time when k = 0.45, subsequently, figure 7 and 8 shows a similar plot, when k = 0.75

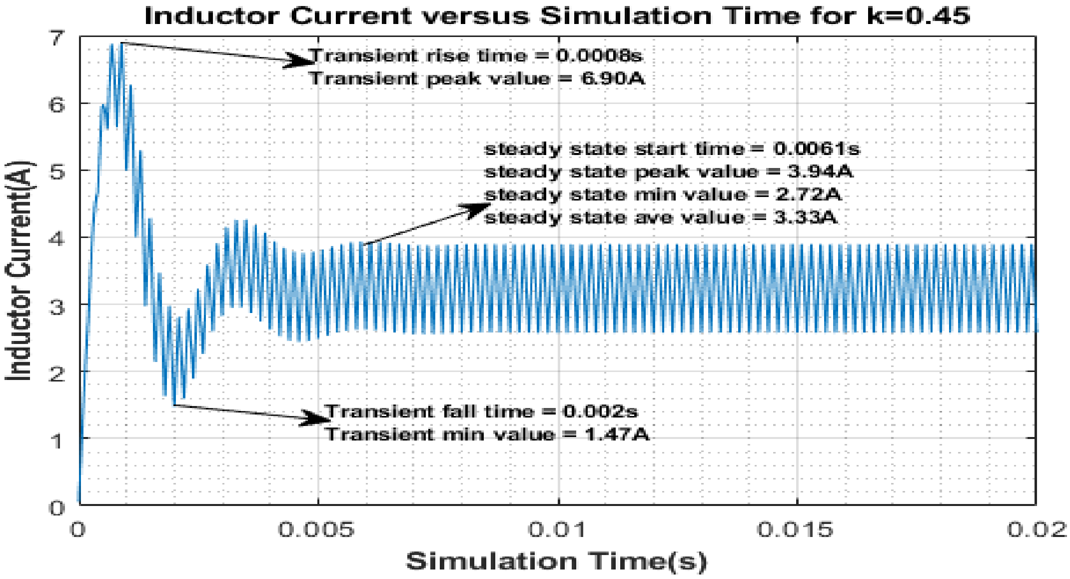


Figure 5 Inductor current versus simulation time for k = 0.45

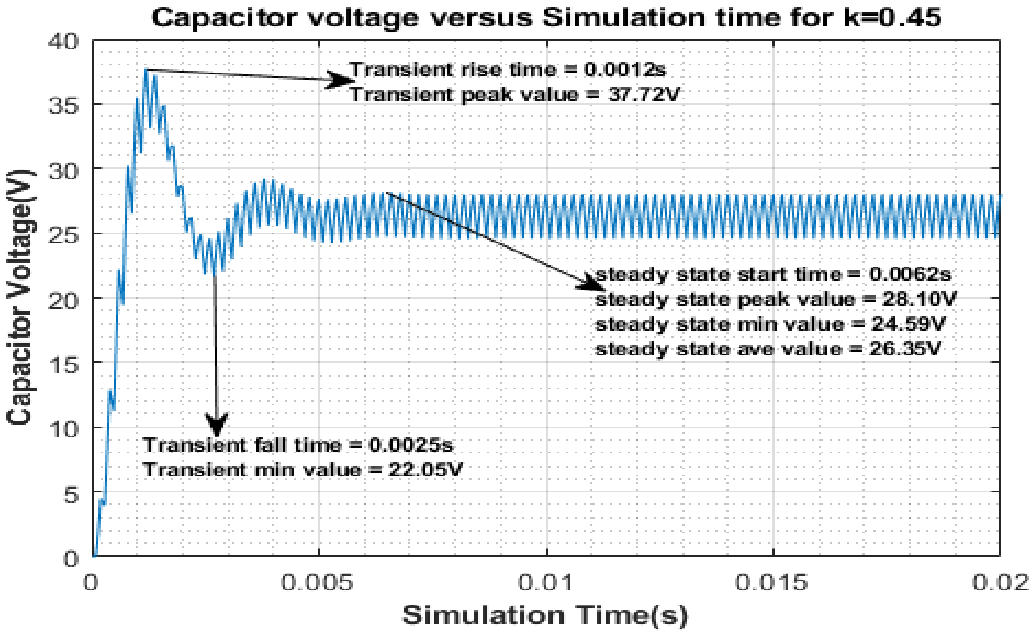


Figure 6 capacitor voltage versus simulation time for k = 0.45

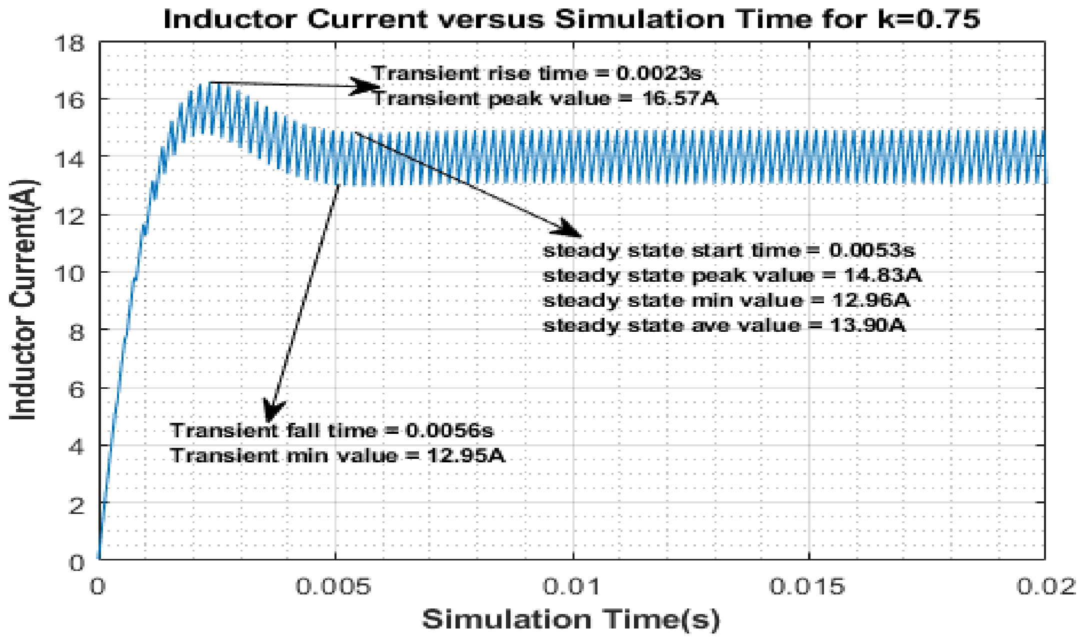


Figure 7 Inductor current versus simulation time for k = 0.75

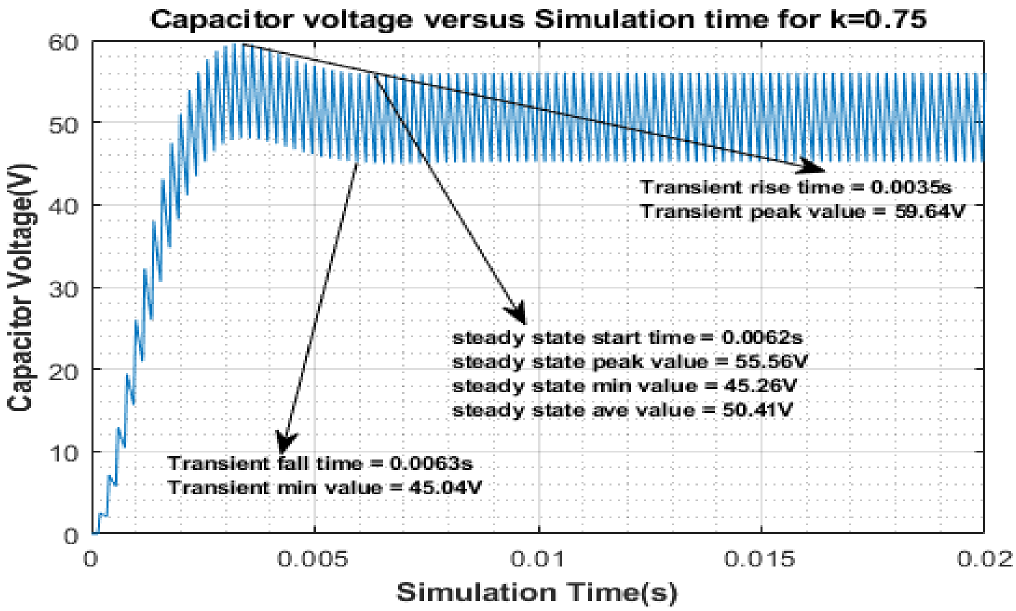


Figure 8 Capacitor voltage versus simulation time for k = 0.75

**Analysing the Inductor Current and Capacitor Voltage against Time plot**

From the Inductor current against simulation time plotted in figure 5 to 8, the plot shows that as the value of k increases, there is a corresponding increase in the output voltage. But, the reverse is the case when it comes to the current, which decreases proportionally with increasing values of k. This proportional increase in voltage and decrease in current is based on a common factor shown below

$$\frac{V_o}{V_i} = \frac{I_L}{I_o} = \frac{1}{1-k} \qquad (14)$$

Considering figure 5 and 6 with k = 0.45, the peak-to-peak ripple at steady state current and the output voltage is evaluated to 36.6% and 13.3% respectively. Similarly, the evaluation of the efficiency is shown below

At steady state, the peak-to-peak voltage and current is given by

$$V_{pp} = V_{\max} - V_{\min} \quad (15)$$

$$I_{pp} = I_{\max} - I_{\min} \quad (16)$$

**At K = 0.45**

For Voltage $V_{\max} = 28.10V, V_{\min} = 24.59V$

$$V_{pp} = V_{\max} - V_{\min} = 28.10 - 24.59 = 3.51V \quad (17)$$

Average voltage, $V_{av} = \dfrac{V_{\max} + V_{\min}}{2} = \dfrac{28.10 + 24.59}{2} = 26.35V$ (18)

Percentage (%) of average voltage $= \dfrac{V_{pp}}{V_{av}} \times 100\% = \dfrac{3.51}{26.35} \times 100\% = 13.3\%$

(18)

Output Voltage, $V_o = \dfrac{V_i}{1-k} = \dfrac{15}{1-0.45} = 27.27V$

Efficiency $= \dfrac{V_{av}}{V_o} \times 100\% = \dfrac{26.35}{27.27} \times 100\% = 96.6\%$

Similarly, for current $I_{\max} = 3.94A, I_{\min} = 2.72A$

$$I_{pp} = I_{\max} - I_{\min} = 3.94 - 2.72 = 1.22A$$

Average current, $I_{av} = \dfrac{I_{\max} + I_{\min}}{2} = \dfrac{3.94 + 2.72}{2} = 3.33A$

Percentage (%) of average current $= \dfrac{I_{pp}}{I_{av}} \times 100\% = \dfrac{1.22}{3.33} \times 100\% = 36.6\%$

Output current, $I_o = \dfrac{V_o}{R_L} = \dfrac{27.27}{15} = 1.818A$

$$I_L = \frac{I_o}{1-k} = \frac{1.818}{1-0.45} = 3.31A$$

Efficiency $= \dfrac{I_{av}}{I_L} \times 100\% = \dfrac{3.33}{3.31} \times 100\% = 100.6\%$

**At K = 0.75**

For Voltage $V_{\max} = 55.56V, V_{\min} = 45.26V$

$V_{pp} = V_{\max} - V_{\min} = 55.56 - 45.26 = 10.3V$

Average voltage, $V_{av} = \dfrac{V_{\max} + V_{\min}}{2} = \dfrac{55.56 + 45.26}{2} = 50.41V$

Percentage (%) of average voltage $= \dfrac{V_{pp}}{V_{av}} \times 100\% = \dfrac{10.3}{50.41} \times 100\% = 20.43\%$

Output Voltage, $V_o = \dfrac{V_i}{1-k} = \dfrac{15}{1-0.75} = 60V$

Efficiency $= \dfrac{V_{av}}{V_o} \times 100\% = \dfrac{50.41}{60.00} \times 100\% = 84.02\%$

Similarly, for current $I_{\max} = 14.83A, I_{\min} = 12.96A$

$I_{pp} = I_{\max} - I_{\min} = 14.83 - 12.96 = 1.87A$

Average current, $I_{av} = \dfrac{I_{\max} + I_{\min}}{2} = \dfrac{14.83 + 12.96}{2} = 13.90A$

Percentage (%) of average current $= \dfrac{I_{pp}}{I_{av}} \times 100\% = \dfrac{1.87}{13.90} \times 100\% = 13.45\%$

Output current, $I_o = \dfrac{V_o}{R_L} = \dfrac{60}{15} = 4A$

$I_L = \dfrac{I_o}{1-k} = \dfrac{4}{1-0.75} = 16A$

Efficiency $= \dfrac{I_{av}}{I_L} \times 100\% = \dfrac{13.90}{16} \times 100\% = 86.88\%$

From the above calculations, it is evident that as the value of k increase the efficiency of the boost converter decrease. This is caused by the increasing ripple caused by the increasing voltages and decreasing current.

**Filter Design**

To reduce the ripple in current and voltages to its minimal, then filter design must come into play. Filter design involves a careful selection of the appropriate capacitance and inductance values, that ensures the ripple is reduced to the desired minimum.

For an inductor L to be designed, with $I_L$ current flowing through it, to reduce the ripple current to within 2% of the inductor peak-to-peak current, then $I_{pp} = 0.02 \times I_{pp}$. The design procedure is show below:

Value of K used is k = 0.5, which is the worst assumed case.

Inductor current $I_L = \frac{I_o}{1-k}$ but $V_o = \frac{V_i}{1-k} = \frac{15}{1-0.5} = 30V$ then $I_o = \frac{V_o}{R_L} = \frac{30}{15} = 2A$

$$I_L = \frac{I_o}{1-k} = \frac{2.00}{0.5} = 4.00A$$

Also, $a = \frac{I_{pp}R_{in}}{V_o} = \frac{I_{pp}r_L}{V_o} = \frac{0.02 \times 4.00 \times 0.2}{30} = 5.33 \times 10^{-4}$

We also have $fL = \frac{R_{in}}{2In[(1+a)/(1-a)]}$

If we take f = 5kHz $L = \frac{0.2}{2In[(1+5.33\times10^{-4})/(1-5.33\times10^{-4})]} \times \frac{1}{5\times10^3} = \frac{0.2}{10.67} = 18.75mH$

For the output(capacitor) ripple voltage $\frac{\Delta V_o}{V} = 0.02 = \frac{kT_p}{R_L C_o}$

Where k = 0.5 (assumed), $T_p$ = 0.2ms, $R_L$ = 15Ω then we will have

$$C_o = \frac{0.5 \times 0.2 \times 10^{-3}}{15 \times 0.02} = 333.3\mu F$$

Substituting this values of the designed inductance and capacitance into the MATLAB program and running with the assumed value of k = 0.5 gives figure 9 and 10. Evaluating the peak-to-peak ripple voltage

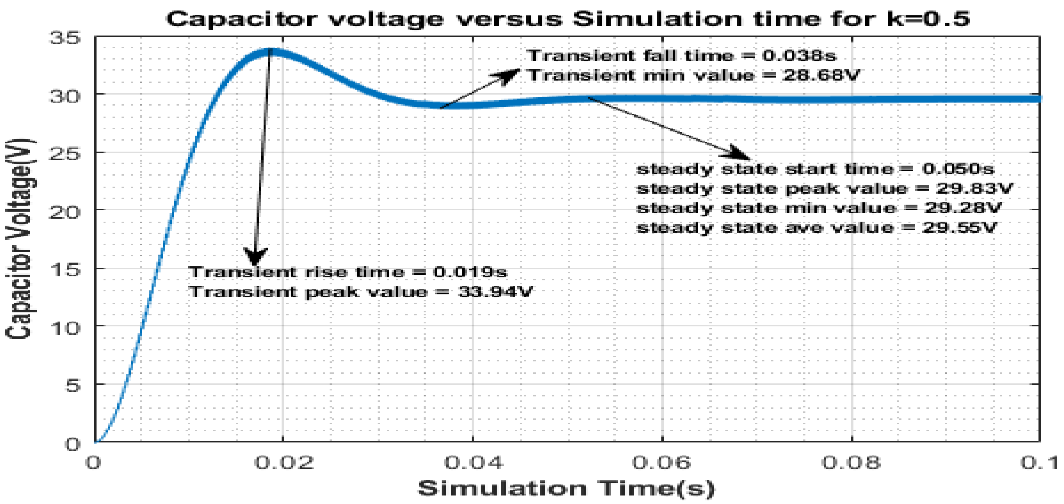


Figure 9: transient and steady state response of the capacitor voltage with improved $C_o$ Value

For Voltage $V_{max} = 29.83V, V_{min} = 29.28V$

$$V_{pp} = V_{max} - V_{min} = 29.83 - 29.28 = 0.55V$$

Average voltage, $V_{av}=\frac{V_{\max}+V_{\min}}{2}=\frac{29.83+29.28}{2}=29.55V$

Percentage (%) of average voltage $=\frac{V_{pp}}{V_{av}}\times 100\%=\frac{0.55}{29.55}\times 100\%=1.86\%$

Output Voltage, $V_o=\frac{V_i}{1-k}=\frac{15}{1-0.50}=30V$

Efficiency $=\frac{V_{av}}{V_o}\times 100\%=\frac{29.55}{30}\times 100\%=98.5\%$

From the figure 9 it is observed that at steady state the peak-to-peak voltage ripple gives 0.55V, this value is approximately 1.86% of the average value at steady state. Also, the voltage efficiency evaluated is 98.5%.

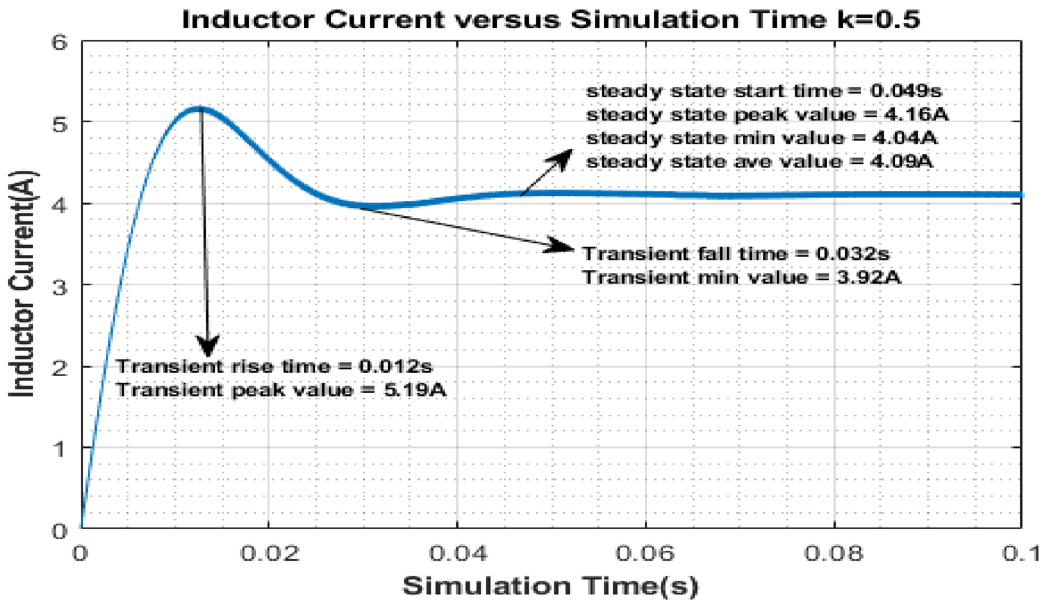


Figure 10: transient and steady state response of the inductor current with improved L Value

current $I_{\max}=4.14A, I_{\min}=4.04A$

$I_{pp}=I_{\max}-I_{\min}=4.14-4.04=0.1A$

Average current, $I_{av}=\frac{I_{\max}+I_{\min}}{2}=\frac{4.14+4.04}{2}=4.09A$

Percentage (%) of average current $=\frac{I_{pp}}{I_{av}}\times 100\%=\frac{0.1}{4.09}\times 100\%=2.44\%$

Output current, $I_o=\frac{V_o}{R_L}=\frac{30}{15}=2A$

$$I_L = \frac{I_o}{1-k} = \frac{2}{1-0.50} = 4A$$

$$\text{Efficiency } = \frac{I_{av}}{I_L} \times 100\% = \frac{4.09}{4} \times 100\% = 102\%$$

Similarly, from figure 10, peak-to-peak ripple current 0.1A, this gives a 2.44% of the average value at steady state. Furthermore, the efficiency evaluated is ideally.

**Comment on the simulation method**

The Runge-Kutta method is a classical 4th order method used for solving differential equation

This is the most often used method for solving a differential equation. It requires four evaluations of the right hand per step $\Delta t$. For solving differential equation, we use

$$k_1 = \Delta t f(t_k, t_k)$$
$$k_2 = \Delta t f(t_k + \frac{1}{2}\Delta t, y_k + \frac{1}{2}k_1)$$
$$k_3 = \Delta t f(t_k + \frac{1}{2}\Delta t, y_k + \frac{1}{2}k_2)$$
$$k_4 = \Delta t f(t_k + \Delta t, y_k + k_3)$$
$$y_{k+1} = y_k + \frac{1}{6}(k_1 + 2\times k_2 + 2\times k_3 + k_4)$$

However, the disadvantages of using this method include error estimation ability and higher computational time [2].

**Boost-Converter + PV Panel**

The introduction of the PV panel into the boost converter circuitry is designed using a capacitor across the terminal of the PV, representing as a voltage source $V_i$, this helps to minimize the input voltage ripples. The design also includes a voltage controlled current source $(I_{pv}(V_i))$, representing the short-circuit current of the PV circuitry generated.

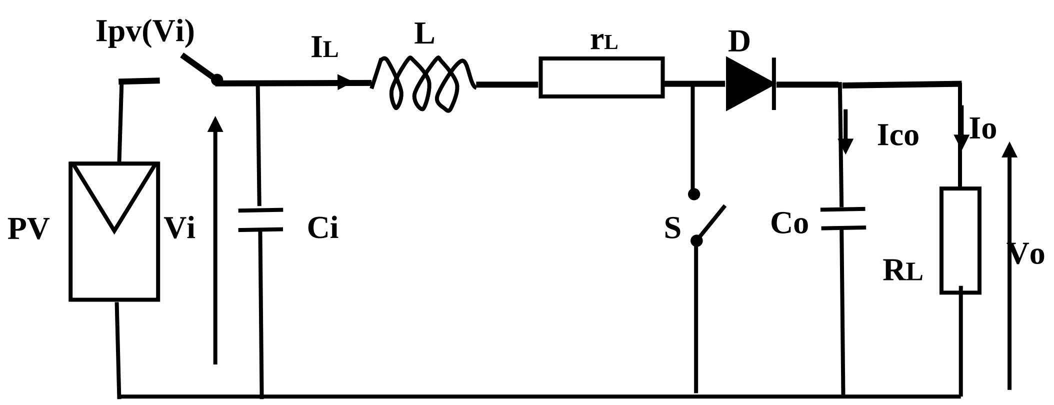


Figure 10: PV + Boost-Converter Model

Figure 10 is a boost (step-up) converter whose input is been fed by a PV panel when the input (PV section) is not supplying current to the load, that is when the above circuit is in its “OFF” state, the equivalent circuit yield is shown in figure 11. Similarly, during the “ON” state the equivalent circuit produce is shown in figure 12

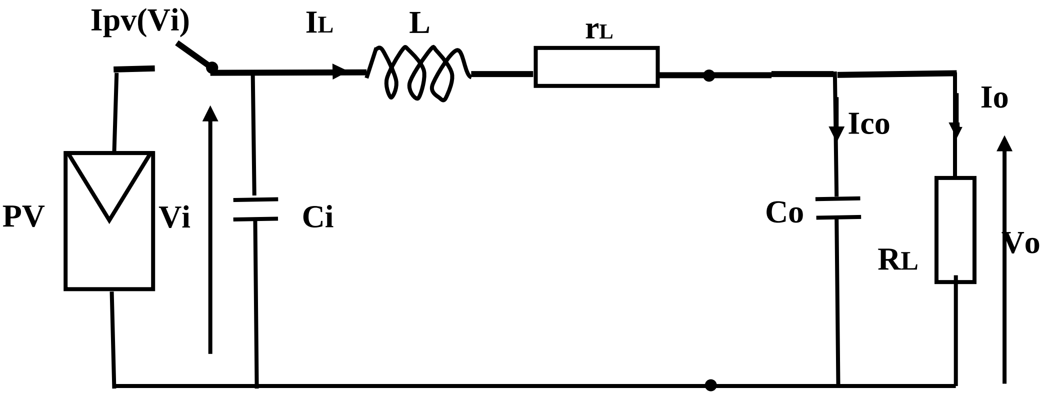


Figure 11: The PV + Boost-Converter during “OFF” state

From the boost converter input

$V_i = r_L I_L + L\frac{\partial I_L}{\partial t} + V_o + I_{pv}(V_i)r_i$ re-arranging the equation gives

$$\frac{\partial I_L}{\partial t} = \frac{1}{L}V_i - \frac{r_L}{L}I_L - \frac{1}{L}V_o - \frac{1}{L}I_{pv}(V_i)r_i$$

Considering the input current

$I_L = I_{pv}(V_i) - C_i\frac{\partial V_i}{\partial t}$ re-arranging we have $\frac{\partial V_i}{\partial t} = -\frac{1}{C_i}I_L + \frac{1}{C_i}I_{pv}(V_i)$

Similarly, the output part of the circuit gives

$I_L = C_o\frac{\partial V_o}{\partial t} + \frac{V_o}{R_L}$ re-arranging gives $\frac{\partial V_o}{\partial t} = \frac{1}{C_o}I_L - \frac{1}{C_oR_L}V_o$

Combining equation, a and b to form the state space equation

$$\begin{bmatrix} \dot{V}_i \\ \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{C_i} & 0 \\ \frac{1}{L} & -\frac{r_L}{L} & -\frac{1}{L} \\ 0 & \frac{1}{C_o} & -\frac{1}{C_oR_L} \end{bmatrix} \begin{bmatrix} V_i \\ I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{C_i} \\ 0 \\ 0 \end{bmatrix} I_{pv}(V_i)$$

The “ON” state for the PV + Boost-Converter model is presented in figure 12

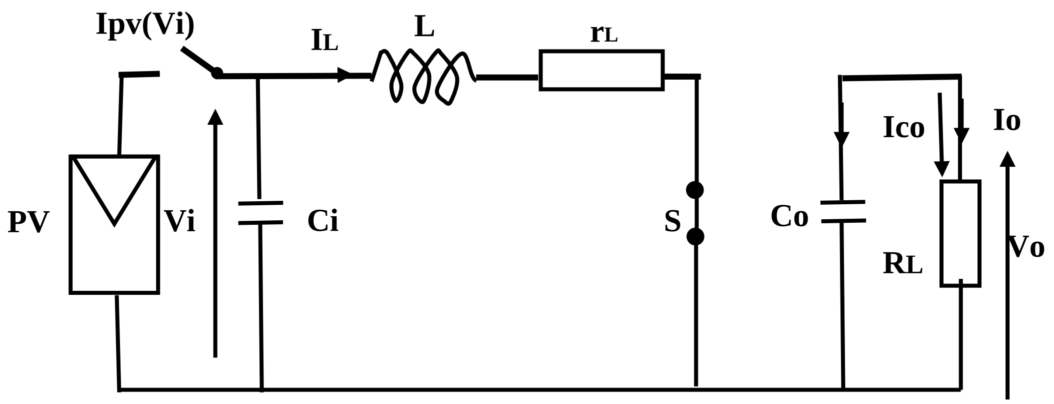


Figure 12: The PV + Boost-Converter during “ON” state

$V_i = (I_{pv}(V_i) - I_L)r_i + r_L I_L + L\frac{\partial I_L}{\partial t}$ re-arranging the equation gives $\frac{\partial I_L}{\partial t} = \frac{1}{L}V_i + \frac{r_L}{L}I_L$

Also, considering the input current

$I_L = I_{pv}(V_i) - C_i\frac{\partial V_i}{\partial t}$ re-arranging we have $\frac{\partial V_i}{\partial t} = -\frac{1}{C_i}I_L + \frac{1}{C_i}I_{pv}(V_i)$

The output section of the circuit gives, from KCL $I_{co} = I_o$ substituting gives

$-C_o\frac{\partial V_o}{\partial t} = \frac{V_o}{R_L}$ re-arranging the equation yields $\frac{\partial V_o}{\partial t} = -\frac{1}{C_o R_L}V_o$

The state space representation for the “ON” state is shown below

$$\begin{bmatrix} \dot{V}_i \\ \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{C_i} & 0 \\ \frac{1}{L} & -\frac{r_L}{L} & 0 \\ 0 & 0 & -\frac{1}{C_o R_L} \end{bmatrix} \begin{bmatrix} V_i \\ I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{C_i} \\ 0 \\ 0 \end{bmatrix} I_{pv}(V_i)$$

Combining both state space equations for “OFF” and “ON” states is shown below

$$\begin{bmatrix} \dot{V}_i \\ \dot{I}_L \\ \dot{V}_o \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{C_i} & 0 \\ \frac{1}{L} & -\frac{r_L}{L} & \frac{-(1-s)}{L} \\ 0 & \frac{(1-s)}{C_o} & -\frac{1}{C_o R_L} \end{bmatrix} \begin{bmatrix} V_i \\ I_L \\ V_o \end{bmatrix} + \begin{bmatrix} \frac{1}{C_i} \\ 0 \\ 0 \end{bmatrix} I_{pv}(V_i)$$

Where s=1 for “ON” state an s=0 for “OFF” state

The PV + boost converter was simulated on MATLAB and the and the plot generated is shown below, the curve was plotted with k=0.55

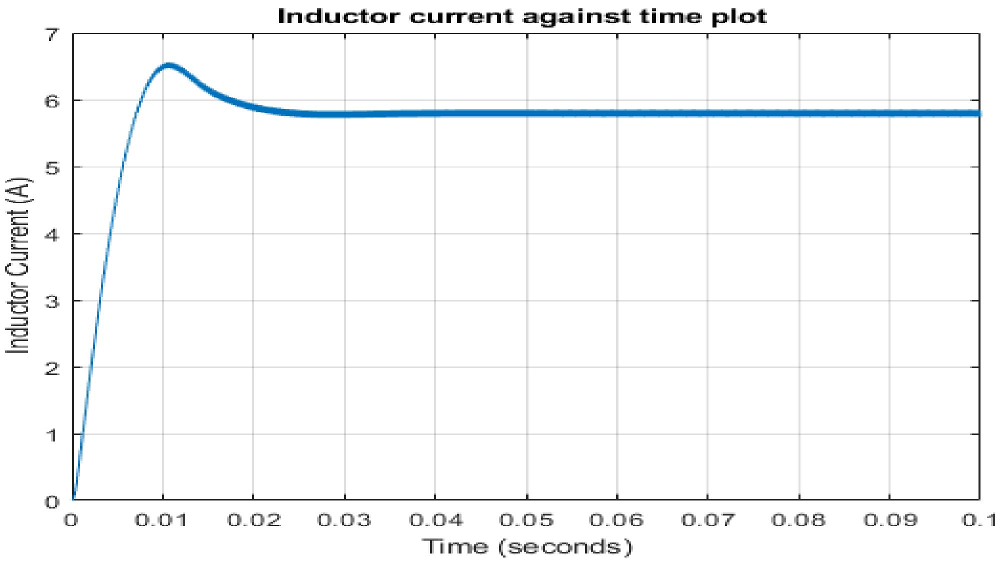


Figure 13: Inductor current versus simulation time with k=0.55

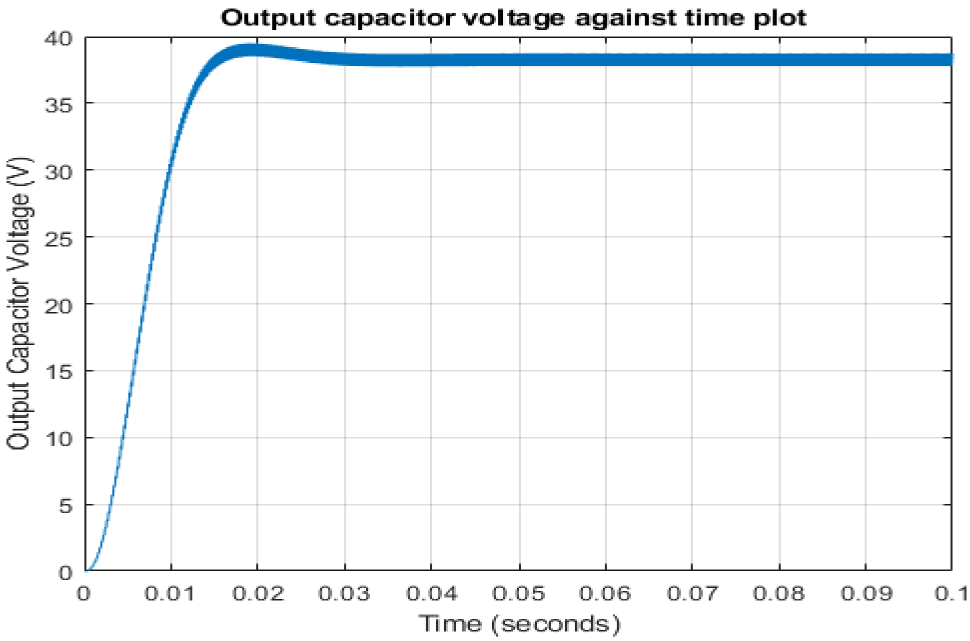


Figure 14: Capacitor Voltage versus simulation time with k=0.55

The flowchart and MATLAB code for the PV + Boost Converter circuit are shown in the appendix section.

**Conclusion**

The plot generated in figure 13 and 14, is different from earlier plots because of the generated input voltage in each case has a corresponding PV short-circuit current computed using Newton-Raphson algorithm. In order to achieve maximum voltage and current in figure 14 and 13 respectively, various values of k were tested, until we got k = 0.55.

# References

[1] "Section 2 - PV + Comverters," School of Electronics and Electrical Engineering, Leeds, [Online]. Available: https://minerva.leeds.ac.uk/bbcswebdav/pid-4795120-dt-content-rid-9733105_2/xid-9733105_2. [Accessed 20 11 2017].

[2] "Numerical Methods," www-personal.umich.edu, [Online]. Available: http://www-personal.umich.edu/~weilu/me574/2/group4/sub/method.htm. [Accessed 19 11 2017].

**Appendix**

Program listing with detailed comment for Boost-Converter is given as shown

```
% MatLab Simulation program for a PV+Boost Converter
%Written Dagogo Orifama, 201177661
%Electronics and Electrical Engineering, University of leeds

%Initiallising the given parameters
Vi = 15; %input voltage
fs= 5000;  %switching frequency
CO = 333.3*1e-6;  %Output capacitor
L= 18.75*1e-3;  %Inductor
rL=0.2;  %%resistance of the inductor
RL=15;   % Load resistance
tp = 1/fs; %switching period
dt = 4*1e-6;  %Sampling time step
k = 0.5;  %Switching duty ratio
IL = 0;     % Initial load current
V0= 0;      %Initial output voltage
x = [IL;V0];    %X-component of the state space model
B = [1/L; 0];   %B-component of the state space model
U = Vi;         %U-component of the state space model

VCSAVE = [];     %Emply array to store voltaga
ILSAVE = [];    %EMpty array to store current
sim_time = []; %Empty arrat to

for t = 0: dt : 500*tp    %switching period interval

    tDiff = mod(t,tp);    %compare t and tp and compute  the difference

    if tDiff < (k*tp)      % Condition for switch to be on
        S = 1;           %Switch ON
    else
        S = 0;           %switch OFF
    end

    A = [-rL/L -(1-S)/L; (1-S)/CO  -(1/(CO*RL))]; % A component of the
state space equation
```

```
    % Start of fourth order Runge-kutta Method
    k1 = dt*(A*x+B*U); %First order term
    k2 = dt*(A*(x+0.5*k1)+(B*U));  %Second order term
    k3 = dt*(A*(x+0.5*k2)+(B*U));  %Third order term
    k4 = dt*(A*(x+k3)+(B*U));      %Fourth order term
    x= x + ((k1+2*k2+2*k3+k4)/6);    % Formulae for finding x

    IL = x(1);                     % Current component of the x result
    V0 = x(2);                     %Voltage component of the x result

    VCSAVE = [VCSAVE V0];            %saves the array of Capacitor 
voltages
    ILSAVE = [ILSAVE IL];          %saves the array of Inductor currents
    sim_time = [sim_time t];       %saves array of simulation time

end

figure (1)                 %Specify figure
plot(sim_time,VCSAVE)    % Plot voltage versus simulation time
title('Capacitor voltage versus Simulation time'); %set graph title
ylabel('Capacitor Voltage(V)'); %set y-axis label
xlabel('Simulation Time(s)');   %set x-axis title
grid on
grid minor

figure (2)
plot(sim_time,ILSAVE)  %Plot Inductor current versus simulation time
title('Inductor Current versus Simulation Time');  %set graph titile
ylabel('Inductor Current(A)'); %Set y-axis label
xlabel('Simulation Time(s)');  %set x-axis lable
grid on
grid minor
```

Program listing with detailed comment for PV+Boost-Converter is given below

```
% MatLab Simulation program for a PV+Boost Converter
%Written Dagogo Orifama, 201177661
%Electronics and Electrical Engineering, University of Leeds

Cin = 200e-6;         %input voltage
Fs= 5000;             %switching frequency
Cout = 300e-6;        %output capacitance
L= 0.0187;            %inductance
rL=0.2;               %inductor resistance
RL=15;                %load resistance
k = 0.55;             %duty ratio

Ts = 1/Fs;           %switching period
dt = 4e-6;           %time step or increment
I_L = 0;             %inductor current
Vout= 0;             %initial output voltage
Vin = 0;
x = [Vin;I_L;Vout]; %x component of state space matrix
B = [1/Cin; 0; 0]; %B component of state space matrix

%Acquring PV information
Ai = 1.72;                   %Ideality factor of the diode
```

```
q = 1.6*1e-19;                %Electron charge
kBoltz  = 1.380658*1e-23;     %Boltzmann constant
Eg= 1.1;                      %Energy gap
Ior = 19.9693*1e-6;           %Reverse saturation current at Tr
Iscr = 3.3;                   %Short circuit current at Tr
ki = 1.7*1e-3;                %temperature coefficent of circuit current
ns = 40;                      %number of series cell
np = 2;                       %number of parallel cell
Rs = 5*1e-5;                  %series resistance
Rp = 5*1e5;                   %parrallel ressistance
Tr = 301.18;                  %Reference temperature
acc = 0.001;                  %Error limit for obtaining Ipv using Newton
terminiation condition

G = 1;
Ta=25;


%compute the remaining PV parameters needed for the simulation
Tc = Ta + 0.2*G + 273.18;                                 %Evaluating
the cell temperature
Isc = (Iscr + ki*(Tc-Tr)) * G;                            % Evaluating
short-circuit current
Is = Ior * (Tc/Tr)^3 * exp( q*Eg*(1/Tr - 1/Tc) / (kBoltz*Ai)) ;
%Evaluating the diode leakage current

Pin_array = [];
Pout_array = [];
Vin_sim = [];
VCSAVE = [];     %array to hold the capacitor voltage
ILSAVE = [];    %array to hold the inductor voltage
sim_time = [];  %array to hold the sampling time

V = 0;                  %set V to zero

%iterate through sampling time of 0 to 500Tp
for t = 0: dt : 500*Ts

    error = 50;
    I = Isc;            %set I to Isc, since this current, Isc is one of
the extreme operating points of the PV
%condition to check that the PV current does not go beyond zero
    while error > acc

     fI = I - np*Isc + np*Is*(exp(q*(V/ns + I*Rs/np)/(Ai*kBoltz*Tc))-1) +
(V*np/ns + I*Rs)/Rp; %re-evaluate the function using the new I value
     dfI = 1 + Rs/Rp + ((q*Is*Rs)/(Ai*kBoltz*Tc)) * exp(q*(V/ns +
I*Rs/np)/(Ai*kBoltz*Tc));          %re-evaluate the function derivative
using the new I value

        In = I - (fI/dfI);  %obtain a new I value

        error = abs((In - I)/I); %re-compute the error
          I = In;
       end
    Ipv = In;
    U = Ipv;
    tDiff = mod(t,Ts); %get the instantaneous time by comparing t with Tp
```

```
    s = 0;     %variable to hold either the ON or OFF state conditions of
the circuit
    if tDiff < (k*Ts)
        s = 1;            %set switch to ON state
    else
        s = 0;    %set switch to OFF state
    end

    A = [0 -1/Cin 0; 1/L -rL/L -(1-s)/L; 0 (1-s)/Cout -(1/(Cout*RL))]; %A
component of state-space equation

    % Start of fourth order Runge-kutta Method
    k1 = dt*(A*x+B*U); %First order term
    k2 = dt*(A*(x+0.5*k1)+(B*U));  %Second order term
    k3 = dt*(A*(x+0.5*k2)+(B*U));  %Third order term
    k4 = dt*(A*(x+k3)+(B*U));      %Fourth order term
    x= x + ((k1+2*k2+2*k3+k4)/6);    % Formulae for finding x

    Vin = x(1);
    I_L = x(2);  %extract the inductor current
    Vout = x(3);    %extract the capacitor voltage
    V = Vin;
    Pin=Vin*I_L;
    Pout=(Vout^2)/RL;

    Vin_sim = [Vin_sim Vin];
    VCSAVE = [VCSAVE Vout]; %store the capacitor voltage for each time
step
    ILSAVE = [ILSAVE I_L];   %store inductor current for each time step
    sim_time = [sim_time t];    %store simulation time
    Pout_array=[Pout_array Pout];
    Pin_array=[Pin_array Pin];
end

Poutave=(max([Pout_array(end-50:end)]) + min([Pout_array(end-
50:end)]))/2;
Pinave=(max([Pin_array(end-50:end)]) + min([Pin_array(end-50:end)]))/2;

figure (1)              %Specify figure
plot(sim_time,VCSAVE)    % Plot voltage versus simulation time
title('Capacitor voltage versus Simulation time'); %set graph title
ylabel('Capacitor Voltage(V)'); %set y-axis label
xlabel('Simulation Time(s)');   %set x-axis title
grid on
grid minor

figure (2)
plot(sim_time,ILSAVE)  %Plot Inductor current versus simulation time
title('Inductor Current versus Simulation Time');  %set graph titile
ylabel('Inductor Current(A)'); %Set y-axis label
xlabel('Simulation Time(s)');  %set x-axis lable
grid on
grid minor
```

```
Start
    Initialize variables
    Initialize arrays VCSAVE, ILSAVE, sim_time,  acc, error
    Increment counter by sampling time dt
    Check
    Is error greater than acc
    Is tDiff greater than the period
     If YES, GO TO **, IF NO
     Evaluate new error
     Increment t by dt
    Check
    Is tDiff < (k*Ts)
    If yes, turn switch ON, i.e s = 1
    If NO, turn OFF switch, i.e, s = 0
    End check
    Evaluate values for k1, k2, k3 and k4
    Update values of x
     Assign component of x to current, IL and voltage, V0
 **  Update VCSAVE, ILSAVE AND sim_time arrays
   Plot sim_time versus VCSAVE and sim_time versus ILSAVE
End
```

Flowchart of PV + Boost -Converter

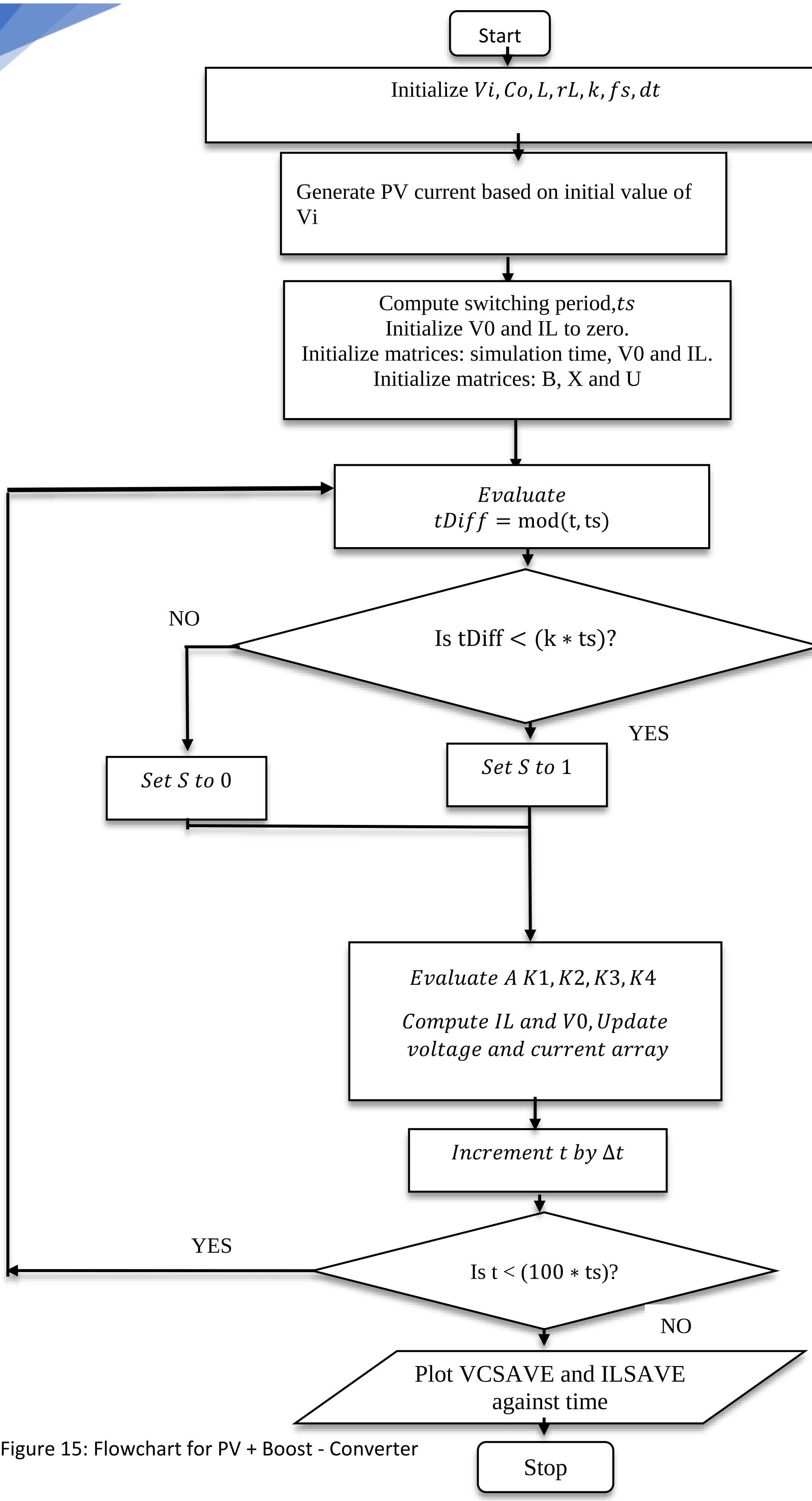


Figure 15: Flowchart for PV + Boost - Converter