AN ADVANCED TIME AVERAGING MODELLING TECHNIQUE FOR POWER ELECTRONIC CIRCUITS

by

Goce Jankuloski

A thesis submitted in conformity with the requirements for the degree of Master of Applied Science
Graduate Department of Electrical Engineering
University of Toronto

© Copyright 2014 by Goce Jankuloski
Abstract

An Advanced Time Averaging Modelling Technique for Power Electronic Circuits

Goce Jankuloski

Master of Applied Science
Graduate Department of Electrical Engineering
University of Toronto

2014

For stable and efficient performance of power converters, a good mathematical model is needed. This thesis presents a new modelling technique for DC/DC and DC/AC Pulse Width Modulated (PWM) converters. The new model is more accurate than the existing modelling techniques such as State Space Averaging (SSA) and Discrete Time Modelling. Unlike the SSA model, the new modelling technique, the Advanced Time Averaging Model (ATAM) includes the averaging dynamics of the converter’s output. In addition to offering enhanced model accuracy, application of linearization techniques to the ATAM enables the use of conventional linear control design tools. A controller design application demonstrates that a controller designed based on the ATAM outperforms one designed using the ubiquitous SSA model. Unlike the SSA model, ATAM for DC/AC augments the system’s dynamics with the dynamics needed for subcycle fundamental contribution (SFC) calculation. This allows for controller design that is based on an exact model.
Acknowledgements

This thesis would not have been possible without the support and mentorship from my supervisor, Peter W. Lehn. A lot of meetings, work and discussion went into this thesis. He was always helpful, supportive, and understanding. It was a great learning experience and unique to have the opportunity to learn from him.

I’d also like to thank my family, for the overwhelming support they have given me over my whole education, especially these past two years. It means a lot for me to culminate my Masters' study with this thesis, which I consider my best work to date.
# Contents

1 Introduction .................................................. 1
  1.1 Background ............................................... 1
  1.2 Literature Review ........................................ 5
  1.3 Thesis Objectives ......................................... 9

2 Converter Modelling ...................................... 11
  2.1 Average Modelling ....................................... 11
  2.2 SSA Modelling Inaccuracy ................................. 16
    2.2.1 Example ............................................ 17
  2.3 SSA Implementation Inaccuracy ........................... 19
  2.4 Summary ................................................ 24

3 DC/DC Modelling .......................................... 26
  3.1 Sliding Window Average ................................. 27
    3.1.1 Elimination of the Delayed Integrator State ....... 32
  3.2 ATAM Application Example ............................... 33
  3.3 Validation ............................................... 35
  3.4 Summary ................................................ 35

4 DC/DC Model Linearization and Control .................. 38
  4.1 Linearization ........................................... 39
4.2 Model Demonstration 42
  4.2.1 Open Loop Performance 42
  4.2.2 Closed Loop Performance 44
4.3 Summary 49

5 DC/AC Modelling 50
  5.1 Modelling Challenges 51
  5.2 Subcycle Fundamental Contribution 52
    5.2.1 Modifying SFC States to Memoryless 55
  5.3 Two Phase Inverter Modelling 59
    5.3.1 Decoupled Fourier Transform 60
  5.4 Model Demonstration 68
    5.4.1 Pure Sine Wave 70
    5.4.2 Oscillator with Negative Sequence 72
  5.5 Summary 73

6 Conclusion 76
  6.1 Future Work 78

7 Appendix 79
  7.1 Capless Buck Converter with Resistive Load 79
    7.1.1 SSA Model 80
  7.2 Capless Buck Converter with Motor Load 81
    7.2.1 New Model 82
    7.2.2 SSA Model 86

Bibliography 91
List of Tables

7.1 Circuit Parameters ............................................. 79
7.2 Circuit Parameters ............................................. 83
List of Figures

1.1 Simple model of a general power electronic converter. The power source, with the control inputs $S(t)$ drive the system output ........................................ 3

1.2 The 2-quadrant chopper converter topology. A clear distinction can be seen between the power source, control inputs, states, and output. ........ 4

1.3 The output current is sampled at interval $T$. Depending on the phase of the sampling, in this case the sample points align with the valleys of the ripple. ................................................................. 6

1.4 The feedback path for PCM control implementation. ....................... 7

1.5 The feedback path for SSA model for control design. ....................... 7

1.6 Digital control based on SSA design. ................................................. 8

1.7 The ideal proposed model for a digital PWM converter receives the duty cycle $d[k]$ and outputs the cycle average current $\langle i(t) \rangle_T$ .............. 9

2.1 Synchronous buck converter showed with general output load. The switches are operated in a complementary fashion. ........................................ 12

2.2 Circuit positions of the buck converter. .............................................. 13

2.3 A duty cycle disturbance of 0.1 is applied at $t = 0.2ms$. The output current response of the circuit is given in green. The circles represent the average output current as predicted by SSA. The plus (+) represents the average output current of the circuit over the last $T$ seconds. .............. 18
2.4 The bode plot of a simple first order LPF. The filter attenuates high frequency components well, while letting low frequency pass. However there is still small non-zero phase present at the low frequencies.

2.5 The bode plots for the slow and fast LPF. The slow LPF attenuates high order frequencies with greater factor than the fast LPF.

2.6 The output current response of the circuit is given in dashed green. The red triangles represent the filtered current by slower LPF. The blue circles represent the filtered current by faster LPF. The black pluses represent the average output current of the converter.

2.7 The faster LPF has relatively accurate transient response but also a steady state error.

3.1 The figure shows how the integral in (3.2) calculates SWA.

3.2 The open loop response of the current to a duty cycle disturbance. The ATAM precisely predicts the simulated current’s average waveform.

4.1 The models’ open loop response to a disturbance in the duty cycle.

4.2 The test converter topology. There is no output capacitor – it is extracted out to the output side.

4.3 The simulation block diagram. The controller acts on the small-signal average output current error, and outputs a small-signal duty cycle. The operating point duty cycle $D$ is added to this before being fed into digital PWM. Digital PWM drives the switches of the converter.

4.4 The average output current response for a load decrease disturbance. The SSA model based controller has higher overshoot and steady state error; LATAM based controller has smaller overshoot and no steady state error.

5.1 The integrator in (5.3) leads to equal sub-integrals which form a straight line for a perfect oscillator.
5.2 Shown here is how the SFC vector is calculated for the second switching period. The SFC of the previous switching event is rotated and added to the fundamental contribution vector of the second period. \( \frac{T_{ac}}{T} = 4 \) 58

5.3 The neutral clamped two phase inverter topology is shown here. 59

5.4 Shown here is how the different switching times in both phases can lead to many different circuit states. 61

5.5 \( \beta \) lags \( \alpha \) by \( \frac{\pi}{2} \) or \( \frac{1}{4} \) of line period. 63

5.6 Figure shows how the duty cycle is calculated for SPWM for \( \alpha \) subsystem. Here \( m_f = 2 \left( \frac{T_{ac}}{T} \right) \) for visualization purposes. For each switching period there is on-off-on period. 66

5.7 Flow chart of the new modelling approach for DC/AC inverter 69

5.8 The evolution of the signal’s components over several cycles. 70

5.9 The SFC of the perfect oscillator is constant in all subintervals as represented by the constant point in this plot. 71

5.10 The evolution of the signal’s components over several cycles. The differing in amplitude represents an unbalanced signal. 72

5.11 The evolution of the fundamental vector over several cycles. It traces out circle around the centre point of 0.01875 + j0. This is the subinterval fundamental contribution vector. 74

7.1 The test converter topology. There is no output capacitor. 80

7.2 The test converter topology. There is no output capacitor. 82

7.3 The closed loop structure for the given example. Plant and controller are all in discrete time. The output of the plant is the sampled average output current. 85

7.4 The root locus of the closed loop. The controllable poles are all inside the unit circle. 86
7.5 The bode plot of the uncompensated closed loop. As can be seen the system has stable margins even with unity feedback. 88

7.6 The output response to a step disturbance in the input. As it can be seen the output never reaches zero steady state error - it settles to a steady state error. 89

7.7 The step response for the newly compensated system. The output returns to zero steady state error withing 0.4 ms. 90
Chapter 1

Introduction

Power converters are an important component of any modern electrical system, be it a large scale utility network or a megawatt scale power supply for cell phones. As such, a detailed analytical understanding of the operation of power converters is paramount.

The main goal of power converters is to allow for the transfer of power, between two systems, with minimal losses and predictable performance [6]. From the perspective of performance, the converter needs to ensure stable operation of the resulting interconnected system and be robust to disturbances that might exist within the system [3].

An accurate mathematical model of a converter is essential for the design of controllers that meet the specified performance criteria. Such models should allow the use of different control design techniques, that aid the designer in meeting the specified performance criteria. Current modelling approaches have performance limitations which will be identified later in this chapter. These limitations motivate the need for the development of an improved modelling approach, which is the main objective of this work.

1.1 Background

A power electronic converter is an electrical circuit used to transfer power between two electrical systems. It may interconnect two DC systems of different voltages, in which
Chapter 1. Introduction

In cases where the converter is typically referred to as a DC/DC converter. Alternatively it may transfer power between a DC and AC system, in which case the converter is typically referred to as a DC/AC converter.

Power electronic converters typically utilize a switching network in conjunction with passive components to achieve average power transfer between the two systems. There are many different classifications of power converters, broadly categorized as hard switched converters and soft switched converters. This thesis will focus on hard switched converters as they are most widely used.

To achieve different output voltages, the duty cycle of the converter is changed. The duty cycle is defined as the amount of time a particular switch is closed during a single switching period. This technique is commonly referred to as Pulse Width Modulation (PWM).

As previously stated, a converter is comprised of a switch network and a set of passive components. At any point in time, each switch is assumed to be either ideally “Off” or “On”. Thus, the switch either blocks voltage with zero current (“Off”), or conducts current with zero voltage drop (“On”). The state of the switch network can be represented by the vector $S(t)$:

$$
S(t) = \begin{bmatrix}
S_1(t) \\
S_2(t) \\
\vdots \\
S_k(t)
\end{bmatrix}
$$

(1.1)

where:

$$
S_i(t) = \begin{cases}
1, & \text{if closed} \\
0, & \text{otherwise}
\end{cases}
$$

(1.2)

$S(t)$ is the switching state vector, and each element represents the state of a particular switch, called its ‘switching function’. At a given time, $t_o$, $S(t_o)$ gives the switch state vector at instant $t_o$. From a control systems perspective the converter can be viewed as an
Figure 1.1: Simple model of a general power electronic converter. The power source, with the control inputs $S(t)$ drive the system output input/output block, between the switching state vector (input) and the current/voltage of interest (output). Figure 1.1 shows this block model of a general converter.

Depending on the switching state vector, the converter would take on a number of different circuit configurations throughout one switching period. However, a power converter generally operates with complementary configured switches. For example, a two switch converter like a two quadrant chopper, as shown in Figure 1.2 would have $S_2$ operate in compliment to $S_1$ at all times. In such a case, the converter would be controlled by a single variable, $d(t)$, which is the duty cycle of the converter. By convention this defines the length of time that $S_1$ is in the “On” state, thus defining the switch state vector for the entire switching period.

For the example of the two quadrant chopper only two different switching state vectors exist throughout the entire switching period. The switches thus create two distinct linear circuits during the switching period. To understand the dynamics of the converter, the evolution of the converters states, such as capacitor voltage or inductor current, must be modelled.

The objective of a converter model is to capture the dynamic relations between control inputs, power sources, and state variables/outputs. This high-level perspective, though
trivial, is important in establishing a framework for discussion and comparison of existing power converter modelling techniques.

For the two-quadrant chopper as shown in Figure 1.2, the output current is defined as the output of the system which needs to be controlled. Moreover the switching state vector is given:

$$\mathbf{S}(t) = \begin{bmatrix} S_1(t) \\ S_2(t) \end{bmatrix}$$  \hspace{1cm} (1.3)

It is assumed that the switches of the converter are bidirectional, and ideal. To ensure Continuous Conduction Mode (CCM) under all conditions, $S_2(t)$ and $S_1(t)$ are operated in a complementary fashion:

$$S_2(t) = \bar{S}_1(t)$$  \hspace{1cm} (1.4)

It is the PWM of the converter that maps the duty cycle of the primary switch, $d(t)$, to the switching function of the same switch. This is a linear mapping and allows the
modelling to focus on $d(t)$ as the input to the converter for all intents and purposes.

### 1.2 Literature Review

Many different modelling techniques have been established in literature [6, 9, 13, 16]. While there are many different approaches to modeling a converter as in Figure 1.1, there are two common methods: predictive control modelling (PCM), and state space averaging (SSA) [6]. As will be seen, it is the short-comings of each of these approaches that is the motivation for this thesis.

#### Discrete-Time Converter Models

Discrete-Time Converter modelling is a well researched topic [2, 7, 14, 21, 27]. Predictive Control Modelling (PCM) as it is also known, is based on a future predictor of the behaviour of the states of the converter (currents/voltages), given known inputs [21]. Since PCM is implemented digitally, it is required that its converter model be fully discretized too. PCM control, samples the states of the system, and computes an action that is discrete with sampling time $T$ (switching period of the converter) [14]. Figure 1.3 shows the result of sampling one possible waveform. The alignment of the sampling with the end of each switching period is shown; but such alignment is not easy to achieve, and any phase shift can introduce error in modelling. Because of this sampling approach, PCM control is straightforward.

PCM takes in the duty cycle $d(kT)$ as the control input, which provides the switching function of switch 1. It is a discrete-time model and as such computes the corresponding output current $i(kT)$ [21]. The input to output transfer function is given by:

$$P(z) = \frac{I(z)}{D(z)} \quad (1.5)$$

where $D(z)$ is the input duty cycle in $z$-domain, and $I(z)$ is the output current in $z$-
Figure 1.3: The output current is sampled at interval $T$. Depending on the phase of the sampling, in this case the sample points align with the valleys of the ripple.

domain. The controller, $C_{PCM}(z)$, receives the error between the current $i(kT)$ and the reference value, and predicts which duty cycle $d((k+1)T)$, will eliminate the error in a finite number of switching periods.

Figure 1.4 shows a common implementation of a digital controller. A controller with modern digital PWM is assumed. The ‘PWM Converter’ block receives a digital input $d(kT)$, which is implemented throughout the subsequent switching period. However, while the input to the block is a digital signal the output of the converter is the continuous time current signal. This current is merely discretized by an A/D block (without anti-aliasing) to provide the precise digital signal $i(kT)$.

The major disadvantage of this PCM control model method is aliasing [21]. Because the current is being sampled – measured instantaneously at regular intervals – the absence of anti-aliasing filter propagates any measurement noise (of any frequency) through the feedback path [7]. This is the main drawback of the PCM method and why, despite the great theoretical performance, this approach has limited application [7].
Chapter 1. Introduction

State Space Averaging

The most widely used modeling method for PWM converters is SSA. It is a small-signal modeling of the converter [5, 1, 15, 17, 20]. SSA exploits the assumption that for the purpose of control, the high frequency dynamics, i.e., switching voltage and current ripple for example can be neglected [6, 23]. That is, only the net change of the states like the current/voltage over a single switching period is important [17]. As long as that information is retained and modeled then presumably the model will be accurate enough for the intended control design.

Refererring to the two-quadrant chopper example, Figure 1.2, the SSA model can be
summarized as in Figure 1.5. The converter model receives a duty cycle command, and computes the corresponding average of the output current. The average in these cases is defined as:

$$\langle x(t) \rangle_T = \frac{1}{T} \int_{t-T}^{t} x(\tau)d\tau$$ (1.6)

namely, it is the average over a single switching period. In the presented form, it can be considered a sliding or moving average.

After continuous time control design the control $C(s)$ is converted to discrete time and implemented as shown in Figure 1.6. As can be seen, the PWM converter again acts as D/A block. More importantly, there is a feedback low pass filter (LPF) that emulates the averaging assumption of the output current as in (1.6).

With sufficient low pass filtering this implementation is free of aliasing problem, and the solution is generally preferred over PCM [10, 19, 24]. SSA model has also been extended for use with resonant converters [26]. However, as will be shown in this thesis, the SSA model does not accurately model the averaging of its states. Further low pass filtering design leads to tradeoffs between steady state and transient performance [18, 22].
Chapter 1. Introduction

Figure 1.7: The ideal proposed model for a digital PWM converter receives the duty cycle $d[k]$ and outputs the cycle average current $\langle i(t) \rangle_T$

1.3 Thesis Objectives

As can be seen from Figure 1.7, however, what is really needed is a digital model that takes in $d[k]$ and outputs the true cycle average current. This proposed model offers coherence between design and implementation and its development is the main objective of the thesis. In implementation, the measured current from the converter would have to be time averaged as per (1.6), and fed-back to the digital controller. This conversion is suitably called Sliding Window Averaging (SWA). This is similar to LPF in SSA, but in this case its dynamics would be included in the model used to design the controller.

When comparing it with the PCM and SSA, the new model removes the aliasing issues present in PCM while also modelling all filtering dynamics unlike SSA. The proposed model follows the logic of SSA. As a result, SSA modelling approach will be discussed in greater detail. Its derivation will be discussed in the next chapter, along with how its lack of full dynamic modelling presents a limitation to the accuracy of the model.

The main objective of this thesis is the development of the newly proposed model. The focus of the development of the model is on DC/DC and DC/AC converters. These two groups of converter topologies are the most commonly used topologies. The accuracy of the developed model is demonstrated and compared against the SSA model.

Chapter 3 is focused on the development of the new model on a general DC/DC
topology. Leveraging, DC/DC converter characteristics, a linear time variant (LTV) model is developed. The methodology is outlined, and a specific example is used to demonstrate the open loop accuracy of the model.

In Chapter 4, the newly developed model for DC/DC converters is used for control design applications, by first undergoing linearization. It is shown that the new linearized model is still more accurate than the SSA model. A time domain simulation also shows the performance improvement by using the newly proposed model.

In Chapter 5, a DC/AC application of the general structure of the new model is examined. The goal is to translate the averaging concept for DC/DC converters, in calculating the fundamental component of the measured AC current, during a subcycle. Through a special modification it is shown that calculating the subcycle fundamental contribution is possible, and these dynamics can easily be augmented in the existing system dynamics.

With such objectives, this thesis is an attempt at bringing coherence between design and implementation of controllers for power converters. This is a gap in the present modelling approaches that has not been addressed properly. With the newly developed modelling, future work can focus on designing and implementing better controllers, thus allowing for better performance out of power converters.
Chapter 2

Converter Modelling

State Space Averaging is the most widely used modelling technique for averaging fixed switching frequency power converter circuits for the purposes of control design. The SSA model is a linearized model that can leverage popular linear control techniques. In this chapter a brief summary of the modelling approach will be given. As will be shown, the assumptions used in SSA cause discrepancies between the model and physical converter behaviour. This further leads to inaccurate control design, thus impacting the performance of the converter.

2.1 Average Modelling

For any given converter, its dynamics can be described by piecewise linear differential equations relating how capacitor voltage and inductor current change with respect to time. For example, a synchronous buck converter (Figure 2.1) can be used to illustrate the difficulty in modelling its switching dynamics.

The synchronous buck converter has two states, namely the inductor current and capacitor voltage. The dynamics of these two elements depends on the switching action of the converter, but at any given instance of time only two differential equations are
The previous equations are linear. Assuming PWM is used for switch modulation, with period \( T \) (or frequency \( F_{sw} \)), then per previous notation the switching state vector \( \mathbf{S}(t) \) is defined. The circuit topology thus changes depending on the value of \( \mathbf{S}(t) \). Figure 2.2 shows the two possible switch conditions, based on the assumption of continuous conduction mode operation.

The dynamics of the two states depend directly on the switching state vector \( \mathbf{S}(t) \). Since it is a piecewise linear function, the dynamics of the states are also piecewise linear:

\[
L \frac{d}{dt} i_L(t) = \begin{cases} v_g(t) - v_C(t), & \mathbf{S}(t) = 1 \\ -v_C(t), & \mathbf{S}(t) = 0 \end{cases} \tag{2.3a}
\]
Chapter 2. Converter Modelling

(a) The buck converter when $S(t) = 1$; primary switch is closed.

(b) The buck converter when $S(t) = 0$; primary switch is open.

Figure 2.2: Circuit positions of the buck converter.
\[
C \frac{d}{dt} v_C(t) = \begin{cases} 
    i_L(t) - i_o(t), & S(t) = 1 \\
    i_L(t) - i_o(t), & S(t) = 0
\end{cases} 
\] (2.3b)

In the case of the inductor current, the switching state affects its dynamics. However, the capacitor voltage is a state which is unaffected by the switching state dynamics. Modelling piecewise linear dynamics is a complicated task, and it is the main challenge of power converter modelling.

SSA is the most commonly used approach to converter modelling. SSA relies on separating state’s dynamics into low frequency and high frequency dynamics.

\[
i_L(t) = \langle i_L(t) \rangle_T + i_L^{hof}(t) 
\] (2.4a)

\[
\langle i_L(t) \rangle_T = \frac{1}{T} \int_{t-T}^{t} i_L(\tau)d\tau 
\] (2.4b)

The higher order frequency component, \( i_L^{hof}(t) \), is due to the switch transitions. The low frequency component, \( \langle i_L(t) \rangle_T \), contains the net or ‘average’ change of the signal over a switching period \( T \). Since control involves tracking average power/current, and higher order frequency components are assumed not to affect the power exchange, the low frequency dynamics are the ones that are of interest. (2.1)-(2.2) can thus be written as:

\[
L \frac{d}{dt} \langle i_L \rangle_T = \langle v_L(t) \rangle_T 
\] (2.5a)

\[
C \frac{d}{dt} \langle v_C \rangle_T = \langle i_C(t) \rangle_T 
\] (2.5b)

Combining these equations with (2.3a) and (2.3b) yields:

\[
L \frac{d}{dt} \langle i_L \rangle_T = \langle d(t) \rangle_T [\langle v_g(t) \rangle_T - \langle v_C(t) \rangle_T] + (1 - \langle d(t) \rangle_T) [-\langle v_C(t) \rangle_T] 
\] (2.6a)

\[
C \frac{d}{dt} \langle v_C \rangle_T = \langle i_L(t) \rangle_T - \langle i_o(t) \rangle_T 
\] (2.6b)
These continuous time equations approximate the low frequency dynamics. They are nonlinear functions of the switching state vector (or duty cycle). As a result they must be linearized around a steady state operating point for control design. In steady state, all averaged capacitor currents and inductor voltages are zero, allowing an operating point to be found. It is this final, linearized averaged model of the circuit about a defined operating point that is called the SSA model of the converter.

Since the model is linearized about an operating point, any of the averaged states can be written as a linear combination of their steady state average value and a small perturbation. There are defined as follows:

\[
\langle v_g(t) \rangle_T = V_g + \hat{v}_g(t) \quad (2.7a)
\]
\[
\langle d(t) \rangle_T = D + \hat{d}(t) \quad (2.7b)
\]

where \( V_g \) represents the steady state input voltage, \( \hat{v}_g(t) \) represents the perturbation away from that voltage; \( D \), represents the steady state duty cycle of switch 1, and \( \hat{d}(t) \) is the perturbation away from that value. Similarly the inductor current, and capacitor voltage are expressed as:

\[
\langle i_L(t) \rangle_T = I_L + \hat{i}_L(t) \quad (2.8a)
\]
\[
\langle v_C(t) \rangle_T = V_C + \hat{v}_C(t) \quad (2.8b)
\]

If the perturbations are small (e.g. \( |\hat{v}_C(t)| << |V_C| \)), then second order terms can be ignored. Then substituting for the average inductor current, and capacitor voltage in (2.6a) and (2.6b) yields the linearized differential equations of the SSA model:

\[
L \frac{d}{dt} \hat{i}_L(t) = \hat{d}(t)V_g + D\hat{v}_g(t) - \hat{v}_o(t) \quad (2.9a)
\]
\[
C \frac{d}{dt} \hat{v}_C(t) = \hat{i}_L(t) - \hat{i}_o(t) \quad (2.9b)
\]
These equations represent the averaged model of the circuit, which can be put in state space form. The state space representation is of the form:

\[
\mathbf{x}(t) = \begin{bmatrix}
\dot{i}_L(t) \\
\dot{v}_C(t)
\end{bmatrix}
\] (2.10)

with the dynamics expressed as:

\[
\frac{d}{dt}\mathbf{x}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) + E\dot{d}(t) \tag{2.11a}
\]

\[
\mathbf{u}(t) = \hat{v}_g(t) \tag{2.11b}
\]

The equations (2.11) represent the SSA model. The duty cycle is the input to the system for control purposes, even though the input voltage is labelled as the input in the above equations. It is used to design a controller to ensure both stability and reference tracking, using conventional design techniques.

### 2.2 SSA Modelling Inaccuracy

As already discussed, the SSA model averages all state variables over a single switching period:

\[
\langle x(t) \rangle_T = \frac{1}{T} \int_{t-T}^{t} x(\tau)d\tau \tag{2.12}
\]

The averaging integral is a bounded (window) average of the state. It represents a window as the averaging is done over switching period of \(T\). By finding the average value of the state, given by (2.12), the state is essentially filtered, of its higher order frequency components. This conceptual filtering action is a basic assumption to SSA, however no dynamics are associated with it. To represent this conceptual filter in a complete model has, to date, not been accomplished. Therefore all SSA models neglect dynamics associated with the averaging process.
2.2.1 Example

The quality of a model is measured by its accuracy in representing the physical process that is modelling. One of the problems with SSA modelling is its inaccuracy. To examine this accuracy, an example is developed. For this example, an asynchronous buck converter, as shown in Appendix 7.1, is used. There is no output capacitor, and the load is a resistor.

The SSA model (Appendix 7.1.1) is used as derived. In addition, given the switching state vector $S(t)$, the output current can be simulated using a detailed time domain simulation tool (in this case, PLECS). The window average of the output current can then be found via simulation using (2.12).

Given a certain disturbance from steady state operation, the accuracy of the SSA model is examined. The model is tested for a step response in the duty cycle (duty cycle disturbance). Its response is compared to the simulated average output current. The results are shown in Figure 2.3.

The results show that the step response of the SSA model does not match the step response of the converter’s average output current. This discrepancy is especially noticeable in the transient region, following the duty cycle step disturbance. In this transient region, the difference between the average output current and the SSA model average output current is largest, and it slowly disappears as the system reaches its new steady state. In steady state, the model accurately predicts the average output current. The more rapid the transient the greater the inaccuracies.

This inaccuracy can further be explained by the absence of an averaging state in the SSA model. It assumes averaging of its states and output for derivation, but does not augment the averaging of the output in the system representation. This is true because as shown in Figure 2.3, SSA model predicts a higher average in the first period after the disturbance. However, averaging dynamics would introduce a delay of at least one switching period which would be shown by the model lagging the actual response.
Figure 2.3: A duty cycle disturbance of 0.1 is applied at $t = 0.2\, ms$. The output current response of the circuit is given in green. The circles represent the average output current as predicted by SSA. The plus (+) represents the average output current of the circuit over the last $T$ seconds.

The model fails to correctly predict its output in the transient response, under a step disturbance in the input. This means, the SSA model does not fully capture the dynamics of the physical converter. This is a clear and elementary limitation of the model.

The discrepancy in the results show that the SSA model may introduce errors during transients, but for the considered example no errors exist in steady state. As mentioned, the SSA model neglects high order frequency components, through filtering, as it assumes they do not contribute to any power exchange. However, it is well known that high order frequency, or harmonics, can produce average power exchange [6]. Consider the current, and voltage flowing through a resistor. Suppose they have a large DC component and small harmonics at frequency $w_o$:

\begin{align}
  v_R(t) &= V_R + v^1_R(t) \\  i_R(t) &= I_R + i^1_R(t)
\end{align}  \tag{2.13}
\[ v_R^1(t) = \cos(w_\omega t) \]  
\[ i_R^1(t) = \cos(w_\omega t) \]  

The power dissipated by the resistor can be calculated:

\[ p_R(t) = v_R(t)i_R(t) = V_R I_R + V_R \cos(w_\omega t) + I_R \cos(w_\omega t) + \cos^2(w_\omega t) \]  
\[ P_R(t) = \langle p_R(t) \rangle_T = \frac{1}{T} \int_{t-T}^{t} \left( V_R I_R + V_R \cos(w_\omega t) + I_R \cos(w_\omega t) + \cos^2(w_\omega t) \right) d\tau \]  
\[ P_R(t) = V_R I_R + \frac{1}{2} [W] \]

As seen, the presence of harmonics introduces average power dissipation. The SSA model averages the states, which means it expects an average power of:

\[ P_R(t) = V_R I_R \]  

SSA thus introduces two sources of error:

- errors during transients, resulting from unmodelled dynamics of the averaging process
- steady state errors resulting from neglecting average power flows associated with HOF terms

Even if the SSA model inaccuracy is overlooked, its implementation is another source of inaccuracy as will be shown in the next section.

### 2.3 SSA Implementation Inaccuracy

The SSA model does not model the high frequency components, but rather relies on filtering to remove them completely, as per (2.12). The practical implementation of (2.12) is another source of discrepancy.
As was seen in Figure 2.3, the model does not necessarily predict the correct state average. Nevertheless, for the implementation, at least, the controller will hopefully receive the converter’s average output current and not the model’s predicted average. In such case, the controller would be acting on the correct current error, even though its performance might not be quite as predicted by the model.

In practice, there are different methods of implementing the averaging of a state as in (2.12). One method used to approximate the window averaging of (2.12) is oversampling. In oversampling multiple measurements of the state are taken during a single switching period. However the most common method of averaging is to use a Low Pass Filter (LPF) in conjunction with a sample and hold. Such LPF would ideally filter all high order frequency terms ($F_{sw}$ and above) and pass all low frequency components (typically components below $F_{sw}/10$).

From a control systems perspective the LPF used in implementation ensures that the designed controller is not exposed to higher order (frequency) terms that are neglected by the model. In practice any LPF has a corner frequency, above which the signal’s frequency components are attenuated. SSA assumes no high frequency components exist whatsoever, hence it assumes infinite attenuation. For simplicity consider a first order LPF:

$$F(s) = \frac{1}{\frac{s}{w_n} + 1}$$  \hspace{1cm} (2.16)

where $w_n$ is the corner frequency. A bode plot of the LPF is shown in Figure 2.4. Below the corner frequency, the LPF passes the signal’s frequency components without any attenuation. Above the corner frequency, the LPF attenuates the frequency components with increasing factor. However, the LPF cannot filter out high order frequency components completely. It must therefore be considered an approximate implementation of (2.12). As it does not filter all high order components, the LPF on the output current of the converter will not produce the average output current precisely. This will be shown next with two exemplary LPFs.
Figure 2.4: The bode plot of a simple first order LPF. The filter attenuates high frequency components well, while letting low frequency pass. However there is still small non-zero phase present at the low frequencies.

Let it be assumed that attenuation greater than $80\,\text{db}$ (or $10^{-4}$) is considered infinite attenuation for practical purposes. Consider, then, a 2nd order LPF that has attenuation of $80\,\text{db}$ for the switching frequency of the previous example (Appendix 7.2):

$$LPF(s) = \frac{1}{\frac{s^2}{w_n} + 2\zeta \frac{s}{w_n} + 1}$$  \hspace{1cm} (2.17)

where $\zeta$ is 0.7, and $w_n$ is $\frac{1}{100}$ of the switching frequency, $w_{sw}$. This LPF is considered slow, as its corner frequency is many times smaller than the switching frequency. In addition, consider another 2nd order LPF with parameters typically used in converter filtering applications. In this case, the corner frequency, $w_n$, is $\frac{1}{3}$ of the switching frequency, a much faster LPF. Both LPFs have their bode plots shown in Figure 2.5. As can be seen in that figure, the slower LPF practically eliminates switching harmonics, at the expense of poor bandwidth, while the faster LPF offers high bandwidth but achieves only a $20\,\text{db}$
Figure 2.5: The bode plots for the slow and fast LPF. The slow LPF attenuates high order frequencies with greater factor than the fast LPF.
Figure 2.6: The output current response of the circuit is given in dashed green. The red triangles represent the filtered current by slower LPF. The blue circles represent the filtered current by faster LPF. The black pluses represent the average output current of the converter.

From this analysis, it would be easy to conclude that the slower LPF is a better approximation to the averaging assumption of the SSA model, and should be used in implementation over the faster LPF. However, as will be seen in the next plot, there’s a trade-off between the attenuation factor of the higher order frequencies, and the approximation to the averaging of the state.

As in the model inaccuracy, consider the asynchronous buck example (Appendix 7.1). The same input step disturbance is applied to the converter in open loop. The output current is then filtered by both LPFs, separately, and their output compared to the converter’s average output current. The results are shown in Figure 2.6. It can be seen that the slower LPF (triangles in Figure) output is very different form the converter’s average output current. It was this slower LPF, though, that had the better attenuation and approximation of the SSA’s averaging assumption. The faster LPF, approximated
Figure 2.7: The faster LPF has relatively accurate transient response but also a steady state error.

the average output current better in the transient region, but has a steady state error that the slower LPF does not. Figure 2.7 shows the faster LPF response only.

There is an associated trade-off between the steady state and transient accuracy of the filters, and their approximation to the SWA (i.e. (2.12)). If the filter attenuates all high frequency terms (i.e. slow filter), then it will approximate (2.12) well in steady state at the expense of not outputting the average value during transients. Similarly if the filter is fast, the transient response of its output matches the converter’s average better, but at the cost of steady state error.

This trade-off in the implementation of the averaging assumption of the SSA model is a further justification for the derivation of an improved model.

2.4 Summary

In this chapter SSA modelling was analyzed in detail. At its fundamental development, SSA modelling relies on the assumption of low switching ripple, namely the averaging
approximation. The net change in the current or voltage during a single switching period is unaffected by the switching ripple. This is modelled using the SWA integral, and it allows for the development of an LTI model representation of the converter. Results showed that the SSA model has limitations, and conventional implementations of SSA based controllers introduce further challenging design trade-offs.

Both the modelling and implementation inaccuracies of the SSA model merit the development of an enhanced modelling approach. The new modelling technique should not rely on the averaging approximation or any assumptions that lead to the addition of unmodelled dynamics, like those of the LPF. In addition the new model must have a better transient and steady error accuracy than SSA. These are the objectives of the thesis, and the next chapter develops such a model.
Chapter 3

DC/DC Modelling

In the previous chapter the limitations of SSA for modelling converters were outlined. The aim of the thesis is to present a new modelling approach that does not have such limitations. This chapter focuses on the development of such models for DC/DC converter application. In DC/DC converters the main control objective is regulation of certain voltages, currents, or a combination thereof, to a reference value. As such, the new model needs to represent the actual average of the quantity of interest, while accurately preserving all system dynamics.

A DC/DC converter transfers power between two voltage levels. This requires regulation of the average of the output current or voltage to track a constant reference value. A suitable model of the converter is one that represents the averaged output variable as a state variable. This averaging state needs to take the switching dynamics of the converter into account. The difference between this approach and the SSA model is that the dynamics of the averaging process are explicitly modelled and not abstracted away and neglected, as is the case in the conventional SSA approach. The new model can be suitably called the Advanced Time Averaging Model (ATAM), as it improves on the SSA approach by including the averaging dynamics in the state space representation of the system.
The first step in developing the ATAM, involves the derivation of a generic method for augmenting linear time varying system equations with an additional state that provides the integral of any state of the original system. Through a proposed discretization process, a new system output may be created that provides the average sliding window average of the state of interest.

### 3.1 Sliding Window Average

The sliding window average integrates a signal, which may be either continuous or discontinuous, between two points in time and precisely computes its average value. Because the window’s end points can be (together) moved to focus on any portion of a signal, it is referred to as a ‘sliding’ window average (SWA). This applies to the task at hand, which is to calculate the average value of a signal during a specified length of time. Of interest is to augment the linear time varying differential equations with an additional state (or set of states) that enable output of the SWA of any desired system state (or set of states). Consider a general LTV system with dynamics:

\[
\dot{x}(t) = A(t)x(t)
\] (3.1)

where \( x \in \mathbb{R}^n \), \( A(t) \in \mathbb{R}^{n \times n} \); suppose the state of interest is the \( m^{th} \) (\( m \in \mathbb{R}^1, 1 \leq m \leq n \)) state \( x_m \). To compute the SWA of this state, the following equation is employed:

\[
\rho(t) = \frac{1}{T} \int_{t-T}^{t} x_m(\tau) d\tau
\] (3.2)

The above equation calculates the signal average over the last \( T \) seconds. Figure 3.1 shows the SWA. The area under the signal over the last \( T \) seconds is integrated and weighted to produce the average of the signal – \( \rho(t) \).

Note that (3.2) is a bounded integral of a state. The goal is to augment the system
Figure 3.1: The figure shows how the integral in (3.2) calculates SWA.
of (3.1) with the signal average. (3.2) can be rewritten:

\[
\rho(t) = \frac{1}{T} \int_0^t x_m(\tau) d\tau - \frac{1}{T} \int_0^{t-T} x_m(\tau) d\tau 
\] (3.3)

Taking the time derivative on both side of the equation:

\[
\dot{\rho}(t) = \frac{1}{T} x_m(t) - \frac{1}{T} x_m(t-T) 
\] (3.4)

The rate of change of the averaging state, \( \rho(t) \), is seen to depend on the original state of interest and a time delayed version of the state. Although the first term of the equation (3.4) is easy to augment into (3.1), the second term, namely the time delayed term, is not trivial. This is due to the fact that time delays cannot be precisely modelled by differential equations of finite order. A further complication that may arise if approximations are made is that, while the SWA output is bounded, the two integrals of (3.3) represent unbounded signals.

Fortunately, there’s a way of augmenting SWA dynamics into the system. A new ‘integrating state’ is defined as follows:

\[
x_{n+1}(t) = \frac{1}{T} \int_0^t x_m(\tau) d\tau 
\] (3.5)

This allows the \( \rho(t) \) signal of (3.3) to be expressed as a function that depends exclusively on the newly defined state:

\[
\rho(t) = x_{n+1}(t) - x_{n+1}(t-T) 
\] (3.6)

Prior to attempting evaluation of (3.7), the augmented system dynamics are defined as:

\[
\dot{x}(t) = A(t)x(t) 
\] (3.7a)
\[ \dot{x}_{n+1}(t) = \frac{1}{T}x_m(t) \quad (3.7b) \]
\[ \frac{d}{dt} x_{n+1}(t) = \frac{1}{T} x_m(t) \quad (3.7c) \]

defining:

\[ x_a(t) = \begin{bmatrix} x(t) \\ x_{n+1}(t) \end{bmatrix} \quad (3.8) \]

(3.7) can be expressed as:

\[ \dot{x}_a(t) = \tilde{A}(t)x_a(t) \quad (3.9) \]

The dynamics of the integrating state, \( x_{n+1} \), have thereby been augmented in the continuous time system, though the SWA value, given by \( \rho(t) \), is not yet available.

The form of SWA dynamics in (3.6) suggest that they are more conveniently represented in discrete time. Assuming a sampling period \( T \), that is equal to the averaging period yields:

\[ \rho(kT) = x_{n+1}(kT) - x_{n+1}(kT - T) \quad (3.10) \]

This represents the SWA dynamics as a discrete time LTI system. Thus it is convenient for the full system (3.9), to be discretized with period \( T \). The system (3.9) remains a linear time varying dynamical system. A general solution can be written of the form:

\[ x(t + T) = \Phi(t, t + T)x(t) \quad (3.11) \]

where the subscript \( a \), has been dropped from notation for convenience. The above equation (3.11), is the solution to (3.1); \( \Phi(t, t + T) \) is a state transition map of the piecewise linear system. Discretizing (3.11) gives:

\[ x(kT + T) = \Phi(kT)x(kT) \quad (3.12) \]
For notational convenience such difference equations will be written in the following form:

\[ x[k + 1] = \Phi[k]x[k] \]  
(3.13)

And similarly (3.10) will be written in form:

\[ \rho[k] = x_{n+1}[k] - x_{n+1}[k - 1] \]  
(3.14)

which is the integration of the state over the last switching period. The advantage now, is that this equation is easily representable in discrete time, though it requires augmentation of a second ‘delay integrator state’ as follows:

\[ x_{n+2}[k + 1] = x_{n+1}[k] \]  
(3.15)

The SWA average is then given by:

\[ y[k] = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 & -1 \end{bmatrix} x[k] \]  
(3.16a)

\[ \rho[k] = y[k] = x_{n+1}[k] - x_{n+2}[k] \]  
(3.16b)

This quantity will be considered as the system output \( y[k] \) for the remainder of this work.

In this section, the system (3.1) was augmented with two additional, integrating and time-delayed integrating, states for the purpose of producing a single SWA. In addition both states, as integrating states, are unbounded. A system with unbounded states is difficult to control, due to the presence of internal state instability. There is a more elegant solution which only requires one additional state.
3.1.1 Elimination of the Delayed Integrator State

Elimination of the delayed integrator state requires re-examination of the continuous time system equation:

$$\dot{x}_a(t) = \hat{A}(t)x_a(t)$$  \hspace{1cm} (3.17)

Recall the last state is the integrating state as represented by:

$$\dot{x}_{n+1}(t) = \frac{1}{T}x_m(t)$$  \hspace{1cm} (3.18)

If the system (3.17) is discretized such that:

$$x_a[k+1] = \Phi[k]x_a[k]$$  \hspace{1cm} (3.19)

Then the last state, namely the integrating state, will have dynamics given by:

$$x_{n+1}[k+1] = \begin{bmatrix} G[k] & 1 \end{bmatrix} x[k]$$  \hspace{1cm} (3.20a)

$$G[k] = \begin{bmatrix} \Phi_{n+1,1}[k] & \Phi_{n+1,2}[k] & \cdots & \Phi_{n+1,n}[k] \end{bmatrix}$$  \hspace{1cm} (3.20b)

This may be re-written as:

$$x_{n+1}[k+1] = x_{n+1}[k] + G[k] \begin{bmatrix} x_1[k] \\ x_2[k] \\ \vdots \\ x_n[k] \end{bmatrix}$$  \hspace{1cm} (3.21)

Therefore in discrete time, the integrating state is the summation of the previous integrating state’s value, $x_{n+1}[k]$, and a contribution from the remaining states, scaled by a vector $G$. To implement sliding window averaging the previous integrating state’s value
may simply be discarded by modifying (3.20a):

\[ x_{n+1}[k+1] = \begin{bmatrix} G[k] & 0 \end{bmatrix} x_n[k] \]  

(3.22)

That is the \((n + 1, n + 1)\) entry of \(\Phi[k]\) is modified from unity to zero. The states of the discrete time system now remain bounded (provided the continuous time system variables remain bounded).

For each state of interest there is now only the addition of a single state that allows for SWA. The procedure can be summarized, from the viewpoint of converter systems:

1. Write out the dynamics of the circuit in all possible switching positions, expressing the system as an autonomous system.

2. Augment these dynamics to include integration states of all the quantities of interest.

3. Write out the evolution of the system over one switching period.

4. Discretize the augmented system with switching period \(T\).

5. Modify the integrating states’ dynamics of the linearized system turning them into SWA values.

### 3.2 ATAM Application Example

Now that there is a way of augmenting a system with the SWA of any of quantity, the new modelling approach can be formulated with a general DC/DC converter example. The converter has different dynamics depending on \(S(t)\). The synchronous buck converter example will be used for development. It is important to note that the method can be readily applied to any topology. Let \(x(t)\), contain the inductor current, and capacitor voltage, as well as all of the voltage/current inputs. This is done to convert the system
to an autonomous form, which makes the derivation of the new modelling technique more convenient. Then in the on, and off position the dynamics can be represented, respectively:

\[
\dot{x}(t) = \begin{cases} 
A_{on}x(t), & S(t) = 1 \\
A_{off}x(t), & S(t) = 0
\end{cases}
\]

\[x(t + T) = e^{\hat{A}_{off}(1-d(t))T}e^{\hat{A}_{on}d(t)T}x(t)\] (3.23b)

where \(A_{on}\) and \(A_{off}\) are \(\mathbb{R}^n\) and capture the dynamics of all states in the on and off switch state respectively; \(d(t)\) is the duty cycle of the primary switch. Equation (3.23b) is the exact evolution of the converter’s dynamics, assuming ideal switches. It is linear time variant (LTV) system.

Without loss of generality, the output of the system will be the SWA of state \(x_1\). Following with the first step in Section 3.1, both the on and off dynamics are augmented with a single state:

\[
\hat{A}_{on} = \begin{bmatrix} 0 & \vdots & 0 \\
A_{on} & 0 \\
1 & 0 & \cdots & 0
\end{bmatrix}
\]

(3.24)

\[
\hat{A}_{off} = \begin{bmatrix} 0 & \vdots & 0 \\
A_{off} & 0 \\
1 & 0 & \cdots & 0
\end{bmatrix}
\]

(3.25)

A single switching period consists of some interval when the switch is on, followed by another interval when the switch is off. This progression can be written as:

\[x[k + 1] = \Phi[k]x[k]\] (3.26a)
\[ \Phi[k] = e^{\tilde{A}_{\text{off}} \left(1-d[k]\right)T} e^{\tilde{A}_{\text{on}} d[k]T} \] (3.26b)

where \( d[k] \) represents the duty cycle for the \( k^{th} \) switching period. The system is linear time variant (LTV) as \( \Phi \) depends on \( d[k] \).

### 3.3 Validation

Before linearizing the new model it is worthwhile to test its accuracy. Namely, whether the augmented state that calculates the SWA, in fact does so with better accuracy than the SSA model. The same buck example from Appendix 7.1 is used. A step in the duty cycle is applied at 0.8 ms from 0.5 to 0.6. The converter is simulated in PLECS, and its output current is averaged using ideal SWA. Similarly the ATAM of the converter is simulated under the same inputs. Its output state, which should be SWA of the output current, is compared with that of the converter.

Results are shown in Figure 3.2. The ATAM exactly matches the average output current, in both steady state and transience. Compared to the SSA model in the previous chapter, which was also simulated under the same conditions, the ATAM is a precise representation of the discrete time dynamics of the converter. The SSA model is not accurate due to its assumptions as already discussed.

### 3.4 Summary

In this chapter a new model approach was developed for modelling DC/DC converters. This new model is called ATAM. It is based on the exact switching dynamics of the converter. Through augmentation of a single state a SWA output may be computed. A buck converter example was used to demonstrate the ATAM procedure. The new model is LTV. For more common control design applications, it is possible to eliminate the time variance through a process of linearization. That is the primary objective of the next
Figure 3.2: The open loop response of the current to a duty cycle disturbance. The ATAM precisely predicts the simulated current’s average waveform.
The model presented in this chapter is significant as it is, to date, the first successful attempt at capturing converter’s output averaging dynamics in a mathematical model. There are no assumptions made with respect to the harmonic composition of the output, nor any other limiting assumptions.
Chapter 4

DC/DC Model Linearization and Control

The newly developed model in the previous chapter augmented a LTV system, with an additional state representing the dynamics of the SWA of the system output. The model is an accurate representation of a DC/DC converter; it makes no assumptions, nor excludes any dynamics. However, as a result of the augmentation technique employed, the new model is linear time variant (LTV). SSA and other modelling approaches are commonly used because they represent the system by an LTI model that is valid about the desired operating point. Many controller design techniques are available for LTI systems and their implementation is straightforward. This chapter demonstrates that existing linearization and control techniques may be readily applied to the derived LTV converter models. This will be demonstrated via control design example for a buck converter. The LTI model will then be validated. The performance of a controller design based on the derived LTI model will be compared with that of a controller design based on the SSA model.
Chapter 4. DC/DC Model Linearization and Control

4.1 Linearization

LTV systems can be linearized about an operating point, though the linearization process is quite distinct from that used for linearization of continuous time nonlinear systems. Much like the SSA model, linearization is acceptable as a converter is typically operated around such an operating point. Recall from the last chapter, the synchronous buck system equations are as follows:

\[
\dot{x}(t) = \begin{cases} 
A_{on}x(t), & S(t) = 1 \\
A_{off}x(t), & S(t) = 0 
\end{cases} 
\]  
(4.1a)

\[
\hat{A}_{on} = \begin{bmatrix}
A_{on} & \vdots \\
0 & \ddots \\
1 & 0 & \cdots & 0
\end{bmatrix} 
\]  
(4.1b)

\[
\hat{A}_{off} = \begin{bmatrix}
A_{off} & \vdots \\
0 & \ddots \\
1 & 0 & \cdots & 0
\end{bmatrix} 
\]  
(4.1c)

\[
x[k] = \Phi[d[k]]x[k-1] 
\]  
(4.1d)

\[
\Phi[k] = e^{\hat{A}_{off}(1-d[k])T}e^{\hat{A}_{on}d[k]T} 
\]  
(4.1e)

For clarity in (4.1d) we explicitly show the dependance of \( \Phi \) on \( d \), where \( d \) may vary with each period. The state that is of interest, namely the output of the system, will be the first state. Equation (4.1d) and (4.1e) are the new model of the chosen system. To linearize this model the following notation is introduced:

\[
\tilde{x}[k] = x[k] - X_{eq} 
\]  
(4.2a)
\[
\tilde{d}[k] = d[k] - D_{eq}
\]

(4.2b)

\[
\tilde{x}[k + 1] = \frac{\partial \Phi}{\partial x} \bigg|_{x_{eq}, D_{eq}} \tilde{x}[k] + \frac{\partial \Phi}{\partial d} \bigg|_{x_{eq}, D_{eq}} \tilde{d}[k]
\]

(4.2c)

\[
\frac{\partial \Phi}{\partial x} \bigg|_{x_{eq}, D_{eq}} = \Phi(D_{eq})
\]

(4.2d)

\[
\frac{\partial \Phi}{\partial d} \bigg|_{x_{eq}, D_{eq}} = \left( -\hat{A}_{off} T e^{\hat{A}_{on} D_{eq} T} + e^{\hat{A}_{off} (1-D_{eq}) T} \hat{A}_{on} D_{eq} T \hat{A}_{on} T \right) X_{eq}
\]

(4.2e)

where the operating point is chosen and represented by the set \((X_{eq}, D_{eq})\). The system’s small signal dynamics are therefore given as follows:

\[
\tilde{x}[k + 1] = A_d \tilde{x}[k] + B_d \tilde{d}[k]
\]

(4.3a)

\[
A_d = \frac{\partial \Phi}{\partial x} \bigg|_{x_{eq}, D_{eq}}
\]

(4.3b)

\[
B_d = \frac{\partial \Phi}{\partial d} \bigg|_{x_{eq}, D_{eq}}
\]

(4.3c)

\[
\tilde{y}[k] = C_d x[k] = \begin{bmatrix} 0 & \cdots & 0 & 1 \end{bmatrix} x[k]
\]

(4.3d)

Since this is now the linearized state space representation of the circuit, and the integrating state has been augmented, the remaining two steps of Section 3.1 can be performed. The SWA state’s dynamics in (4.3a) are modified as to perform SWA. So if:

\[
\tilde{x}_{n+1}[k + 1] = \begin{bmatrix} A_{d(n+1,1)} & A_{d(n+1,2)} & \cdots & A_{d(n+1,n)} & 1 \end{bmatrix}
\]

(4.4)

This is replaced by:

\[
\tilde{x}_{n+1}[k + 1] = \begin{bmatrix} A_{d(n+1,1)} & A_{d(n+1,2)} & \cdots & A_{d(n+1,n)} & 0 \end{bmatrix}
\]

(4.5)

With the SWA state augmented to the DC/DC system, the new state space representation encapsulates the dynamics of the circuit linearized around the desired operating point.
This linearized model, can be suitably called Linearized Advanced Time Averaging Model (LATAM) of the DC/DC system. In terms of output to the system (4.3a), the SWA output variable is provided as:

\[
\tilde{y}[k] = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \tilde{x}[k]
\] (4.6)

Again the steps to derive LATAM representation of a general DC/DC topology can be summarized:

1. Write out the dynamics of the circuit in all possible switching positions, expressing the system as an autonomous system.
2. Augment these dynamics to include integration states of all the quantities of interest.
3. Write out the evolution of the system over one switching period.
4. Discretize the augmented system with switching period \( T \).
5. Linearize this system around a desired operating point.
6. Modify the integrating states’ dynamics of the linearized system turning them into SWA values.

Note that the last two steps are interchangeable, the linearization can be done first, or the discrete model’s dynamics can be modified for the augmented states to make them memoryless. If linearization is not needed, then that step can be skipped.

While the advantages of this modelling approach will be outlined in greater depth in the ensuing sections, it can already be seen that this approach offers some benefits. The first is the fact that this approach does not suffer from aliasing issues. The true average is calculated through the augmented state (SWA), which rejects high order frequency
components of the state. Second, the model presented is of the form \( d[k] \) to \( y[k] \). This is the output and input of a conventional digital controller for converters.

4.2 Model Demonstration

With the development of the LATAM for DC/DC, its accuracy and performance need to be tested and compared to SSA. The focus of this section will be on using the LATAM, for the DC/DC topology, in designing a controller. The performance of this controller will be compared against that of one designed using SSA model. In addition, the accuracy of both modelling techniques will be compared in an open loop scenario. The closed-loop performance determines the degree of improvement over SSA. Since the new modelling does not suffer from aliasing issues and models all system latencies, a better performance is expected.

4.2.1 Open Loop Performance

As seen already in previous chapters, one way of validating a model’s accuracy is to check its open loop response under an input disturbance, and compare it with the converter’s response. From Chapter 2, it was shown that the ATAM output precisely matched the simulated converter’s response (sampled). Thus, ATAM will be used as the benchmark, and representation of the converter’s response. The same asynchronous buck example is used for this open-loop comparison, as detailed in Appendix 7.1.

The results are shown in Figure 4.1a. The three model responses are shown: ATAM (converter), SSA, and the LATAM. At time 2 ms, a disturbance in the duty cycle is applied. The ATAM output is what the LATAM of the converter, and SSA model aim to achieve. From the figure it can be seen that LATAM is closer to the actual average output current during the disturbance. Figure 4.1 shows the models’ error to really differentiate between the performance accuracy of the two models. This is an encouraging result. It
(a) The output average current, open loop response to a disturbance in the duty cycle.
+ shows the response of the converter (ATAM). As can be seen the LATAM response (○) is more accurate compared to the SSA model (□) in the transient. At steady state, the responses are identical.

(b) The error of each model’s response is shown here. As can be seen the LATAM offers much better accuracy than the SSA model in the transient region.

Figure 4.1: The models’ open loop response to a disturbance in the duty cycle.
validates the linearization of the ATAM. By doing so the new model is not only more accurate than the SSA in open-loop but it also models the ‘filtering’ (SWA) of the output current as a state.

4.2.2 Closed Loop Performance

As seen before, the new DC modelling approach revolves around the linearization of the ATAM of a DC/DC converter around a chosen operating point. Thus, any inaccuracies in the modelling will be only because the linear time variant model was linearized. However, the claim is still that in spite of this linearization, the end result will be a more accurate model.

For the purpose of comparing the different modelling techniques under closed loop a modified example has been chosen, and is shown in Figure 4.2. The converter is still asynchronous buck; the load has been changed to a constant voltage source – or a motor.

Figure 4.2: The test converter topology. There is no output capacitor – it is extracted out to the output side.
The topology (Figure 4.2) has been chosen for several reasons. First, to keep the modelling simple, the output capacitor has been removed. Having it there only adds an additional state that has to be modelled. Further, the control objective for this topology will be maintaining the output current at a certain reference value. This is equivalent to a constant power exchange with the output, as the output is a motor with fixed voltage. The load has been changed from resistive to a motor, mainly because the damping characteristics at a resistor aid to reduce the effect of any imposed disturbances on the current. The goal is to do a fair comparison between the new model and the SSA, thus removing unaccounted factors and minimizing variables.

As the control objective is keeping the output current constant reference value, the disturbance response of the controller will be compared. Specifically load disturbance (load voltage change), will be applied in order to see how the controller responds.

Model Setup

Before designing the controller, the new model for the presented topology (Figure 2.1) is derived. The SSA model is needed as well for its own controller design. Detailed derivation of both methods is presented in Appendix 7.2. Exemplary circuit parameters are also chosen in Appendix 7.2.

With the two models of the given topology derived, the comparison analysis can be performed. The main comparison is closed-loop disturbance rejection. Building on the accuracy of the ATAM, the most important comparison is the use of the new modelling to design a better controller. As stated the control objective will be to maintain the output current at a constant reference value. The SSA model will be also used to design a controller, whose performance will be essentially the benchmark of this test. Next, the controllers for each model are designed.
Controller Design

Digital controllers are commonly used for converter applications. For the given example, a conventional digital controller will be implemented, and the whole system simulated in a detailed time domain simulation. The system setup is shown in Figure 4.3. The controller receives the error in the output current, and outputs the corrective duty cycle. This duty cycle is processed through a digital PWM, which then converts it into $S(t)$, the switching state. The switching state signal controls the behaviour of the converter.

The calculate the deviation from the reference output current, the converter’s output current is measured. Depending on whether a SSA, or LATAM controller is being implemented, a LPF or SWA in hardware, respectively, is implemented in the feedback as shown on the same figure. SWA in hardware can be accomplished in different ways. One common method is to oversample the measured output current (many samples in one switching period), and average the samples for one switching period [4]. This is a simple, and accurate method that can be easily implemented in a DSP.

The LATAM is a discrete time model of the system. There are many different controller design techniques that can be applied to such model. For simplicity, a root locus method will be used. In addition, with both models, a Proportional Integrator (PI)
controller will be used for fair comparison. A digital PI controller can be approximated as:

\[ C(z) = K_p + \frac{K_i}{z - 1} \]  

(4.7)

The controller design for the new model is shown in Appendix 7.2.1. The root locus method for discrete time systems leads to a discrete controller with gains \( K_i = 0.401 \) and \( K_p = 1.889 \). The control design for the SSA model is shown in Appendix 7.2.2 with gains \( K_i = 0.339 \) and \( K_p = 1.669 \).

Notably, the gains of the two controllers are relatively similar. It is the use of the LPF vs. SWA, that will determine the performance difference between the two models.

**Results**

The controllers are implemented, and the output current response is recorded for a load disturbance. The load's voltage is decreased from 5V to 2V. The average output current response for both controllers is shown in Figure 4.4.

The results for this disturbance test, show that the LATAM based controller performs better. The overshoot of the average output current is smaller than the case for the SSA model based controller. Also, the SSA model based controller results in a steady state error. The LATAM based controller does not.

The response of the SSA model based controller shows that, the controller is relatively aggressive. This can be seen in Figure 4.4, at around 2.15 milliseconds; showing as a high frequency oscillation in the response. Such oscillations are a sign of a controller that is as fast as can be without pushing the system into instability. The steady state error, as discussed before, is due to the optimization between fast LPF and transient response and steady state inaccuracy. The new model has no such trade-offs.

The fact that the SSA model based controller is designed under aggressive criteria, and still is outperformed by the LATAM based controller also shows the advantage of using LATAM and SWA over SSA and LPF.
Figure 4.4: The average output current response for a load decrease disturbance. The SSA model based controller has higher overshoot and steady state error; LATAM based controller has smaller overshoot and no steady state error.
4.3 Summary

In this chapter the new model, ATAM, was linearized and used for control design. It was shown that the controller performance based on the LATAM performed better than that of the SSA model. The open-loop comparison indicated that the LATAM more closely outputs the average value of its output than the SSA model.

More importantly a closed loop test was carried out. A controller designed using LATAM was tested under different load disturbances. A second controller designed using SSA modelling was also tested under the same load disturbances. The LATAM controller performed better – with smaller overshoot and without any steady state error.

Further to this, the presented modelling approach, based on its construct does not suffer from aliasing. It also captures the complete dynamics of average output controlled converters. Because SWA is used, there’s no implementation calibration in the same way LPF needs calibration depending on the different switching frequencies.
Chapter 5

DC/AC Modelling

In Chapter 3 an Advanced Time Averaging Model was developed for DC/DC converters. DC/AC converters are another widely used type of converter. They are used for interfacing renewable sources such as solar and wind, to the utility and as back to back converters for changing the frequency of the AC.

The focus of this chapter will be on developing a new modelling approach for DC/AC converters. In DC/AC converters, the main control objective is controlling the AC side current or voltage to a certain reference fundamental sinusoidal value. Comparing to the DC/DC, the controller needs to track a sinusoidal reference as opposed to constant. First, however, a mathematical model is needed for calculating AC current/voltage injected by the converter. This corresponds to calculating the fundamental (frequency of the AC side) component of that signal, through Fourier Decomposition. The new model needs to be able to model the incremental contribution to the fundamental component of that signal during each switching period or subcycle. Past work [11, 12, 25] has focused on deriving a full DC/AC model including all harmonics by augmenting the Fourier decomposition integrals. This research will be leveraged in developing the new model.

As will be seen, the new modelling approach leads to a non-LTI model because of the calculation of the fundamental. Designing a controller from the new model is non-trivial
and does not allow for the use of simple LTI control design techniques. Thus, design of a DC/AC converter controller is considered outside the scope of this thesis. However, the accuracy of the new modelling approach is verified. The accuracy of the calculation of the fundamental contribution during each switching period is checked. For that purpose, two examples will be used in validating the new model's accuracy.

5.1 Modelling Challenges

The objective of the chapter is to develop a modelling approach that does not suffer from aliasing issues and allows for more accurate controller design. DC/AC converters exchange power between DC and AC. As most inverters are grid tied (AC side), to accomplish the desired power exchange, the converter needs to be controlled in order to appropriately shape its AC side current. The reference signal for the controller is sinusoidally changing - whereas the DC/DC controller tracked a constant reference signal.

The main control objective in the DC domain was to ensure that certain current/voltage average values follows a reference value. Analogous to this in DC/AC converters, is to ensure that the converter’s AC current fundamental component follows a certain reference. Shaping the AC current really refers to ensuring the converter injects current with the correct amount of fundamental frequency. Determining the magnitude and phase of the fundamental component of the AC current (or voltage) in a converter topology is a challenge. Switching frequency is larger than the grid frequency. To calculate the fundamental component of any signal:

$$X^1(t) = \frac{1}{T_{ac}} \int_{t-T_{ac}}^{t} x(\tau)e^{-jw_o\tau} d\tau$$  \hspace{1cm} (5.1)

where $T_{ac}$ is the period of the fundamental frequency, which is much longer than the switching period; $w_o$ is the fundamental frequency in rad/s. To calculate the total fundamental component of the signal, a whole fundamental period is needed. This is a
challenge in AC modelling, as the control decisions must happen at the switching frequency. The new model therefore employs a method for calculating the ‘fundamental component contribution’ of a signal during a subcycle of the fundamental period.

5.2 Subcycle Fundamental Contribution

To create a new modelling approach, like in the DC/DC case, the first challenge is developing a way of calculating the fundamental contribution of a signal over a single switching period. To illustrate, consider the space vector associated with a balanced ac quantity. In the $\alpha\beta$ frame the space vector may be expressed as: $x(t) = \cos(w_o t) + j \sin(w_o t)$. Per (5.1):

$$X^1(t) = \frac{1}{T_{ac}} \int_{t-T_{ac}}^{t} (\cos(w_o t) + j \sin(w_o t)) e^{-jw_o \tau} d\tau \quad (5.2)$$

The integral inside simplifies to a constant value of unity, as the multiplication of positive and negative rotating signals is constant. This integral can be split into subintervals (subcycles) of switching period, $T$:

$$X^1(t) = \frac{1}{T_{ac}} \int_{t-T}^{t} 1 d\tau + \frac{1}{T_{ac}} \int_{t-2T}^{t-T} 1 d\tau + \cdots + \frac{1}{T_{ac}} \int_{t-(m-1)T}^{t-mT} 1 d\tau \quad (5.3)$$

For a perfect oscillator, the subcycles have equal contribution to the fundamental frequency (Figure 5.1)

Thus evaluating (5.1) for a subcycle time, gives the subcycle fundamental contribution (SFC), represented as $X_{sw}^1(t)$.

$$X_{sw}^1(t) = \frac{1}{T_{ac}} \int_{t-T}^{t} x(\tau)e^{-jw_o \tau} d\tau \quad (5.4)$$

where $X_{sw}^1(t) = X_{sw, re}^1(t) + j X_{sw, im}^1(t)$. This integral, will need to be augmented into the
Figure 5.1: The integrator in (5.3) leads to equal sub-integrals which form a straight line for a perfect oscillator system so that the SFC of the signal of interest is available as a state at the end of each switching period. Fortunately, equation (5.4) is of a special form [8]. First, consider a general differential equation of the form:

\[
\frac{dy}{dt}(t) = m \cdot y(t) + h \cdot q(t) \quad (5.5)
\]

the general solution to this is:

\[
y(t) = e^{mt}y(t - t_o) + \int_{t-t_o}^{t} e^{m(t-\tau)}h \cdot q(\tau)d\tau \quad (5.6)
\]

The above equation describes the integral that calculates SFC with the following substitutions:

\[
y(t - t_o) = 0 \quad (5.7a)
\]
\[ y(t) = X_{sw, re}^1(t) + jX_{sw, im}^1(t) \quad (5.7b) \]
\[ m = jw \quad (5.7c) \]
\[ q(t) = x(t) \quad (5.7d) \]
\[ h = \frac{1}{T_{ac}} e^{-j\omega_o T} \quad (5.7e) \]
\[ t_o = T \quad (5.7f) \]

As a result (5.4) can be written as a state whose dynamics are represented as:

\[
\frac{dy}{dt}(t) = jw \cdot y(t) + \frac{1}{T_{ac}} e^{-j\omega_o T} x(t) \quad (5.8)
\]

\[
\frac{d}{dt} \left( X_{sw, re}^1(t) + jX_{sw, im}^1(t) \right) = jw( X_{sw, re}^1(t) + jX_{sw, im}^1(t) ) + \frac{1}{T_{ac}} e^{-j\omega_o T} x(t) \quad (5.9)
\]

Or in real form, where the complex states are split into real and imaginary:

\[
\begin{bmatrix}
\frac{d}{dt} X_{sw, re}^1(t) \\
\frac{d}{dt} X_{sw, im}^1(t)
\end{bmatrix}
= \begin{bmatrix}
0 & -\omega_o \\
\omega_o & 0
\end{bmatrix}
\begin{bmatrix}
X_{sw, re}^1(t) \\
X_{sw, im}^1(t)
\end{bmatrix}
+ \frac{1}{T_{ac}}
\begin{bmatrix}
\cos(\omega_o T) \\
-\sin(\omega_o T)
\end{bmatrix}
x(t) \quad (5.10)
\]

The augmented system is composed of the state space representation of the system with two additional states needed for the calculation of SFC. The two new states to the augmented system solely act as integrators to find the magnitude and phase of the voltage/current of interest (i.e. fundamental component), and are not dependent on any system dynamics. Notice that the initial conditions in (5.7a) assume that at the start of the subinterval (or switching period) the SFC states are initialized to zero. This is important to note because once the system is discretized similar modification as the SWA for DC/DC case (Section 3.1) will be necessary in order for the above equations to be valid. As it stands now, the states would keep integrating, when a sliding integration of a switching period is required.
5.2.1 Modifying SFC States to Memoryless

Before doing a specific DC/AC converter example, the issue regarding the reset of the SFC states needs to be addressed. Similar to the DC/DC case for SWA the dynamics of SFC in continuous time integrate and calculate the cumulative fundamental contribution. The dynamics will need to be modified once the system is discretized so that SFC states only calculate the current switching period’s fundamental contribution. Similarly to SWA it will need to become a memoryless state. In order to see the steps needed, a sinusoidal signal will be considered for which the fundamental component is the only one present.

Let:

\[ x_1(t) = \cos(w_o t) \] (5.11)
\[ x_2(t) = \sin(w_o t) \] (5.12)

Here a second state is needed \( x_2(t) \), in order to write the dynamics of the sinusoidal signal. The goal is to calculate the fundamental contribution of each subcycle of the sinusoid \( (x_1(t)) \). Two additional states are introduced for SFC calculation:

\[ x_3(t) + jx_4(t) = \frac{1}{T_{ac}} \int_{t-T}^{t} x_1(\tau)e^{-jw_o \tau} d\tau \] (5.13)

In state space form this can be written as per (5.10):

\[
\frac{d}{dt} \begin{bmatrix} x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} 0 & -w_o \\ w_o & 0 \end{bmatrix} \begin{bmatrix} x_3(t) \\ x_4(t) \end{bmatrix} + \frac{1}{T_{ac}} \begin{bmatrix} \cos(w_o T) \\ -\sin(w_o T) \end{bmatrix} x_1(t) \] (5.14)

Finally the whole system:

\[
\frac{d}{dt} x(t) = Ax(t) \] (5.15a)
\[ A = \begin{bmatrix} 0 & -w_o & 0 & 0 \\ w_o & 0 & 0 & 0 \\ \frac{1}{T_{ac}} \cos(w_o T) & 0 & 0 & -w_o \\ -\frac{1}{T_{ac}} \sin(w_o T) & 0 & w_o & 0 \end{bmatrix} \]  \hspace{1cm} (5.15b)

As in DC/DC, and as will be seen for the DC/AC case, the value of interest is at the end of the switching period – since it’s then when control action can take place. Mathematically this is equivalent to discretizing the system in (5.15a)-(5.15b):

\[ \Phi = e^{AT} \]  \hspace{1cm} (5.16a)

\[ x[k+1] = \Phi x[k] \]  \hspace{1cm} (5.16b)

where \( x[k] = x(kT) \) The structure of the \( \Phi \) matrix is as follows:

\[ \Phi = \begin{bmatrix} \mathcal{R} & 0 \\ \mathcal{M} & \mathcal{R} \end{bmatrix} \]  \hspace{1cm} (5.17)

where:

\[ \mathcal{R} = \begin{bmatrix} \cos(w_o T) & \sin(w_o T) \\ -\sin(w_o T) & \cos(w_o T) \end{bmatrix} \]  \hspace{1cm} (5.18)

and \( \mathcal{M} \in \mathbb{R}^{2 \times 2} \) is the Fundamental Contribution Matrix (FCM). It relates the states of the original system to the SFC states - in other words how much each one contributes to the calculation of the fundamental. The discrete dynamics of the SFC states are:

\[ \begin{bmatrix} x_3[k+1] \\ x_4[k+1] \end{bmatrix} = \mathcal{M} \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix} + \mathcal{R} \begin{bmatrix} x_3[k] \\ x_4[k] \end{bmatrix} \]  \hspace{1cm} (5.19)

The SFC states can be considered the coordinates of a vector; Cumulative Fundamental Contribution (CFC) vector, \( \mathbf{H}_k \), on a 2-d plane. The subscript indicates the switching
period during which the SFC states (vector coordinates) are computed. The position of
the CFC vector is the contribution from the states of interest ($F_k$) and a rotated version
of the previous position of the CFC vector ($H_{k-1}$). Let $F_k$ represent the fundamental
contribution vector (SFC vector) during the $k^{th}$ switching period:

$$F_k = M \begin{bmatrix} x_1[k-1] \\ x_2[k-1] \end{bmatrix}$$

(5.20)

Then $H_k$, which is represented by the SFC states, can be written as:

$$H_k = M \begin{bmatrix} x_1[k-1] \\ x_2[k-1] \end{bmatrix} + R F_{k-1} + R^2 F_{k-2} + \cdots + R^{k-1} F_1$$

(5.21a)

$$H_k = F_k + RH_{k-1}$$

(5.21b)

Since the goal is to only have the fundamental contribution during the current switching
period, the dynamics have to be modified as to eliminate the addition of the previous
vectors. In addition, though, the current contribution from the states of interest need
to be unrotated by the number of switching periods that have already passed. This is
because, unlike the DC averaging case, the frame of reference for each SFC is rotating
by $R$. This is evident in the dynamics of (5.21a) that the previous vectors are added to
the present vector only after a corrective rotation is applied; thus bringing all vectors in
the same frame of reference. Figure 5.2 shows this graphically for an arbitrary example.
By looking at (5.20) and applying the inverse of the rotation, the following equation is
derived:

$$F_{k,\text{stat}} = R^{-(k-1)} M \begin{bmatrix} x_1[k-1] \\ x_2[k-1] \end{bmatrix}$$

(5.22)

$F_{k,\text{stat}}$ represents the SFC during the current subinterval. By multiplying (5.20) by $k$
inverse rotations, the SFC is brought to the frame of reference at time $k = 1$. In other
words, SFC states are reset every switching period. Further because the SFC is really the desired output value, the final augmented system is described as:

\[ \dot{x} [k] = \begin{bmatrix} x_1 [k] \\ x_2 [k] \end{bmatrix} \]  

\[ \hat{\Phi} = \mathcal{R} \]  

\[ \hat{x} [k + 1] = \hat{\Phi} \hat{x} [k] \]  

\[ \hat{y} [k] = \Gamma \hat{x} [k] \]  

\[ \Gamma = \mathcal{R}^{-(k-1)} \mathcal{M} \]

These equations describe the modified discrete system dynamics for the calculation of the SFC of the states of interest – the perfect oscillator. Thus at any switching period,
5.3 Two Phase Inverter Modelling

For the full development of the new ATAM modelling, a two phase inverter has been chosen as shown in Figure 5.3. A two-phase inverter is analyzed, as opposed to a three-phase inverter, for two reasons. First, the two phases are orthogonal, meaning each phase represents the $\alpha$ and $\beta$ axis respectively. This, as will be seen later, simplifies
the construction of the system model (and matrices) as there is no coupling between the states on different axis. Second, the switching events between phases are uncoupled – a switching sequence in phase $a$, does not affect the current in phase $b$ and vice versa. Suppose, as per Figure 5.3:

\[
v_o(t) = v_\alpha(t) + jv_\beta(t) \tag{5.25a}
\]
\[
i_o(t) = i_\alpha(t) + ji_\beta(t) \tag{5.25b}
\]

Assuming the AC side voltage is ideal, its dynamics can be written as:

\[
\begin{bmatrix}
\frac{dv_\alpha}{dt} \\
\frac{dv_\beta}{dt}
\end{bmatrix}
= \begin{bmatrix}
0 & -w_o \\
w_o & 0
\end{bmatrix}
\begin{bmatrix}
v_\alpha \\
v_\beta
\end{bmatrix} \tag{5.26}
\]

The objective in the new modelling approach is to augment the system with states that contain the SFC of the output current $i_o(t)$. One of the challenges with this is that the dynamics of $i_o(t)$ depend on the switching events in phase $a$, and phase $b$. This means that depending on the the switching events for both phases, up to six subintervals, varying in length, are created so that the evolution of the output current can be computed. Figure 5.4 shows this with an arbitrary example of switching times.

The presence of variably long subintervals creates additional computation that’s complex to model, as the length of different switch positions, changes between each switching period. It’s not as simple as DC/DC, where the sequence is always the same (on-off-on...).

### 5.3.1 Decoupled Fourier Transform

The output current can be decoupled into its $\alpha$, and $\beta$ phase. For each of its component currents, the switching order closely resembles the DC/DC case. In fact it is precisely the same. As such, decoupling the output current into its phase components helps in keeping the model and calculations relatively simple. The same motivation exists to decouple the
Figure 5.4: Shown here is how the different switching times in both phases can lead to many different circuit states.

SFC calculation of the output current. As in Section 5.2:

\[
I_\alpha^1 + jI_\beta^1 = \frac{1}{T_{ac}} \int_{t-T_{ac}}^{t} i_\alpha(\tau) e^{-j\omega_\alpha \tau} d\tau
\]  

(5.27)

This equation forces the dynamics of the \( \alpha \) and \( \beta \) components to be coupled together in order to calculate the SFC of the output current. Since the goal is to decouple this calculation as well, (5.27) is re-written into two equations splitting it into the two phase currents:

\[
I_{\alpha, re}^1 + jI_{\alpha, im}^1 = \frac{1}{T_{ac}} \int_{t-T_{ac}}^{t} i_\alpha(\tau) e^{-j\omega_\alpha \tau} d\tau \]  

(5.28a)

\[
I_{\beta, re}^1 + jI_{\beta, im}^1 = \frac{1}{T_{ac}} \int_{t-T_{ac}}^{t} i_\beta(\tau) e^{-j\omega_\alpha \tau} d\tau \]  

(5.28b)
where the equations calculate the SFC of the output current through its phase \( \alpha \), and \( \beta \) respectively. Solving for them separately, as was done for a general signal in Section 5.2, their solution can be written:

\[
\frac{d}{dt} \left( I_{1,\alpha, re}(t) + j I_{1,\alpha, im}(t) \right) = jw(I_{1,\alpha, re}(t) + j I_{1,\alpha, im}(t)) + \frac{1}{T_{ac}} e^{-j\omega T_{i\alpha}}(t) \quad (5.29a)
\]

\[
\frac{d}{dt} \left( I_{1,\beta, re}(t) + j I_{1,\beta, im}(t) \right) = jw(I_{1,\beta, re}(t) + j I_{1,\beta, im}(t)) + \frac{1}{T_{ac}} e^{-j\omega T_{i\beta}}(t) \quad (5.29b)
\]

Following the development of Section 5.2.1 and the example of the perfect oscillator, both systems (\( \alpha \), and \( \beta \) dynamics) can be re-written so that the fundamental contribution for each system is:

\[
C_{1,\alpha}^1[k] = \hat{y}_{\alpha,1}[k] + j \hat{y}_{\alpha,2}[k] \quad (5.30a)
\]

\[
C_{1,\beta}^1[k] = \hat{y}_{\beta,1}[k] + j \hat{y}_{\beta,2}[k] \quad (5.30b)
\]

where the notation is consistent with Section 5.2.1 – \( \hat{y}_\alpha \) refers to the output vector of the modified discrete system of the \( \alpha \)-frame, \( \hat{x}_\alpha \) refers to the state vector of the modified discrete system of the \( \alpha \)-frame. The same notation applies for the \( \beta \)-frame. The SFC of both frames as per above equations, can be combined together so that the fundamental contribution (SFC) of the whole system for the state of interest is calculated; that is the solution to (5.27). This can be easily done. As shown in Figure 5.5, \( \beta \)-frame lags the \( \alpha \)-frame by \( \frac{\pi}{2} \). Since the SFC coefficients in (5.30a) and (5.30b) are independent, the total system’s SFC coefficient is as follows:

\[
C_{tot}^1[k] = C_{\alpha}^1[k] + jC_{\beta}^1[k] \quad (5.31)
\]

With the decoupling of the SFC calculation, the two-phase inverter example can now be fully developed. Following the circuit in Figure 5.3 the dynamics for the on and off
Figure 5.5: β lags α by $\frac{\pi}{2}$ or $\frac{1}{4}$ of line period.

The switch states on the α, and β leg are:

\[
L \frac{d}{dt} i_\alpha(t) = \begin{cases} 
\frac{V_{dc}}{2} - v_\alpha(t), & \text{if } S_\alpha(t) = 1 \\
-\frac{V_{dc}}{2} - v_\alpha(t), & \text{otherwise}
\end{cases}
\]  \hspace{1cm} (5.32a)

\[
L \frac{d}{dt} i_\beta(t) = \begin{cases} 
\frac{V_{dc}}{2} - v_\beta(t), & \text{if } S_\beta(t) = 1 \\
-\frac{V_{dc}}{2} - v_\beta(t), & \text{otherwise}
\end{cases}
\]  \hspace{1cm} (5.32b)

The system dynamics for $\alpha$:

\[
\mathbf{x}_\alpha(t) = \begin{bmatrix} 
i_\alpha(t) \\
V_{dc} \\
v_\alpha(t) \\
v_\beta(t) \\
I_{\alpha,re}(t) \\
I_{\alpha,im}(t)
\end{bmatrix}
\]  \hspace{1cm} (5.33a)
Chapter 5. DC/AC Modelling

\[ A_{\alpha, on} = \begin{bmatrix}
0 & \frac{1}{2L} & -\frac{1}{L} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -w_o & 0 & 0 \\
0 & 0 & w_o & 0 & 0 & 0 \\
\frac{1}{T_{ac}} \cos(w_o T) & 0 & 0 & 0 & 0 & -w_o \\
-\frac{1}{T_{ac}} \sin(w_o T) & 0 & 0 & 0 & w_o & 0 \\
\end{bmatrix} \] (5.33b)

\[ A_{\alpha, off} = \begin{bmatrix}
0 & -\frac{1}{2L} & -\frac{1}{L} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -w_o & 0 & 0 \\
0 & 0 & w_o & 0 & 0 & 0 \\
\frac{1}{T_{ac}} \cos(w_o T) & 0 & 0 & 0 & 0 & -w_o \\
-\frac{1}{T_{ac}} \sin(w_o T) & 0 & 0 & 0 & w_o & 0 \\
\end{bmatrix} \] (5.33c)

\[
\frac{d}{dt} x_\alpha(t) = \begin{cases} 
A_{\alpha, on} x_\alpha(t), & \text{if } c_\alpha(t) = 1 \\
A_{\alpha, off} x_\alpha(t), & \text{otherwise}
\end{cases} \] (5.33d)

Similarly for \( \beta \):

\[
x_\beta(t) = \begin{bmatrix} 
i_\beta(t) \\
V_{dc} \\
v_\alpha(t) \\
v_\beta(t) \\
I_{\beta, re}(t) \\
I_{\beta, im}(t)
\end{bmatrix} \] (5.34a)
Assuming SPWM for the duty cycle, during a single switching period, there will be three distinct subintervals, where the primary switches are on-off-on. Figure 5.6 shows the reason behind this.

Following the above notation, after a single switching period the evolution of the state vector can be written as:

\[
\mathbf{x}_\alpha((k+1)T) = e^{\mathbf{A}_{\text{on},\alpha}D_{3,\alpha}[k]T}e^{\mathbf{A}_{\text{off},\alpha}\bar{D}_{2,\alpha}[k]T}e^{\mathbf{A}_{\text{on},\alpha}D_{1,\alpha}[k]T}\mathbf{x}_\alpha(kT)
\]

(5.35)

where \(D_{1,\alpha}[k]\) represents the first ‘on’ part of the \(k^{th}\) switching period; \(\bar{D}_{2,\alpha}[k]\) represents the ‘off’ duration of the switch during the \(k^{th}\) switching period. Sampling this, with
Figure 5.6: Figure shows how the duty cycle is calculated for SPWM for $\alpha$ subsystem. Here $m_f = 2 \left( \frac{T_{ac}}{T} \right)$ for visualization purposes. For each switching period there is on-off-on period.
period $T$, results in:

$$
\Phi_\alpha [k] = e^{A_{on,\alpha}D_{3,\alpha}[k]T} e^{A_{off,\alpha}D_{2,\alpha}[k]T} e^{A_{on,\alpha}D_{1,\alpha}[k]T}
$$

(5.36a)

$$
x_\alpha [k + 1] = \Phi_\alpha [k] x_\alpha [k]
$$

(5.36b)

For this system, $\Phi_\alpha$ changes between switching periods, whereas for the perfect oscillator example in Section 5.2.1 it didn’t. As a result the FCM is a function of the switching period.

$$
\Phi_\alpha [k] = 
\begin{bmatrix}
D_\alpha [k] & 0 \\
M_\alpha [k] & R
\end{bmatrix}
$$

(5.37)

such that, the new modified discrete system can be written as:

$$
\hat{x}_\alpha [k] = 
\begin{bmatrix}
x_{\alpha,1}(kT) \\
x_{\alpha,2}(kT) \\
x_{\alpha,3}(kT) \\
x_{\alpha,4}(kT)
\end{bmatrix}
$$

(5.38a)

$$
\hat{\Phi}_\alpha [k] = D_\alpha [k]
$$

(5.38b)

$$
\hat{x}_\alpha [k + 1] = \hat{\Phi}_\alpha \hat{x}_\alpha [k]
$$

(5.38c)

$$
\hat{y}_\alpha [k] = \Gamma_\alpha [k] \hat{x}_\alpha [k]
$$

(5.38d)

$$
\Gamma_\alpha [k] = R^{k-1} M_\alpha [k]
$$

(5.38e)

The above equations (5.38a)–(5.38e) describe the full system dynamics of phase $\alpha$ of the converter as well as the SFC states needed to calculate the coefficient of the fundamental Fourier transform of the $\alpha$ output current ($i_\alpha$). The above steps follow for phase $\beta$. The equations are identical as (5.38a)–(5.38e), with the respective $\beta$ matrices, and switching times. Finally the SFC for the total output current $i_\alpha$ at the end of each switching period
is available:

\[
C_{\alpha}^1[k] = \hat{y}_{\alpha}[k] \tag{5.39a}
\]

\[
C_{\beta}^1[k] = \hat{y}_{\beta}[k] \tag{5.39b}
\]

\[
I_\alpha^1[k] = I_{\alpha}^1[k] + jI_{\beta}^1[k] = C_{\alpha}^1[k] + jC_{\beta}^1[k] \tag{5.39c}
\]

This two-phase inverter example has helped in establishing the procedure for calculating the SFC of the output current, but it can be modified for any current/voltage of interest. To put this whole procedure in perspective, a flowchart of it is shown (Figure 5.7).

Although the two-phase inverter is a subset of inverter topologies, it helps in demonstrating the new modelling approach of inverters. There are no assumptions with respect to the development of the modelling that limit it to two-phase topologies and the proposed methodology in fact can be applied to any dc/ac converter with any number of phases.

### 5.4 Model Demonstration

In this section the new AC modelling approach is demonstrated. Instead of taking an inverter topology, the system under consideration will be an ideal oscillator. The signal is 100\% fundamental component – it has no harmonics. The Fourier transform of an ideal oscillator is easy to calculate and use as validation for the results given by the new model.
Chapter 5. DC/AC Modelling

Circuit Dynamics

Split into $\alpha$, $\beta$ subsystems

Get switching times for $\alpha$

Augment with SFC state for $I_{1\alpha}$

Get $\Phi_{\alpha}[k]$, $M[k]$, $\hat{y}_{\alpha}[k]$

$\forall k$

Get $\Phi_{\beta}[k]$, $M[k]$, $\hat{y}_{\beta}[k]$

$\forall k$

$\forall k$

Figure 5.7: Flow chart of the new modelling approach for DC/AC inverter
Figure 5.8: The evolution of the signal’s components over several cycles.

5.4.1 Pure Sine Wave

As per Section 5.2.1, the complete model for the example of perfect oscillator is:

\[
\hat{x}[k] = \begin{bmatrix} x_1[k] \\ x_2[k] \end{bmatrix}
\]  \hspace{1cm} (5.40a)

\[
\hat{\Phi} = R
\]  \hspace{1cm} (5.40b)

\[
\hat{x}[k+1] = \hat{\Phi} \hat{x}[k]
\]  \hspace{1cm} (5.40c)

\[
\hat{y}[k] = \Gamma \hat{x}[k]
\]  \hspace{1cm} (5.40d)

\[
\Gamma = R^{k-1} M
\]  \hspace{1cm} (5.40e)

\[
R = \begin{bmatrix} \cos(w_o T) & \sin(w_o T) \\ -\sin(w_o T) & \cos(w_o T) \end{bmatrix}
\]  \hspace{1cm} (5.40f)

The system is simulated with the standard parameters as specified \((T = \frac{T_{ac}}{80})\). The complex fundamental Fourier coefficient is plotted as a vector, with its real and imaginary components as the basis.

As the Figure 5.9 shows the SFC vector is constant for all switching periods. This is
Figure 5.9: The SFC of the perfect oscillator is constant in all subintervals as represented by the constant point in this plot.
in agreement with (5.3). During each subinterval the vector is $0.0125 + j0$. Comparing this to the evaluation of the SFC integral:

$$X^1(t) = \frac{1}{T_{ac}} \int_{t-T}^{t} 1d\tau = \frac{1}{T_{ac}} \tau|_0^T = \frac{T}{T_{ac}} = 0.0125$$  \hspace{1cm} (5.41)

The same result is obtained. Therefore for the perfect oscillator example, the new modelling and SFC states give the correct results.

### 5.4.2 Oscillator with Negative Sequence

The next step in the validation is to take a look at SFC calculation of a signal that has harmonics. One such signal is an unbalanced oscillator. It is shown in Figure 5.10. It can be written as:

$$x(t) = A_1\cos(w_0t) + jA_2\sin(w_0t)$$ \hspace{1cm} (5.42)

The amplitude difference between the $\alpha$ and $\beta$ components of the signal introduces
negative sequence of the fundamental. Again the modelling procedure is the same as the previous example. The steps will be omitted here. The output of the new system with the simulation of the system for number of switching periods. This is shown in Figure 5.11.

To compare this to the actual result, note that the signal presented can be written as:

$$x(t) = \frac{1}{2} (e^{j\omega t} + e^{-j\omega t}) + \frac{j2}{2j} (e^{j\omega t} e^{-j\omega t})$$

$$x(t) = 1.5e^{j\omega t} - 0.5e^{-j\omega t}$$  \hspace{1cm} (5.43a) \hspace{1cm} (5.43b)

In other words the signal consists of positive rotating vector $+\omega$ with amplitude of 1.5, and smaller negative rotating vector $-\omega$ with amplitude of 0.5. This matches the plot of Figure 5.11. In the figure, the SFC is shown which is the fundamental component divided into the number of subintervals, in this case 80. The centre of the circle is fundamental due to the positive rotating vector, and the rotation around the centre point is due to the negative rotating vector.

In this section the validity of the AC modelling, specifically the SFC states has been established. For two simple examples, it was shown that indeed the model is able to calculate the exact SFC. This is a significant achievement, and one that has not been previously made. The modelling for DC/AC converters is just a foundation on which other works can build, specifically in designing more accurate controllers. This is an important stepping stone towards that goal.

### 5.5 Summary

This chapter showed the development and validation of the new modelling approach, ATAM, for DC/AC. Specifically, the validity of the augmented SFC states was verified.
Figure 5.11: The evolution of the fundamental vector over several cycles. It traces out circle around the centre point of $0.01875 + j0$. This is the subinterval fundamental contribution vector.
Given simple examples of signals for which the SFC is easily calculated, the output of the new modelling was matched against the expected results. The model showed exact precision as expected. As mentioned in the introduction of the chapter, the ATAM for DC/AC is also LTV. Control design applications were left out of this thesis, and would be the focus of future works. The progress up to that model, though, is significant and allows for future works to focus mainly on solving that challenge.
Power converters are an integral part of the utility, and portable electronics. They allow for efficient transfer of power from differing voltage levels and type of current. It is this unique ability and versatility that makes them very useful. Understanding how power converters work, and having a mathematical model is crucial for good controller design that gives stable and reliable converter performance.

Currently, SSA is a popular modelling method for converters. It allows for fast model derivation and, since the model is LTI, controller design is straightforward. PCM is another approach to modelling a converter. It is a convenient model, as the controller is implemented digitally as well. It also allows for fast deadbeat control design. However, the PCM can suffer from significant aliasing problems and, as a result, is seldom employed in practice.

SSA is more widely used, even though it has its weaknesses. As was seen in Chapter 2, the averaging approximation upon which it is derived leads to modelling inaccuracy. The main reason behind this is the exclusion of an averaging state in the model. The averaging upon which the SSA model is based should result in a model lag, rather than a model lead. Implementation of the averaging assumption introduces further inaccuracy. Conventional implementation uses a LPF to filter switching dynamics. The addition of a
LPF in the system without modelling its own dynamics leads to deteriorated controller performance, when compared to the theoretical. Specifically LPF leads to a trade-off between good transient and good steady state response.

The issues that present modelling techniques face is the motivation for this thesis. A new modelling approach was needed that models the full dynamics of the system. Specifically, the averaging dynamics need to be part of the model that the controller is designed for.

In Chapter 3, a new model, ATAM, was developed for DC/DC converters. The model augments the system dynamics with a SWA, that precisely models the averaging dynamics of the system output. The model was demonstrated for its accuracy with a time domain simulation of an example converter topology. The model, which is LTV, is the first to capture all dynamics and be in a mathematical form that can be used with advanced control techniques.

In Chapter 4, the ATAM for DC/DC was linearized and LATAM was developed. Through another simulation, it was shown that the LATAM is more accurate than SSA. Further, a simple controller design example showed that the LATAM based controller implementation performed better under load disturbance than the conventional SSA based controller with LPF.

In Chapter 5, an extension of the ATAM was developed for DC/AC converters. For DC/AC, the main challenge is calculating the fundamental component contribution, of the state, during a subcycle (subinterval). Using the same idea of including the averaging dynamics into the model, a new model was developed that accomplishes this. The dynamics of calculating the SFC were augmented into the original system’s dynamics. The SFC calculation was shown to be accurate with an sine wave example. Further, it was shown how a two phase inverter dynamics can be decoupled into its phases, for easier computation, and then reconstructed back together once the SFC was calculated separately.
6.1 Future Work

The thesis developed a new model, ATAM, for modelling DC/DC and DC/AC power converters. For DC/DC, the model is more accurate and this advantage was evident in the controller performance example. For DC/AC the model includes the dynamics and calculates the SFC, which is needed for control design. The thesis was successful in accomplishing all of its objectives.

The new ATAM for DC/DC and DC/AC is the foundation for future control design application to PWM converters. An advanced control design technique developed based on the ATAM of the converter, would allow for enhanced static and dynamic performance of the converters. With the advantages of the new model, future works are left to focus on the application and controller design task.
Chapter 7

Appendix

7.1 Capless Buck Converter with Resistive Load

Consider a buck converter with no capacitor, and a resistive load, as shown in Figure 7.1. The circuit parameters are given in Table 7.1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_g$</td>
<td>10 V</td>
</tr>
<tr>
<td>$R$</td>
<td>2.5 Ω</td>
</tr>
<tr>
<td>$F_{sw}$</td>
<td>50 kHz</td>
</tr>
<tr>
<td>$w_{sw}$</td>
<td>$2\pi \times 50000$ rad/s</td>
</tr>
<tr>
<td>$L$</td>
<td>0.25 mH</td>
</tr>
<tr>
<td>$D$</td>
<td>0.5</td>
</tr>
</tbody>
</table>
7.1.1 SSA Model

Model Derivation

The SSA model is derived for the given converter topology and parameters. The states are defined first:

\[ x(t) = i_o(t) \quad (7.1a) \]
\[ u(t) = v_g(t) \quad (7.1b) \]
\[ y(t) = x(t) \quad (7.1c) \]

\[ \frac{d}{dt} x(t) = A_{on} x(t) + B_{on} u(t) \quad (7.1d) \]
\[ \frac{d}{dt} x(t) = A_{off} x(t) + B_{off} u(t) \quad (7.1e) \]

\[ A_{on} = A_{off} = -\frac{R}{L} \quad (7.1f) \]
\[ B_{on} = \frac{1}{L} \quad (7.1g) \]
\[ B_{off} = 0 \quad (7.1h) \]
Based on the operating point let:

\[
A = DA_{on} + (1 - D) A_{off} \quad (7.2a)
\]
\[
B = DB_{on} + (1 - D) B_{off} \quad (7.2b)
\]
\[
C = DC_{on} + (1 - D) C_{off} \quad (7.2c)
\]

The linearized, small-signal model is then written as:

\[
\frac{d}{dt} \tilde{x}(t) = A\tilde{x}(t) + B\tilde{u}(t) + [(A_{on} - A_{off})X + (B_{on} - B_{off})U]\tilde{d}(t) \quad (7.3a)
\]
\[
\tilde{y}(t) = C\tilde{x}(t) + E\tilde{u}(t) + [(C_{on} - C_{off})X + (E_{on} - E_{off})U]\tilde{d}(t) \quad (7.3b)
\]

After evaluating all parameters the system can be written as:

\[
\frac{d}{dt} \tilde{x}(t) = -10000\tilde{x}(t) + 2000\tilde{u}(t) + 40000\tilde{d}(t) \quad (7.4a)
\]
\[
\tilde{y}(t) = \tilde{x}(t) \quad (7.4b)
\]

### 7.2 Capless Buck Converter with Motor Load

Consider a buck converter with no capacitor, and a motor load (constant voltage) as shown in Figure 7.2.

The parameters of the circuit are established (and \( T = \frac{1}{F_{sw}} \)) in Table 7.2. The objective is be the control of the output current. The modelling approach will be based upon this goal.
7.2.1 New Model

Model Derivation

The procedure outlined in Section 3.1 is followed for the derivation of the new model for the shown topology. Let:

\[
x(t) = \begin{bmatrix}
i_a(t) \\
i_{av}(t) \\
v_g(t) \\
v_o(t)
\end{bmatrix}
\]  \hspace{1cm} (7.5)

where \( i_{av}(t) \) represents the augmented SWA state. Then the dynamics are given as:

\[
x((k + 1)T) = e^{\hat{A}_{off}T}e^{\hat{A}_{on}d(t)T}x(t)
\]  \hspace{1cm} (7.6)
Table 7.2: Circuit Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_g$</td>
<td>10 V</td>
</tr>
<tr>
<td>$V_o$</td>
<td>5 V</td>
</tr>
<tr>
<td>$F_{sw}$</td>
<td>50 kHz</td>
</tr>
<tr>
<td>$w_{sw}$</td>
<td>$2\pi$ 50 000 rad/s</td>
</tr>
<tr>
<td>L</td>
<td>0.25 mH</td>
</tr>
<tr>
<td>D</td>
<td>0.5</td>
</tr>
</tbody>
</table>

where:

$$\hat{A}_{on} = \begin{bmatrix} 0 & 0 & \frac{1}{L} & -\frac{1}{L} \\ \frac{1}{T} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$  \( (7.7) \)

$$\hat{A}_{off} = \begin{bmatrix} 0 & 0 & 0 & -\frac{1}{L} \\ \frac{1}{T} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$  \( (7.8) \)

In discrete form:

$$x[k+1] = \Phi[k]x[k]$$  \( (7.9) \)

This system describes the exact dynamics of the circuit. It is therefore referred to as the exact switching model. The system is linearized around the operating point (Table 7.2).

The final results are:

$$\ddot{x}[k+1] = A_d \ddot{x}[k] + B_d \ddot{d}[k]$$  \( (7.10a) \)

$$\ddot{y}[k] = C \ddot{x}[k]$$  \( (7.10b) \)
\[ A_d = \begin{bmatrix} 1 & 0 & 0.04 & -0.08 \\ 1 & 0 & 0.03 & -0.04 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \] (7.10c)

\[ B_d = \begin{bmatrix} 0.8 \\ 0.4 \\ 0 \\ 0 \end{bmatrix} \] (7.10d)

\[ C_d = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} \] (7.10e)

\[ \dot{x} = x[k] - X \] (7.10f)

There are couple remarks to be made. First, the SWA state was augmented as a second state, and not at the end. This will be make things easier when designing the controller, especially for pole placement. On point, the last states, the voltage input and output are really not states in the sense they can be controlled. They will be excluded from the feedback control. They’re included here in order to derive the full system dynamics and because the input voltage is actually needed for the estimator design.

**Controller Design**

For the controller design a root locus method is used. The system in (7.10) can be written as a discrete time transfer function:

\[ P(z) = \frac{2(z + 1)}{5(z)(z - 1)} \] (7.11)

This transfer function can be used in a root locus design method. With a digital
Figure 7.3: The closed loop structure for the given example. Plant and controller are all in discrete time. The output of the plant is the sampled average output current.

controller, the feedback is presented in Figure 7.3. The closed loop transfer function is:

\[ T(z) = \frac{P(z)C(z)}{1 + P(z)C(z)} \]  \hspace{1cm} (7.12)

The controller will be a PI. As such its transfer function is:

\[ C(z) = K_p + \frac{K_i}{z - 1} \]  \hspace{1cm} (7.13)

Since the pole is fixed for a PI controller, the only design choice is the placement of the closed loop zero and the associated controller gains.

Figure 7.4 shows the root locus plot of the closed loop for a chosen controller. Unlike continuous time root locus design, the goal is to ensure all closed loop poles are inside the unit circle. Suitably, a zero has been placed at 0.8 on the real axis. The complex conjugate poles have been placed for a good damping performance. The final controller transfer function is therefore given by:

\[ C(z) = 1.86 + \frac{0.372}{z - 1} \]  \hspace{1cm} (7.14)
Figure 7.4: The root locus of the closed loop. The controllable poles are all inside the unit circle.

7.2.2 SSA Model

Model Derivation

As in the previous section, the SSA is derived for the given converter topology and parameters. The states are defined first:

\[ x(t) = i_o(t) \] (7.15a)
\[ u(t) = v_g(t) \] (7.15b)
\[ y(t) = x(t) \] (7.15c)
\[ \frac{d}{dt} x(t) = A_{on} x(t) + B_{on} u(t) \] (7.15d)
\[ \frac{d}{dt} x(t) = A_{off} x(t) + B_{off} u(t) \] (7.15e)
\[ A_{on} = A_{off} = 0 \] (7.15f)
Based on the operating point let:

\[ A = DA_{on} + (1 - D) A_{off} \]  \hspace{1cm} (7.16a)

\[ B = DB_{on} + (1 - D) B_{off} \]  \hspace{1cm} (7.16b)

\[ C = DC_{on} + (1 - D) C_{off} \]  \hspace{1cm} (7.16c)

The linearized, small-signal model is then written as:

\[
\frac{d}{dt} \tilde{x}(t) = A \tilde{x}(t) + B \tilde{u}(t) + [(A_{on} - A_{off})X + (B_{on} - B_{off})D] \tilde{d}(t) \]  \hspace{1cm} (7.17a)

\[
\tilde{y}(t) = C \tilde{x}(t) + E \tilde{u}(t) + [(C_{on} - C_{off})X]D] \tilde{d}(t) \]  \hspace{1cm} (7.17b)

After evaluating all parameters the system can be written as:

\[
\frac{d}{dt} \tilde{x}(t) = 2000 \tilde{u}(t) + 40000 \tilde{d}(t) \]  \hspace{1cm} (7.18a)

\[
\tilde{y}(t) = \tilde{x}(t) \]  \hspace{1cm} (7.18b)

For this whole modelling to work, there needs to be a LPF that enacts the assumptions under which the model was derived. The LPF chosen is second order, with cutoff frequency which is \( \frac{1}{3} (w_n) \) of the switching frequency. Also its damping ratio is chosen to be 0.7 (\( \zeta \)) based on related in-lab testing. In other words, the transfer function of it:

\[
LPF(s) = \frac{1}{s^2 + \frac{\zeta}{w_n} s + \frac{1}{w_n}} \]  \hspace{1cm} (7.19)
Controller Design

Designing a controller for the SSA model is done commonly through the use of bode plots. Since the dynamics of the plant are known and the feedback LPF has also been chosen, one can do a bode plot of the uncompensated closed-loop. Figure 7.5 shows this. For good control design the closed-loop system’s bode plot is analyzed based on its gain margin (GM) and phase margin (PM). The GM needs to be as small as possible, while PM above 45° is acceptable. Based on these design criteria, a continuous time compensator $C(s)$ is calculated. However the true criteria by which the compensator is judged, is the disturbance rejection of the whole system. In the uncompensated case, the disturbance rejection response is shown in Figure 7.6. This is obviously unacceptable.

Using MATLAB one can input the model’s dynamics and optimize $C(s)$ based on
Figure 7.6: The output response to a step disturbance in the input. As it can be seen the output never reaches zero steady state error - it settles to a steady state error.

having the best disturbance rejection. Doing this leads to the following compensator:

$$ C(s) = \frac{16695 \left(1 + 9 \times 10^{-5}s\right)}{s} $$  

(7.20)

Because the controller is being implemented digitally, the above compensator is discretized, with sample rate equal to the switching frequency:

$$ C(z) = \frac{1.669z - 1.336}{z - 1} $$  

(7.21)

The new compensated disturbance rejection is shown in Figure 7.7.
Figure 7.7: The step response for the newly compensated system. The output returns to zero steady state error within 0.4 ms.
Bibliography


