Autonomous model predictive control for realization of smart inverters at the grid-edge

by

Mitchell Thomas Easley

B.S., Kansas State University, 2018

A THESIS

submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Mike Wiegers Department of Electrical and Computer Engineering
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2020

Approved by:

Co-Major Professor
Dr. Mohammad Shadmand

Approved by:

Co-Major Professor:
Dr. Haitham Abu-Rub

Copyright

Abstract

The continuous increase in feasibility of renewable energy and renewable energy assets has opened the door to a future grid dominated by renewable-based power electronics converters, rather than electric machines powered by fossil fuels. Simultaneously, increases in practical processor clock speed enables new, advanced control techniques for such devices. The sampling rate of digital controllers for power converters is now able to be set within the same order of magnitude as the converter's power semiconductor switching frequency, and even greater. This has created the possibility for the controller to perform online decision-making. This is realized in a control technique called model predictive control, more specifically, *finite-set* model predictive control. In finite-set model predictive control, the control will evaluate each of its available control actions and select that which achieves *best performance*. This evaluation of possible control actions requires making a prediction about each control action based on an internal model of the controlled system. To define *best performance*, the control contains objectives and their associated reference value, or value which is considered optimal. The control action with predictions aligned most closely with the reference are selected as the next-up control action. The ability to individually evaluate and select potential control actions presents transient responses faster than can be seen in control systems that incorporate typical linear controllers and modulator-based switching. It also allows for single-loop multi-objective control and the possibility to program grid-tied inverters with enhanced system awareness, as the controller can note performance that results from its decided control action. In this thesis, I present finite-set multi-objective model predictive control for multiple grid-tied power electronics converters. I introduce practical enhancements to the finite-set model predictive control paradigm which can remove the controller design stage, a generally tedious and ambiguous for predictive controllers. I show how hierarchical objective

tracking makes it possible to retain fast controller sampling rates on converter topologies with especially large control sets. Finally, I introduce solutions which enables real time model alignment, fault tolerant operation and situational awareness of the converter. These enhancements to predictive control can ensure that the smart inverters of tomorrow’s grid are fast, aware, and reliable.

Table of Contents

List of Figures

List of Tables

List of Algorithms

Acknowledgements

I am grateful for my co-advisor Dr. Mohammad Shadmand, who made my work possible and who pushed me to improve every step of the way. Without the opportunities he gave and chances he took for me, I would not be where I am today. I would like to also thank my co-advisor Dr. Haitham Abu-Rub. Dr. Abu-Rub was incredibly supportive throughout my graduate studies and especially during my time as a visiting researcher in Qatar. He felt like an uncle, making Qatar feel a little more like home, and his kindness and intellect will never be forgotten. Also, I want to thank the other members of my supervisory committee, Dr. Hongyu Wu and Dr. Warren White, for their guidance and support. My first ever published research work was with Dr. Wu, and my successful experience with him gave me great confidence as I entered my graduate studies. I became inspired by Dr. White's shrewdness and curiosity, and his rigorous nonlinear controls course was perhaps the most enjoyable academic course I have taken.

I want to thank my research colleagues at Kansas State, of whom are a very bright and inquisitive team which I was lucky to be a part of. The support we shared as a group taught me the power of unity. I must thank my mother and father for everything. Their undying support and love gave me great strength through it all. Finally, I need to thank my partner, Susan, who has always given me incredible support, love, and encouragement. Only one thing is certain: I could not have succeeded without her.

This thesis was made possible by NPRP-EP grant # X-033-2-007 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the author.

Chapter 1 - Introduction

1.1 Background

Renewable-based energy technology has seen considerable growth over the last few decades [1, 2]. This can be attributed to advancements in renewable energy technology that has increased efficiency and overall viability, and to worldwide climate-conscious initiatives such as the Paris Climate Agreement [3]. The global trend of steady increases in renewable-based power generation facilities, including photovoltaic (PV) based power conversion, suggests renewable energy technology will expand its market share. Presently, PV power generation has a unique characteristic of widespread residential installation, as new businesses emerge to provide plug-and-play PV solutions to customers. Normal residential consumers can now act as active participants in the energy market; they have moral and financial incentives to do so, as they can both align their actions with the aforementioned climate-conscious initiatives and reduce their energy expenses by generating their own power and occasionally back feed to the grid.

These trends of increasing market share of renewables in the energy market and increased residential participation leads many researchers to consider the possibility of a power-electronics dominated grid [4]. That is, a grid driven by renewable distributed energy resources (DERs), rather than large, centralized power plants where power is generated from electric machines. Such changes in the grid infrastructure come with unprecedented challenges, challenges which must be met both by standards of grid-connected power converters and their controls. Specifically, the lack of coordination among the oncoming DERs creates risk of grid instability in a power electronics dominated grid. For instance, the European 50.2 Hz problem refers to the risks associated with the European standard which PV installations are set to disconnect from the grid, or island themselves, as a protective feature. In such a scenario, the disconnection of many PV installations in a power

electronics dominated grid could lead to a sudden and drastic imbalance between generation and demand that cannot be compensated by online generation reserve [5]. Similar conditions can arise when grid or bus voltage falls below its nominal value, referred to as a voltage *sag*. These can occur from downstream short-circuit faults, overloading (insufficient power generation to match power demand), and the starting of large motors [6]. PV inverters will disconnect if the sensed grid voltage falls below its nominal value for an extended period. Thus, successful realization of a power electronics dominated grid demands advanced, grid-supporting control of DERs.

The risks of a power electronics dominated grid can be mitigated with the next generation of grid-connected inverters, often referred to as *smart inverters*. Smart inverters are most often defined as inverters that support grid stability through enhanced control functionality. Such a device will perform high quality power conversion with situational awareness to adjust its behavior. Figure 1.1 illustrates the distinction between a smart inverter and normal grid-connected inverters. With a normal grid-connected inverter, the power it injects to the grid depends exclusively on the power that can be harvested. In the case of PV inverters, the power electronics interface simply retains maximum power point tracking (MPPT) mode. If the grid undergoes the aforementioned fault conditions, it is able to disconnect from the grid as a protective feature. However, this creates an unreliable power source, and high penetration of inverters with such limited functionality would increase the volatility of the grid's stability. Oppositely, a smart inverter can adjust is active power setpoint, and inject and absorb reactive power. The active and reactive power setpoints can either be adjusted internally by sensing and interpreting the grid condition or can be commanded from a superordinate control scheme. The latter assumes a coordinated *fleet* of smart inverters. This adjustment in behavior occurs through various modes of operation which fall under the term *fault ride through*. Fault ride through occurs in a brief interval

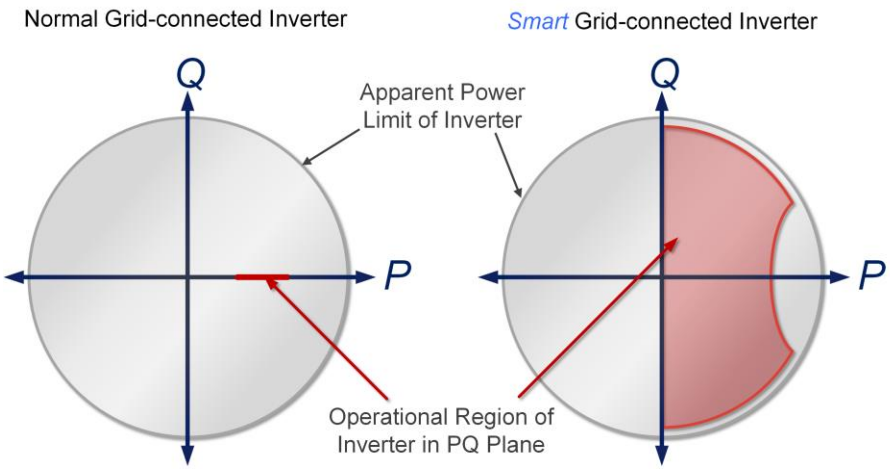


Figure 1.1: Active and reactive power (PQ) region of operation for normal grid-connected inverters and smart inverters.

between sensing abnormal grid conditions and disconnection from the grid. Rather than immediate isolation from the grid, the smart inverter will retain connection to the grid and offer dynamic grid support. The grid-supporting behavior will vary according to the faulted condition, such as high/low voltage ride through and frequency ride through. With high penetration of smart inverters in the grid, conditions that would cause inverters to trip could be mitigated and avoided altogether. Note that the apparent power may not be able to reach the inverter's physical apparent power rating, but this depends on the surrounding grid infrastructure. Similarly, the active power (at high power factor) may be limited according to its DC link voltage/peak grid voltage ratio and grid impedance, but further discussion is outside the scope of this work. Smart inverters are merely defined according to the grid-supporting features they offer, and thus a multitude of publications have been produced proposing control schemes to enable grid-supporting features of smart inverters. Though the control of industrialized grid-connected power converters is proprietary, several advanced control techniques have been proposed in literature, comprising the state of the art. A comprehensive review of grid-supporting functionalities and proposed methodologies for smart inverters is provided in [7].

Finite-set model predictive control (MPC) is an appealing control technique to realize smart inverters. Firstly, finite-set MPC generally has fast dynamic response when compared to classical control techniques which incorporate linear controllers [8]. Fast dynamic response is critical for smart inverters, as inverters are allotted a brief period of time before islanding to offer grid-supporting functionality. The faster the smart inverter's active and reactive setpoints can be adjusted to offer grid support, the more likely a trip-invoking fault scenario can be avoided altogether in a power electronics dominated grid. When implementing a linear control in power electronics applications, there tends to be a tradeoff between transient performance (settling time during transient conditions) and subharmonic oscillation of the control action at steady state. For instance, a larger integral compensation can quickly adjust the control action but tends to marginalize the stability of the control. Such a tradeoff is not inherent to control schemes which implement MPC. The improvement in dynamic response that can be seen with finite-set MPC is demonstrated in the next chapter. Secondly, finite-set MPC can produce high power quality with higher converter efficiency than when employing a pulse width modulation (PWM) based switching scheme. That is, when equalizing the average switching frequency of finite-set MPC with the fixed switching frequency of a PWM-based switching scheme, the total harmonic distortion (THD) of the injected current will be lower when using the finite-set MPC scheme. This is intuitive when realizing the finite-set MPC undergoes online decision making to determine the proper control action to make the current reach closest to its reference in the next sampling instant. Linear controllers must be tuned to minimize subharmonic content of its output modulation signal, which contributes with the modulator's switching harmonics to distort the inverter's output current. It must be noted that finite-set MPC implements a variable switching frequency and must be averaged over time to understand the frequency of switching via its average switching behavior.

Nonetheless, if the average switching frequency is less than the modulator's fixed switching frequency, the inverter incurs fewer switching losses over time, improving its efficiency. Lastly, with MPC's employment of an overall cost function several objectives can be included in the control scheme in a single loop fashion, rather than nested control loops that are typically seen in classical multi-objective control schemes.

In MPC, the control action is decided by predicted behavior of the controlled variables, also known as control objectives. The predictions in MPC are computed using a mathematical model of the overall system, in which the behavior of the control objectives can be predicted at the next sampling instant. MPC is parsed into continuous control sets and discrete (or finite) control sets. The phrase *finite-set MPC* is most often used, rather than *discrete-set MPC*. Perhaps this is to emphasize that no optimization technique is used in the finite-set MPC paradigm; the entire control set is evaluated in an exhaustive fashion to determine the optimal control action. In the field of power electronics, MPC schemes with continuous control sets typically utilize MPC to generate a modulation index [9], a signal which is applied to comparators for switching the semiconductor gate/base logic via pulse-width modulation [10]. In the space of finite-set MPC, the output or control action is either a space vector [11] which has one or perhaps multiple associated switching sequences (unique combinations of gate logic for each semiconductor in the inverter topology), or a distinct switching sequence. Each control proposal in this thesis implements finite-set MPC (or a similar finite-set optimal control) where the output of the controller is a switching sequence [12, 13]. MPC is a branch within optimal control, as the selected control action optimizes (minimizes) a cost function, typically denoted as J [14, 15]. MPC can incorporate multiple objectives simply by adding its cost for each switching sequence in the cost function. However, in multi-objective control, it is often necessary to adjust the scaling of each control objective, either for normalization

of control objectives or to emphasize tracking of certain objectives over others, which creates ambiguity in the controller design stage [16]. Only recently has MPC, especially finite-set MPC been within the scope of power electronics controls. Upon the original conception of MPC, achieving the practical sampling frequencies for its use in power electronics was not yet possible and was limited to systems with time constants much slower than the power electronics domain [17]. However, the steady increase in processor clock speeds over the past several decades has brought it to power electronics. That is, the state of the art digital processing technologies allow economically-sound embedded systems to be produced which permits the computational burden of MPC at sufficiently high sampling rates. Nonetheless, the finite-set MPC paradigm is computationally expensive, and not practical to realize in all considerable topologies within the field of power electronics. Researchers must enhance the familiar finite-set MPC paradigms to retain the benefits of MPC for their desired applications. For instance, utilizing model redundancies to reduce unnecessary online computations, shrewdly reducing the control set for different converter set points, adjusting the optimization paradigm of finite-set MPC, among others. Smart redesign of the familiar finite-set MPC paradigms will enable smart inverters of tomorrow's grid to fully exploit the benefits that MPC has to offer.

1.2 Scope and Outline of Thesis

In this thesis, I will present control solutions for grid-connected smart inverters using finite-set MPC. Each chapter will detail a novel MPC scheme for a unique grid-connected inverter topology. These chapters also aim to solve some of the well-known drawbacks in practical implementation of finite-set MPC for power electronics. In Chapter 2, a hierarchical MPC framework is proposed and implemented on a grid-connected cascaded multilevel inverter. The hierarchical framework eliminates the ambiguity in designing the overall cost function in model

predictive control. The control optimizes each objective hierarchically, and the designer specifies acceptable tracking behavior of each objective, rather than tuning the weight factors of each objective. This technique enables the inclusion of objectives with volatile magnitudes, allowing a unique objective to be applied which supports equal power draw from the isolated voltage sources. In Chapter 3, an optimal control technique, which is remarkably similar to finite-set MPC, is proposed for a grid-connected cascaded multilevel inverter. The multi-objective control performs similar to a comparable finite-set MPC technique but is able to operate at twice the sampling rate when tested experimentally due to its computational efficiency. An adjustment of the MPC's predictive equation allows the control to refer to a time-variant control set which increases linearly with the number of H-bridges in the topology. This varies remarkably from the exponentially growing control set. The control implements a three-dimensional lookup matrix to optimize grid current in minimal switching events, while cycling through redundant, optimal switching sequences. This equalizes power draw from each isolated voltage source in the cascaded multilevel inverter topology. In Chapter 4, a complete PV-to-grid control scheme is proposed using a quasi Z-source inverter. The quasi-Z-source inverter is capable of voltage boosting and power inversion by implementing a *shoot-through* switching sequence. The control includes a low voltage ride through mode to support the grid via reactive power injection during sensed voltage sags. A 1kW photovoltaic string is controlled using a flexible power point tracking algorithm. This allows the PV power harvesting to be reduced, allowing the control to retain a specific apparent power setpoint or current amplitude. The model predictive control uses an autonomously tuned weight factor technique, in which the weight factors are adjusted according to the minimal cost seen for its respective objective over a moving window. Further, the control objectives are normalized in the cost function using a rolling root mean square (RMS) computation technique. This intuitive

technique removes the weight factor tuning in the control design stage and is shown to improve objective tracking in the presented application, both in steady-state and transient conditions. In Chapter 5, the topology is extended to cascading quasi-Z-source cells, thus referred to as a quasi-Z-source cascaded multilevel inverter. With the control set increasing to twenty-five switching sequences, a computationally-efficient model predictive control scheme is proposed. Objective tracking is achieved hierarchically by constraining the control set for output current according to switching sequences which optimize input current of both cells. The control set is reduced further by applying a *voltage window*, in which the output voltage cannot change more than one voltage level per sampling instant. This is shown to reduce online cost computations by over ninety percent on average. In Chaper 6, the main findings of the thesis are summarized, and I briefly discuss suggested future work, which includes ideas that I have published but must be expanded upon for successful implementation in hardware.

Chapter 2 - Hierarchical Model Predictive Control of Grid-connected Cascaded Multilevel Inverters[1]

2.1 Problem Statement

Multilevel converters are long-studied and a widely accepted class of power converters. When compared to two-level converter topologies, multilevel topologies exhibit improved harmonic content of the output voltage waveform, reducing either the necessary switching frequency or output filtering requirements, and reduced common-mode voltage [18, 19]. They are often proposed in high-power applications, as the increased number of series-connected semiconductor switches increases the voltage rating of the converter [20]. Among these, the cascaded multilevel inverter (CMI), also known as the cascaded H-bridge inverter, consists of series-connected H-bridges, as was first proposed in [21]. The modularity of the CMI deems it more reliable and capable of fault-tolerant operation than other multilevel topologies such as the neutral point clamped inverter and flying capacitor inverter. The CMI topology has been proposed for a wide range of applications, including photovoltaic inverters [22], motor drives [23], and static VAR compensators [24]. Traditional control techniques for the CMI incorporate linear controllers and PWM switching techniques such as phase-shifted PWM, space vector PWM, and subharmonic multilevel PWM [25, 26]. In general, linear controllers lack the fast dynamic response that can be realized with modern, computationally-extensive control techniques. Further, incorporating them into a multi-objective control system generally requires cascading control loops, which is challenging to design and increases complexity.

[1] At the time of writing, the content of this chapter has been accepted as an Institute of Electrical and Electronics Engineers (*IEEE*) manuscript in an upcoming edition of the *Journal of Emerging and Selected Topics in Power Electronics*. The content from the aforementioned manuscript is taken with permission.

MPC is becoming a topic of greater interest in power electronics and has recently achieved adoption in industry [27, 28]. The subset known as finite-set MPC is the most intuitive implementation of MPC, as control actions are considered directly, and eliminates the need of a modulator [8, 29-31]. The phrase "control action" here refers to either a specfic sequence of gate signals applied to the semiconductor switches (referred to as optimal switching sequence MPC), or the space vector that results for a specific sequence of gate signals (referred to as optimal space vector MPC). Besides intuitive implementation, MPC tends to work well as an inner control loop due to its fast dynamic response [11, 32, 33]. Finite-set MPC characteristics can be leveraged to tackle constrained multi-objective control problem challenges within the power electronics space [34]. Thus, finite-set MPC is a potential solution towards resilient power electronics at the grid-edge to enhance the power distribution system resiliency in a straight forward manner [35]. Modern power converters are commonly required to provide auxiliary services which requires implementation of multi-objective control schemes. As such, finite-set MPC is a promising solution towards grid-enhancing power converters with advance functionality.

Despite these benefits, there are challenges in the MPC formulation that have not been fully addressed [8, 36, 37]. In other words, what allows multi-objective MPC to boast its simplicity is what creates difficulty in the design stage. The cost function, while simple to implement, leaves its user to determine how they should weight the objectives in the cost function. Discussion in literature on how to design the weight factors is limited to trial and error techniques for finding an optimal set-point [16, 38]. These techniques require the user to observe the tracking performance of each control objective and decide what is the best behavior. To be succinct, the optimal weight factor ratio of multi-objective MPC tends to be laborious to navigate and difficult to define, particularly for cost functions with more than two objectives. Additionally, a static weight factor

ratio will likely not have optimal performance for all considerable scenarios [39], e.g. in a grid-connected inverter, the control object references should alter due to a grid-fault. The concept of an adaptive weight factor ratio for a power quality compensator was proposed in [40], where the weight factor of each control objective's cost in the overall cost function adapts to the predicted optimal error of each term. Still, such a control provides no guaranteed tracking performance of its objectives and requires normalizing its control objectives according to a pre-defined operating point. Thus, some of the difficulties that arise when implementing a multi-objective finite-set MPC stem from its use of a single cost function. Furthermore, conventional finite-set MPC with an overall cost function is not designed to drive any parameter error to fall below an acceptable tolerance. The designed weight factor ratios by themselves do not reveal nor confirm any tracking capability of the control; only through extensive tests can the control be evaluated which is a challenging design strategy. Any reported weight factor ratios may become unfit for a user who wants to emulate a controller under new conditions. Ultimately, it is desirable and simpler for end users to define their multi-objective controllers directly based on desired tracking performance of the objectives, while still being generally aware of tradeoffs associated with multi-objective optimization.

In this chapter, I propose a method to achieve multi-objective model predictive control without using an overall cost function for a CMI at the grid-edge. The proposed approaches enables users to design the controller according to acceptable tracking performance. This is achieved by addressing each control objective hierarchically instead of combining all control objectives in a single cost function, the objectives are evaluated in sequence. Thus, the proposed control structure eliminates the need to design weight factors for a generic cost function. By ranking control objectives and defining an acceptable bound of error, the objectives are tracked in a descending

fashion. Control actions that do not satisfy the error bound of an objective will be removed before considering subsequent control objectives. This will create stronger assurance of controller tracking for critical objectives, while still allowing for local minimization of less critical/slower dynamic control objectives.

The proposed concept is presented generally for finite-set MPC of N objectives, then described for the presented case study that enables resilient cascaded multi-level inverter at grid-edge. The proposed MPC approach is well-suited for CMI with advanced functionalities. The controller achieves active and reactive power injection with switching event minimization while simultaneously balancing the power drawn from the individual dc voltage sources. The represented dc sources could be battery cells connected to the grid through the proposed CMI. The latter of these control objectives is made possible by the proposed hierarchical approach to the control objective tracking, whereas traditional MPC requires complex logic external to the cost function [41, 42]. These approaches quickly become impractical as the number of cascaded bridges increase, as the number of switching sequences increase exponentially. Finally, the control is modularized for each phase, making the CMI robust to unbalanced grid conditions. Although it is beyond the scope of this work, the proposed control scheme can be integrated with energy management algorithms to optimize the power drawn from battery cells while considering current stresses on the battery cells during grid fast transients.

The remainder of this chapter is organized as follows: Section 2.1 explains the grid interactive CMI, the foundation of the predictive model, and reference generation for MPC cost function. Section 2.3 details the hierarchical MPC concept, and explains how it can be implemented on a controller with sequential logic. In Section 2.4, the concept is demonstrated for the presented case study in simulation, to investigate the control procedure in detail. In section 2.5,

the control is implemented in a hardware experiment for one phase, where multiple transients are induced on the system. Finally, Section 2.6 summarizes the findings of the chapter.

2.2 System Description

As mentioned in the introduction, the presented case study for the hierarchical model predictive control framework is a CMI at the grid-edge. Figure 2.1 illustrates the CMI topology and summarizes the control scheme. A second order generalized integrator (SOGI) phase locked loop (PLL) detects the grid voltage angle [43]. The SOGI orthogonal signal generation technique is particularly beneficial for its inherent filtering of the grid voltage, making the reference current signal robust to grid voltage harmonics [44]. The reference current is assembled in the rotating reference (*dq*) frame, which is then converted to the reference grid current in stationary frame. The *dq* frame conversion is made possible by the orthogonal signal generation capability of the SOGI, where the original and quadrature signals are inputs to the Park transformation. The reference current is determined using equations for single-phase active and reactive power in the *dq* frame:

$$P_k^* = \frac{1}{2}\left(v_{d,k} i_{d,k}^* + v_{q,k} i_{q,k}^*\right) \quad Q_k^* = \frac{1}{2}\left(v_{q,k} i_{d,k}^* - v_{d,k} i_{q,k}^*\right) \tag{2.1}$$

P^*_k and Q^*_k are the active/reactive power set-points for the CMI. The subscript k indicates a discrete sampling instant. $v_{d,k}$ and $v_{q,k}$ are the grid's components in the rotating reference frame, and $i^*_{d,k}$ and $i^*_{q,k}$ are decoupled components of the reference current to be solved. This equation is rearranged to calculate the reference current components in the *dq* frame, and then converted to the original time-variant frame using the inverse Park equation:

$$i_{d,k}^* = \frac{2\left(P_k^* v_{d,k} + Q_k^* v_{q,k}\right)}{{v_{d,k}}^2 + {v_{q,k}}^2} \quad i_{q,k}^* = \frac{2\left(P_k^* v_{q,k} - Q_k^* v_{d,k}\right)}{{v_{d,k}}^2 + {v_{q,k}}^2} \tag{2.2}$$

$$i_k^* = i_{d,k}^* \sin(\theta_k) + i_{q,k}^* \cos(\theta_k) \tag{2.3}$$

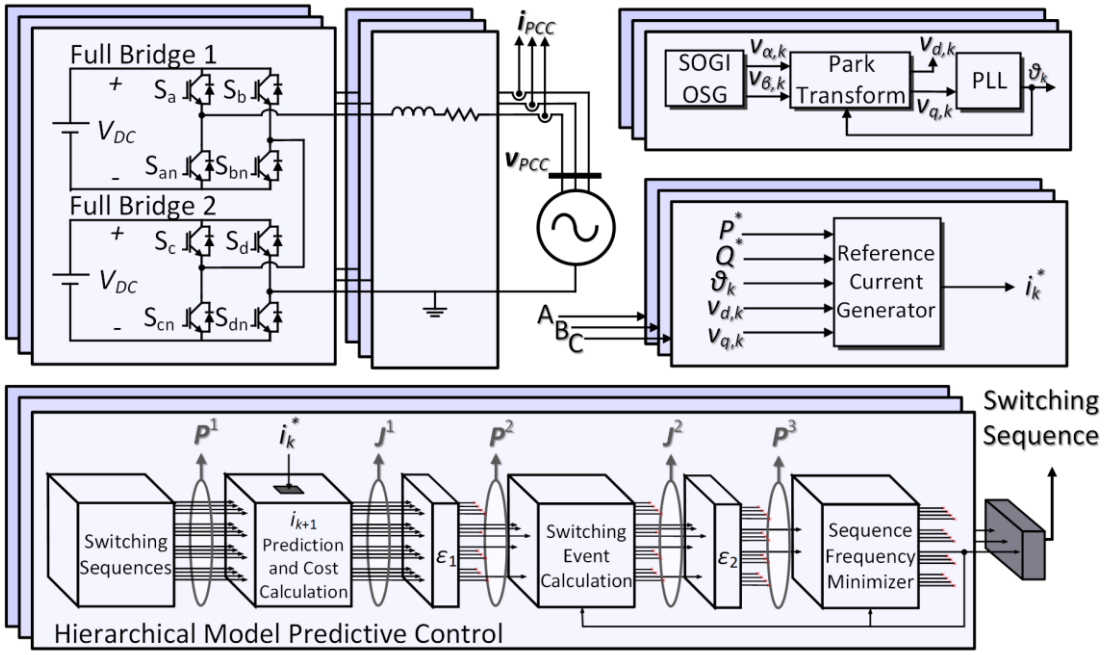


Figure 2.1: Hierarchical model predictive control developed for grid-connected cascaded multilevel inverter.

where θ_k is the grid angle detected by the PLL and i^*_k is the time-variant reference current. For current control, the finite-set MPC evaluates each of the switching sequences, or switching states, and compares it to the reference. The output current predictions derive from the AC-side KVL equation:

$$v_{inv} = r(i) + L\frac{di}{dt} + v_g \tag{2.4}$$

where L and r are the filter inductance and equivalent series resistance, respectively. v_g and v_{inv} are the grid and inverter voltages, respectively. This equation is discretized by approximating the differential using forward Euler, assuming constant inductance, and is rearranged to create an explicit solution for the one-step ahead prediction of the output current:

$$i_{k+1}^M = \left(1 - \frac{r}{L}T_S\right)i_k + \frac{T_s}{L}\left(v_{inv,k+1}^M - v_{g,k}\right)$$
$$M \in \mathbb{Z} : M \in [-2 \;\; 2] \tag{2.5}$$

where M is the output voltage level of the considered switching sequence; the applied output voltage is the output voltage level M multiplied by the DC link voltage V_{DC}. The associated cost term for injected current is defined as

$$\boldsymbol{J}^1 = \left| i_{k+1} - i_k^* \right| \tag{2.6}$$

where $\boldsymbol{J}^1$ is the cost vector of injected grid current. This will be explained more in Section 2.3. The control also considers changes in the switches gate logic as a control objective to reduce the switching events and as a result the average switching frequency. The cost is defined as:

$$\boldsymbol{J}^2 = \sum_{i=a,b,c,d} 2\left| S_{i,k+1} - S_{i,k} \right| \tag{2.7}$$

The gate logic for each switching sequence is provided in Table 2.1. The summation in (2.7) is multiplied by two to account for the other switch in the leg (S_{an}, S_{bn}, etc.). There is a third control objective, referred to as *sequence frequency* minimization, but further discussion of it is left for Section 2.3. The proposed sequence frequency minimization adds a unique feature to CMI via the proposed hierarchical MPC scheme that enables self-power balancing of its H-bridge cells, i.e. balances the power drawn from battery cells.

Table 2.1: Switching Sequences for One Phase of 5-Level CMI

Sw. Seq.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
S_a	0	0	0	1	1	1	0	0	0	1	0	1	1	1	0	1
S_{an}	1	1	1	0	0	0	1	1	1	0	1	0	0	0	1	0
S_b	0	0	1	0	1	1	0	1	1	1	0	0	0	1	1	0
S_{bn}	1	1	0	1	0	0	1	0	0	0	1	1	1	0	0	1
S_c	0	1	1	0	0	1	0	0	1	0	1	0	1	1	0	1
S_{cn}	1	0	0	1	1	0	1	1	0	1	0	1	0	0	1	0
S_d	0	1	0	1	0	1	1	0	1	1	0	0	1	0	1	0
S_{dn}	1	0	1	0	1	0	0	1	0	0	1	1	0	1	0	1
M	0	0	0	0	0	0	-1	-1	-1	-1	1	1	1	1	-2	2

2.3 Proposed Hierarchical Predictive Control

Hierarchical model predictive control is first described in a conceptual way. To aid in understanding the concept, and to address a few exceptional cases of the control, then I describe how the control is actualized in a discrete controller.

2.3.1 Hierarchical Model Predictive Control: Conceptualized

The hierarchical predictive control paradigm is presented for N control objectives. The control is explained using two types of vectors, called cost vectors and argument vectors. Cost vectors are denoted with $\boldsymbol{J}$, as is standard in the field of convex optimization [14, 15]. The argument vectors are denoted as $\boldsymbol{P}$. For each control objective, there is an associated pair of cost and argument vectors. The rank of the vectors' associated objective is denoted by the vectors' superscript. For example, the cost and argument vectors associated with the primary objective are denoted by $\boldsymbol{J}^1$ and $\boldsymbol{P}^1$, respectively. If a subscript i is specified, then the expression is referring to the i^{th} element in the vector. $\boldsymbol{J}^1{}_3$ is referring to the third cost (cost of switching sequence three) of the primary objective. Each element has an associated switching sequence. Each objective is denoted by its rank within the hierarchy. As described earlier, switching sequences that are not projected to meet the specified cost tolerance are removed from the optimization set of the subsequent objective. This is denoted in (2.8):

$$\boldsymbol{P}^n = \left\{ i : \boldsymbol{J}_i^{n-1} < \varepsilon_{n-1} \right\} \tag{2.8}$$

Thus, $\boldsymbol{P}^n$ contains the arguments of the optimization set for the subsequent objective n. Cost vectors $\boldsymbol{J}_{S\times 1}$ hold the projected cost of each switching sequence that remains in the optimization set. The value $\varepsilon_{n\text{-}1}$ is the cost tolerance of the previous control objective (objective n-1). Each control objective, with the exception of the last control objective, will use a cost tolerance for removing candidate-switching sequences. Switching sequences are evaluated for each control objective

according to the objective's $\boldsymbol{P}$ vector. The $\boldsymbol{P}$ vector of an objective details all the switching sequences that will have its cost measured for the objective.

$$\begin{gathered}\boldsymbol{P}^N \subseteq \boldsymbol{P}^{N-1} \subseteq \cdots \subseteq \boldsymbol{P}^2 \subseteq \boldsymbol{P}^1 \\ \boldsymbol{P}^1 = \{x \in \mathbb{N} \mid x \leq S\}\end{gathered} \tag{2.9}$$

where S equals the number of possible switching sequences. For the CMI, this is equal to 4^H, where H is the number of H-bridges in cascade [45]. Also shown in (2.9) is the time-invariant setting of $\boldsymbol{P}^1$. Definitely, $\boldsymbol{J}^1$ will be computed for each possible switching sequence, as nothing else has allowed switching sequences to be excluded. This means $\boldsymbol{P}^1$ will hold a constant size, which is simply an array of incrementing natural numbers from 1 to S. From (2.9), we see the $\boldsymbol{P}$ vectors may decrease in size to objective N, but are not guaranteed to. This reduction in the optimization set is dependent on the chosen cost tolerances. In general, hierarchical MPC cannot guarantee a consistent reduction in cost computations, but also cannot exceed that of a traditional finite-set MPC. This will be demonstrated in the following section. Upon evaluation of all switching sequences with respect to each control objective, the controller will have determined a reduced optimization set, equal to $\boldsymbol{P}^N$. Again, there is no guarantee to the size of $\boldsymbol{P}^N$, only a guaranteed upper bound. However, since all sequences within are deemed sufficient for proceeding objectives, the control selects the sequence within this set which optimizes the N^{th} objective:

$$s_{k+1} = \arg\min\{\boldsymbol{J}^N\} \tag{2.10}$$

Thus, for the lowest rank objective, a cost tolerance need not be defined.

2.3.2 Hierarchical Model Predictive Control: Actualized

For implementation in a microcontroller with sequential logic, Algorithm 2.1 outlines how the controller can be implemented. Note that the argument vector $\boldsymbol{P}^1$ is fixed, since all switching sequences will be evaluated for the primary objective. In block 1 of the algorithm, $\boldsymbol{P}^2$ is

Algorithm 2.1: HMPC, *S* Switching Sequences, *N* Objectives

Function $[s_{k+1}]$ = HMPC **(***measurements***)**
Initialization: sampling at T_s , Define $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_{N-1}$
$\boldsymbol{J}^1 \leftarrow \boldsymbol{J}^2 \leftarrow \ldots \leftarrow \boldsymbol{J}^N \leftarrow [0\ 0\ \ldots\ 0]_{1xM}$, $\boldsymbol{P}^1 \leftarrow [1\ 2\ \ldots\ M\ ]_{1xS}$
$\boldsymbol{J}^1_{opt} \leftarrow \boldsymbol{J}^2_{opt} \leftarrow \ldots\ \boldsymbol{J}^N_{opt} \leftarrow \infty$, $i \leftarrow 1$
1: Argument Reduction for Primary Objective
for each $x \in \boldsymbol{P}^1$ **do**
compute $\boldsymbol{J}^1_x$
if $\boldsymbol{J}^1_x \leq \varepsilon_1$ **then**
$\boldsymbol{P}^2_i \leftarrow x$, $++i$
end if, end for, reset i
if length($\boldsymbol{P}^2$) = 0 **then** → Optimize $\boldsymbol{J}^1$ (Algo. 2.2)
else → descend …
2: Argument Reduction for Secondary Objective
for each $x \in \boldsymbol{P}^2$ **do**
compute $\boldsymbol{J}^2_x$
if $\boldsymbol{J}^2_x \leq \varepsilon_2$ **then**
$\boldsymbol{P}^3_i \leftarrow x$, $++i$
end if, end for, reset i
if length($\boldsymbol{P}^3$) = 0 **then** → Optimize $\boldsymbol{J}^2$ (Algo. 2.2)
else → descend …
⋮
***N*-1: Argument Reduction for Objective *N*-1**
for each $x \in \boldsymbol{P}^{N-1}$ **do**
compute $\boldsymbol{J}^{N-1}_x$
if $\boldsymbol{J}^{N-1}_x \leq \varepsilon_{N-1}$ **then**
$\boldsymbol{P}^N_i \leftarrow x$, $++i$
end if, end for, reset i
if length($\boldsymbol{P}^N$) = 0 **then** → Optimize $\boldsymbol{J}^{N-1}$ (Algo. 2.2)
else → descend …
***N*: Optimization of N^{th} Objective**
for each $x \in \boldsymbol{P}^N$ **do**
if $\boldsymbol{J}^N_x < \boldsymbol{J}^N_{opt}$ **then**
$s_{k+1} \leftarrow x$, $\boldsymbol{J}^N_{opt} \leftarrow \boldsymbol{J}^N_x$
end if, end for
end *N* hierarchical if statements, Return s_{k+1}, **End function**

constructed, and is of varying length. The bottom two lines of block 1 suggest a scenario not yet considered: what if none of the switching sequences allow any of the cost to be less than its associated tolerance ε? A switching decision must still be made. The most obvious workaround is

to choose the switching sequence which minimizes the cost, despite the cost exceeding the defined bound. Effectively, this is what is done. However, it is possible to continue optimizing subordinate objectives if there are redundancies in the minimized cost. Redundant switching sequences are especially prevalent in the CMI topology [25]. As an example, let us consider the following scenario: at instant *k*, it is predicted that no switching sequence will allow the injected current to meet the defined tolerance. Additionally, when optimizing for grid current, it is determined that applying an output voltage level of zero will minimize cost. For the five-level CMI, there are six switching sequences which can achieve this voltage level. Thus, these six switching sequences can be evaluated for subordinate objectives, while still ensuring minimization of cost in injected current. Such logic is described in Algorithm 2.2, which is nested in blocks one through N-1 in Algorithm 2.1.

Algorithm 2.2 is written in for an arbitrary objective U, where $U \in [1, N\text{-}1]$. Block 1 acts in a very similar way to block N of Algorithm 2.1. Essentially, objective U is optimized with the optimization set developed from objective U-1. If a redundant optimal sequence is found, the vector ***redundancies*** is filled with logic-high values in alignment with the redundant sequences. The variable *redundancy* acts as a flag to trigger block two of the algorithm. If no redundant optimal sequences are detected, the control breaks out of Algorithm 2.2 and Algorithm 2.1 and returns the optimal sequence. However, if there are redundant optimal switching sequences with respect to objective U, the control moves to block two, which fills the argument vector $\boldsymbol{P}^U$ with the detected redundant sequences. In this scenario, the control returns to Algorithm 2.1, to evaluate objective U+1.

For the case study implemented in this chapter, the primary, secondary, and tertiary objectives are (respectively): injected grid current, switching events, and a term referred to as

Algorithm 2.2: Nested in Algorithm 2.1, Triggered if Cost Tolerance Cannot be Met for Objective *U*

Initialization: ***redundancies*** = $[0\ 0\ \ldots\ 0]_{1xS}$
1: Optimizing J^U, Tracking Redundant Optimal Sequences
for each $x \in P^{U-1}$ **do**
if $J^U_x < J^U_{\text{opt}}$ **then**
$s_{k+1} \leftarrow x$, *redundancy* $\leftarrow 0$, reset ***redundancies***
elseif $J^U_x = J^U_{\text{opt}}$ **then**
redundancy $\leftarrow 1$, $\boldsymbol{redundancies}_x = 1$
$\boldsymbol{redundancies}_{sk+1} \leftarrow 1$
end if, end for
2: Checking for Redundant Optimal Sequences
if *redundancy* = 1 **then**
for each $x \in P^{U-1}$ **do**
if $\boldsymbol{redundancies}_x = 1$ **then**
$P^U_i \leftarrow P^{U-1}_x$, $++i$
end if, end for
else break hierarchical if statements, Return s_{k+1}, End function
end if, End of Nested Algorithm

sequence frequency. Sequence frequency is the frequency in which the controller selects a particular switching sequence. Applying the sequence frequency objective allows the controller to eliminate bias among redundant switching sequences. To better understand the purpose of this objective, we will introduce a traditional finite-set MPC for the presented case study. It optimizes injected grid current and minimizes switching events with the following cost function:

$$\begin{aligned} \boldsymbol{J} &= \left|i_{k+1} - i_k^*\right| + \lambda \sum_{i=a,b,c,d} 2\left|S_{i,k+1} - S_{i,k}\right| \quad \lambda \ll 1 \\ s_{k+1} &= \arg\min(\boldsymbol{J}) \end{aligned} \tag{2.11}$$

The weight factor applied to the switch minimization term is made sufficiently small so as not to affect the current objective, which only depends on the output voltage level *M*. The optimization procedure of such a control is similar to block *N* of Algorithm 2.1, where $\boldsymbol{J}$ replaces $\boldsymbol{J}^N$, and $\boldsymbol{P}^1$ replaces $\boldsymbol{P}^N$. For implementation of this control, redundant switching sequences are ignored. That is, when there are two or more switching sequences that equally optimize the control objective(s),

the control will consistently select only one of the sequences. If $\boldsymbol{P}^1 i$ and $\boldsymbol{P}^1{}_{i+1}$ minimize $\boldsymbol{J}$ equally, the function will always return s_{k+1} as $\boldsymbol{P}^1{}_i$. This is common for finite-set predictive control [34, 46]. For the traditional predictive control outlined by (2.10), consider the scenario where the previous switching sequence was switching sequence sixteen (M=2), and to optimize injected grid current, the controller will implement M = 1. There are four switching sequences to achieve this; two sequences apply positive voltage across cell one (sequences twelve and thirteen), the other two apply positive voltage across cell two (switching sequences eleven and fourteen). Considering both injected grid current and switching events in the cost function, the controller will always select the first switching sequence such that M = 1 (switching state eleven) when the previous switching state was switching state sixteen, since $\boldsymbol{J}_{11} = \boldsymbol{J}_{12} = \boldsymbol{J}_{13} = \boldsymbol{J}_{14}$. Undue bias toward this switching sequence induces unequal power draw from the isolated DC sources, which has practical consequences such as uneven discharge rates of grid-connected battery storage systems. When applying switching events in the cost function, it cannot be said with certainty how the power draw characteristics will behave for traditional finite-set MPC, since this depends on the changes in the selected output voltage over time, which depends on the output filter, sampling frequency, and DC link voltages. However, without an objective to regulate the selection of redundant switching sequences, the power draw characteristics are likely to be distinct for each source. This will be demonstrated in Section 2.5.

For hierarchical MPC, applying an objective to remove such biases is simple. This objective is applied by creating a tertiary cost vector $\boldsymbol{J}^3$ and incrementing $\boldsymbol{J}^3{}_i$ each time the controller selects switching sequence i. Mitigation of bias among the possible switching sequences improves the equalization of power draw from the isolated DC sources. An objective such as this is difficult to apply directly in a traditional finite-set MPC, since the cost increases without bound.

Over time, this objective would ‘dominate’ the cost function, as its magnitude of the cost quickly exceed that of all other objectives. With the hierarchical controller, this is impossible, as this objective can only be optimized once the superordinate objectives fall within their respective bounds.

While discussing this control, the feasibility of finite-set MPC (and the proposed HMPC) for alternative applications should be addressed. First, it should be established that the proposed HMPC can easily be expanded for CMI’s with a larger number of series-connected H-bridge modules. If there are N-series connected H-bridges, the possible control actions and maximum size of the argument vectors can be defined automatically from N. However, the number of possible control actions to implement increases exponentially with N. In the selected application (with N = 2) there are 16 possible control actions. This can be easily implemented at a high controller sampling frequency. For instance, a sampling frequency of 50kHz is used in the hardware experiment of Section 2.5, and the sampling rate was limited primarily because of the analog-to-digital converters for sensed measurements. To consider a topology with 100 H-bridge modules (N = 100), there would be 4^{100} unique control actions. Needless to say, evaluating this number of control actions is not possible with a practical embedded system at a reasonable sampling frequency. This is inherent to any finite-set control, including traditional finite-set MPC. Mitigation of this issue for such applications would require a technique which carefully eliminates a substantial number of control actions without removing the benefits of finite-set MPC or introducing critical drawbacks, but further discussion is outside the scope of this work.

2.4 Illustration of Concept

The main objective of this section is the detailed analysis of the control procedure which is not feasible in the hardware implementation of the system. For instance, we cannot collect

intermediate variables, e.g. cost and argument vectors, in real time at each discrete instant while still implementing a practical controller sampling frequency. To overcome this, we first simulate the hardware experiment. The sampling frequency, filter size, and grid voltage are equal to that of the hardware experiment in Section 2.5. In the first subsection, the tracking of each objective is studied in depth. In the second subsection, we examine the effect of model inductance error on current tracking.

2.4.1 Objective Tracking Analysis

Data is collected for one phase of the modular control. Figure 2.2 shows injected grid current for one phase during a step-change in cost tolerances of objective one and two. At $t = 0.1$s, the cost tolerance of injected grid current is reduced from 0.8A to 0.3A, while the cost tolerance of switching events is increased from three to five. Figure 2.3 shows the reference current, injected current and cost tolerance bounds during this step change. An obvious reduction in current ripple about the peaks is observed, and the controller is able to maintain the current within this bound. Figure 2.4 shows the number of switching events, changes in gate signal logic, during this transition. Since a relaxation of switching event error bound has been implemented, the average number of switching events per sampling instant increases by roughly thirty percent. Figure 2.5 is provided to better understand the decision making procedure of the hierarchical MPC. The same parameters are present that were shown in Figure 2.3, but the next-state predictions at each discrete sampling instant. The next-state prediction of the selected switching sequence is highlighted for each instant. Note that there are only five next-state predictions per instant, yet some predictions are represented by a darker point. This demonstrates that there exist only five unique current predictions among the sixteen possible switching sequences, one unique prediction for each possible output voltage level, and thus many predictions overlap. This agrees with Table 2.1,

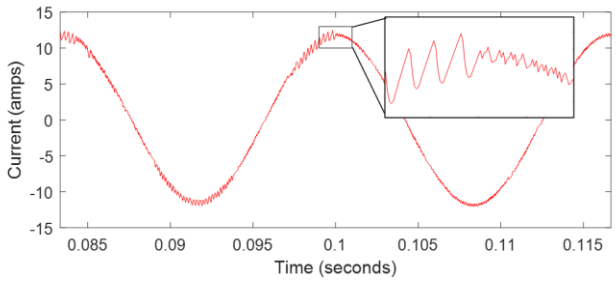


Figure 2.2: Injected phase current. At t = 0.1s, ε_1 is reduced from 0.8A to 0.3A.

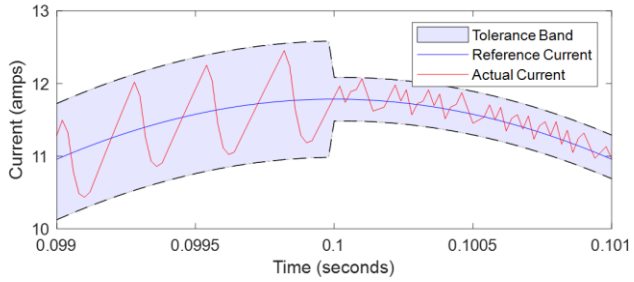


Figure 2.3: Injected phase current, reference current, and tolerance band as ε_1 is reduced from 0.8A to 0.3A.

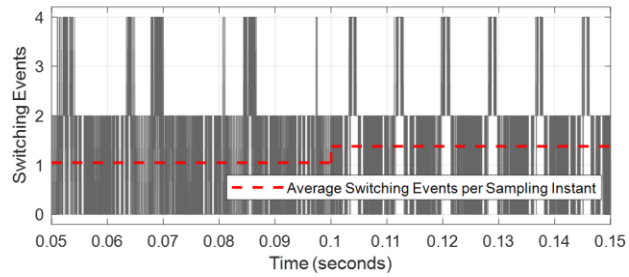


Figure 2.4: Number of switching events as ε_1 is reduced from 0.8A to 0.3A, and ε_2 increases from 3 to 5. These changes result in about a 30% increase in the average number of switching events per discrete sampling instant.

showing each switching sequence and its output voltage level M. The instances in this figure occur before the transition in cost tolerance. Prior to t = 0.09928s, we see the controller continues implementing M = 2, despite the fact that it is not optimizing the injected current objective. Since these control actions fall within the bounds of its cost tolerance, switching sequences such that $M \in [0, 2]$, are still being considered as the controller moves to calculating switching event computations. With the cost tolerance of objective two set to three (switching events) switching sequences that require more than one bridge gate inversion are removed. From examining Table

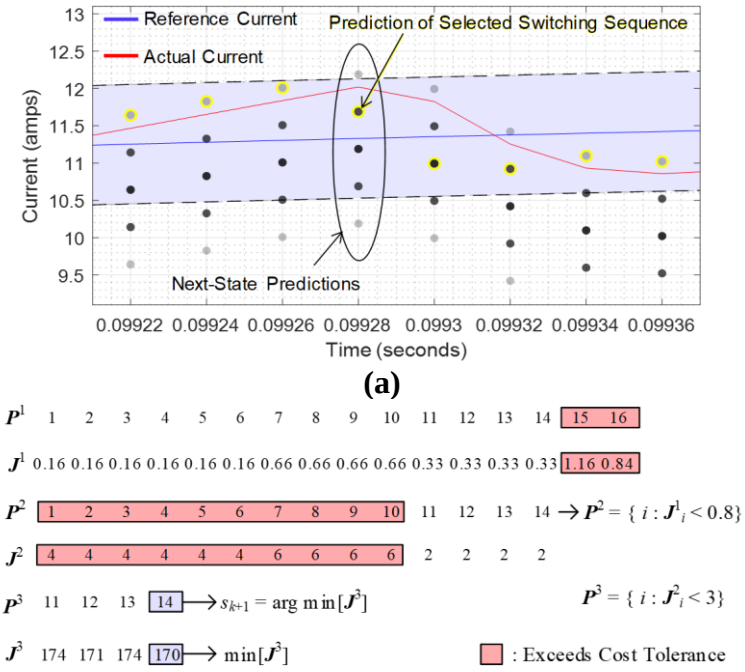


Figure 2.5: (a) Injected phase current, reference current, tolerance band, and next-state current predictions for each discrete instant, evaluated with ε_1 equal to 0.8A. The prediction of the selected switching sequence is highlighted. (b) Cost and argument vectors at instant t = 0.09928s. Cost tolerance of objective 1 and 2 are 0.8A and 5, respectively.

2.1, we see this will eliminate zero-voltage levels, but retains all sequences such that M = 1, as well as the previous switching state. Thus, switching sequences eleven, twelve, thirteen, fourteen, and sixteen remain in the optimization set. From there, the controller selects the switching state that has been chosen the least. From this, we conclude the M = 1 switching sequences have been chosen more frequently at this point, thus, the control continues to implement switching sequence sixteen. A transition in switching sequence occurs at t = 0.09928s. We see that the presently implemented voltage level of two is no longer in the acceptable boundary. The cost and argument vectors at the end of this iteration are shown in Figure 2.5. After computing $\boldsymbol{J}_1$, switching sequences fifteen and sixteen which produce output voltage levels of negative two and positive two, respectively, exceed ε_1. Thus, $\boldsymbol{P}^2$ has both sequences removed from the optimization set. However, since the currently-selected switching state is sixteen, and the switching event tolerance

is three, all negative and zero output voltage levels are removed, as they require at least four gate signal transitions. The only switching sequences that remain in the set are those such that $M = 1$, thus $\boldsymbol{P}^3$ contains the eleven through fourteen. Finally, the controller selects among the reduced set according to the level that has been selected the least, which is switching state fourteen. Thus, the controller implements switching state fourteen at $t = 0.0993$s. Note that the size of the cost vectors do not change size; Figure 2.5 only includes the cost vector elements that were computed.

In Figure 2.6, the current, reference current, and reduced tolerance band ($\varepsilon_1 = 0.3$A) are shown after $t = 0.1$s with the next-state current predictions. It is clear that fewer voltage levels tend to fall within this error bound. In general, the argument vectors $\boldsymbol{P}^2$ and $\boldsymbol{P}^3$ will be smaller in this scenario. Thus, it will be more difficult to optimize their associated objectives. This is also evidenced in Figure 2.7, which shows the power draw characteristics of the individual H-bridges for each set of cost tolerances. In Figure 2.7a, the larger cost tolerance for injected grid current induces greater equalization of power draw from the two sources, with a difference in power draw of roughly 5W. The controller is better able to even the selection of switching sequences. Figure 2.7b shows the power draw of each cell for the reduced ε_1. The difference in power draw increases to 65W, as a result of the size reduction in $\boldsymbol{P}^3$. As will be shown in the hardware results, this deviation is far less than that of traditional MPC.

As discussed in Section 2.3, hierarchical MPC does not guarantee a consistent reduction in iterative computation when compared to traditional finite-set MPC. However, hierarchical MPC will not exceed the computation of traditional MPC. To understand this, the iterative computations are directly tracked in Figure 2.8. Here, a *computation* is considered an addition, subtraction, multiplication, division, or comparison. A comparison is done when searching for an *arg min*, or finding objectives which fall within their respective tolerances. This tracking is done for a high-

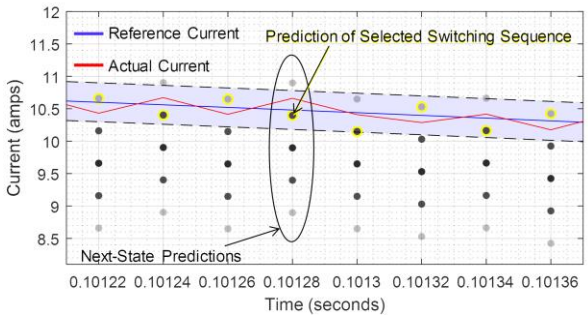
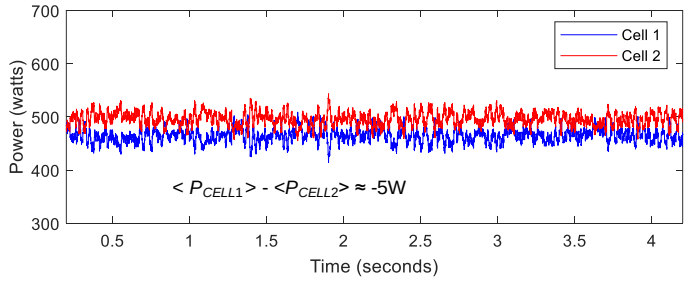
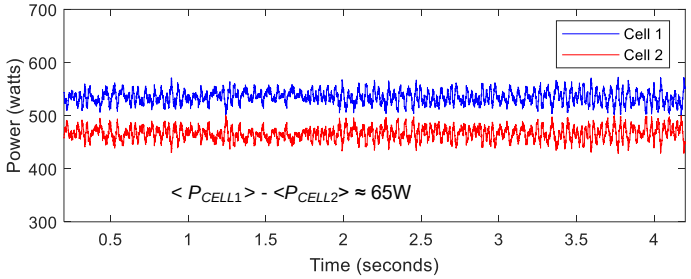


Figure 2.6: Injected phase current, reference current, tolerance band, and next-state current predictions for each discrete instant, evaluated with ε1 equal to 0.3A. The prediction of the selected switching sequence is highlighted.



(a)



(b)

Figure 2.7: Power injection from each of the isolated sources for (a) ε1 = 0.8A (b) ε1 = 0.3A.

level explanation of how computations differ for hierarchical MPC and not for a definitive comparison of feasible sampling frequencies; this would require an assumed architecture of the embedded system. Further, computations are only considered for the optimization portion of the control (i.e. computations for reference signal generation are not included). Prior to the change in cost tolerances at t = 0.1s, we see computation varies between 170 and 315, with an average of about 239. There is a fixed minimum number of computations associated with computing J^1 and P^2. The remaining computations are dependent on the length of P^2 and P^3. For larger tolerances,

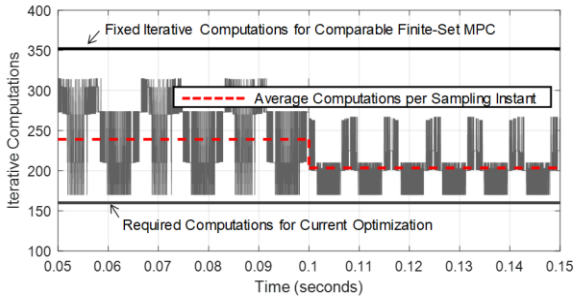


Figure 2.8: Number of iterative computations as ε_1 is reduced from 0.8A to 0.3A, and ε_2 increases from 3 to 5. These changes result in an average iterative computation count of roughly 239 to about 203.

the length of succeeding argument vectors will increase, and iterative computation will increase accordingly. Thus, following the reduction in ε_1 at t = 0.1s, we see a reduction in iterative computation. This is a result of P^2 tending to be smaller, as the current optimization constraint has become more difficult to satisfy. Note that the comparable finite-set MPC has a fixed number of iterative computations of 352. For the selected error tolerances, the hierarchical MPC has a consistently reduced number of computations.

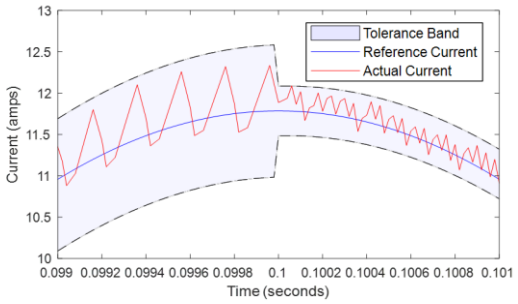
2.4.2 Effect of Model Parameter Error on Current Tracking

Accurate next-state current prediction is dependent on accurate estimation of the model parameters. For finite-set model predictive current control with an inductive filter, model inductance error is found to be much more critical than error on the equivalent series resistance (ESR) of the filter [47]. Thus, we have restricted our focus to relative error on the model inductance.

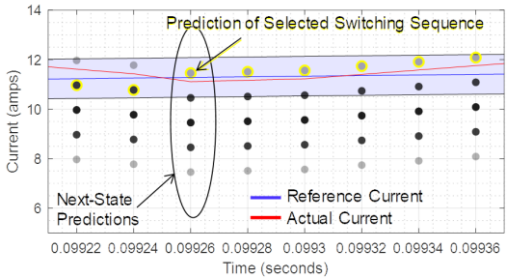
First, a relative error of negative fifty percent is applied to the model inductance. That is, the inductance within the HMPC algorithm is half of the physical inductance. The same transient from subsection A is applied here; namely, ε_1 is reduced from 0.8A to 0.3A, and ε_2 is increased from three to five at time t = 0.1s. The injected current, reference current, and cost tolerance bounds are shown in Figure 9a. It is evident that the current is maintained well within the defined bound.

In fact, the error magnitude appears lower on average than was seen without model inductance error. The reason the bound is maintained can be explained when looking back to (5) which computes the next-state current prediction for each output voltage level. With a reduced model inductance L, the prediction overestimates the change induced on the current from the estimated voltage across the filter. In Figure 9b, ε_1 is 0.8A, and the difference in predicted currents for each output voltage level is around 1A. The current predictions have "spread out", from the model-aligned inductance from Figure 6, where the predictions were roughly 0.5A apart. Less output voltage levels are determined to be within the defined error tolerance bound, and the current stays well within the bound as a result. In Figure 9c, for most sampling instances, there is no output voltage level which satisfies the cost tolerance ε_1. As a result, the control must select the output voltage which minimizes the current error magnitude. This allows the controller to maintain the defined boundary.

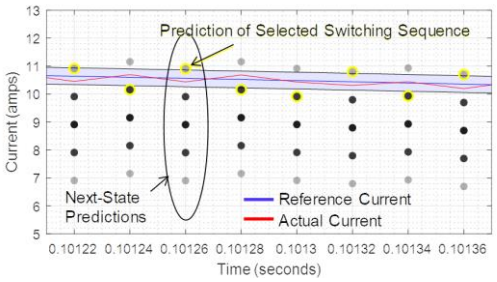
Next, a relative error of 100 percent is applied to the model inductance. This is considered a worst-case scenario. The inductance value in the controller is set to twice that of the physical filter inductance. The previous transient is again applied at $t = 0.1$s, as shown in Figure 10a. Unlike the scenario of underestimated model inductance, the overestimated model inductance is unable to keep the current within the defined bound. With positive relative error on the model inductance, the control underestimates the change in inductor current. In Figure 10b, we see the current predictions closer together for each sampling instant. As a result, output voltage levels which would not normally reside within the acceptable boundary are kept within the optimization set for the secondary and tertiary objectives. Since there are more zero voltage switching sequences, they tend to be selected less frequently. As a result, the control tends to select switching sequences with



(a)



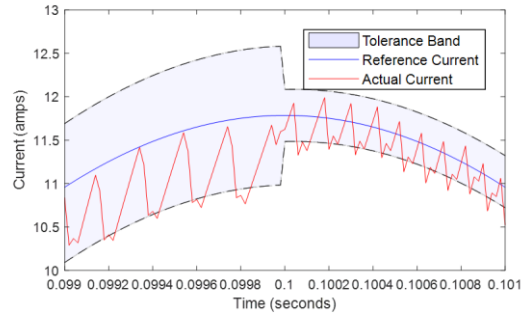
(b)



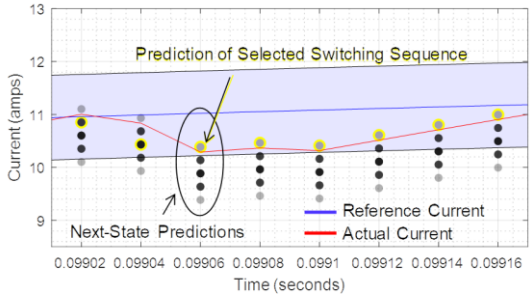
(c)

Figure 2.9: Current tracking for negative fifty percent relative error in model inductance (a) Injected phase current, reference current, and tolerance band as ε1 is reduced from 0.8A to 0.3A (b) Injected phase current, reference current, tolerance band, and next-state current predictions, evaluated with ε_1 equal to 0.8A (c) ε_1 equal to 0.8A.

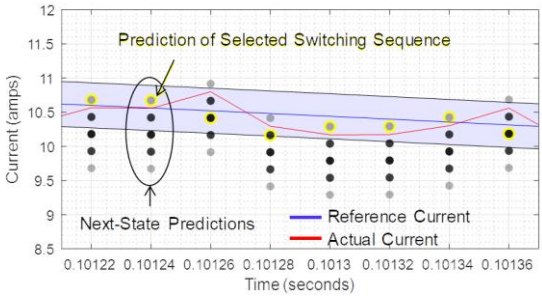
lower voltage levels to satisfy the tertiary objective. Thus, near the peak of the reference current, the injected current tends to settle below the reference. In Figure 10c, the reduced cost tolerance boundary contains more voltage levels than was noted for reduced model inductance. Whereas reduced model inductance still allowed for the current to be constrained within the desired boundary set by ε_1, the increased model inductance cannot. It is worth mentioning that this case study demonstrates a significant model parameter error as a worst-case scenario which is rarely considered in practice, This extreme model parameter error demonstrates the acceptable performance of the proposed controller.



(a)



(b)



(c)

Figure 2.10: Current tracking for 100 percent relative error in model inductance (a) Injected phase current, reference current, and tolerance band as ε1 is reduced from 0.8A to 0.3A (b) Injected phase current, reference current, tolerance band, and next-state current predictions, evaluated with ε_1 equal to 0.8A (c) ε_1 equal to 0.3A.

2.5 Results and Discussion

The proposed hierarchical model predictive control scheme is tested experimentally. For the case studies, a five-level CMI is tied to a $120V_{LN}$ (RMS) grid. Table 2.2 details parameters of the testbed, shown in Figure 2.11. DC power supplies provide the DC link voltages of the H-bridges, and a four-quadrant (power-bidirectional) grid emulator is tied to the output of the CMI. The control is implemented on a rapid control prototyping device, the dSPACE CP1103. The CP1103 has embedded analog-to-digital converters, and thus all measurements shown in this

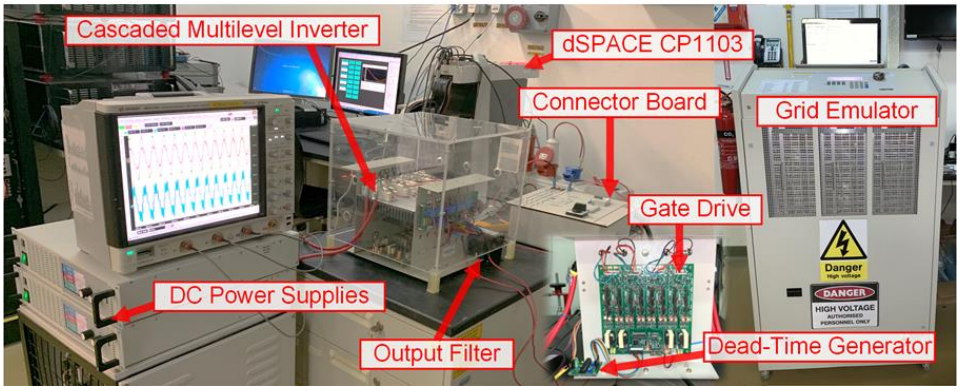


Figure 2.11: Hardware setup for experimental validation of proposed hierarchical model predictive control scheme for CMI.

Table 2.2: System Specifications

Parameter	Value
DC-link voltages	165V
Filter Inductance	2.5mH
Filter Resistance	0.2Ω
Rated line-neutral Grid Voltage	$120V_{RMS}$
P^*	1kW
Q^* (normal grid condition)	0kVAR
Q^* (grid voltage sag)	0.25kVAR
Controller Sampling Frequency	50kHz
Imposed dead-time	1ms
ε_1 [Hierarchical MPC]	0.2A
ε_2 [Hierarchical MPC]	5 switching events
λ [Traditional MPC, governed by (10)]	1e-6
k_p (PR controller of PWM-based control)	0.1
k_r (PR controller of PWM-based control)	10

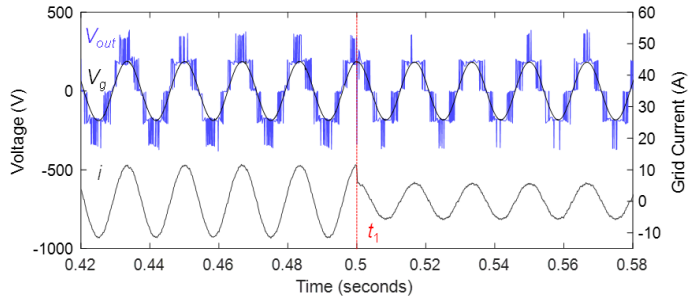
section come from stored values sensed/computed by the CP1103. To store data in real-time, the sampling rate of the collected measurements are 12.5kHz (one fourth of the controller's sampling frequency). The cost tolerances ε_1 and ε_2 are set to 0.2A and 5 switching events, respectively. When validating the control to system transients, its dynamic and steady-state response is compared to a comparable, traditional finite-set control scheme and a standard PWM current control. In all three control schemes, the current reference is developed identically, which is described in Section 2.2.

In Section 2.5.1, the control is compared to standard finite-set MPC. In Section 2.5.2, the proposed control is compared to a current control scheme which uses a multilevel subharmonic PWM switching scheme, and the modulation signal is developed from a proportional-resonant (PR) control.

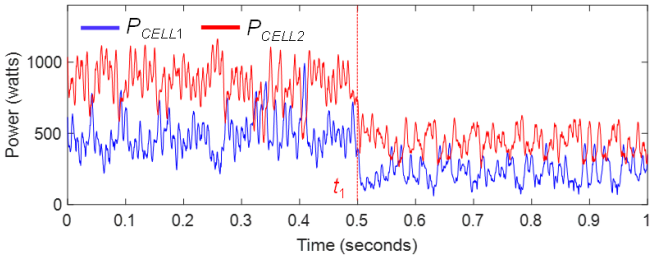
2.5.1 Comparison against Traditional Finite-set MPC

The standard model predictive current control implements the cost function and control action defined in (2.10). Its weight factor λ tied to switching event minimization is defined in Table 2.2 as 1e-6. Note that this is done so the control will only reduce switching events once the optimal output voltage level *M* is selected. It is expected that the control will select the same switching sequences for all λ less than 1e-4 but greater than zero. This statement is equivalent to saying it is expected that the cost difference of the current predictions between different voltage levels will always exceed 0.8mA, as the greatest cost produced by the switching event term in $\boldsymbol{J}$ cannot exceed 8e-4. Thus, the designed cost function will behave in a similar fashion to the proposed hierarchical control but does not include the sequence frequency objective.

Both the proposed control and finite-set MPC are tested for a step change in power reference, from 1kW to 0.5kW. In Figure 2.12, the traditional finite-set MPC implements a reduced power reference at t_1. Within a few samples, the control settles the current amplitude in alignment with the reduced power reference, as shown in Figure 2.12a. In Figure 2.12b, the power-draw characteristics of each voltage source is shown. There is evident double-frequency power ripple. This is also evidenced on the DC-link voltages, which are used to compute the output voltage in Figure 2.12a. Further, there is a notable distinction in power draw among the voltage sources. In particular, the voltage source of the lower H-bridge has notably larger power draw on average than that of the upper H-bridge. Without an objective to equalize the selection of switching



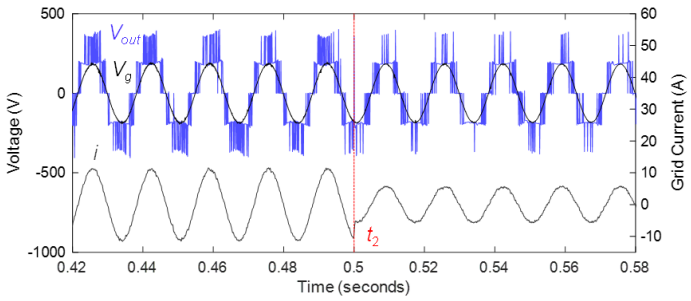
(a)



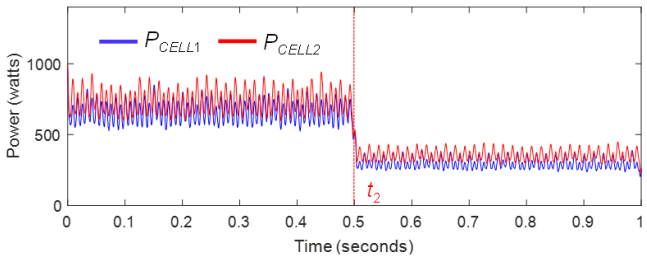
(b)

Figure 2.12: Dynamic response of traditional finite-set MPC for reduction in power reference at t_1 from (a) output voltage, grid voltage, and output current (b) power draw characteristics.

sequences, the standard MPC selects only one switching sequence for each output voltage level. This distinction is dependent on how the switching sequences are defined and compared in the control algorithm, unlike the proposed control. In Figure 2.13a, the proposed control is placed under the same power reference reduction at t_2. The control reduces the output current to the reduced reference current amplitude within a few sampling instants, as was seen with the traditional finite-set MPC. In Figure 2.13b, the power draw characteristics of each H-bridge are shown. The double-frequency power ripple that was seen in traditional finite-set MPC still occurs for the proposed control. However, there is only a slight distinction in power draw between the voltage sources. This was noted in Section IV, where implementing a small ε_1 created a slight distinction in power draw. Thus, the proposed control successfully retains the fast reference tracking of traditional finite-set MPC while substantially reducing the distinction in power draw characteristics.



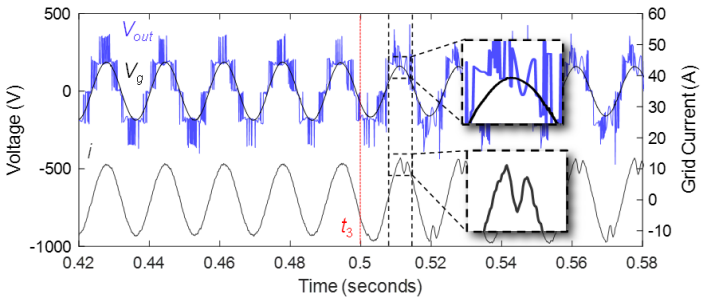
(a)



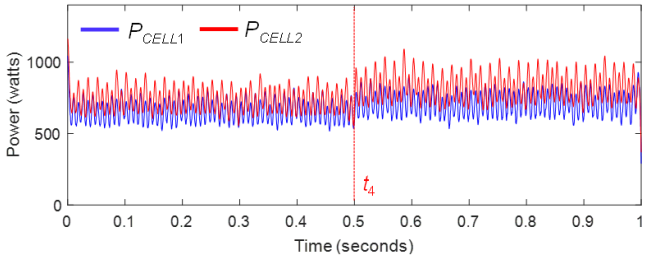
(b)

Figure 2.13: Dynamic response of proposed control for reduction in power reference at t_2 (a) output voltage, grid voltage, and output current (b) power draw characteristics.

Next, each control is tested for a sag in grid voltage. As noted in Table 2.2, the reactive power reference (Q^*) is zero under normal grid conditions. However, in the event of a significant grid voltage sag, the inverter injects 250VAR to support the grid voltage. In Figure 2.14, the dynamic response of traditional finite-set MPC is shown for a twenty percent sag in grid voltage at t_3. Shortly after t_3, there is a notable adjustment in the output current. Specifically, the amplitude of the current has increased and lads the grid voltage. This suggests the current reference has been adjusted to enable reactive power injection. However, in Figure 2.14a, substantial distortion occurs at the current peaks. Further, it is noted that the output voltage of the CMI falls below the grid voltage during this interval of current distortion. This voltage oscillation at the DC side is a result of the erratic and disparate power drawn from the voltage sources. In Figure 2.14b, the power draw characteristics of the finite-set MPC is shown. Not only is the power draw largely distinct for each voltage source, but there is notable fluctuations in the power drawn over time, marginalizing the stability of the DC-link voltages. This unregulated power draw is inherent to the finite-set MPC,



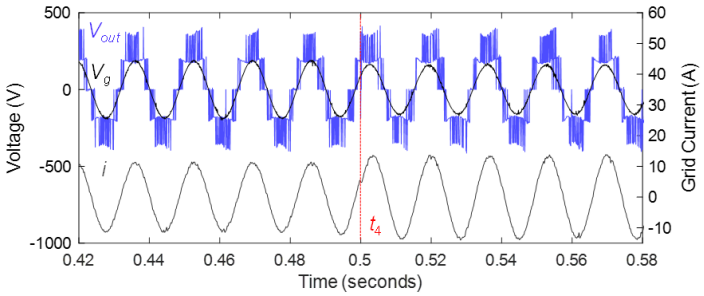
(a)



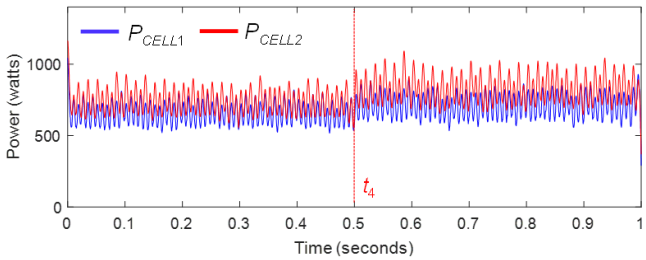
(b)

Figure 2.14: Dynamic response of traditional finite-set MPC for twenty percent grid voltage sag at t_3 (a) output voltage, grid voltage, and output current (b) power draw characteristics.

making the traditional predictive control unable to reach the desired output current. The same grid voltage sag is applied to the proposed HMPC at t_4. In Figure 2.15a, the current is noted to increase in amplitude and lag the grid voltage, as was noted in traditional finite-set MPC. However, the proposed HMPC does not experience the current distortion previously noted for the traditional finite-set MPC. In Figure 2.15b, there is a slight increase in overall power draw from the voltage sources, presumably caused by a slight reduction in efficiency from the increased current demand of the converter. However, stable and near-equal power draw characteristics are observed from the voltage sources, and thus the control is able to meet the demanded active and reactive power. The problem observed using the traditional finite-set MPC can be mitigated with increased DC-link capacitance. However, the stable and equalized power draw characteristics of proposed control helps to reduce the resultant voltage ripple across the DC-link, when compared to a traditional finite-set MPC scheme. With the proposed control, the sequence frequency objective can be applied to the control as $\boldsymbol{J}^3$, and thus the control tends to select redundant switching sequences



(a)



(b)

Figure 2.15: Dynamic response of proposed control for twenty percent grid voltage sag at t_4 (a) output voltage, grid voltage, and output current (b) power draw characteristics

more evenly. Furthermore, the proposed hierarchical approach mitigated the trial and error design stage of the weight factors of traditional MPC.

The total harmonic distortion of the grid current is found to be 3.75% when injecting 1kW of active power at steady-state The distortion of the comparable finite-set MPC is computed as 3.78%, thus there is no notable distinction in the error of the injected grid current between the two strategies. An FFT analysis of the injected current for the proposed control scheme is provided in Figure 2.16. What is most notable is the continuous nature of the harmonic content, which is inherent finite-set model predictive control, which operates with a variable switching frequency.

2.5.2 Comparison against PR current control scheme

The proposed control is also compared to a classical, PWM-based current control scheme for the CMI topology, shown in Figure 2.17. The modulation technique is referred to as multilevel subharmonic PWM [26]. A proportional-resonant controller is used to provide a modulation signal that matches the grid frequency. The gains of the controller, k_p and k_r, were tuned to 0.1 and 10,

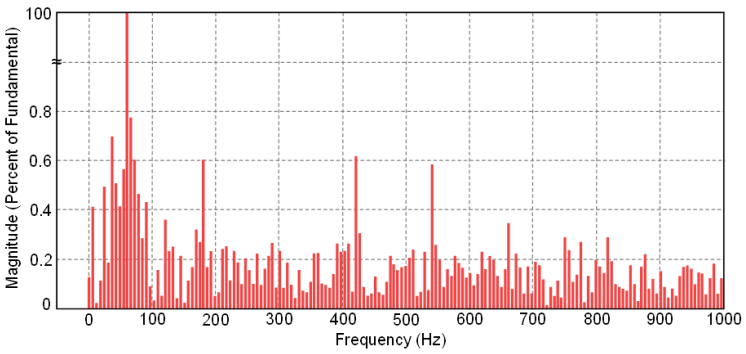


Figure 2.16: FFT analysis of injected current for proposed control scheme at 1kW power injection.

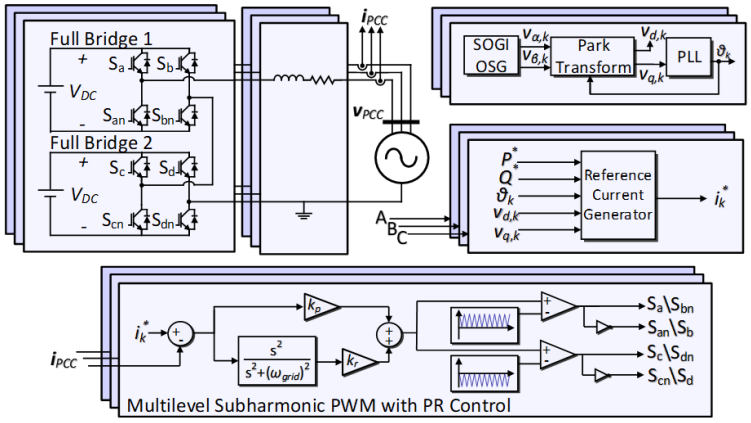
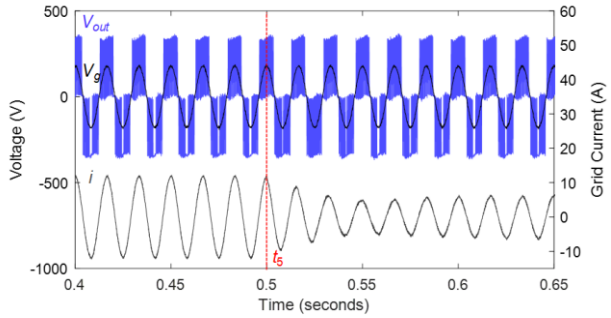
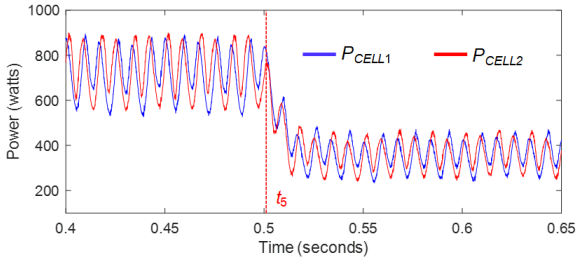


Figure 2.17: Cascaded H-bridge topology and proportional-resonant based current control with subharmonic pulse-width modulation.

respectively. There was a tradeoff when tuning the gains of transient response and power quality at steady-state. The carrier signals are set to 20kHz. Since finite-set MPC has a variable switching frequency, we could not ensure the switching frequencies of both control scheme were equal. In Figure 2.18a, the output current is tracked for a fifty percent reduction in reference. power at t_5. First, it is noted that the output voltage acts as a three-level switching scheme. The subharmonic PWM scheme operates each H-bridge with bipolar modulation. Thus, two cascaded H-bridges operate like a three-level inverter. The new reference current is tracked within a few grid cycles, or around 50ms. In Figure 2.18b, it is noted that the power draw of each H-bridge is roughly equal. The issue of unequal power draw is not inherent to this control scheme, as neither H-bridge operate with a zero-voltage output level. In Figure 2.19, the same transient is applied to the proposed



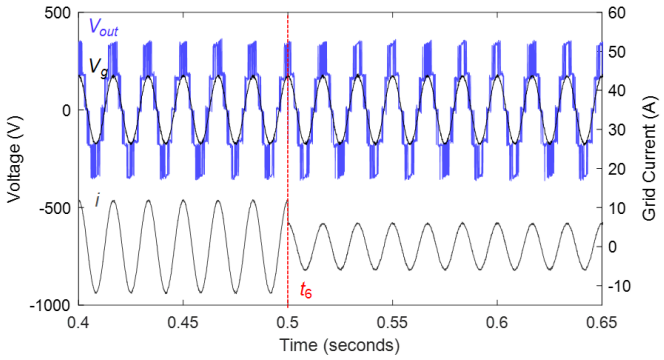
(a)



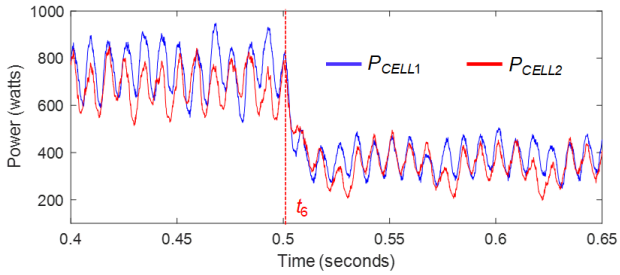
(b)

Figure 2.18: PR based current control with subharmonic pulse-width modulation with a reduction in power reference t_5 (a) output voltage, grid voltage, and output current (b) power draw characteristics.

hierarchical MPC at t_6. In Figure 2.19a, the new reference current is tracked in under 200µs. The control applies negative output voltage during this time to drive the current down to the updated reference. This modulation signal of the PR-based control scheme is unable to reduce sufficiently fast to match the transient response of the proposed control. In Figure 2.19b, the power draw characteristics are shown for the proposed control. There is only a slight distinction between the power draw of the two H-bridges. This is because the tertiary objective equalizes the rate of selection of redundant switching sequences, which mitigates selection bias in standard finite-set MPC. Further, the current THD of the PR-based control was computed as 5.84% at reduced power, while the proposed control produced current with 3.92% THD. Thus, the proposed control realizes significantly faster transient response and improved power quality when compared to the traditional control scheme, but is able to exhibit similar power draw characteristics that traditional finite-set MPC cannot. Table 2.3 documents the main findings of the results section. As noted in Section 2.4, the difference in power draw can be reduced further in the proposed control by



(a)



(b)

Figure 2.19: Proposed hierarchical MPC with a reduction in power reference at t_6 (a) output voltage, grid voltage, and output current (b) power draw characteristics.

Table 2.3: Control Comparison at 1kW Power Injection

Control Technique	Current THD	Settling time (P^* reduction)	$\lvert\langle P_{CELL1}\rangle - \langle P_{CELL2}\rangle\rvert$	Control Design Effort
Proposed Control	3.75%	~160µs	~65W	Low
Traditional FS- MPC	3.78%	~160µs	~325W	Medium
PR current control	5.57%	~51ms	~7W	High

increasing the current tolerance ε_1. Thus, in summary as presented in Table 2.3, the proposed hierarchical MPC is highly superior in power balancing feature comparing to tradition MPC. In comparison to PR current control, the proposed hierarchical MPC is highly superior in current THD and dynamic response measures. Finally, the proposed hierarchical can be implemented in a straight forward manner which highlight its low control design effort requirement comparing to traditional MPC and PR current control techniques. This is due to the fact that the proposed controller mitigated the control parameter tuning effort which is needed in PR current control and weight factor tuning effort which is needed in traditional MPC.

2.6 Conclusion

This chapter presented a new approach within the scope of finite-set MPC framework for power electronic converters. The presented predictive control approach does not include a generic cost function. Rather, control objectives are ranked and given a cost tolerance, and switching sequences that do not meet the specified cost tolerance are removed from the optimization set of all subsequent objectives. Not only does this remove the need to select among the equivocal weight factor design procedures, but it allows for the implementation of non-standard control objectives. The proposed hierarchical MPC is leveraged to enhance the operation of the cascaded multilevel converter at grid-edge with a self-power balancing feature. This is demonstrated with an incrementing cost objective referred to as sequence frequency, which equalizes the controller's selection of redundant switching sequences. Applying this objective is shown to improve equality in behavior, power draw, among the H-bridges in the CMI. The theoretical analysis and experimental results demonstrate the difference in power draw among the H-bridges is reduced by over 75% when evaluated against a comparable implementation of conventional MPC. When compared to a traditional PR-based control scheme, the proposed control scheme is shown to exhibit similar power draw characteristics, while maintaining superior dynamic response and power quality. Finally, the tuning stage is mitigated in the proposed control scheme which makes it superior to both PR-based control and conventional MPC schemes. Illustration of concept via simulation and hardware experiments verify the control's ability to track its respective P and Q set points in case of a grid-fault as a required feature for grid-supporting power converters in the future high penetrated grid with power converters. Finally, it is demonstrated that the proposed hierarchical MPC computational cost is significantly reduced comparing to the traditional MPC formulation.

Chapter 3 - Computationally-efficient Optimal Control of Cascaded Multilevel Inverters with Power Balance for Energy Storage Systems[2]

3.1 Problem Statement

With the sustained increase in renewable energy technology, energy storage systems are being considered to address the inherent drawbacks of renewable energy. Namely, the variability of renewables and its potential effect of grid stability and power quality at high penetration [48, 49]. A popular realization of energy storage systems is through the use of strings of battery modules in a cascaded topology, referred to as a CMI. This topology consists of series-connected full-bridge inverters, which have a battery module connected to their DC link [50-52]. The CMI enables direct connection to medium and high-voltage grids without the use of transformers and without the necessity of acquiring semiconductors with high voltage-blocking capability [25, 53]. Additionally, the reduced harmonic content of the output voltage waveform reduces filter requirements. The same benefits apply for battery modules, which can operate at voltages far lower than the grid voltage.

However, the compound topology inherent to CMI tends to limit the range of control techniques that are feasible. Particularly, cost optimization-based control schemes such as MPC lack feasibility for CMIs with a large number of voltage levels, as the optimization set is too large

[2] At the time of writing, the content of this chapter is undergoing third-round revisions as a post conference manuscript in the *IEEE Transactions on Industrial Electronics*. Some content has been retained with permission from the original conference article entitled "Computationally-efficient Hierarchical Optimal Controller for Grid-tied Cascaded Multilevel Inverters" in *IEEE Energy Conversion Congress and Exposition* by M. Easley, M. Hosseinzadehtaher, A. Fard, M. B. Shadmand, and H. Abu-Rub, 2019.

to run a practical sampling frequency of the digital controller. Nonetheless, MPC is often regarded as a control technique with generally fast dynamic response and easy inclusion of constraints and nonlinearities in the objective function. The subset of MPC known as *finite-control set* MPC, although perhaps the most intuitive solution, is the most computationally-demanding subset. Whereas *continuous-control set* MPC includes offline optimization and online searching of the optimal action, finite-control set MPC schemes predict the behavior of a set of possible control actions and implements that which is most aligned with the reference. This is distinct in that a switching modulator is eliminated, yet the optimization is done online [8]. Within the subset of finite-control set MPC, a further distinction can be made based on the nature of the control set. Common in three-phase applications, MPC schemes which choose among a set of space vectors is referred to as an optimal space vector (OSV) control [54, 55], while the remainder evaluate a set of possible switching states; this is known as optimal switching sequence (OSS) finite-set MPC [12]. With the number of non-redundant space vectors in a converter smaller than its number of possible switching sequences, OSS-based MPC proves the most computationally demanding within MPC [8]. To mitigate the computational burden inherent to finite-set MPC, authors have suggested modified MPC schemes which reduce the optimization set based on the previous control action. For OSV MPC, this is done by considering only adjacent voltage vectors in $\alpha\beta$ frame, referred to as the *nearest neighbor method* [45]. A similar idea is proposed for OSS MPC of multilevel inverters by applying a *voltage window* [56] which is essentially the nearest neighbor method applied to a one-dimensional control set. However, both techniques reduce the control set, which can negatively effect dynamic performance. In [57-59], the next-state current is replaced with a reference current in the predictive equation, creating a reference output voltage (in the $\alpha\beta$ frame for a three-phase system). By creating a reference output voltage and considering the output

voltage vectors time-invariant, next-state predictions can be removed. Though this has been proposed for OSV MPC, eliminating the next-state prediction has yet to be proposed for OSS MPC, the most computationally demanding subset of MPC. Additionally, removing redundant switching sequences, as is done in OSV MPC, has undesirable consequences in applications with multiple power sources. Depending on the redundant switching sequences that are implemented by the converter, power can be unintentionally drawn at uneven rates from the sources [60]. For the application of grid-tied batteries, this will create a divergence in the state-of-charge among the battery modules.

This work aims to eliminate all aforementioned challenges of finite-set MPC. The presented control includes a method of optimization similar to finite-set model predictive current control. Specifically, the control exploits the fact that next-step current is affected only by output voltage. In doing so, current control is achieved with a control set that increases linearly with the number of H-bridges in the inverter, rather than increasing exponentially, as is seen in traditional finite-set MPC. This distinction creates a remarkable reduction in online computations, thus increasing achievable sampling frequency when compared to traditional finite-set MPC on a given control platform/embedded control system. The method of reducing the control set involves an objective function which includes time-invariant control actions, eliminating the next-state predictions, the "predictive element" of predictive control. Thus, the proposed control is referred to as an optimal control scheme, where optimal control is a superset of predictive control. Still, the control selects a specific switching sequence at each sampling instant, a distinction shared by finite-set MPC. Further, switching event minimization is achieved as a secondary objective. This is done offline, by computing the number of switching events between switching sequences *during construction of the lookup matrix*. A collection of switching sequences that achieve the optimal

output voltage in minimal switching events is realized using a lookup matrix, where the previous switching sequence and optimal output voltage serve as the address for an optimal switching sequence (or list of switching sequences) within the matrix. The control rotates through the list of redundant optimal switching sequences at each call to its respective matrix address. This equalizes the time that each full-bridge implements a non-zero output voltage, which in turn equalizes the average power drawn from each isolated power source. This is particularly useful for grid-tied battery applications, as the control scheme supports equal state of charge for all batteries. By deriving current references according to an active and reactive power reference per phase, the power drawn is equalized not only for each source in a phase, but for all sources in the three-phase system. This holds true even for phase-to-neutral grid voltage sags.

The remainder of this chapter is organized as follows: Section 3.2 explains the formulation of the optimal output voltage selection. Section 3.3 explains the structure of proposed look-up matrix, dimension of the lookup matrix, and the cycling of redundant states. Section 3.4 compares proposed method with finite-set MPC. In Section 3.5 the proposed system is tested against transients in grid voltage and compared with MPC. Finally, the chapter is summarized with the conclusion in Section 3.6.

3.2 System Description

An overview of the topology and proposed control is shown in Figure 3.1. Each phase is controlled separately. This allows the optimization set to be reduced and allows the controller to respond to grid voltage imbalance. Without loss of generality, the control formulation is described for a single phase. The converter implements current control, but the current reference is generated from a desired active and reactive power injection. To implement instantaneous active and reactive power equations in single-phase, an OSG is applied to the sensed phase voltages. This permits the

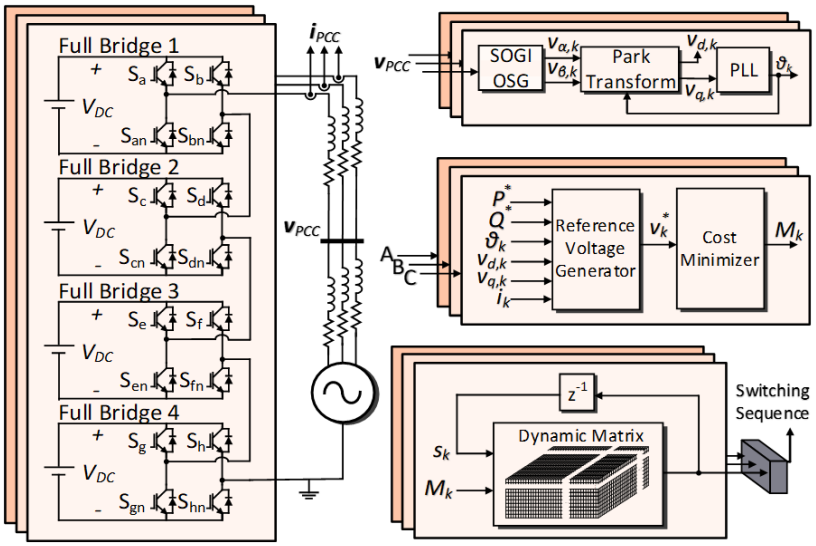


Figure 3.1: 9-level CMI with proposed hierarchical optimal control

use of active and reactive power equations in the rotating reference frame. As with the control scheme proposed in Chapter 2, the reference current is developed from a desired active and reactive power control in the rotating reference frame, demonstrated in (2.1) – (2.3). The phase angle of the grid voltage, θ_k, is computed using a phase-locked loop which regulates the term $v_{q,k}$ to zero. The determination of the optimal control action is formulated by considering the AC KVL equation of the circuit:

$$v_{inv} = Ri + L\frac{d}{dt}i + v \tag{3.1}$$

v_{inv} is the voltage applied at the output of the inverter. R and L are the equivalent series resistance (ESR) and inductance of the output filter, respectively. The single prediction method which is proposed in [57] for controlling three-phase grid voltages in the $\alpha\beta$ frame, is modified to find a reference line-to-neutral voltage for each phase, next-state predictions for grid current as is the well-established predictive current control method. By discretizing the differential in (3.1) with the measured and reference current, the optimal output voltage is computed:

$$v^*_{inv,k} = v_k + Ri_k + L(T_S)^{-1}\left[i^*_k - i_k\right] \tag{3.2}$$

where $v^*_{inv,k}$ is the optimal output voltage at discrete time instant k. In traditional finite-set MPC, the current in the next time instant is estimated for all control actions. With this technique, the output voltage that would produce the reference current in the next state is estimated, the benefit being that no prediction is necessary for the realizable output voltages.

The number of output voltage levels that a CMI can produce is $2M+1$, where M is the number of full-bridges in series. Considering the output voltage levels as the control set, the control set is now $[-M\ M]$, $M \in \mathbb{Z}$. In this chapter, this set of integers is considered as the control set, called vector $\boldsymbol{M}$. The output voltage to implement in the next time instant is computed as:

$$M_k = \arg\min \left| \boldsymbol{M} - \frac{v^*_{inv,k}}{\langle V_{DC,k} \rangle} \right| \tag{3.3}$$

where $<V_{DC,k}>$ is the average of the measured DC link voltages. It is assumed that the DC link voltages will be regulated to nearly equal values. By adjusting the control set to the optimal output voltage level M_k, a remarkable reduction in online computation has been achieved. For N H-bridges in a CMI, there are $2N+1$ voltage levels and 4^N unique switching sequences. A quantitative measure of increased computational efficiency is provided in Section 3.4. The following section will discuss how the lookup table is used to select a particular switching sequence from the optimal output voltage M_k.

3.3 Matrix Structure and Operation

In Section 3.2, it was revealed how finite-set MPC can be adjusted into a computationally-efficient optimal current control technique. Computational-efficiency is claimed here because finite-set current control is enabled while replacing the original control set, which has an exponential relationship with the converter topology, to a control set with a linear relationship. In this section, the remaining problems with finite-set MPC discussed in the introduction are

addressed. Namely, how the optimal output voltage level is translated to a specific switching sequence while minimizing switching events and addressing the issue of power imbalance.

For the remainder of the article, it will be useful to reference particular switching sequences (a specific combination of logic applied to each gate in the CMI). However, with the 9-level CMI, there exist 256 unique switching sequences, which is far too many to define explicitly in a table. Thus, a nomenclature is defined here to make referenced switching sequences deducible: the gate logic value of the top switch in each inverter leg (S_a, S_b, …, S_h) will be concatenated. Consider this sequence of logical values a binary number. Now, convert this value to its decimal equivalent and add one. Adding one simply allows the names of switching sequences to start at *switching sequence one* rather than *switching sequence zero*. An example of this nomenclature is provided in Figure 3.2. Note that the value of the lower switches in each leg (S_{an}, S_{bn}, …, S_{hn}) need not be considered when defining switching sequences, as they will be the logical opposite of the upper switches in their respective legs (e.g. if S_a is one/closed, S_{an} must be zero/open).

Offline switching event minimization can be achieved because the number of switching events between two distinct switching sequences can be determined a priori. For example, it is known that, to transition from switching sequence 256 to switching sequence 128, two switching events must occur (the gate logic of S_a and S_{an} must invert). Further, the output voltage level of a switching sequence can be determined as:

$$M = \sum_{i=a,c,e,g} S_i - \sum_{x=b,d,f,h} S_x \tag{3.4}$$

where the value of a switch S_x is denoted as one for on/closed and zero for off/open. Thus, given the deducibility of a switching sequence's output voltage level and the number of switching events required to transition to different switching sequences, it is concluded that switching event minimization can be achieved offline. What remains to be determined is *how* the number of

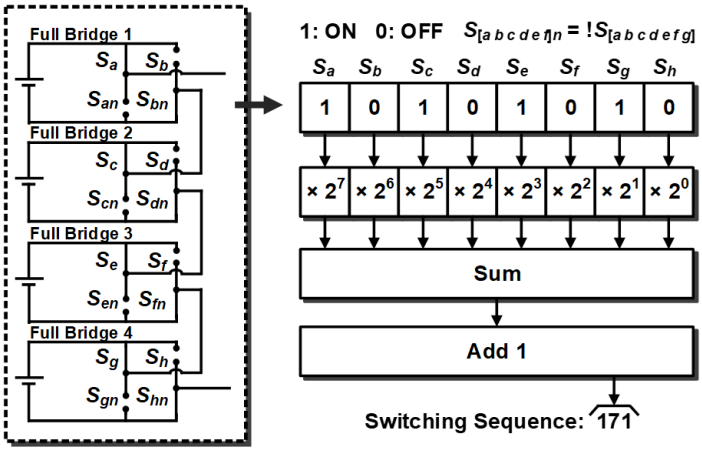


Figure 3.2: Nomenclature for each switching sequence in the optimal control scheme

switching events between switching sequences should effect the switching sequence selection. In traditional finite-set MPC, this is decided by factoring each cost term in the objective function with a *weight factor*. This will be detailed in Section 3.4. In this work, switching event minimization is considered a secondary objective to current control. This means that, once an optimal output voltage level M_k is determined, the set of switching sequences to consider in the next state is reduced to those which produce the optimal output voltage level. It is called a secondary objective because, although it can be locally minimized, the number of switching events will never alter the selected output voltage level M_k. For this reason, the control is considered a *hierarchical* optimal control. Current is globally optimized, while the number of switching events is minimized locally in the remaining control set.

Although switching events can be computed offline, as discussed previously, the control requires knowledge of the previously-selected switching sequence (s_k) for switching event minimization. Additionally, the selected switching sequence must produce the optimal output voltage level M_k. Thus, the control requires an indication of s_k (previously chosen switching sequence) and M_k (optimal output voltage level) to locate the optimal switching sequence. At this point, it is worthwhile to note that both s_k and M_k fall within a finite range of discrete variables

(integers from 1 to 256 and from -4 to 4, respectively). This is the intuition for the operation of the lookup matrix. A lookup matrix (or two-dimensional array) is apt for this problem, as we can input the optimal output voltage level M_k and the previous switching sequence s_k and retrieve the optimal switching sequence. This switching sequence would implement M_k in minimal switching events. This can be achieved by indicating M_k via lookup matrix row number, and s_k via column number, with s_{k+1} indicated by the value within that row-column combination, or *matrix address*. Thus, with the lookup matrix, switching event minimization is achieved without online computations.

The final issue of finite-set MPC to be addressed is power imbalance among the isolated voltage sources. First, the cause of this issue must be explained. To do so, consider a time instant k in which switching sequence 1 is implemented; all upper switches (S_a, S_b, …, S_h) are set to logic-low, and the resultant inverter output voltage is zero. During current optimization at time instant k, M_k is computed as one, meaning the controller has determined that an output voltage of $+V_{DC}$ will bring the current closest to i^*. This requires a change in the output voltage level of +1. To change the voltage level by one with minimal switching events, the logic gates of one leg must be inverted. In this scenario, inverting the gate logic of any of the left-most full-bridge legs will create the appropriate output voltage level (inverting the logic of S_a/S_{an}, S_c/S_{cn}, S_e/S_{en}, or S_f/S_{fn}). Thus, using the switching sequence nomenclature previously established, switching sequences 3, 9, 33, and 129, are optimal and redundant. That is, each produce the optimal output voltage level M_k in minimal switching events. Note here that these switching sequences each apply positive voltage across a different full-bridge output. In turn, power will be drawn from a different battery module for each switching sequence. Thus, by ignoring all but one of these redundant switching sequences, power will be drawn exclusively from one of the battery modules. If one redundant sequence is chosen at each address of the lookup matrix, this will result in unequal power draw from the

batteries. In finite-set MPC, only one of these switching sequences will be selected in this scenario. Traditionally, it is the first-seen optimal switching sequence. Thus, in this scenario, finite-set MPC would only select switching sequence 3, which draws power from full-bridge 4's battery module. Since the control actions of a finite-set MPC are non-deterministic, it cannot be said with certainty how exactly power draw will vary among the isolated battery modules. Even if the sequences are carefully selected such that non-zero voltage is applied across each full-bridge an equal number of times in the lookup table, the frequency in which addresses are called will vary, thus unequal power draw will remain inevitable. Thus the power imbalance that occurs from finite-set MPC in this topology is caused by the existence of redundant optimal switching sequences that are ignored during optimization. Figure 3.3 shows the number of redundant optimal switching sequences exist at each matrix address. The number of redundant switching sequences tends to increase as the magnitude of M_k decreases. This is because each H-bridge has two switching sequences that produce zero output voltage. Oppositely, it is noted that for all addresses at row one and row nine, there is only one switching sequence; just one switching sequence can achieve an output voltage level of four (switching sequence 171), and likewise for negative-four (switching sequence 86). Additionally, it is intuitive that if the output voltage level of s_k is equal to M_k, then there will only be one optimal switching sequence, and s_{k+1} will be set to s_k. That is, if $M_k = M_{k-1}$, then no switching events must occur.

The power imbalance among the isolated voltage sources is mitigated by implementing all redundant switching sequences over time. This can be done by implementing a multidimensional array. This is simply a matrix with non-unitary depth; this data structure can also be considered a 3D matrix. The row and columns work exactly as previously described with the lookup table, except the additional layers will be used for selection of redundant switching sequences. Figure

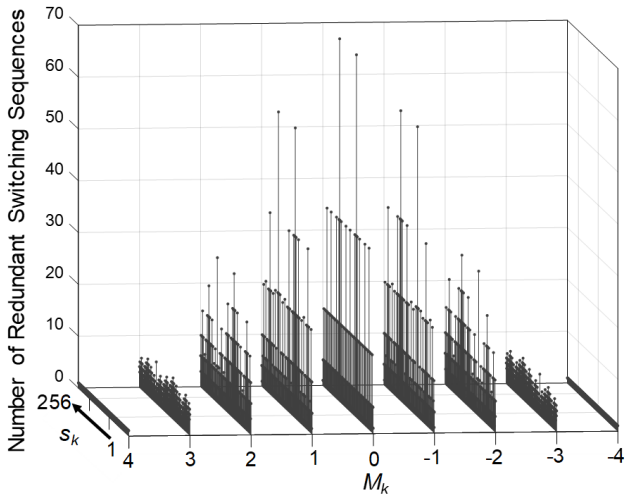


Figure 3.3: Number of redundant switching sequences at each matrix address.

3.4 illustrates the concept more clearly. Each matrix address has a list of redundant optimal switching sequences. The first two layers decide which of the redundant switching sequences are in queue for selection. Layer one of each matrix address is an integer equal to the length of the list in its associated stack (*stack* referring to the elements in the data structure which share a row and column number/matrix address). Layer two acts as a pointer, containing the location of the next switching sequence to implement when the associated matrix address is called. Figure 3.5 explains exactly how the matrix is used. Once M_k is computed, it is converted to the associated row number by adding five (row number one is associated with M_k equal to negative four). The previous switching sequence determines the column number. Then, the layer number (z-coordinate) that will be implemented is determined by layer two, making layer two referred to as the *pointer layer*. The value within the layer number being pointed to is the selected switching sequence according to the previously defined nomenclature. The value within layer two of the selected matrix address is then incremented. This will ensure that, the next time this matrix address is called, a new redundant switching sequence will be selected. This is visualized in block six of the algorithm. The value in layer two was *n*, thus the switching sequence that was selected was contained in cell

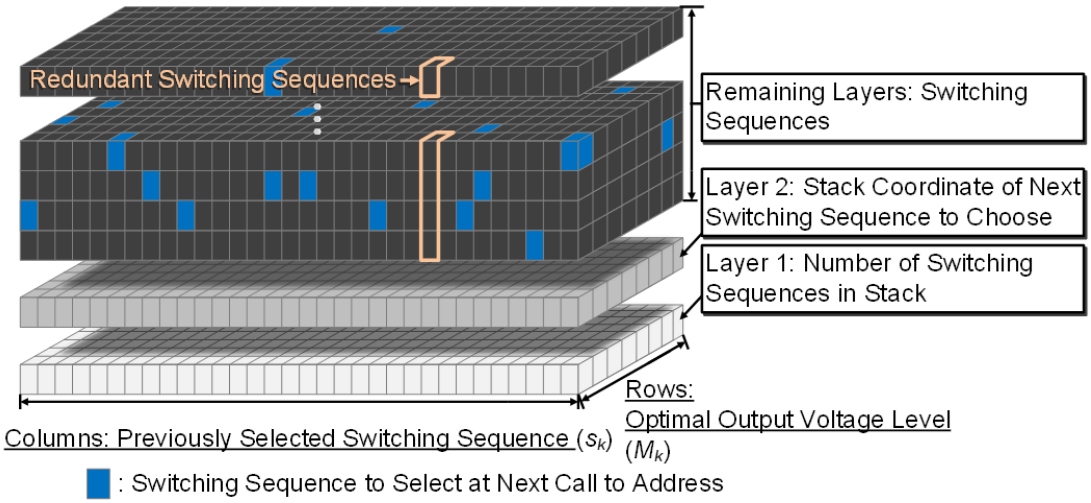


Figure 3.4: Structure of three-dimensional lookup matrix.

n of the stack. Then, layer two points to the cell above it (cell *n*+1) for the next time this address is called. The conditional block that follows is related to what was shown in Figure 3.3. That is, these lists of redundant switching sequences vary in length. However, the matrix must have a fixed length, width, and depth. Thus, the depth of the matrix must be large enough to contain the longest list (a list of seventy switching sequences). However, most lists are not this long. Pointing to layer numbers that extend beyond the lists must be prevented. This is the purpose of layer one, which specifies the length of the list at its address. If the value of layer two points beyond the limits of the list, the value is reset to three, which contains the first switching sequence in the list. Thus, the control rotates through the list at each address.

Here, it must be acknowledged that using a multi-dimensional array as described is not the most memory-efficient solution, since many of the matrix elements contain null blocks, or places in memory that have been allocated but will never be written to nor read from. Null blocks are necessary to use a multi-dimensional array as the data structure. The memory-efficient solution is to convert the matrix into a group of lists. This way, each stack would have length equal to the number of redundant sequences at its address, removing the null blocks. Figure 3.6 demonstrates

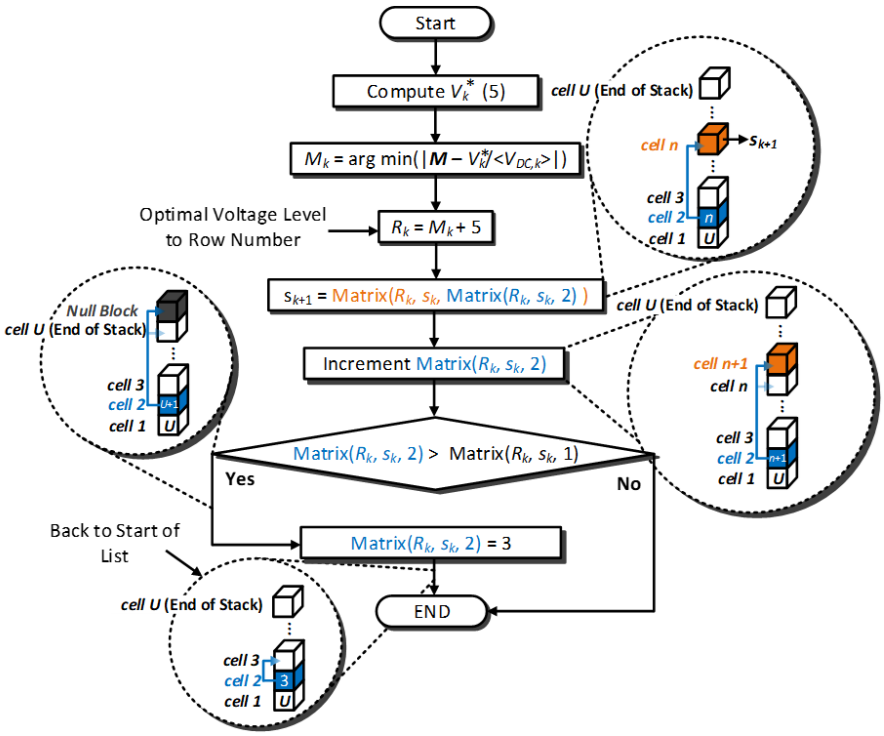


Figure 3.5: Algorithm describing how switching sequences are selected from the lookup matrix.

how changing the algorithm to implement a group of lists would reduce the necessary random access memory (RAM). Note that, for CMI topologies with more than four full-bridges, the variables must be defined using two-byte variables (presumably 16-bit unsigned integers). This is because the number of possible switching sequences exceeds 256, which is the total number of values that can be distinguished in a single byte. Despite the memory that can be saved by converting to a group of lists, the control technique is described using a multi-dimensional array, to aid in visualizing the control technique and to avoid making the concept more convoluted. The algorithm described can be easily modified to the memory-efficient implementation.

3.4 Computational Comparison with Finite-Set MPC

To better comprehend the reduction in on-line computation to reach an appropriate control action with the proposed control technique, the operations are compared against finite-set MPC. First, the finite-set MPC of which the proposed control is compared against must be defined.

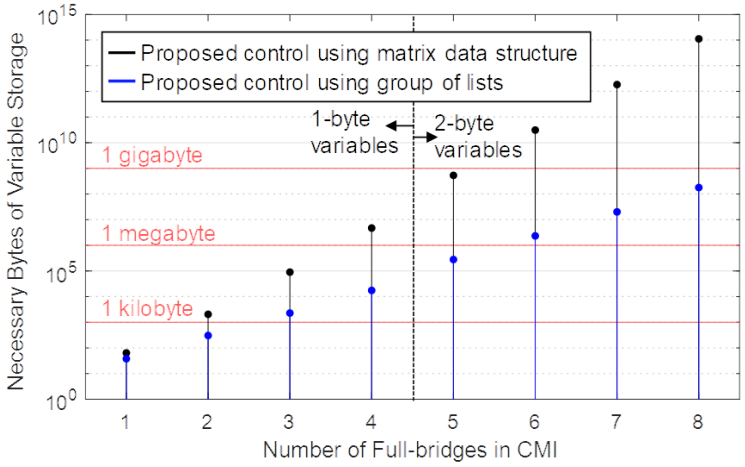


Figure 3.6: Logarithmic plot of the number of bytes that must be allocated to the proposed control scheme vs. the number of H-bridges in the CMI topology.

The finite-set MPC that is being considered will contain the same objectives as the proposed control: current-control with minimal switching events. The reference current is developed exactly as it is for the proposed control using (2.1) – (2.3), and thus the techniques will only differ in how the control action is decide. In traditional MPC, each control action is individually considered, meaning a prediction will be computed for each possible switching sequence in finite-set MPC. The next-state prediction is derived from (3.1), of which the differential term is discretized using the Forward Euler approximation. Solving for the next-state prediction leads to:

$$\boldsymbol{i}_{k+1} = (1 - RT_s(L)^{-1})i + T_s(L)^{-1}(\boldsymbol{v}_{inv,k+1} - v_k) \tag{3.5}$$

where $\boldsymbol{i}_{k+1}$ is a vector of next-state current predictions according to each possible switching sequence, and $\boldsymbol{v}_{inv,k+1}$ is a vector of the output voltage that would results from each possible switching sequence, which is defined by:

$$\boldsymbol{v}_{inv,k+1} = V_{DC1}(S_{a,k+1} - S_{b,k+1}) + V_{DC2}(S_{c,k+1} - S_{d,k+1}) + V_{DC3}(S_{e,k+1} - S_{f,k+1}) + V_{DC4}(S_{g,k+1} - S_{h,k+1}) \tag{3.6}$$

where the gate signals are defined for all 256 switching sequences. Further, since switching event minimization is another control objective, defined as the changes in gate logic between possible

switching sequences and the previously selected switching sequence. This will be included in the overall cost function $\boldsymbol{J}$, which is defined as:

$$\boldsymbol{J} = \left|\boldsymbol{i_{k+1}} - i^*\right| + \lambda \times 2 \sum_{i=a,b,...,h} \left|S_{i,k+1} - S_{i,k}\right| \tag{3.7}$$

The factor of two accounts for the lower switches' gate signals which must change along with their respective upper switches (if S_a changes, so must S_{an}). The λ is referred to as a weight factor, and its magnitude is chosen according to the desired effect of switching events on the control action. Note here that there are only nine unique next-state current predictions: one for each voltage level. Thus, switching event minimization can be considered among a single next-state current prediction by making λ sufficiently small. So long as the cost of switching events does not exceed the differences between next-state current predictions, the optimization will work similarly to the proposed optimal control scheme, where switching event minimization acts as a secondary control objective. The control action which minimizes $\boldsymbol{J}$ will be chosen for time k+1:

$$s_{k+1} = \arg\min(\boldsymbol{J}) \tag{3.8}$$

With both control techniques detailed, Table 3.1 summarizes the number of floating-point operations needed for one iteration of the finite-set MPC and the proposed control, as a function of H: the number of full-bridges in the CMI. The column labeled *Value Comparisons* is related to the size of the control set of which each control minimizes its objective loss. Figure 3.7 is a

Table 3.1: Computation Comparison with Finite-set MPC
***H*: Number of Full-Bridges in CMI**

Control Algorithm	Additions	Subtractions	Divisions	Multiplications	Value Comparisons
FS-MPC	$4^H(H+1) + 4H + 1$	$4^H(H+3) + 4H$	2^{2H+1}	$4^H(H+3) + 1$	4^H
Proposed Control	$H+3$	$2H+1$	$H+2$	2	$2H+1$

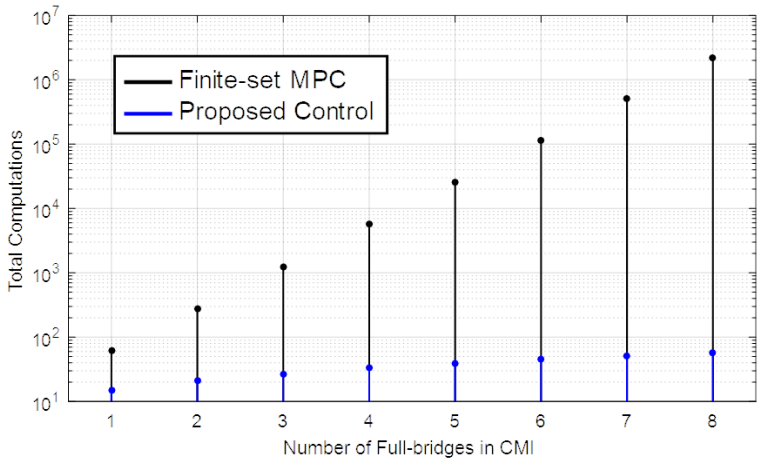


Figure 3.7: Logarithmic plot of the total computations of the proposed control algorithm and a comparable finite-set MPC algorithm vs. the number of full-bridges in the CMI topology.

logarithmic plot which visualizes how the total computations increase with the number of full-bridges in the CMI topology. The critical difference in the number of operations is that the proposed control compares voltage levels in real-time, while the finite-set MPC compares switching sequences in real time. The number of voltage levels increases linearly with the number of full-bridges [25], while the number of switching sequences increases exponentially. Here it is evident how standard finite-set MPC loses its feasibility as the CMI includes more full-bridges. Note that the total computations should not necessarily be considered directly proportional to the execution time nor to potential sampling frequency of the controller. The speed of each type of floating-point operation will vary according to the selected processor, and the speed of one iteration will depend of the selected processor(s) and number of processors used in the CMI's embedded system. Thus, a theoretical discussion on explicit improvements in run-time is avoided. However, an empirical metric is available, as both control techniques are realized on hardware in Section 2.5. The control schemes are implemented using the dSPACE CP1103 rapid control prototyping hardware. The graphical user interface is *ControlDesk*. Upon loading the control algorithms into the hardware, *ControlDesk* offers a metric called *turnaround time*. This metric

represents the time taken to execute a single iteration of the control algorithm, providing insight into the real-time sampling frequency that can be used without causing overrun. However, the *turnaround time* is not suitable for comparing the two control algorithms alone, as the controller's analog-to-digital converters and reference signal generation creates substantial overhead on total execution time. It is possible to measure the execution time of an individual subsystem. This metric will be distinguished with the label *control execution time*. Both metrics are measured. Upon loading the two control algorithms into the CP1103, around 8,000 samples are collected for each metric. The standard deviations and averages for both turnaround time and controller execution time are provided in Table 3.2. As shown in Table 3.2, the model predictive control takes nearly 28μs to compute, while the proposed control finds the optimal switching sequence in under 5.6μs, making the proposed control roughly five times faster for this topology.

3.5 Results and Discussion

The proposed control scheme is validated and compared with finite-set MPC experimentally for one phase. Figure 3.8 provides an overview of the hardware setup Table 3.3 presents system parameters, both for the proposed optimal control and the comparable finite-set MPC scheme. The plots that follow are exported measurements from the dSPACE CP1103 discussed in Section 3.4. However, for the controller to successfully execute in real time, the sampling rate of the exported data is reduced by a factor of four. In other words, the control samples at 50kHz, but the exported data is sampled at 12.5kHz. A bi-directional grid emulator is used for testing the control under changes in grid condition. First, the proposed control is shown for changes in grid voltage and power reference. Then, it is compared with finite-set MPC to demonstrate the comparable tracking capability and balancing of power draw from each H-bridge in the topology.

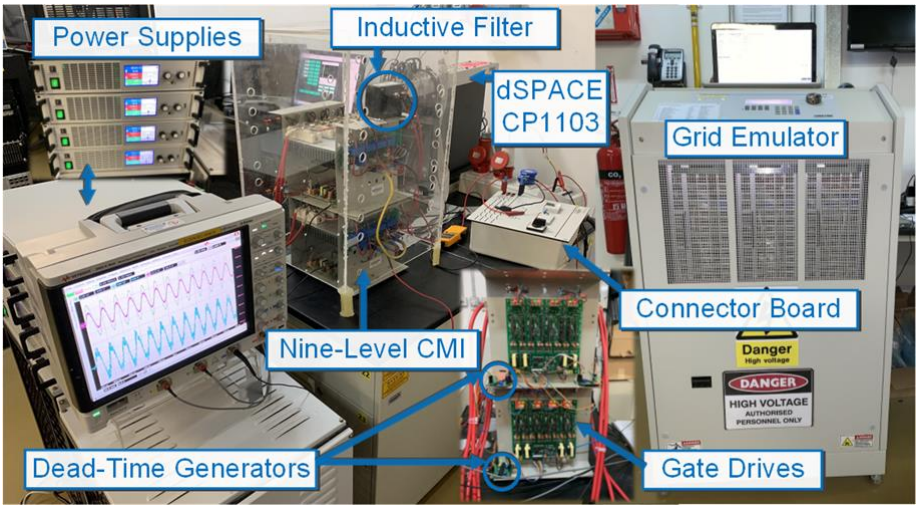


Figure 3.8: Hardware setup for experimental validation of both the proposed optimal control and comparable finite-set MPC.

Table 3.2: Controller Speed Comparison with MPC

Timing on CP1103	Finite-Set MPC	Proposed Control
Controller Execution Time (Average / Standard Deviation)	27.95 µs / 128.7 ns	5.578 µs / 26.09 ns
Turnaround Time (Average / Standard Deviation)	35.87 µs / 158.0 ns	13.57 µs / 99.22 ns

Table 3.3: System Parameters for Hardware Experiment

Parameter	Value
P^* (Single-Phase Active Power Reference)	1 kW
Q^* (normal grid condition)	0 VAR
Q^* (grid voltage sag)	250 VAR
DC Link Voltages	80V
f_S (proposed control scheme)	50kHz
f_S (comparison with MPC)	25kHz
Filter Inductance	2.5mH
Filter ESR	0.2Ω
Imposed dead-time	1µs

The control's response to dips in grid voltage are shown in Figure 3.9, in which a ten percent grid voltage is induced at t_1. To retain 1kW power injection, the active current reference is increased, as evidence by the increase in grid current amplitude. Figure 3.10 shows the system

response to the grid voltage returning to normal condition, where the control responds by reducing i^*_d from about 13A to 11.5A.

Next, a more severe dip grid voltage is considered. Figure 3.11 shows a twenty percent sag in grid voltage at t_3. For more substantial grid voltage sags, the control is programmed to inject 250VAR. Thus, i^*_q is stepped up from zero to around 3.7A. Meanwhile, i^*_d also increases to about 14.7A to retain 1kW power injection during the voltage sag. This leads to a notable increase in grid current amplitude following t_3. Reactive power injection is evidenced by the waveform of the injected grid current following the grid voltage. In Figure 3.12, the grid voltage returns to healthy condition at t_4. The control responds quickly by reducing i^*_q to zero and returning i^*_d to 11.7A, as evidenced both by the reduction in grid current amplitude and immediate phase-alignment of the grid voltage and injected grid current.

As discussed previously, the power reference is considered a fixed value, or determined from a higher-level control. A step-change in the active power reference of the CMI is considered to emulate such changes. Figure 3.13 demonstrates the response of the system to a reduction in active power reference. At t_5, P^* is reduced from 1kW to 0.5kW. The CMI quickly tracks the new current reference, which has a new active current amplitude of 5.88A. In Figure 3.14, the active power reference returns to 1kW at t_6. Again, the control tracks the new, increased current reference.

The proposed control is now compared to a traditional finite-set MPC. For the topology presented, the finite-set MPC is unable to sample at 50kHz. Table 3.2 from Section 3.4 showed that the turnaround time for the MPC is about 36μs. To avoid any over-run errors, the sampling frequency of the MPC had to be reduced to 25kHz (sampled every 40μs). Further, so the comparison is not biased, we are including results in which the proposed optimal control is also sampled at 25kHz, rather than the 50kHz previously presented. An increase in current distortion

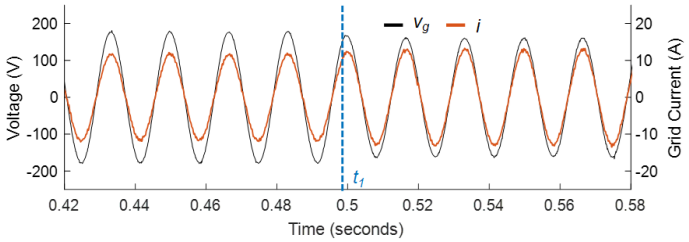


Figure 3.9: Grid voltage and current. A ten percent voltage sag occurs at t_1.

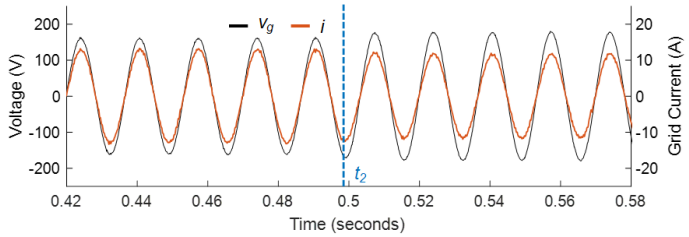


Figure 3.10: Grid voltage and injected current. A recovery in ten percent grid voltage sag occurs at t_2.

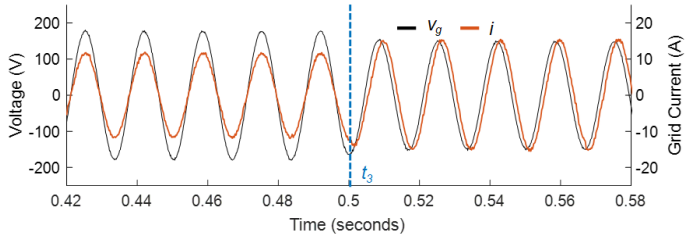


Figure 3.11: Grid voltage and current. A twenty percent voltage sag occurs at t_3.

is apparent, as a result of the reduced sampling rate, both for the finite-set MPC and the proposed control. Additionally, the exported measurement data from the CP1103 experiences a similar reduction in sampling rate, to 6.25kHz. The control schemes are tested at steady-state and for a step-change in active power reference. Figure 3.15 presents the finite-set MPC. The control is able to track the current reference, and can quickly adjust the reduction in active power reference at t_7. However, Figure 3.15b shows the power drawn from each battery module for the finite-set MPC, and it is revealed that the power draw characteristics are drastically varied among the cells. As discussed previously, the finite-set MPC will sequence through a list of all possible control actions (switching sequences), and retain the first action which optimizes its control objectives. Although in many instances there will be redundant switching sequences, the control will only implement

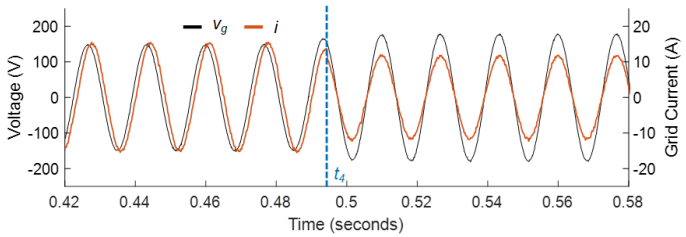


Figure 3.12: Grid voltage and injected grid current. A recovery in twenty percent grid voltage sag occurs at t_4.

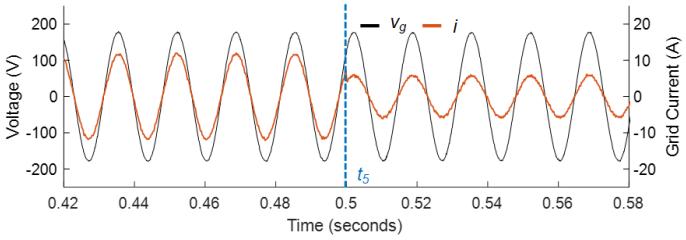


Figure 3.13: Grid voltage and injected grid current. A decrease in active power reference from 1kW to 0.5kW occurs at t_5.

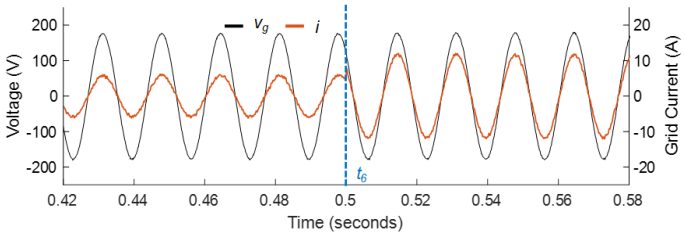
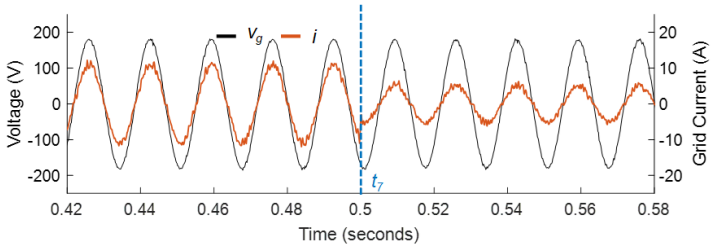
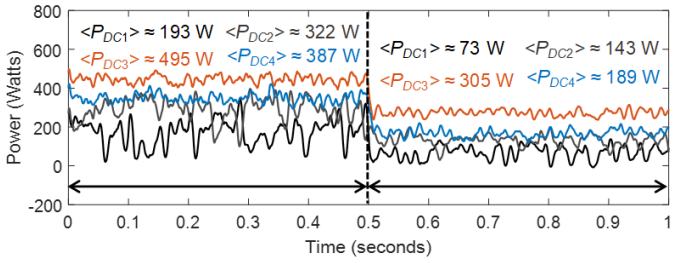


Figure 3.14: Grid voltage and injected grid current. An increase in active power reference from 0.5kW to 1kW occurs at t_6.

the first-seen switching sequence. In this implementation, the control evaluates in numerical order starting from switching sequence one to switching sequence 256. Based on the defined nomenclature, the higher the full-bridge number, the less significant its switches are in defining the binary number. Said another way, the upper switches in full-bridge one (S_a, S_b) are the most significant bits, and the upper switches in full-bridge two (S_g, S_h) are the least significant bits. Thus, switching sequences when S_a and S_b are switched on tend to be considered less-often, tending toward lesser power draw. It was also mentioned in Section 3.3 that this cannot be avoided, even when reducing the control set for finite-set MPC, because the frequency in which each voltage level is selected cannot be easily predicted or controlled. In Figure 3.16a, the same transient as was



(a)



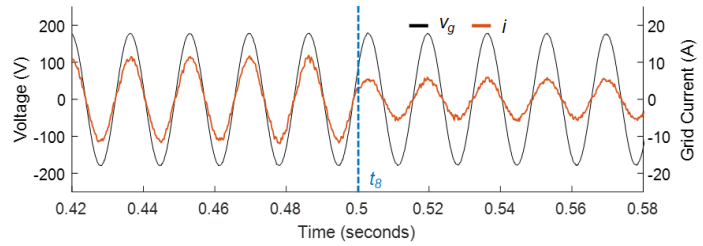
(b)

Figure 3.15: Step change in P^* occurs at t_7 using FS-MPC (a) Phase grid voltage and current (b) Power draw from each DC link.

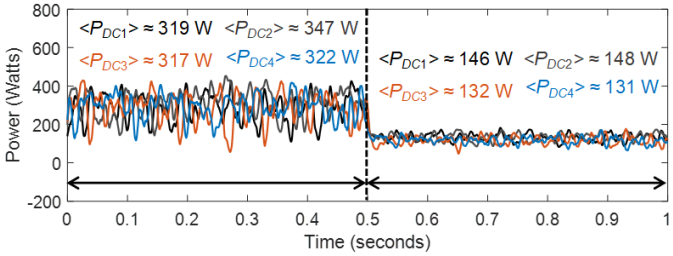
shown in Figure 3.13 is shown again, only this time at the reduced sampling rate. The current tracking capability is similar to the finite-set MPC. Additionally, Figure 3.16b shows the power draw characteristics of each battery module. A much more equalized power draw among the cells is noted, which occurs both before and after a reduction in the active power reference. Slight inequalities in the average power draw is noted. However, this is likely caused by slight mismatch in the DC voltages and the reduction in sampling rate. Because there is no bias among the redundant switching sequences, this difference in average power draw will reduce as the measured time interval increases.

3.6 Conclusion

An optimal current control technique for grid-connected CMI has been proposed which addresses the major setbacks of the finite-set MPC control paradigm for the proposed application. The proposed control achieves equivalent current tracking as is attained using finite-set MPC, but sheds substantial computation associated with next-state current predictions. Further, by invoking time-invariant parameters in the topology, online computations become a linear function of the number



(a)



(b)

Figure 3.16: Step change in P^* occurs at t_8 using the proposed control scheme (a) Phase grid voltage and current (b) Power draw from each DC link.

of full-bridges, rather than an exponential function seen by the comparable finite-set MPC. This reduced the control's execution time by a factor of five, allowing the proposed control scheme to run at twice the sampling rate of the finite-set MPC scheme during experimentation. The control also implements switching event minimization as a secondary objective using a lookup matrix. Further, the lookup matrix contains a third dimension, which contains a variable list of redundant optimal switching sequences at each address of the lookup matrix. The matrix is supplemented with an algorithm that cycles through the list at each matrix address. This eliminates the unintentional bias towards redundant optimal switching sequences that is seen in finite-set MPC. By removing this bias, the power drawn from the isolated battery modules is balanced on average.

Chapter 4 - Autonomous Model Predictive Controlled Smart Inverter with Proactive Grid Fault Ride-Through Capability[3]

In this chapter and the following chapter, the arbitrary discrete instances *k* are denoted by brackets, rather than included in the subscript as was done in the previous chapters. This is done to ensure the formulation of the rolling RMS measurement can be clearly defined, and because subscripts are already used to denote particular impedance elements within the presented inverter topologies.

4.1 Problem Statement

The interest of PV power generation has been steadily increasing, as evidenced by the substantial market growth from the last two decades [1]. To improve the commercial viability, the effects of high PV penetration of the grid must be addressed. Stability is a critical issue when considering a high PV-penetrated grid [61]. Voltage swelling can occur as a result of rapid increases in PV power generation/solar irradiance, referred to as *overloading* [62]. Further issues occur when the PV system transitions to islanded mode in the event of grid faults [1]. It is desirable to seek alternatives to disconnection during grid faults. The concept of low-voltage ride through (LVRT) is a proposed method of grid voltage support, where the grid-connected inverter will inject reactive power prior to disconnection upon sensing a dip in grid voltage. By applying such a control feature, the smart inverter can turn PV generation from a stability hazard to a stability asset. In recent years, there has been discussion to update grid codes pertaining to PV inverters to include this capability, and has been implemented in some grid codes [63-65]. However, adjusting the

[3] The content of this chapter has been obtained with permission from an Early Access version of a journal publication titled exactly as this chapter "Autonomous Model Predictive Controlled Smart Inverter with Proactive Grid Fault Ride-Through Capability" in *IEEE Transactions on Energy Conversion* by M. Easley, S. Jain, M. B. Shadmand, and H. Abu-Rub, 2020.

active and reactive power set-points to produce these advanced grid-support features makes for a more difficult control problem.

Creating the control scheme for grid-connected PV inverters is a complicated control problem in of itself, as cascaded control loops are generally required. This is because grid-connected PV strings are typically introduced to the grid through a two-stage converter scheme. That is, a DC/DC boost converter followed by a DC/AC inversion stage [66, 67]. The first stage performs the MPPT and boosts the PV voltage to meet the DC-bus voltage requirement, while the DC/AC inversion stage regulates DC-bus voltage and ensures PV-to-grid power flow. Converters which contain an impedance network, termed impedance source inverters or Z-source inverters (ZSI), have been proposed for such an application [68-70]. The arguments for these topologies are the possibility of removing the DC-DC converter or removing the need for a transformer on the AC side, potentially reducing the overall converter expenditure, improving the efficiency [71]. The permitted *shoot-through state* can adjust the input impedance of the converter, making it possible to perform a MPPT control, and can step up/down the DC source voltage to meet the desired DC link voltage level. The voltage-fed quasi impedance source inverter (qZSI) is generally considered more practical among the Z-source topologies for PV applications, as it permits continuous input current, positively influencing the lifetime of the PV string [72, 73]. Despite the implementation of a single-stage topology, dual-stage control structures are still proposed, on account of the contrast in dynamics at the AC and DC sides of the circuit. In [74], such a control structure is proposed, and even includes both a current-control loop for grid-connected applications and voltage-control for islanded mode. More traditional control schemes with cascaded structure such as these tend to suffer from slower dynamic response. Additionally, incorporating advanced, grid-supporting functionality such as LVRT capability is not straightforward.

MPC is a potential candidate to address the challenges in control of grid-tied PV inverters with advanced functionalities, as it has become a competitive control scheme [8, 75, 76]. It has also been recently implemented in industry [27]. With MPC, particularly the subset known as optimal switching sequence, the switching state is selected via minimizing a cost function, which can contain several control objectives of different natures. Thus, implementing adjustable active and reactive power set-points is actualized by adjusting references of the cost function, and thus will have no bearing on the control structure. However, what allows multi-objective MPC to boast its simplicity is what causes difficulty in the design stage. To adjust the *impact* of each control objective in the cost function, each penalty term is multiplied by a weight factor. This means implementing a multivariable cost function generally requires preliminary tuning. The optimization of the weight factors is a design burden tied to traditional MPC. Discussion in literature on how to design the weight factors is limited to trial and error techniques for finding an optimal set-point [38]. This also requires the user to define *optimal behavior* when designing the controller, which is not straightforward for multi-objective optimization. To be succinct, the optimal weight factor ratio of multi-objective MPC tends to be laborious to navigate and difficult to define. Additionally, a static weight factor ratio will likely not have the best achievable performance for all considerable scenarios i.e. MPPT versus LVRT operation modes and the transient periods therein. Such issues exist for those who attempt to implement previously proposed predictive control schemes for the qZSI, such as [77, 78]. The authors in [79] acknowledge the issue of static weight factors during transient conditions, and address the preliminary design of the weight factors for each mode must still be done, which is especially difficult for the implemented cost function which contains four penalty terms. In [80], a predictive control scheme for a four-leg qZSI is proposed, but is not considered for grid-connected

applications. Predictive current control of the qZSI is also considered in [81], where the authors address the difficult cost function design by optimizing the objectives hierarchically, ultimately cutting the iterative computation in half. Still, the considered application is limited to a resistive load with an ideal DC source. In [82], the authors propose a predictive control for a battery-assisted qZSI, which reduces iterative computation by using the predictive control to decide if shoot-through should be implemented in the next-state. This concept is extended to a cascaded battery-assisted qZSI topology in [83], where the control is generalized to any number of qZSI converters (cells) in series. However, output current control is not implemented in the predictive control, and requires a proportional resonant controller and modulation scheme for output current tracking. The authors in [84] propose a time-averaged model of the qZSI dynamics to produce an optimal shoot-through duty ratio and modulation index. Unlike finite-set MPC, this control implements a modulator and thus does not include the fast-dynamic performance associated with direct computation of an optimal switching sequence. Additionally, the proposed control in [84] is for isolated qZSI (off-grid) with constant DC Source and R-L load.

In order to include the aforementioned control features while addressing the issues of traditional MPC, this chapter proposes an autonomous model predictive control (AMPC) solution to improve the capabilities and robustness of grid-connected PV systems. Specifically, the active/reactive power of the smart inverter are decoupled, and the set-points are adjusted according to the grid condition. Three reactive power injection (RPI) strategies are shown, which adjust the active power reference [85]. The active power is adjusted using a flexible power point tracking technique, which can pull the PV module's voltage above its maximum power point to reduce power injection. The chapter also proposes a method which removes any tuning of the weight factors by autonomously adjusting the weight factors according to the controller's tracking

performance of each objective. This, in addition to the seamless transitions between the MPPT and LVRT modes of operation according to the grid's condition, make the proposed control considered autonomous. As a case study, the proposed control is implemented using a qZSI. The buck-boost capabilities of the inverter make it possible to use as a single-stage solution for grid-connected photovoltaics This work is considered an extension of the work in [86], with additional analysis, experimental verification, and more advanced predictive control which features the auto-tuning weight factor technique. The tracking improvements from the auto-tuning feature are demonstrated experimentally, comparing to the original, static weight factor ratio in [86]. Additionally, this proposed control scheme implements an index variable, s, in the cost function. This variable is used to eliminate iterative computations associated with redundant next-state predictions. Without any degradation of tracking capabilities, five of the ten next-state prediction calculations are removed using this technique, enhancing the control's computational efficiency.

Beyond the introduction, Section 4.2 explains the decoupled control scheme and its formulation. Section 4.3 presents the auto-tuning weight factor feature. Section 4.4 includes several experiments to validate the performance of the proposed control scheme. Finally, the proposed system and its performance are summarized in the conclusion section.

4.2 System Description and Control Strategy

Figure 4.1 shows the qZSI-based PV system along with the proposed control scheme. The control works in two modes: (i) normal operation mode when grid voltage is within 90%-110% [85] of its rating, in this mode the predictive MPPT technique is triggered to harvest the maximum power from PV; (ii) the LVRT mode during grid voltage sags of more than 10%, in this mode the qZSI will move from MPP operation to support grid as an ancillary service from PV inverter.

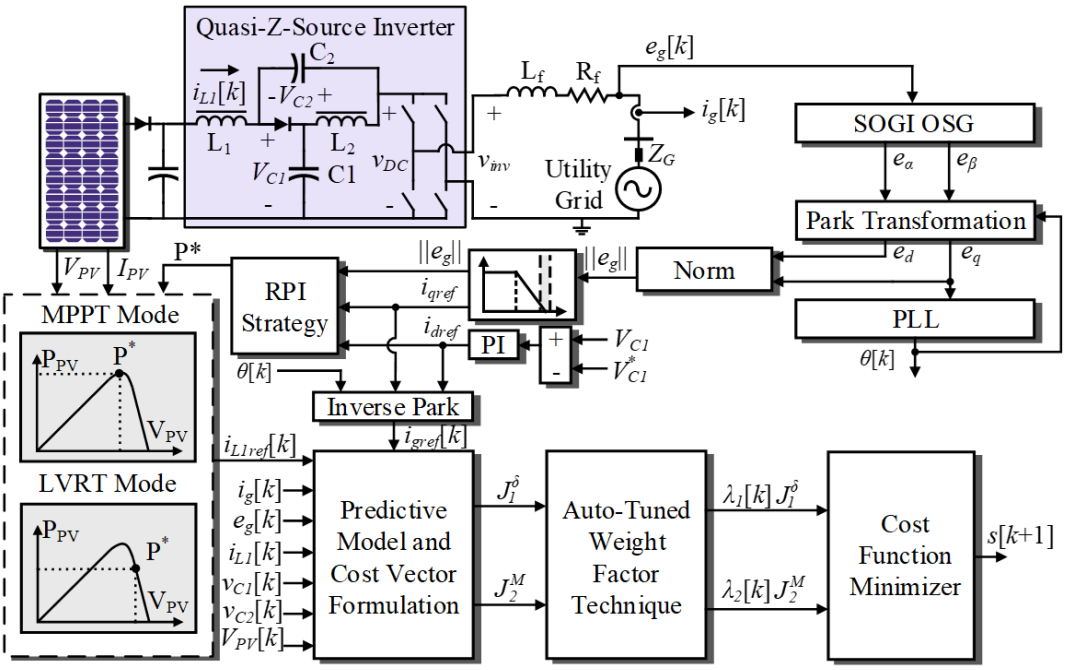


Figure 4.1: Proposed power electronic interface for PV applications with LVRT capability.

The qZSI, shown in Figure 4.1, is analyzed for active states and shoot through state separately to develop the system model which are further required for designing the predictive controller. Considering full switching period to be T, active state time is taken as T_1 and shoot through time is taken as T_0. By applying volt-sec balance on $L1$ and $L2$ and equating the average voltage over a switching cycle to zero, the capacitors $C1$ and $C2$ voltages and dc-link voltage is given by:

$$V_{C1} = \frac{(1-D)\times V_{PV}}{1-2D} \tag{4.1}$$

$$V_{C2} = \frac{D\times V_{PV}}{1-2D} \tag{4.2}$$

$$V_{DC} = \frac{V_{PV}}{1-2D} \tag{4.3}$$

where D is the shoot through duty ratio, defined as T_0/T. Gernerally, qZSI control requires multiple system elements to be controlled simultaneously. For a qZSI, V_{C1}, i_{L1} and i_g are controlled,

however, the proposed algorithm does not include V_{C1} in the cost function; a PI controller is used for the regulation of V_{C1} by appropriately generating the $i_{d,ref}$ (active power component of reference grid current) needs to be injected into the grid by qZSI which is achieved by MPC.

4.2.1 Quasi-Z-Source Inverter Modeling

By using the voltage-current relationship of an inductor and applying the Forward Euler method to the dynamic inductor current equations [72], the discretized model for i_{L1} is determined in (4.4):

$$i_{L1}[k+1] = i_{L1}[k] + T_S(L1)^{-1}\left(V_{PV}[k] - \delta V_{C1}[k] + (1-\delta)V_{C2}[k]\right) \quad \delta \in [0,1] \tag{4.4}$$

where $i_{L1}[k+1]$ is the predicted value of inductor current and δ is non-shoot through indicator as described earlier. Thus, the two-element sub-cost vector is defined as:

$$J_1^{\delta} = \left|i_{L1}[k+1] - i_{L1,ref}[k]\right| \times \frac{1}{i_{L1,RMS}[k]} \tag{4.5}$$

$i_{L1,RMS}[k]$ is used to normalize the cost and is discussed further in the following section.

The KVL equation for the grid side (4.6) is used for the discretization of i_g in (4.7) using Euler forward method:

$$v_{inv} = Ri_g + L\frac{di_g}{dt} + e_g \tag{4.6}$$

$$i_g[k+1] = i_g[k]\left(1 - \frac{R}{L}T_S\right) + \frac{T_S}{L}(v_{inv}[k] - e_g[k]) \tag{4.7}$$

$$v_{inv}[k] = M \times v_{DC}[k] = M \times \left(V_{C1}[k] + V_{C2}[k]\right) \quad M \in [-1,0,1] \tag{4.8}$$

Seeing that the output current prediction is dependent on the output voltage level M, the output current cost vector is defined:

$$J_2^{M} = \left|i_g[k+1] - i_{g,ref}[k]\right| \times \frac{1}{i_{g,RMS}[k]} \tag{4.9}$$

$i_{g,rms}[k]$ is a rolling RMS measurement and its formulation is discussed in Section 4.3.2. Note that the cost vectors are dependent on the next-state predictions, which are dependent either on δ or M. Ultimately, these two variables are coupled, as shoot-through constrains the output voltage level. The true independent variable is switching state s shown in Figure 4.2, but the cost vectors $J^{\delta}{}_1$ and $J^{M}{}_2$ are defined according to δ and M respectively instead of s to reduce the vector lengths, thus reducing iterative computation.

Finally, the sub-cost vectors, with their corresponding weight factors, are summed in the cost function vector:

$$\begin{aligned} s_{opt} &= \arg\underset{s[k+1]}{\text{minimize}}(J[s]) \\ J[s] &= \lambda_1 J_1^{\delta} + \lambda_2 J_2^{M} \end{aligned} \tag{4.10}$$

where s unites M, δ, and the gate signals associated with each switching state. Formulation of the time-variant weight factors λ_1 and λ_2 is discussed in Section 4.3. Although (10) requires a second pass through the sub-cost vectors to compute J, the next-state predictions do not need to be recomputed for cost function optimization. Control in both normal and LVRT modes are briefly explained in Sections 4.2.2 and 4.2.3.

4.2.2 Normal Grid Mode

The system will operate in this mode when the grid voltage magnitude is within ±10% of the rated grid voltage. The MPPT algorithm, shown in Figure 4.3, is used to generate the PV current reference or average input inductor current i_{L1} reference for operation at MPP. The detailed explanation of this algorithm is provided in Section 4.2.4. As illustrated in Figure 4.1, the PI controller will generate the real power component current reference i_{dref} to be injected by the qZSI to the grid. In this mode, reactive power component i_{qref} is kept at zero, making the qZSI work at

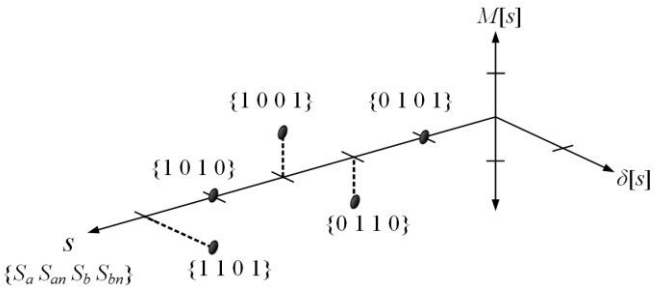


Figure 4.2: Total optimization set. The switching states are defined and their relationship with *M* and *δ* are shown explicitly.

unity power factor. The i_{dref} and i_{qref} are then converted to the phase current i_{gref} and fed to the MPC cost function along with the $i_{L1,ref}$ from the MPPT algorithm.

4.2.3 LVRT Mode

This mode of operation is triggered for grid voltage sags of more than 10%. SOGI based PLL [87] is used to detect the voltage magnitude as well as the phase for the control algorithm as shown in the control schematic of Figure 4.1. The proposed system is tested for three reactive power injection strategies: constant average active power, constant peak current, and constant active current. The equations used to develop the current references for these three strategies are given in Table 4.1. I_N is the nominal current rating of the inverter and the Euclidean distance between the reactive and active current components. Each term in the equations of Table 4.1, except for I_N and P, is in per unit (p.u.). Note that all the RPI strategies respond equally to voltage sags, thus these RPI strategies vary only in how the active current injection is compensated. In these equations, γ represents the ramping rate of reactive power injection. More explicitly, the inverter will inject reactive power associated with the current rating when the grid drops to or below (1-1/γ) p.u.. A γ value of 2 p.u. has been suggested in literature, but these standards vary and can be adopted based on country of operation [85]. Specific implementation of these RPI strategies in cohesion with the MPPT is discussed in the next section.

Table 4.1: Current References in LVRT Mode

	Constant Average Power	Constant Peak Current	Constant Active Current (I_d)
I_d	I_N/v_g	$[1-\gamma^2(1-v_g)]^{1/2}I_N$	$2P/v_{g,rated}$
I_q	$\gamma(1-v_g)I_N$	$\gamma(1-v_g)I_N$	$\gamma(1-v_g)I_N$

4.2.4 MPPT with Flexible Power Point Tracking

The MPT is achieved by creating the $i_{L1,ref}$ which is an input to the J^{δ}_1 calculation. Although the input capacitance distinguishes I_{PV} from i_{L1}, their DC components are equal, thus $i_{L1,ref}$ is a suitable output of the MPPT. The flexible power point on the MPPT algorithm is what allows the LVRT mode and MPPT modes to be distinguished. Although the reactive current references from Table I are applied directly to the formulation of J^{M}_2 using a lookup table, the active current component must be created indirectly, as the active current reference is developed by the PI controller which regulates V_{C1}. Specifically, to reach the active current references outlined in Table 4.1 during grid voltage sags, the power drawn from the PV string must be reduced (except for the constant average active power strategy). With a reduction in power sent across the DC bus, the active current reference will decrease to maintain the voltage across C1. Thus, each reactive power injection strategy holds a unique power reference to reach the active current specified in Table 4.1. This is explained below:

Constant average active power: During voltage sags, the reference power is held constant at P_{MPP}. With a reduction in grid voltage, the active power injected decreases initially. This results in more energy being stored in C1, causing an initial voltage swell at the DC link. To return V_{C1} to the reference value, the PI increases i_{dref}. This, in addition to the i_{qref}, will substantially increase the output current amplitude.

Constant active current: To maintain the active current component during grid voltage sags, V_{C1} should not experience a transient. This can be achieved by reducing the harvested PV power such that it is commensurate with the grid voltage sag. Thus, the power reference is defined as:

$$P^*[k] = P_{MPP} \times e_{g,PU}[k] \tag{4.11}$$

where the per-unit grid voltage $e_{g,PU}[k]$ is:

$$e_{g,PU}[k] = \frac{\sqrt{e_d[k]^2 + e_q[k]^2}}{120\sqrt{2}} \tag{4.12}$$

$e_d[k]$ and $e_q[k]$ are the decoupled components of the grid voltage. Although this power reduction ideally does not create a transient in the active current component, the non-zero response time of the MPPT to the grid voltage sag will create a brief swell on the DC-link voltage and a corresponding i_{dref} swell. The settling time of this transient at the DC-link is dependent on the MPC sampling time and current step-size of the MPPT algorithm.

Constant peak current: To maintain the injected peak current that occurred prior to the grid voltage sag, the harvested PV power must decrease in concordance both with the grid voltage sag and the iqref. For this strategy, P* is constructed as:

$$P^*[k] = \frac{120\sqrt{2}e_{g,PU}[k]}{2}\sqrt{\left(\sqrt{2}i_{g,rated}\right)^2 - i_{q,ref}^2[k]} \tag{4.13}$$

Similar to the constant active current strategy, a voltage sag will initially create a swell on the DC link voltage and i_{dref}. However, the reduction in PV harvesting will be substantial enough such that C1 starts drawing out of its energy reserve, creating a brief sag in DC link voltage. Thus, the PI will reduce i_{dref} below its value before the swell to compensate.

A reference power term P^* is not used in traditional MPPT algorithms, as the reference power is the maximum extractable power. To adhere to the RPI strategies, a conditional statement is applied prior to the traditional MPPT, shown in Figure 4.3. The extracted PV power is compared

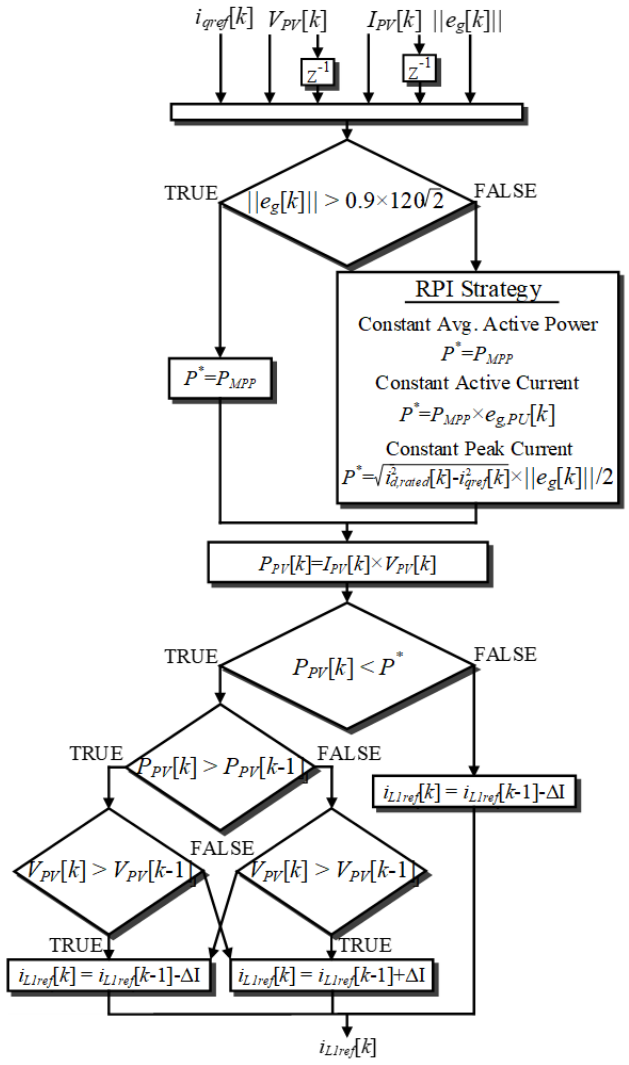


Figure 4.3: RPI strategies and MPPT algorithm with flexible power point.

with P^*. If the extracted PV power exceeds P^*, the power point tracker immediately reduces the $i_{L1,ref}$. Reducing $i_{L1,ref}$ will pull the PV string to the right of the power-voltage characteristic curve. For a current-controlled MPPT, this side of the MPP is more stable, as the $|dP_{PV}/dI_{PV}|$ is lower.

4.3 Auto-Tuning of Weight Factors and Online Normalization

4.3.1 Auto-Tuning Algorithm

When considering multiple parameters in MPC, designing the cost function weight factors is a laborious task. These parameters, often with different units and orders of magnitude, are

conglomerated into a single cost function subject to minimization, and finding a systematic approach for optimizing these weight factors is still a topic of discussion in literature [8, 16]. This section proposes an auto-tuning algorithm for the weight factor in the MPC cost function (4.10). The method of weight factor auto-tuning is based on an intuitive concept: poorly-tracked objectives should have increased weight in the cost function. This algorithm evaluates the tracking of each control objective, and makes the weight commensurate with the tracking error. The algorithm starts by collecting the minimized cost for each objective. Note this is distinct from the traditional optimization in MPC, as finite-set MPC typically only considers the overall cost function, or the sum of the cost terms. The two optimal sub-costs are defined as:

$$\zeta_1[k] = \min(J_1^{\delta}) \tag{4.14}$$

$$\zeta_2[k] = \min(J_2^{M}) \tag{4.15}$$

This auto-tuning procedure is first explained as a general procedure, then the implementation is described.

The auto-tuning weight factor procedure is explained for only one objective without loss of generality. Consider a threshold of acceptable objective error ε. In other words, if it is possible for the qZSI to keep a normalized control objective less than ε from its reference, then the objective is considered sufficiently tracked and no remedial action is necessary. This condition can be written concisely as:

$$\zeta_1[k] \le \varepsilon \Rightarrow \lambda_1[k] = \nu \tag{4.16}$$

where ν is an arbitrary value attached to both weight factors. Note that the actual value of ν is not important, as only the weight factor ratio will affect system performance. If the condition in (4.16) is not satisfied, then a larger value of the weight factor should be selected to place more emphasis

on J^{δ}_1, promoting its reduction for the next sampling time. The evaluation of λ_1 when $\zeta_1[k]$ exceeds ε is as follows:

$$\begin{aligned} \zeta_1[k] \le 2\varepsilon &\Rightarrow \lambda_1 = 2\nu \\ \zeta_1[k] \le 3\varepsilon &\Rightarrow \lambda_1 = 3\nu \\ &\vdots \\ \zeta_1[k] \le n_1\varepsilon &\Rightarrow \lambda_1 = n_1\nu \qquad n_1 \in \mathbb{N} \end{aligned} \tag{4.17}$$

Thus, each objective is weighted in J according to the controller's tracking performance.

Applying a conditional loop of indefinite size as is shown in (4.17) can be problematic for a controller operating at a fixed sampling rate. However, this concept can be applied to the controller in a definite time by recalling that the term v is arbitrary. By setting v equal to one, λ_1 can be computed directly:

$$\begin{aligned} (n_1 - 1)\varepsilon \le \zeta_1[k] \le n_1\varepsilon &\Rightarrow \lambda_1[k] = n_1 \\ roundup\left(\frac{\zeta_1[k]}{\varepsilon}\right) = n_1 = \lambda_1[k] & \\ \lambda_1[k] = roundup\left(\frac{\zeta_1[k]}{\varepsilon}\right) & \end{aligned} \tag{4.18}$$

where *roundup* refers to a function which rounds the inner argument up to the nearest integer. Thus, the indefinite loop has been removed while maintaining the original auto-tuning concept.

Although an adaptive and performance-based weight factor is desirable, next-state predictions can have substantial instantaneous error for lower sampling rates. Considering a moving window of tracking performance can improve stability by reducing the effect of high frequency error attributed to discrete predictions. The weight factors are computed using a moving average of the value computed in (4.19). The weight factors are stored in a vector of length N. Thus ζ[k] to $\zeta[k\text{-}N\text{+}1]$ contribute to the associated objective's weight factor $\lambda[k]$. Rather than placing the elements at each end from newest to oldest, a pointer variable cycles through the vector,

replacing the variable it is pointing to with ζ[k]. As long as the pointer follows a cycling pattern, ζ[k] will always replace ζ[*k*-*N*+1], regardless of the pointer's location in the vector. The moving average can be updated efficiently with the following equation:

$$\lambda_x[k] = \frac{1}{N}\sum_{i=0}^{N-1}\zeta_x[k-i] = \lambda_x[k-1] + \frac{\zeta_x[k]}{N} - \frac{\zeta_x[k-N+1]}{N} \qquad x \in [1,2] \tag{4.19}$$

Thus, (4.19) is used to update the weight factor for both objectives with the number of computations independent of the window size.

4.3.2 Adaptive Normalization of Control Objectives

Rather than hard-programming an expected operating point of the control objectives or dividing by the rated values, the objectives are normalized using previous measurement data. Specifically, the root-mean-square (RMS) is computed over a moving window. The RMS is computed efficiently by first updating the mean squared sum of each objective:

$$i_{x,MSS}[k] = \frac{f_g}{f_s}\sum_{i=0}^{\frac{f_s}{f_g}-1}\left|i_x[k-i]\right|^2 = i_{x,MSS}[k-1] + \frac{f_g\left|i_x[k]\right|^2}{f_s} - \frac{f_g\left|i_x[k-\frac{f_s}{f_g}+1]\right|^2}{f_s} \qquad x \in [L1, g] \tag{4.20}$$

where it is assumed the discrete sampling frequency f_s is an integer multiple of the grid frequency f_g. If this is not the case, then the term f_s/f_g can be rounded up to the nearest integer, with f_g/f_s replaced with the reciprocal of the rounded number. It should also be noted that f_s/f_g can be replaced with $\alpha f_s/f_g$, $\alpha \in \mathbb{N}$, while still considering practical constraints such as the controller's available RAM. Once the mean squared sum is updated, the root mean square is calculated:

$$i_{x,RMS}[k] = \sqrt{i_{MSS}[k]} \quad x \in [L1, g] \tag{4.21}$$

Figure 4.4 summarizes one iteration of the predictive control scheme.

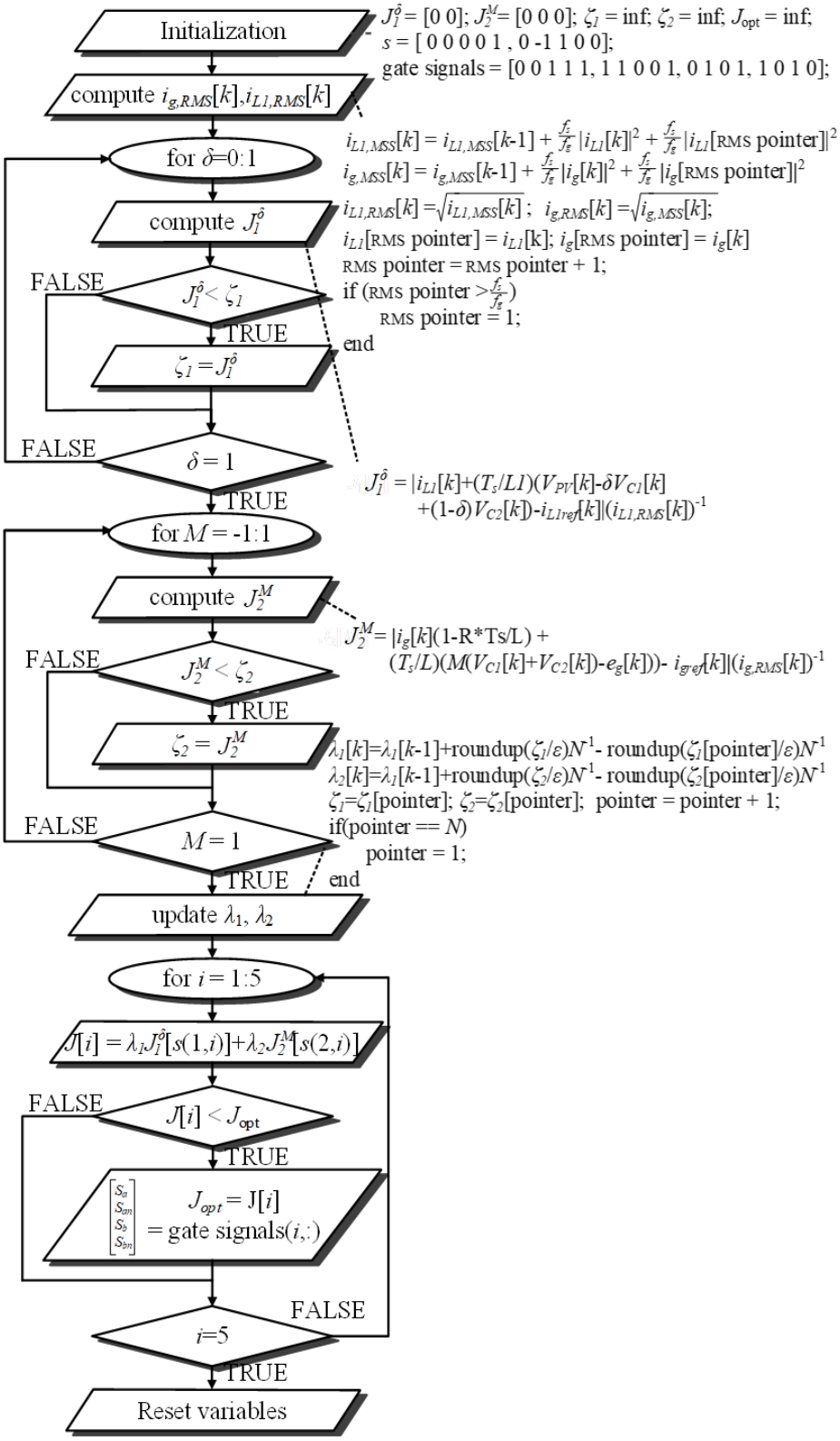


Figure 4.4: Flowchart of the predictive model, auto-tuning technique, and cost-vector minimization.

4.4 Results and Discussion

The single-phase smart inverter using the qZSI topology is tested experimentally with the proposed control scheme, shown in Figure 4.5. Table 4.2 includes several specifications of the prototype.

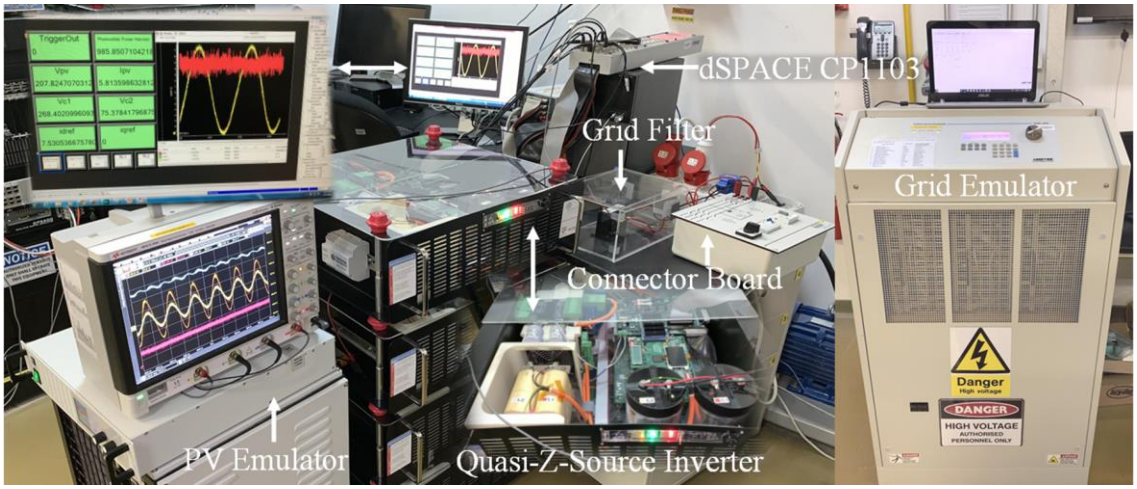


Figure 4.5: Hardware prototype of single-phase grid-connected qZSI.

Table 4.2: System Specifications

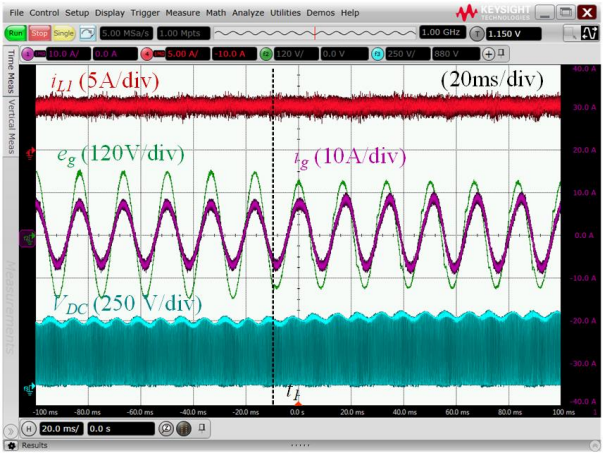
Parameter	Value
Grid Frequency	60Hz
MPC Sampling Frequency	50kHz
MPPT Sampling Frequency	6.25kHz
PI controller Sampling Frequency	3.125kHz
PI controller proportional/integral gains	0.008/0.08
RMS window size	834
weight factor window size N	100
ε	1e-4
V_{MPP}/V_{OC}	200V/239V
I_{MPP}/I_{SC}	5A/5.4A
L_1/L_2	0.4µH
C_1/C_2	1.5mF
C_{PV}	200µF
L_f	2.5mH
R_f	200 mΩ
Ramping rate γ	4 p.u.
$i_{g,rated}$	$8.33A_{RMS}$
Grid Voltage	120 V_{RMS}
V^*_{C1}	275V

The qZS is implemented with GB100XCP12-227 Silicon Carbide IGBTs. The control algorithm is implemented on the dSPACE CP1103, with a sampling time of 20μs. The gate signals from the CP1103 are applied directly to the qZSI gate driver from the digital I/O port, meaning no dead-time is implemented. This is generally risky, yet the impedance network prevents undesired current spikes during unintentional shoot through. The output of the qZSI is connected to a bidirectional grid emulator allowing grid voltage sags to be tested. The power supply used to emulate a 1kW PV string is the EA-PSI 9000 2U. The ramping rate γ is set to a large value of 4 p.u. to make the RPI easier to visualize on the oscilloscope.

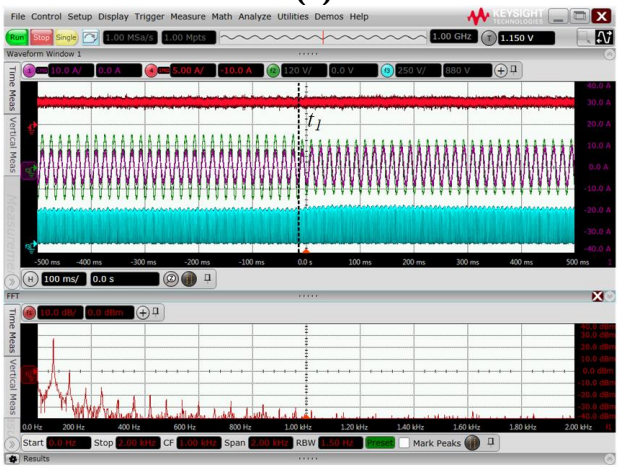
The single-phase qZSI is interfaced with single-phase $120V_{RMS}$, 60Hz grid to verify the performance of the proposed system experimentally. The prototype is tested for the constant average active power strategy, constant active current, and constant peak current strategy. Then, the system is shown as the grid voltage recovers from a voltage sag using the constant average active power strategy.

4.4.1 LVRT Model in Constant Average Active Power Strategy

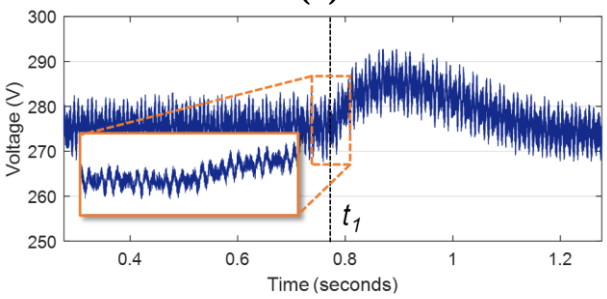
Figure 4.6 shows the system dynamics during a grid voltage sag, where the RMS grid voltage decreases from 120V to 100V at t_1. i_{L1} is shown to visualize changes in PV power harvesting. Using the constant average active power strategy, P^* is fixed at 1000W. Thus, no transient in i_{L1} is seen after t_1. The reactive power injection is easily identifiable with the grid current lagging the grid voltage. A sudden increase in grid current amplitude is noted, caused by the step-change in i_{qref}. With the reduction in grid voltage and unchanged active current amplitude, the power injection briefly drops. This initial reduction in power injection creates a swell at the DC link with the excessive energy stored on $C1$. Thus the PI controller begins increasing i_{dref}. The full DC link voltage is shown over an extended interval in Figure 4.6b. As i_{dref} increases, the DC



(a)



(b)



(c)

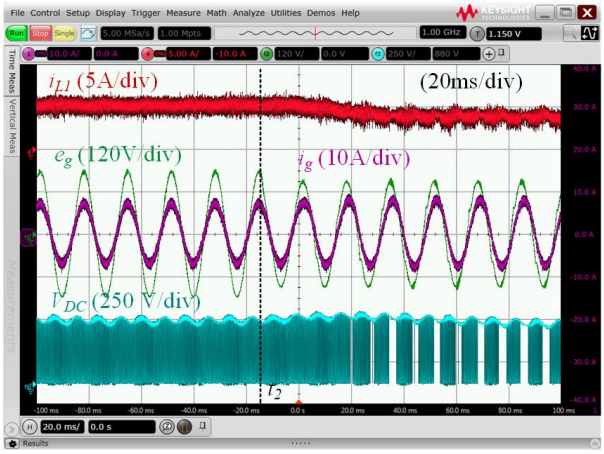
Figure 4.6: Voltage sag using constant average active power strategy: (a) grid voltage and current, inductor L1 current, and dc-link voltage dynamic response, (b) extended view of dc-link voltage dynamics and FFT plot of injected grid current, (c) V_{C1} dynamic response.

link voltage reaches steady-state. This settling time is dependent on the gains of the PI controller. The primary trade-off in designing the gains of the PI is settling time of the DC link voltage (and i_{dref}) during grid voltage transients and sub-harmonic content of the injected grid current. In other words, the settling time of the DC link voltage could be reduced by increasing the sampling

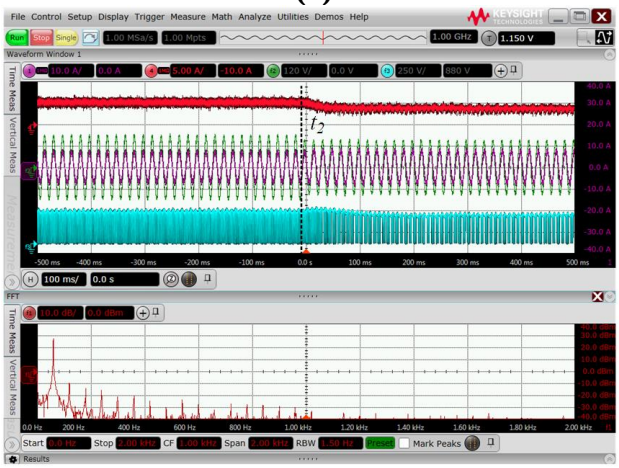
frequency of the PI or increasing the integral gain. However, increasing the sensitivity of the PI controller would create substantial steady-state oscillation on i_{dref} and increase sub-harmonic distortion of the injected current. These oscillations are a result of voltage ripple on C1. An FFT of the injected grid current is also shown in Figure 4.6b, using one second of measurements centered at t_1. Although the most substantial frequency content exists at harmonics of the grid frequency, the frequency content of the injected grid current is continuous. This is caused by the variable switching frequency inherent to finite-set MPC.

4.4.2 LVRT Mode in Constant Active Current Strategy

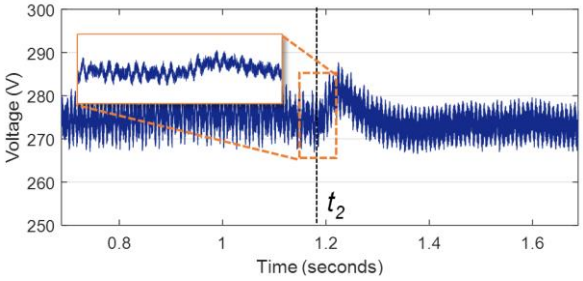
The grid voltage transient occurs using the constant active current strategy at t_2 in Figure 4.7. Commensurate with the grid voltage sag, the power reference decreases to about 833W. This can be seen by the reduction in i_{L1} in Figure 4.7a. The brief swell in DC link voltage still occurs at t_2, despite a reduction in power reference. This is expected, as the flexible power point tracking algorithm requires several milliseconds to reach steady-state. In this time, the difference in power harvesting and power injection is absorbed by C1, as discussed earlier. However, the eventual reduction in harvested power in addition to the initial increase in i_{dref} causes a brief sag in DC link voltage before settling. This is shown in Figure 4.7b. By making the reduction in harvested power commensurate with the reduction in grid voltage, the active current reference is unchanged at steady-state. However, the reactive current reference will still create an increase in grid current amplitude. Note also there are distinct intervals in which shoot-through is not implemented, evidenced by V_{DC} not reaching zero volts. The MPC is driving down the input current by avoiding the shoot-through state. This is intuitive when considering the dynamic equation for v_{L1}. By placing the steady-state values of V_{PV} and V_{C1} into these equations, it is evident that shoot-through creates positive di_{L1}/dt, while the opposite is true for non-shoot through states. Thus, the shoot-through



(a)



(b)



(c)

Figure 4.7: Voltage sag using constant active current strategy: (a) grid voltage and current, inductor L1 current, and DC-link voltage dynamic response (b) extended view of DC-link voltage dynamics and FFT plot of injected grid current (c) V_{C1} dynamic response.

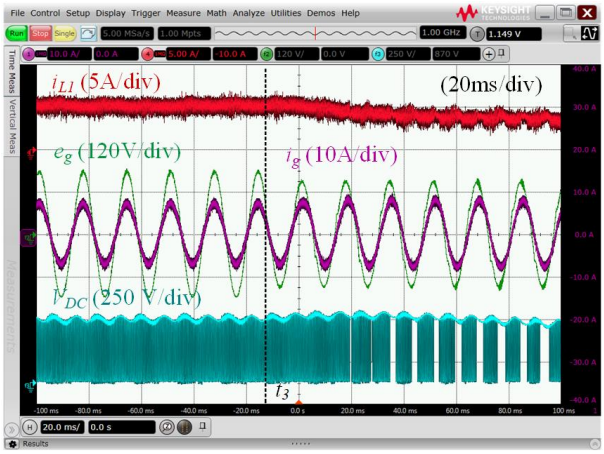
duty ratio is expected to decrease for reductions in harvested PV power. Figure 4.7b also shows the FFT of the injected grid current amplitude. Although the oscilloscope does not provide a total harmonic distortion measurement to compare against the constant average active power strategy, a reduction in the subharmonic content of the injected current is observed, caused by a less

substantial increase in grid current amplitude at t_2. This can be judged by observing the left-most side of the FFT plots prior to the 60Hz peak.

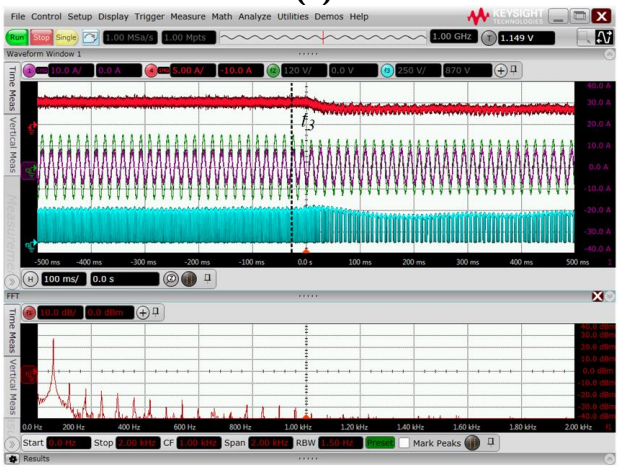
4.4.3 LVRT Mode in Constant Peak Current Strategy

The grid voltage sag is induced at t_3 using the constant peak current strategy, shown in Figure 4.8. i_{qref} is set to 3.5A for the presented grid voltage sag and ramping rate. Based on (4.13), the power reference is reduced to about 795W. With the power reference close to that of the constant active current strategy, it is difficult to distinguish these two strategies. We expect a slightly more dramatic decrease in i_{L1}, in addition to a slightly larger grid voltage sag as the harvested power drops. Note that in Figure 4.8b, the DC link voltage appears to settle at a peak voltage slightly below what it was before t_3. This is expected; reducing i_{L1} to reduce the harvested power creates a reduction in the shoot-through duty ratio D, as discussed previously. Additionally, a reduction in shoot-through duty ratio causes a reduction in the DC link's peak voltage, as evidenced by (4.3). The DC link voltage and V_{C1} have equal DC components. The steady-state reduction in D is expected to reduce the DC link voltage peak to maintain equality between the average DC link voltage and V_{C1}. Additionally, V_{C2} decreases for reductions in D, as shown in (4.2). Thus, the noted changes at the DC link voltage are required to regulate V_{C1} during the transient in harvested power.

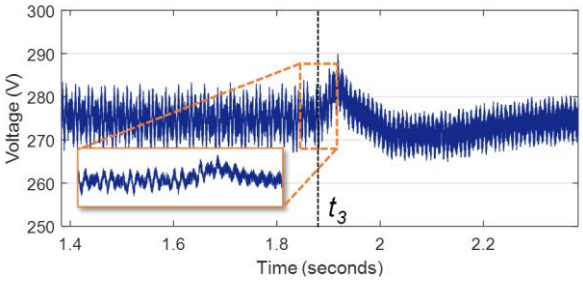
Despite the RPI strategy being labelled as *constant peak current*, there is a notable increase in grid current amplitude after t_3. Although the power reference is reduced to maintain the peak current at steady state, this initial amplitude spike results from the differences in how the active and reactive current references are generated. i_{qref} is generated automatically, based on the severity of the grid voltage sag, whereas the i_{dref} relies on the cascaded settling times of the MPPT algorithm and the PI controller. This difference in time causes the initial spike in grid current amplitude. At



(a)



(b)



(c)

Figure 4.8: Voltage sag using constant peak current strategy: (a) grid voltage and current, inductor L1 current, and dc-link voltage dynamic response, (b) extended view of dc-link voltage and FFT plot of injected grid current, (c) V_{C1} dynamic response.

steady-state the current amplitude is unchanged. Although the PI controller and MPPT algorithm can be altered to reduce this time, a non-zero interval of amplitude spike is inevitable, regardless of P^*. Figure 4.8c shows the FFT of the grid current during this transient. The most notable

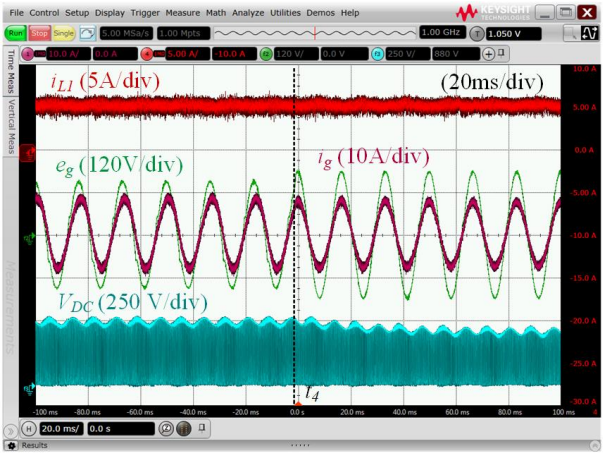
difference in this FFT plot compared to those of the other two RPI strategies is the reduced low-frequency content, as a result of the less severe transient in grid current amplitude.

4.4.4 LVRT Mode to Normal Grid Condition

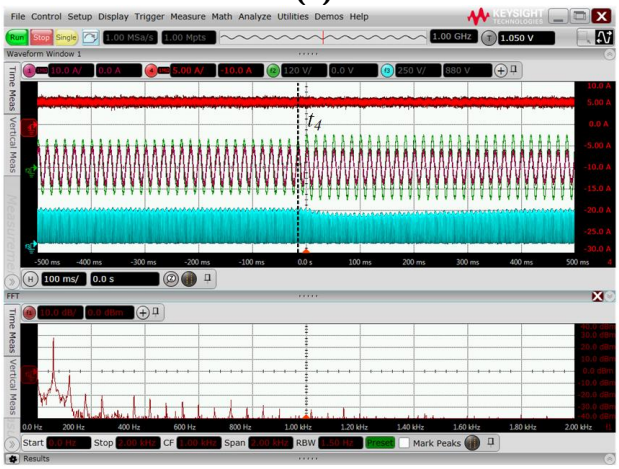
In Figure 4.9a, the qZSI dynamics are shown for a recovery in grid voltage using the constant average active power strategy. The grid voltage increases from 100VRMS to 120VRMS. The grid current immediately aligns with the grid voltage, as i_{qref} drops to zero. After t_4, Figure 4.9b shows a brief sag in the DC link voltage. i_{dref} was raised during the grid voltage sag. As the grid voltage returned to normal condition, the increased active current injection pulled additional energy from $C1$, resulting in the sag. The DC link voltage returns to normal as the PI controller reaches V^*_{C1}.

4.4.5 Solar Irradiance Transient

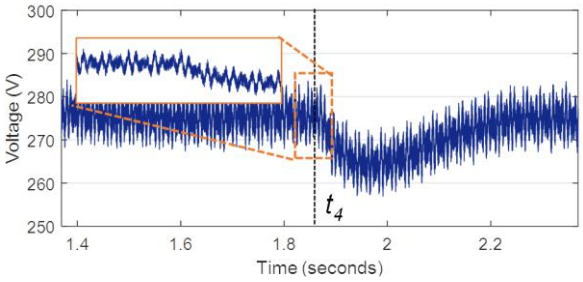
The qZSI response to a solar irradiance transient from 1000W/m^2 to 400W/m^2 at t_5 is shown in Figure 4.10. A substantial dip in V_{PV} occurs after t_5, shown in Figure 4.10a. This dip occurs from the rapid change in the I-V characteristic curve. Specifically, while the MPPT is adjusting its input current reference, the PV array is operating at its short-circuit current. Thus, this dip is proportional to the settling time of the MPPT algorithm. After dropping around 100V, it settles slightly below 200V at the new MPP. In Figure 4.10b, we see the injected current drop substantially following t_5, dropping close to zero before settling around a 2.5 peak amplitude. This can be explained when seeing the dip in V_{C1} in Figure 4.10c, as the energy injected to the AC side is not matching that which is being pulled from the PV array, thus energy is rapidly drawn from $C1$. The PI control responds by reducing the active current reference near to zero, until power can once again be harvested from the DC side. Once i_{L1} settles at the new MPP, power harvesting from the PV array continues, and V_{C1} begins to rise accordingly. In Figure 4.10d, the weight factors λ_1 and λ_2, which



(a)



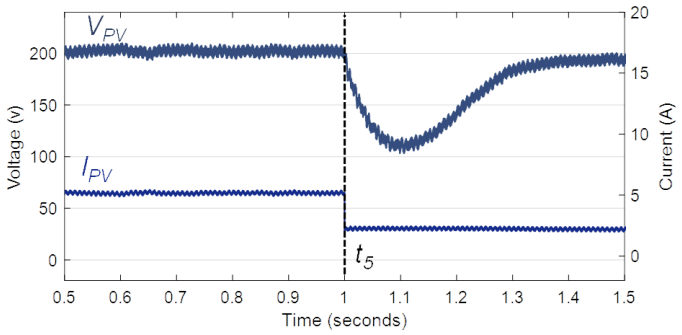
(b)



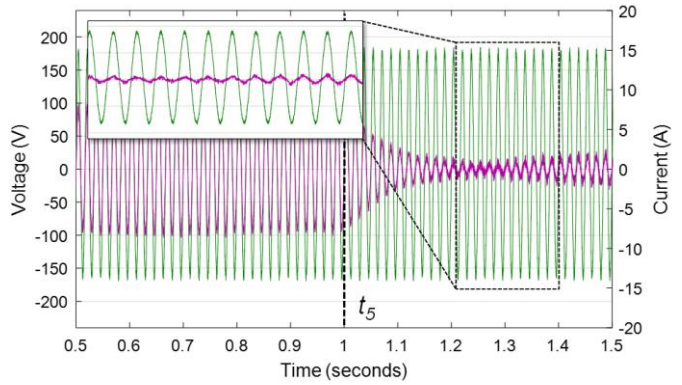
(c)

Figure 4.9: Voltage recovery using constant average active power strategy: (a) grid voltage and current, inductor L1 current, and dc-link voltage dynamic response, (b) extended view of DC-link voltage and FFT plot of injected grid current, (c) V_{C1} dynamic response

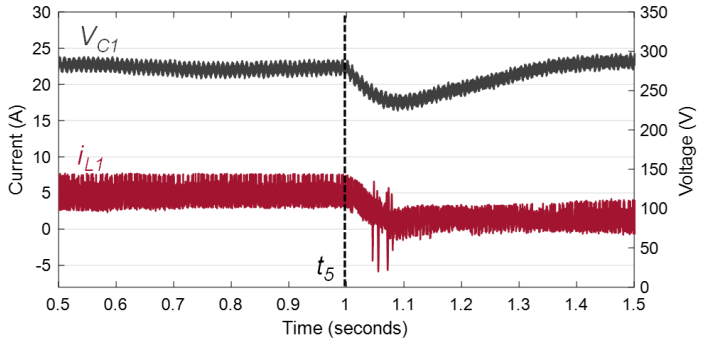
respectively weight the input and output current objectives, have settled at a ratio near 2:3, until t_5. Once the solar irradiance transient occurs, the input current reference quickly drops from the MPPT, but is not immediately tracked, causing large spikes in λ_1. This causes i_{L1} to dominate the cost function J, allowing it to hold precedence for this transient. As the grid current decreases in



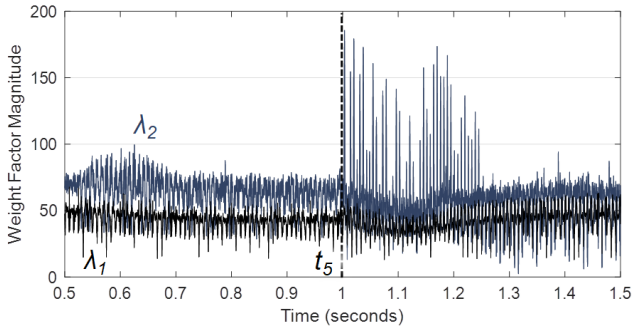
(a)



(b)



(c)



(d)

Figure 4.10: Solar irradiance transient from 1000W/m^2 to 400W/m^2: (a) PV voltage and current, (b) grid voltage and current, (c) V_{C1} and i_{L1} inductor $L1$ current, and dc-link voltage dynamic response, (d) weight factor adaptation.

magnitude, its normalized factor, determined by the rolling RMS from (4.20), is not immediately adjusted, causing the normalized tracking error to decrease. The weight factor λ_2 decreases accordingly, and both weight factors reach steady-state as their normalizing factors are adjusted according to the new magnitudes of i_{L1} and i_g.

4.4.6 Comparison with Static Weight Factors

Finally, the tracking performance of the auto-tuned weight factors is compared with the static weight factors implemented in [86]. Observing metrics such as transient performance (i.e. settling time during transient conditions) of the two control schemes does not directly reveal the performance of the MPC itself, only of the entire system, as transient performance is highly dependent on the reference values, which are merely inputs to the predictive control. Rather, the exact tracking performance is compared for both systems. To conduct this comparison, a term referred to as *tracking error* is defined as:

$$x_\varepsilon[k] = \left| x^*[k-1] - x[k] \right| \quad x \in [i_{L1}, i_g] \tag{4.22}$$

Objective tracking error is decidedly the most apt metric, as this value demonstrates how close the controlled variables reached to their references in the next sampling period. To compare tracking errors of each control objective, the difference in tracking errors is shown for the two control schemes. For clarity, the measurements in Figure 4.11 are defined as:

$$x_\Delta[k] = x_{\varepsilon,auto}[k] - x_{\varepsilon,static}[k] \quad x \in [i_{L1}, i_g] \tag{4.23}$$

where $x_{\varepsilon,auto}$ and $x_{\varepsilon,static}$ are the tracking errors for the proposed control and the control of the static weight factor ratio implemented in [86]. Thus, a negative x_Δ would imply better tracking by the proposed, autonomously-tuned control. In Figure 4.11, x_Δ is shown under normal condition at full solar irradiance, and for a step-change in the flexible power point tracking reference from 1kW to 0.5kW which occurs at t_6. While the magnitude is larger for $i_{L1,\Delta}$ than $i_{g,\Delta}$, both are negative on

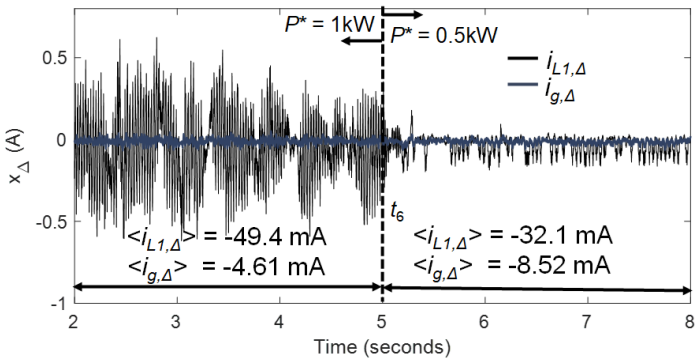


Figure 4.11: Tracking error comparison of the proposed auto-tuned weight factors and the static weight factor ratio from [86].

average. The results reveal better overall tracking for both objectives using the proposed control. After t_6, the difference decreases slightly for both values. This is due to a decrease in magnitude of both control objectives with the reduction in power harvesting from the PV array. Precisely, the average tracking error of the input current was reduced by nearly 10%, while the average output current tracking error was reduced by about 4% which shows significant improvement in tracking performance. The weight factor ratios for both control schemes are shown in Figure 4.12. Like the autonomous weight factor tuning feature itself, the reason for better tracking is intuitive. The proposed control weights each objective according to normalized anticipated tracking error. Thus, an objective which is poorly tracked is quickly compensated via greater weight in the cost function. It is also worthwhile to mention that reaching to the weight factor in [86] involved substantial testing, whereas such a design stage is not required for the proposed control.

4.5 Conclusion

A smart inverter is actualized using the proposed AMPC scheme, and is shown to seamlessly transition between LVRT and MPPT modes according to the grid condition. The controller is able to retain stability during large transients in solar irradiance. Finally, when comparing to a carefully designed, yet static, weight factor ratio in [86], the auto-tuned weight

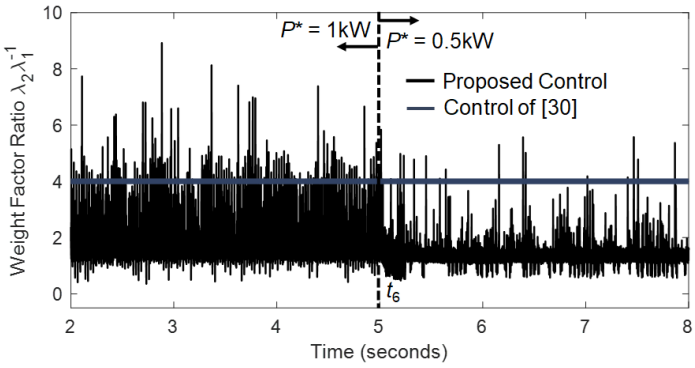


Figure 4.12: Ratio of the proposed auto-tuned weight factors and the static weight factor ratio from [86].

factors exhibited consistently better tracking performance. Additionally, this auto-tuned weight factor in the cost function eliminates the trial-and-error design stage in conventional finite-set MPC. Experiments are provided to verify the system performance for three reactive power injection strategies over LVRT mode, while the reactive power injected adheres to recent grid codes such as the E. ON standard for the chosen RPI strategy. There remain areas for further discussion and analyses as future work. First, practical efficiency of the power stage creates a substantial disparity between the PV power reference and the actual power injected to the grid. The power reference of the flexible power point tracking algorithm should be chosen to account for inefficiencies in the power conversion stage. Second, a brief increase in current amplitude was seen for the constant peak current RPI strategy. This was caused by the non-zero time to reduce the harvested PV power. The reactive current reference should be adaptive to align with the DC-side dynamics to avoid initial current spikes, rather than using a simple lookup table.

Chapter 5 - Computationally-efficient Distributed Predictive Controller for Cascaded Multilevel Impedance Source Inverter with LVRT Capability[4]

5.1 Problem Statement

Voltage Source inverters (VSI) are widely used in PV grid-connected inverter applications. They can only be operated in buck mode and thus generally require multi-stage solution to interface low voltage PV strings. The first stage performs DC/DC boost conversion and MPPT of PV, and the second stage performs DC/AC conversion using a VSI [66, 67, 88-91]. The use of CMI topology is practical for PV energy harvesting systems, as the MPPT can be modularized for each VSI cell thus requiring less voltage across each VSI [92]. This would also able to address the mismatch in PV module voltage-current characteristics, caused by either manufacturing differences or inconsistencies in solar irradiance. In this chapter, both issues will be generally referred to as PV mismatch. Additionally, the diminished harmonic content of the CMI output voltage can reduce output filter requirements to adhere to grid THD standards [93, 94]. However, the double stage configuration of the CMI cells decreases the overall system efficiency, incurs low transient response, etc. [95-98]. Removing the DC/DC boost conversion stage and coupling the PVs to their DC bus would mitigate this, but will lead to unbalanced DC link voltage under PV mismatch scenarios [99] and diminish the MPP operating capability.

[4] The content of this chapter has been taken with permission from a journal publication titled "Computationally Efficient Distributed Predictive Controller for Cascaded Multilevel Impedance Source Inverter with LVRT Capability" in *IEEE Access* by M. Easley, S. Jain, M. B. Shadmand, and H. Abu-Rub, 2019.

The voltage-fed ZSI [72] has some benefits over the traditional VSI/CSI and can achieve both buck and boost operations and thus can directly couple the PV with the grid in a single-stage manner. The ZSI incorporates an impedance network between the DC source and the inverter bridge. This allows for an additional switching state, often called the shoot-through state, enabling buck/boost capabilities. The qZSI is an improvement over the ZSI, as it prevents discontinuous current at the PV side, increasing the lifespan of the PVs [71]. The qZSI can replace the traditional full-bridge inverter cells of the CMI, referred to as the quasi-Z-source cascaded multilevel inverter (qZS-CMI). However, incorporating an appropriate shoot-through duty ratio requires implementation of a complex switching modulator. Additionally, distributed generation (DG) should provide ancillary services to achieve a resilient utility grid, this feature requires a complex multi-loop control scheme when using classical control approaches particularly for qZS-CMI [100, 101].

Finite-set MPC is a potential candidate to address the challenges in the control of grid-tied qZS-CMI with advanced functionalities such as ancillary grid services. Finite-set MPC eliminates the need for a switching modulator. In addition, multi-objective control schemes can be implemented in a straightforward manner. However, traditional finite-set MPC optimizes its control objectives via an overall cost function, in which all predicted errors on control objectives are evaluated and summed for each achievable switching state. This requires tuning the weight factors of each control objective. There is no standard method for this procedure, and suggested procedures such as branch-and-bound [38] are particularly arduous for MPC schemes with more than two objectives. Traditional finite-set MPC is also computationally burdensome, especially for cascaded multilevel inverter topologies and other topologies with large sets of switching states, and thus may not achieve feasible sampling frequencies.

The analysis of qZS-CMI topologies is investigated in [73, 102-104]. In [105], MPC for a grid-tied qZS multilevel inverter is proposed; however non-linearities and fluctuations of PV sources are not considered in the control scheme. Further, it does not investigate the ability to provide ancillary grid services, which will be necessary for a high PV-penetrated grid. The power electronics interface (PEI) for grid-tied renewable energy sources (RES) should be able to support the grid resiliency in addition to the extraction and transfer of power from RES to grid. E.ON-Netz grid code [63] mandates the low voltage ride through (LVRT) capability of PEI, which requires the PEI to retain grid connection and inject reactive power during grid voltage sag for a pre-defined amount of time. The E.ON suggests that PEI must inject entirely reactive power if grid voltage drops below 50% of the rated value [63].

Although MPC for the qZS-CMI topology has not been studied in literature to our knowledge, MPC techniques have been discussed for the CMI [45, 106, 107]. In [45] , the authors discuss MPC of a three-phase CMI with an RL load. The authors note the problem of increased calculations with multilevel topologies and mitigate this for their application by removing redundant voltage vectors with high common-mode voltage. Only current control is implemented for this MPC scheme, and redundant switching states are determined according to the output voltage vector. However, when MPC is used for both AC and DC-side control in a grid-connected application, this method of switching state elimination is not applicable.

This chapter proposes a computationally efficient decoupled active and reactive power control scheme via MPC framework for a single phase qZS-CMI with 5-level output voltage. The decoupled power control scheme is advantageous to regulate the required reactive power and adjust active power injection independently during LVRT grid fault conditions. The novel control structure eliminates the overall cost function and applies a hierarchical objective structure that has

been optimized offline to remove superfluous cost computations for practical realization of MPC for multilevel inverters. The remainder of this chapter is organized as follows: Section 5.2 explains the grid-tied qZS-CMI as a primer to the control system. Section 5.3 describes the predictive model and reference signal generation, which are the inputs to the controller. In Section 5.4, the novel and highly efficient predictive controller is detailed. In Section 5.5, the controller performance is verified through several case studies. Finally, a summary of the findings are provided with the conclusion in Section 5.6.

5.2 Proposed System Description

Figure 5.1 shows the qZS-CMI along with its predictive decoupled active and reactive power control scheme. Five-level qZS-CMI is interfaced with single-phase 120VRMS, 60Hz grid. The controller works in two modes of operation: i) normal grid mode when grid voltage sag is not more than 10% of its rating, in this mode the MPPT technique is triggered to harvest the global maximum power from the PV cells; and ii) the LVRT mode for grid voltage sags of more than 10% threshold. In this mode, the qZS-CMI will move from MPP operation to support grid based on desired RPI strategies [85]. These modes are the same as what is discussed in the previous chapter from section 4.2.2 to 4.2.4, and thus are not detailed here. However, it should be noted that each PV string operates according to its own flexible power point tracking algorithm; at the output of each flexible power point tracker is an input current reference. Thus, in this topology, there are two input current references, $i_{L1,1}$ and $i_{L1,2}$, that are tracked using the efficient predictive control.

5.3 Predictive model and reference generation

A distributed predictive control strategy is proposed for the qZS-CMI; the i_{L1} control is modularized for each PV string, while the injected grid-side current is accumulated from each qZSI cell. In general, a finite-set model predictive controller has three components: the predictive

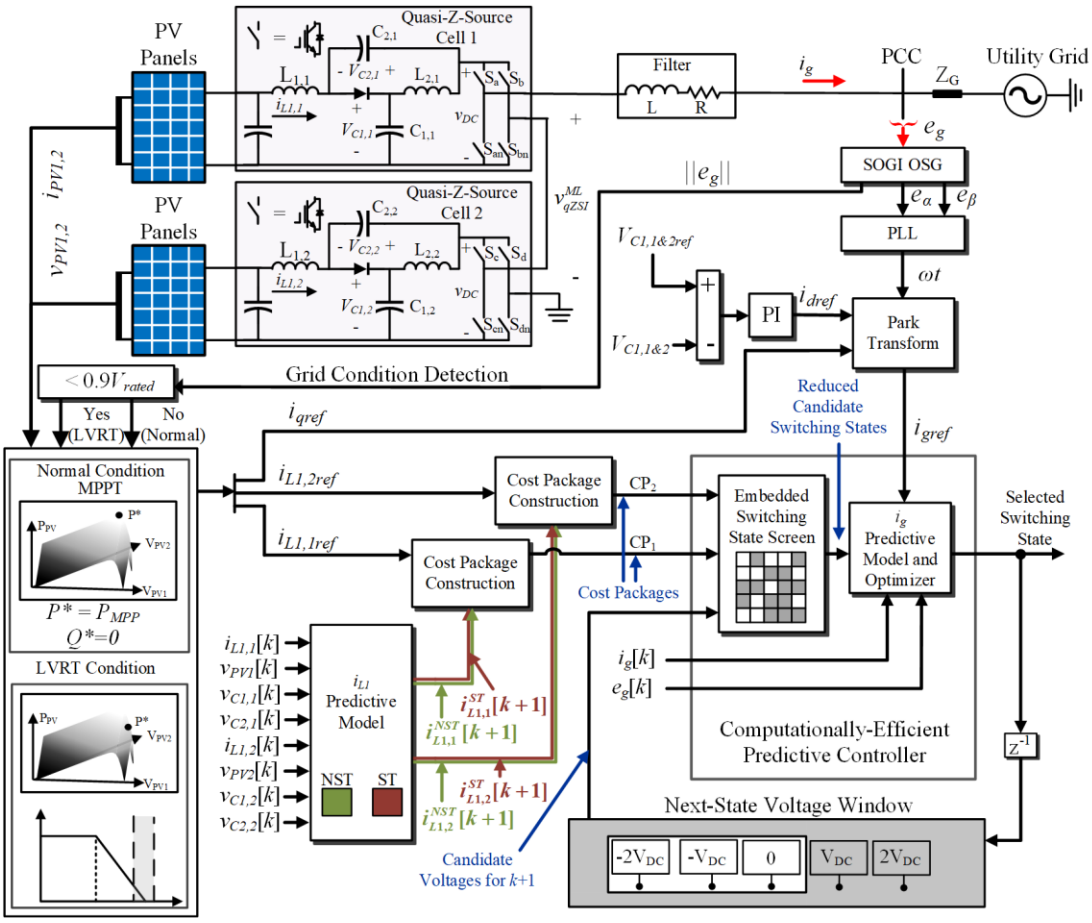


Figure 5.1: Proposed quasi-Z-source cascaded multilevel inverter and predictive control with LVRT capability for PV applications.

model, reference signal generation, and cost function optimization. The proposed controller does not implement an overall cost function, thus the first two components are discussed in this section, and the optimization is discussed in the following section.

5.3.1 Predictive Model

Each qZSI cell is analyzed for both of its active and shoot-through states of operation which is required to develop predictive model equations for the control implementation. Table 5.1 shows the dynamic equations of qZSI in both active/non-shoot-through and shoot-through states. The grid current predictive model derives from the AC-side KVL equation:

Table 5.1: Dynamic Equations of qZS Cells

State	v_{L1}	v_{L2}	v_{DC}	v_{diode}
Active/Null	$V_{PV} - V_{C1}$	$-V_{C2}$	$V_{C1} + V_{C2}$	0
Shoot through	$V_{PV} + V_{C2}$	V_{C1}	0	$-V_{C1}-V_{C2}$

$$v_{qZSI}^{ML} = Ri_g + L\frac{di_g}{dt} + e_g \tag{5.1}$$

where v^{ML}_{qZSI} denotes the voltage vector which is the summation of the voltage vectors by both qZSI cells as shown in Figure 5.1. i_g is the injected current to the grid, L is the filter inductance with R as its equivalent series resistance. Discretizing the differential term in (5.1) with the forward Euler approximation and rearranging terms gives the next-state prediction of i_g:

$$i_g[k+1] = i_g[k]\left(1-\frac{R}{L}T_S\right) + \frac{T_S}{L}(v_{qZSI}^{ML}[k] - e_g[k]) \tag{5.2}$$

The predicted inductor current i_{L1} for each cell is given by:

$$i_{L1,j}^{\delta\in\{0,1\}}[k+1] = i_{L1,j}[k] + \frac{T_S}{L_{1,j}}\left(v_{PV,j}[k] - \delta \times v_{C1,j}[k] + (1-\delta)\times v_{C2,j}[k]\right) \tag{5.3}$$

where j denotes the qZSI cell number, $i_{L1,j}^{\delta\in\{0,1\}}[k+1]$ is the cell j input inductor current prediction and δ is non-shoot-through indicator, i.e. δ is equal to zero in shoot-through mode.

5.3.2 Reference Signal Generation

The proposed MPC scheme includes three references: input inductor current of cell 1 qZSI ($i_{L1,1}$), input inductor current of cell 2 qZSI ($i_{L1,2}$), and injected current to the grid (i_g). As shown in Section 4.2.3, the reference signals for $i_{L1,1}$ and $i_{L1,2}$ are generated by the MPPT algorithm based on the power extraction requirement and grid condition. Furthermore, the grid current reference is generated by its two quadrature components i.e. *d* component for the real power and *q* component for the reactive power injection, respectively. The total grid current *d*-component (real power) is

calculated via separate PI controllers for both qZS cells; these PI controllers generate the active current references (d-component) by regulating the $V_{C1,1}$ and $V_{C1,2}$ voltages of qZSI cells. The reference for $V_{C1,1}$ and $V_{C1,2}$ voltages is 150V, adding up to over twice the grid voltage having $120V_{rms}$ to fulfill high demand of reactive power. The *q*-component is given by the grid codes as explained in Sections 5.1 and 5.2 based on the amount of grid voltage sag [85, 108]. Finally, these *d* and *q* current references with angle information from PLL will be converted into (5.4), which is the AC grid current reference for the MPC controller.

$$i_{gref} = i_{dref} \sin(\omega t) + i_{qref} \cos(\omega t) \tag{5.4}$$

5.4 Efficient Controller Design and Analysis

All feasible switching states of the qZS-CMI are listed in Table 5.2 for prediction of control objectives. These include all active, zero, and shoot-through states. The control scheme takes advantage of the cost redundancy of the $i_{L1}[k+1]$ predictions, as the predictions are equal for all switching states except shoot-through. In this section, i_{L1} next-state predictions for shoot-through and non-shoot-through modes will be referred to as $i_{L1}^{ST}[k+1]$ and $i_{L1}^{NST}[k+1]$, respectively, for clarity and conciseness. The control algorithm is discussed in more detail in the following subsections.

5.4.1 Voltage Window

When implementing a multilevel inverter, it is preferable to reduce harmonic content of the output voltage without interfering with grid current optimization. This is achieved by applying a voltage window for each switching state. This control will only consider voltage levels within one step of the previous output voltage level. This has the added benefit of reducing the number of switching considerations. The voltage window is applied by using the selected switching state

Table 5.2: Switching States for the qZS-CMI

Switching States	S_a	S_{an}	S_b	S_{bn}	S_c	S_{cn}	S_d	S_{dn}	Cell 1 Output Voltage	Cell 2 Output Voltage
1	0	1	0	1	0	1	0	1	0 (Z)	0 (Z)
2	0	1	0	1	1	0	1	0	0 (Z)	0 (Z)
3	0	1	1	0	1	0	1	0	$-V_{DC,1}$	0 (Z)
4	1	0	0	1	0	1	1	0	$+V_{DC,1}$	$-V_{DC,2}$
5	1	0	1	0	0	1	0	1	0 (Z)	0 (Z)
6	1	0	1	0	1	0	1	0	0 (Z)	0 (Z)
7	0	1	0	1	0	1	1	0	0 (Z)	$-V_{DC,2}$
8	0	1	1	0	0	1	0	1	$-V_{DC,1}$	0 (Z)
9	0	1	1	0	1	0	1	0	$-V_{DC,1}$	0 (Z)
10	1	0	1	0	0	1	1	0	0 (Z)	$-V_{DC,2}$
11	0	1	0	1	1	0	0	1	0 (Z)	$+V_{DC,2}$
12	1	0	0	1	0	1	0	1	$+V_{DC,1}$	0 (Z)
13	1	0	0	1	1	0	1	0	$+V_{DC,1}$	0 (Z)
14	1	0	1	0	1	0	0	1	0 (Z)	$+V_{DC,2}$
15	0	1	1	0	0	1	1	0	$-V_{DC,1}$	$-V_{DC,2}$
16	1	0	0	1	1	0	0	1	$+V_{DC,1}$	$+V_{DC,2}$
17	1	1	0	1	1	0	0	1	0 (ST)	$+V_{DC,2}$
18	1	0	0	1	1	1	0	1	$+V_{DC,1}$	0 (ST)
19	1	1	0	1	1	1	0	1	0 (ST)	0 (ST)
20	1	1	0	1	0	1	1	0	0 (ST)	$-V_{DC,2}$
21	0	1	1	0	1	1	0	1	$-V_{DC,1}$	0 (ST)
22	1	1	0	1	0	1	0	1	0 (ST)	0 (Z)
23	1	1	0	1	1	0	1	0	0 (ST)	0 (Z)
24	0	1	0	1	1	1	0	1	0 (Z)	0 (ST)
25	1	0	1	0	1	1	0	1	0 (Z)	0 (ST)

Z: Zero state, ST: Shoot-Through state

as a feedback to the controller, to determine which voltage levels can be considered for the following time step.

5.4.2 Cost Package Construction

For every time step, one cost package is constructed for each qZS cell and sent to the supervisory controller. The cost packages contain all needed information for the supervisory controller to automatically filter out switching states that are unable to optimize the input current objective or adhere to the voltage window. The cost packages and previous output voltage constitute the addresses of associated reduced optimization sets. Algorithm 5.1 evaluates the cost

Algorithm 5.1: Cost Package Construction

Function [CP_1 ,CP_2] = Cost Package Construction **(**$v_{c1\&2}[k]$, $i_{PV1\&2}[k]$, $i_{L1,1\&2ref}[k]$**)**

Initialization: sampling at T_s

1: Find input current costs for each cell

Finding predicted i_{L1} in next state for each cell:

$\delta = 1$, indicating non-shoot-through (NST) mode

compute $i_{L1,1}^{NST}[k+1]$ and $i_{L1,2}^{NST}[k+1]$ from (5.3)

$\delta = 0$, indicating shoot-through (ST) mode

compute $i_{L1,1}^{ST}[k+1]$ and $i_{L1,2}^{ST}[k+1]$ from (5.3)

Finding i_{L1} costs for each control action:

$g_{NST,1} \leftarrow |i_{L1,1}^{NST}[k+1] - i_{L1,1ref}[k]|$, $g_{ST,1} \leftarrow |i_{L1,1}^{ST}[k+1] - i_{L1,1ref}[k]|$

$g_{NST,2} \leftarrow |i_{L1,2}^{NST}[k+1] - i_{L1,2ref}[k]|$, $g_{ST,2} \leftarrow |i_{L1,2}^{ST}[k+1] - i_{L1,2ref}[k]|$

2: Construct Cost Package for each cell

if $g_{ST,1} < g_{NST,1}$ **then**

$CP_1 \leftarrow \begin{bmatrix} 'ST' & g_{ST,1} \\ ; & \\ 'NST' & g_{NST,1} \end{bmatrix}_{2\times2}$

else

$CP_1 \leftarrow \begin{bmatrix} 'NST' & g_{NST,1} \\ ; & \\ 'ST' & g_{ST,1} \end{bmatrix}_{2\times2}$

end if

if $g_{ST,2} < g_{NST,2}$ **then**

$CP_2 \leftarrow \begin{bmatrix} 'ST' & g_{ST,2} \\ ; & \\ 'NST' & g_{NST,2} \end{bmatrix}_{2\times2}$

else

$CP_2 \leftarrow \begin{bmatrix} 'NST' & g_{NST,2} \\ ; & \\ 'ST' & g_{ST,2} \end{bmatrix}_{2\times2}$

end if, Return CP_1, CP_2, **End function**

of the i_{L1} objectives for both ST and NST scenarios. The cost packages rank each scenario by row and contains the cost of each scenario in the second column. In most cases, the grid current optimizer only needs to know which state should be implemented for i_{L1} optimization, but knowing the control cost of the second-rank objective is needed when i_{L1} optimization and the voltage window constraint conflict, as is explained in the following subsection.

5.4.3 Efficient Supervisory Predictive Control

The control algorithm uses the previous voltage level and the input current control objective to filter out several switching states. Figure 5.2 outlines the supervisory control algorithm, and it specifies the switching states that are remaining to optimize injected grid current. The control paths all differ to satisfy the voltage window and i_{L1} control constraints. For example, consider the scenario when the previous output voltage was $+2V_{DC}$ and Algorithm 5.1 has determined both input currents are optimized in the non-shoot-through state. The shoot-through switching states must be eliminated. Additionally, to satisfy the voltage window constraint, only the switching states which produce $+V_{DC}$ or $+2V_{DC}$ can be considered. Thus, the remaining switching states available for grid current optimization are limited to states 11-14 and 16, which produce an output voltage of $+V_{DC}$ and $+2V_{DC}$, respectively. Now consider a similar scenario, except Algorithm 5.1 has determined that cell 1 must implement shoot-through for $i_{L1,1}$ optimization. To allow for shoot-through of cell 1 and adhere to the voltage window constraint, a positive DC-link voltage level must be applied across cell 2; state 17 is the only state that satisfies both conditions. Thus, state 17 is chosen automatically, with grid current optimization bypassed. There is one scenario in which the voltage window and input current control objectives cannot be met simultaneously. This occurs when $v^{ML}{}_{qZSI}[k]$ is equal to $\pm 2V_{DC}$, and both i_{L1} objectives are optimized by implementing shoot-through. This is because the DC-link voltage is zero when both cells are in shoot-through, which is not within the voltage window for $v^{ML}{}_{qZSI}[k]$ equal to $\pm 2V_{DC}$. Adhering to both i_{L1} objectives would violate the voltage window constraint, as the output voltage would be zero. To address this, the controller must decide which i_{L1} objective should be defied. The decision is made by comparing the second-rank cost of each qZS cell, and the cell with the highest rank cost implements shoot-through. That is, the cell with its input current objective more

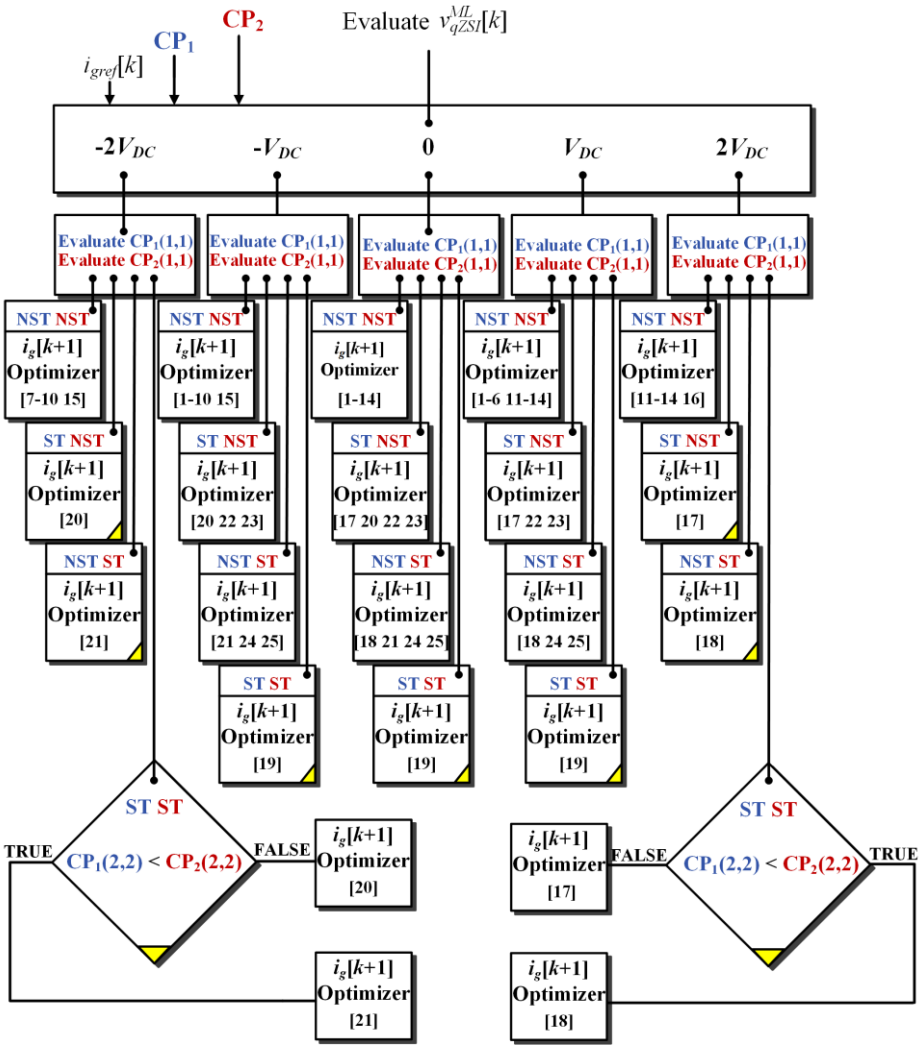


Figure 5.2: Control paths of embedded switching state screen (yellow triangle in path indicates no grid current optimization is done).

closely optimized in non-shoot-through will defy its i_{L1} objective. As shown in Figure 5.2, the second-row, second-column element of both cost packages are compared, and the cell with the highest second-rank cost implements shoot-through.

5.4.4 Comparison with Traditional Finite-Set MPC

For the proposed system, traditional finite-set MPC considers 5^N switching states for every time step, where the system has N qZS cells. To optimize the same control objectives as in the proposed control, the cost function is:

$$J = \lambda_1 \left\| i_{L1,1ref}[k] - i_{L1,1}[k+1] \right\| + \lambda_2 \left\| i_{L1,2ref}[k] - i_{L1,2}[k+1] \right\| + \lambda_3 \left\| i_{gref}[k] - i_g[k+1] \right\| \quad (5.5)$$

where the $\lambda_{1\text{-}3}$ are scaling terms, commonly referred to as weight factors. Using this control approach would require preliminary design of the weight factors. As shown in Table III, the cost computations required for this control scheme is $5N(\beta+1)$, where β is the number of control objectives. For the proposed five-level qZS-CMI, traditional finite-set MPC requires 100 cost function computations. In the proposed control, the number of cost computations is dependent on the cost packages and the previous voltage state. As shown in Figure 5.2, there are $N^2(2N+1)$ control paths for the proposed controller, which can vary in the number of necessary switching states to consider for grid current optimization. In the proposed scheme, half of the control paths do not require any optimization. As shown in Table 5.3, these iterations require only four cost computations, which are necessary to develop the cost packages. The control path with the most cost computations needed is the case where the previously implemented output voltage was zero and both i_{L1} objectives are optimized in the non-shoot-through state. Table 5.3 shows the average control path has over a ninety percent reduction in cost computation. It is notable that no functionality is lost in this architecture, as it only exploits cost redundancy of the input current predictive model, and applies higher priority to input current, as is typically done in predictive control of quasi-Z-source inverters.

5.5 Results and Discussion

The performance of the proposed control scheme is tested during solar irradiance transients and grid voltage sags. Each RPI strategy presented in Section 4.2 is showcased, now for the qZS-CMI. The experiments are implemented with the dSPACE MicrolabBox control platform. A qZS-CMI with two 1kW PV modules connected to each inverter cell is used for the case studies. An RPI ramping rate of 2 p.u. is implemented for the case studies in each scenario, where reactive

Table 5.3: Cost Computation Comparison

Considered Control Path	i_{L1} Cost Comp.	i_g Cost Comp.	Overall Cost Function Comp.	Total Cost Comp.	Comp. Reduction from Traditional FS-MPC
Lowest Comp. path	4	0	0	4	96%
Highest Comp. path	4	14	0	18	82%
Average for all paths	4	3.3	0	7.3	92.7%
Traditional FS-MPC	50	25	25	100	0%

Table 5.4: System Specifications

Parameter	Value
Sampling Time	20µs
Filter Inductance	4mH
Filter Equivalent Series Resistance	50mΩ
Impedance Network Inductances	1.5mH
Impedance Network Capacitances	1000µF
Input Capacitance	2200µF
P_{MPP}	1000W
V_{MPP}	85.6V
Rated Grid Voltage	$120V_{RMS}$

current saturates at $16.7A_{RMS}$ for a fifty percent voltage sag. Additional system specifications are provided in Table 5.4. For each LVRT mode case study, a 20% grid voltage sag is induced, and the RPI is increased to 0.64kVAR, while active power injection varies for each RPI strategy. The major evaluation criteria considered in the case studies are: a) ability to adjust active and reactive power injection to the grid according to the desired RPI strategy during LVRT mode; b) seamless transition between LVRT and normal grid modes of operation; c) ability to operate at global MPP with mismatch in PV modules i.e. unbalance solar irradiance of PV modules connected to different qZSI cell.

Figure 5.3 - Figure 5.6 show the system dynamics response to grid voltage sag for average active power, constant peak current, and constant active current control strategies respectively. The grid-side voltage and current, $i_{L1,1}$, and $v^{ML}{}_{qZSI}$ are captured for these case studies. In Figure

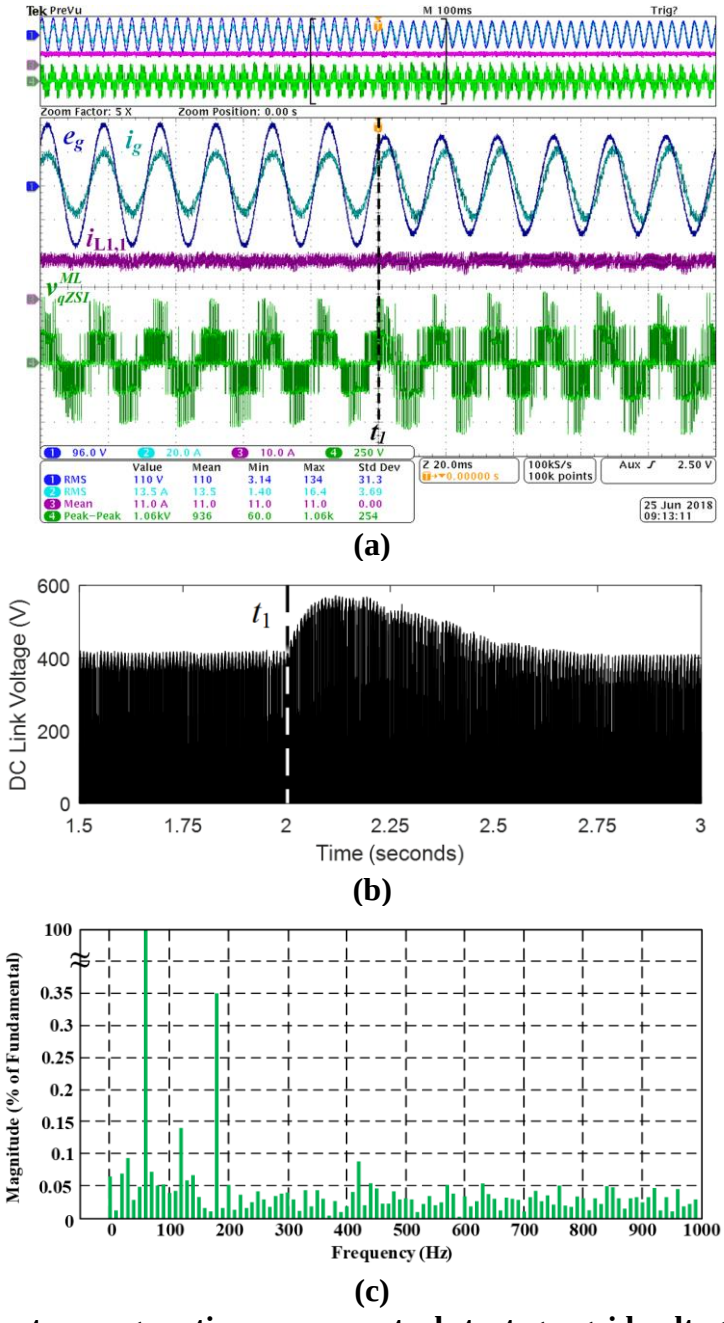
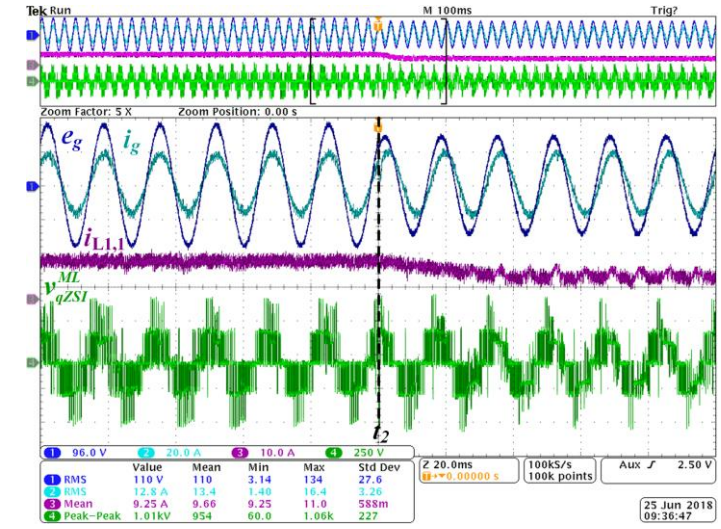


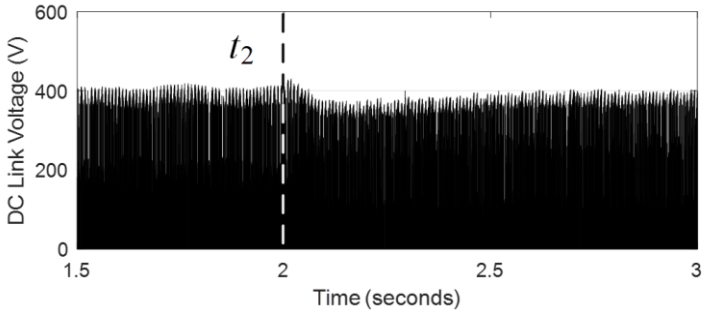
Figure 5.3: Constant average active power control strategy, grid voltage sag occurred at t_1 (a) Grid voltage, injected grid current, qZS cell 1 input current and output voltage (b) DC link voltage (c) FFT of injected current 500ms after voltage sag transient, THD of 1.61%.

5.3, a 20% voltage sag is triggered at t_1, transitioning the control from MPPT to the constant average active power LVRT mode. The inductor current $i_{L1,1}$ is shown to remain constant after the voltage sag trigger, showing that the average active power injection of roughly 2kW remains constant. The MPPT algorithm of each cell's PV array remains at its MPP. Additionally, the grid

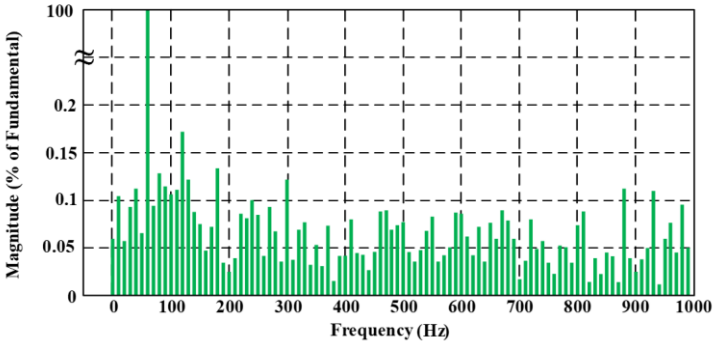
voltage sag and resultant i_{qref} creates a voltage swell across the DC link, shown in Figure 5.3b. A voltage swell is also seen across both $C1,1$ and $C1,2$. The PI controller responds by increasing the active current component i_{dref} from 20.6A to about 26.8A, and the DC link voltage returns to 400v. Thus, grid-side current amplitude increases with the voltage sag to maintain constant active power as well as inject reactive power. The FFT plot of injected current during the sag is shown in Figure 5.3c. The measured THD is 1.61%, well within the IEEE-519 standard of 5% [109]. Figure 5.4 shows the constant peak current control strategy when the voltage sag occurs at t_2. The injected active power decreases to about 1.32kW. The reduction in power extraction from the PV arrays causes a reduction in DC link voltage, as shown in Figure 5.4b. Consequently, a voltage reduction is seen across $C1,1$ and $C1,2$. The PI controller reduces i_{dref} to about 18.9A to return DC link voltage to normal condition. Grid current peak remains constant during voltage sag, but the grid current leads the voltage after instant t_2 while i_{L1} decreases. This indicates that the required reactive power is injected, while active power decreases according to constant peak current strategy. The FFT plot for the injected current during the voltage sag is shown in Figure 5.4c. An increase in THD is noted when compared to the constant average active power RPI strategy, due to the change in the grid current magnitude. Figure 5.5 shows RPI for constant active current after a voltage sag occurs at t_3, where a reduction in i_{L1} is seen according to this RPI strategy. For this RPI strategy, active power injection decreases to 1.6kW. The MPPT algorithm of each qZS cell moves the PV operation point to right of the MPP, until reaching a power limit of 800W. The reduced power extraction from the PV arrays and the reactive power injection counteract their effects on the DC link voltage, and thus no transient is seen in the DC link voltage, as shown in Figure 5.5b. Thus, the voltage across $C1,1$ and $C1,2$ is maintained, and i_{dref} is unchanged. Therefore, applying (4.12), the MPPT algorithm successfully maintains the active current injection during grid voltage



(a)



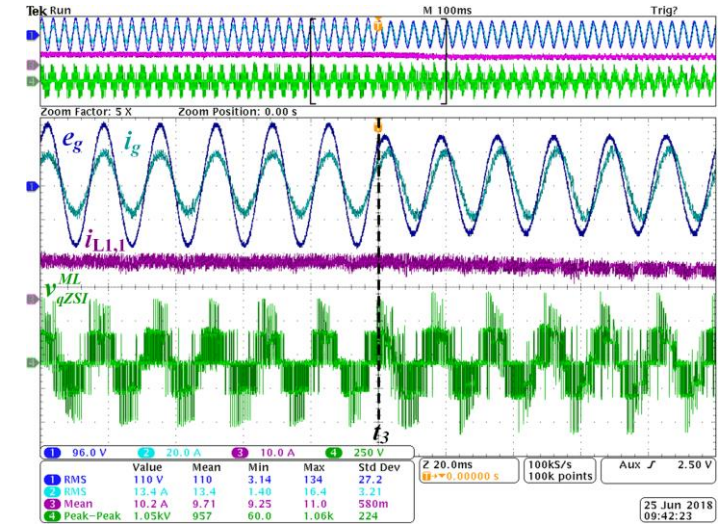
(b)



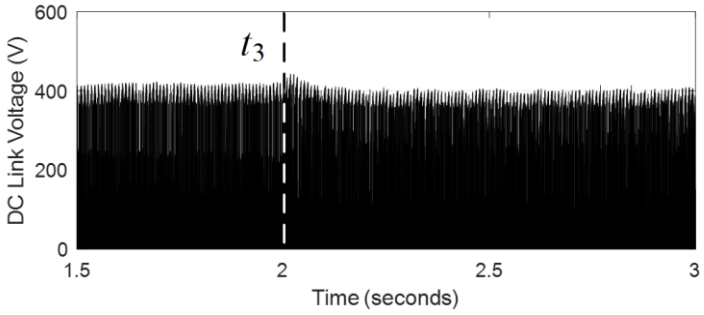
(c)

Figure 5.4: Constant peak current control strategy, grid voltage sag occurred at t_2 (a) Grid voltage, injected grid current, qZS cell 1 input current and output voltage (b) DC link voltage (c) FFT plot of injected current 500ms after voltage sag transient; THD of 2.65%.

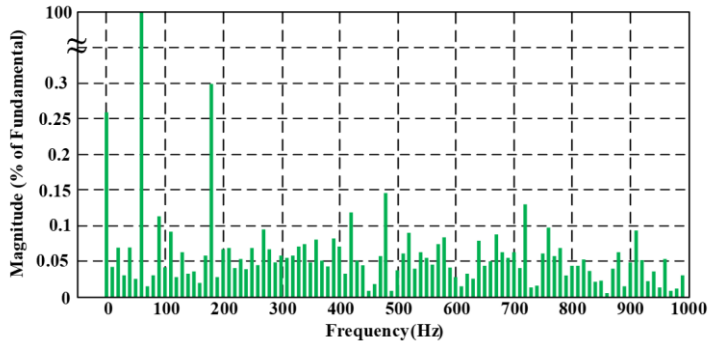
transients. The FFT plot of grid current is shown in Figure 5.5c. The measured THD is lower compared with the constant peak current strategy, due to larger grid current amplitude. For each FFT plot, it is noteworthy that the FFT analysis shows a uniform distribution of harmonic content across the frequency spectrum; this is a result of the variable switching frequency inherent to finite-



(a)



(b)



(c)

Figure 5.5: Constant active current control strategy, grid voltage sag occurred at t_3. (a) Grid voltage, injected grid current, qZS cell 1 input current and output voltage (b) DC link voltage (c) FFT plot of injected current 500ms after voltage sag transient; THD of 2.46%

set MPC In Figure 5.6, the grid recovers from voltage sag at t_4, transitioning rapidly from average active power strategy in LVRT to MPPT mode. This is evidenced by the reduction in grid-side current amplitude and the phase alignment of grid-side current and voltage after t_4. These experiments verify the controller performance based on aforementioned evaluation criteria (a) and

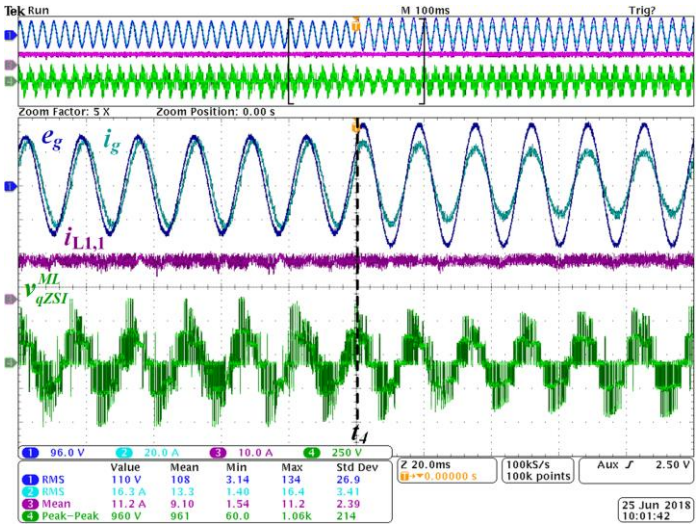


Figure 5.6: System recovery from LVRT to normal grid conditions at instant t_4 using constant average active power control strategy.

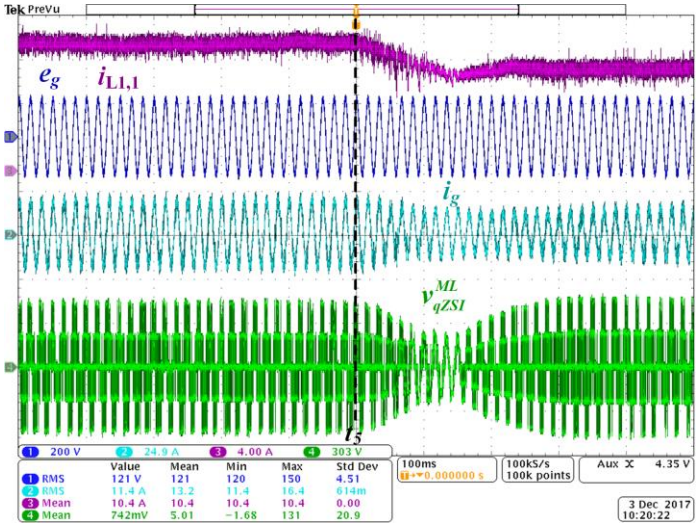


Figure 5.7: Step change in solar irradiance from 1000W/m^2 to 800W/m^2 at instant t_5 in normal grid mode.

(b). Figure 5.7 shows grid-side parameters during a step change in solar irradiance from 1000W/m^2 to 800W/m^2 at t_5 in normal grid mode. The reduction in power to the grid causes a sag in DC link voltage, and the PI controller responds by reducing i_{dref}. Figure 5.8 shows the PV-side parameters for the step-change in solar irradiance at instant t_5. As it is captured, the PV current and voltage adjusted to track the new MPP in less than 180ms after the transient occurred at t_5, this case study

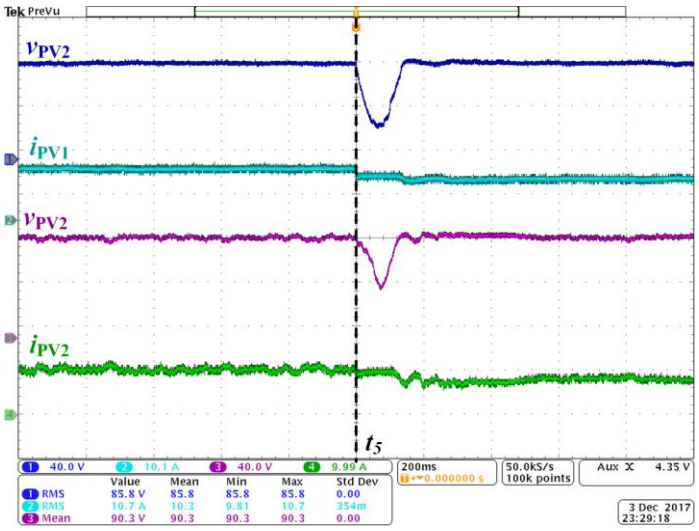


Figure 5.8: PV-side parameters during a step change in solar irradiance from 1000W/m^2 to 800W/m^2 at instant t_5 in normal grid mode.

demonstrates the fast-dynamic response of the proposed control scheme to step change in solar irradiance.

Finally, the system is tested under PV mismatching conditions to verify the controller performance according to aforementioned evaluation criteria (c). In Figure 5.9 at t_6, the solar irradiance of the PV string connected to the qZSI cell 1 is reduced from 1000W/m^2 to 800W/m^2, while the PV string of the qZSI cell 2 operates at 1000W/m^2. The grid-side peak current and PV string 1 current, I_{PV1}, are reduced according to the new MPP operation, while the PV string 2 current is maintained according to its MPP operation at 1000W/m^2. The solar irradiance of PV string 1 returns to 1000W/m^2 at t_7. This case study demonstrates the robustness of the proposed control scheme to unbalance solar irradiance at cells of qZSI-CMI. As it is shown, each qZSI cell can independently boost its voltage, making the inverter more robust and preventing uneven voltage levels. The system maintains stability during PV string mismatch conditions.

5.6 Conclusion

A power electronics interface and control scheme for a cascaded multilevel impedance-source inverter is presented in this chapter. It can operate each PV string independently and

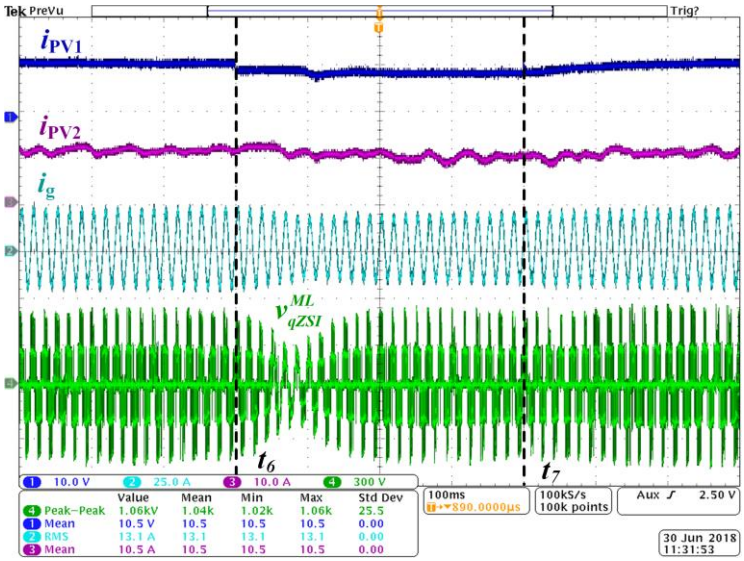


Figure 5.9: PV mismatching due to unbalance solar irradiance level of PV cells, transient from 1000W/m^2 to 800W/m^2 in solar irradiance of PV cell 1 during period t_6 to t_7 while the solar irradiance of PV cell 2 is kept constant at 1000W/m^2.

employs RPI during abnormal grid conditions to support grid stability. The efficient control scheme reduces over ninety percent of the cost computation on average, when compared to a finite-set MPC scheme with the same control objectives. With the decoupled active and reactive power control, the proposed system can support unity power factor of the grid during voltage sags. Additionally, the system is shown to remain stable during solar irradiance imbalances and transients in solar irradiance.

Chapter 6 - Conclusion and Future Work

This chapter will summarize the proposals and findings discussed in this thesis. I will move to recommended ideas for investigation in future work.

6.1 Summary of Findings

Model predictive control has proven to be a fast and promising solution for realizing smart inverters, and its slow emergence in industrial power electronics devices ensures it will make its way into industry. This thesis has introduced modifications to finite-set model predictive control for various applications, demonstrating how often-toted setbacks of finite-set MPC can be solved. Notably, the issues of the ambiguous cost function design and, for some applications, impractical computational burden are addressed. Following the introduction, I proposed an alteration of finite-set MPC which removes the ambiguity in the design stage of finite-set MPC. Using hierarchical model predictive control, the desired tracking performance of each objective is defined during design. Assuming multiple objectives, the designer must rank each objective and apply an associated cost tolerance (or acceptable error) on the objectives. The concept is proposed generally, then is applied to a grid-connected cascaded H-bridge inverter. By *isolating* the cost terms, the control can include objective with volatile magnitude. This is shown with an objective referred to as *sequence frequency* minimization, or an objective which merely tries to balance the sequence in which redundant, yet optimal, switching sequences are selected. This was shown to reduce the mismatch in power draw among the cascaded H-bridge cells, which goes unregulated in the standard finite-set MPC implementation. Further in this work, it is demonstrated how finite-set MPC provides faster transient response than control methods which utilize linear controllers. This is because linear controllers with an integral component generally exhibit a tradeoff between transient settling time and stability, or steady-state oscillation.

In chapter 3, the concept of hierarchical objective tracking is extended further. While the first chapter was more aimed at introducing and demonstrating the concept of hierarchical finite-set MPC, I like to consider the following work a control technique that is form-fit to the application of the cascaded H-bridge topology. The control aims to address the exponentially growing control set, or exponentially growing set of possible switching sequences, that occurs as H-bridges are included in the topology. The control exploits the fact that the grid current depends only on the inverter's output voltage. A modification to the next-state current prediction produces an optimal output voltage, which can then be optimized among the list of possible output voltage levels. This creates a dramatic reduction in computation. Further, since the voltage level of a switching sequence and the number of switching events between switching sequences can be computed offline, switching event minimization is embedded in the control via a lookup matrix, further reducing computation. The control is supplemented with an algorithm which cycles through lists of redundant and optimal switching sequences. This equalizes the power draw among the isolated cells. When applied to a nine-level cascaded H-bridge, the control selects an optimal switching sequence approximately five times faster on the rapid control prototyping hardware. It was also shown to hold similar tracking performance, when compared to traditional finite-set MPC. Thus, the proposal addresses concerns of impractical computational burden for finite-set MPC of high-level cascaded H-bridge topologies.

In the following chapter, the quasi-Z-source inverter is introduced for a full photovoltaic-to-grid control solution. The control implements flexible power point tracking at the input side. This enables an intentional reduction in power draw from the photovoltaic string to account for reactive power injection in the event of a grid voltage sag. Three reactive power injection techniques previously proposed in literature are implemented, each of which produce a different

power draw from the photovoltaic side. Each reactive power injection technique is tested on the experimental setup. This chapter includes an additional alternative to preliminary cost function design in MPC. Specifically, the weight factors are adjusted in real time, according to the controller's tracking performance of each objective. The weight factors are determined based on the lowest-cost next state prediction of each objective, applied over a moving window. Further, to address the adjustments in amplitude, the control objectives are normalized online using a moving-window RMS calculation. The control was shown to improve tracking performance of each control objective, when compared to a previously implemented static weight factor ratio.

In chapter 5, a model predictive control solution is proposed for a similar application, only the topology includes cascading quasi-Z-source cells. As was discussed for the case of the cascaded H-bridge, adding series-connected inverters dramatically increases the control set of the finite-set model predictive control. The number of online computations is reduced significantly by exploiting the fact that only two unique next-state predictions exist for each cell, and thus is not necessary to compute for all 25 switching sequences. Further, the possible output voltage is constrained such that the change in output voltage level cannot change by more than one between sampling instances. This further reduces the control set, and the average online cost computations are reduced by over ninety percent on average. The case studies done in this chapter are not unlike that of chapter 4, where three reactive power injection techniques are tested during emulated grid voltage sags.

6.2 Recommended Future Work

I have proposed some works and even published conference articles on a few topics that either have not yet been implemented on hardware or must be extended prior to considering

hardware implementation. Thus, they are excluded from the main body of this thesis and included here as recommendations for future work.

6.2.1 Auto-tuned Model Parameters in Finite-set Predictive Control

Finite-set model predictive control requires making next-state predictions. Often, the predicted variable is sensed directly to compute a prediction. An example is the next-state output current predictions that are made in chapters 2,4, and 5. Also required is (typically hard-programmed) model parameters, such as the impedance of the output filter and time between discrete sampling instances. However, unlike the sampling rate, model parameters are often not fixed. Physical inductors have inductance which varies with temperature, age, and current. Thus, filter components can become misaligned with the nominal value implemented in the control. This misalignment can result in inaccurate next-state predictions and ultimately poorer tracking of the control objectives. The work proposed in [47] presents an intuitive method to align the control's inductance to that of the physical system. The model parameters are incrementally adjusted according to error in the next-state predictions of the output current. It is implemented on a grid-connected active rectifier with predictive control of active and reactive power using an inductive filter. The concept can be applied to higher order filters by using multiple sensor measurements to produce an overdetermined system of equations, which can then be used to adapt the filter components using the least-squares solution. However, before the concept can be successfully implemented outside of simulation, the control must address DC error on the sensor measurements. The control is adjusted according to steady-state error in next-state predictions, and DC error on sensor measurements could disrupt proper model tuning.

6.2.2 Self-healing Model Predictive Control of Cascaded Multilevel Inverters

Detection and location of open-circuit switch faults is possible using a similar technique to that of the auto-tuned model parameter technique. Each switching sequence has an associate output voltage level, using the current and previous output current sensor readings, the *actual* applied output voltage can be estimated. Small errors (differences between the expected, applied output voltage and the estimated output voltage) are expected, due to discretization error of the modeled differential equation. However, substantial errors can be associated with open-circuit switch faults, as unintentional logic-low gate signals tie an H-bridge output voltage to zero. In [110], I propose a diagnostic algorithm that initiates upon sensing a substantial error reading. The diagnostic algorithm then collects all switching sequences which produce substantial, or *anomalous*, voltage errors. The diagnostic algorithm ensures all switching sequences are tested by cycling through redundant switching sequences (with respect to current tracking). Once all switching sequences have been tested, the algorithm checks if any particular switches were sent logic-high values during most of the anomalous instances. If there is above ninety percent commonality between a switch receiving logic-high values and instances of anomalous error, the switch is considered faulted. For the cascaded multilevel inverter topology, operation can continue with a faulted switch; the control set must be reduced to account for the uncontrollable, permanently logic-low switch. Further the definition of anomalous behavior is computed online, by measuring the mean and standard deviation of the absolute value of the error measurements over a moving window. Anomalous behavior is defined as five standard deviations larger than the error measurements captured within the moving window. In the application discussed in the paper, this algorithm is proposed in a grid-connected CMI, where the inverter's control sends a faulted-state signal to a higher-level control to verify the validity of phasor measurement unit readings for cyber-physical attack detection.

Further discussion of the application is outside the scope of this section. However, the algorithm must be adapted slightly for hardware implementation. When ran in simulation, there is no accounting for implementation, sensing, and computational delays. That is, once a switching sequence is applied, its effect on the output current is immediately sensed and registered by the controller. For hardware implementation, practical delays must be accounted for, so a sensor reading can be associated with the implementation of the correct switching sequence.

References

[1] Y. Yang, P. Enjeti, F. Blaabjerg, and H. Wang, "Wide-scale adoption of photovoltaic energy: Grid code modifications are explored in the distribution grid," *IEEE Industry Applications Magazine,* vol. 21, no. 5, pp. 21-31, 2015.

[2] E. Kabir, P. Kumar, S. Kumar, A. A. Adelodun, and K.-H. Kim, "Solar energy: Potential and future prospects," *Renewable and Sustainable Energy Reviews,* vol. 82, pp. 894-900, 2018.

[3] R. J. Salawitch, T. P. Canty, A. P. Hope, W. R. Tribett, and B. F. Bennett, *Paris climate agreement: Beacon of hope*. Springer Nature, 2017.

[4] B. Kroposki *et al.*, "Achieving a 100% renewable grid: Operating electric power systems with extremely high levels of variable renewable energy," *IEEE Power and Energy Magazine,* vol. 15, no. 2, pp. 61-73, 2017.

[5] R. Brundlinger *et al.*, "Lab tests: Verifying that smart grid power converters are truly smart," *IEEE Power and Energy Magazine,* vol. 13, no. 2, pp. 30-42, 2015.

[6] M. H. Bollen, "Understanding Power Quality Problems," in *Voltage sags and Interruptions*: IEEE press, 2000, ch. Voltage Sags - Characterization.

[7] X. Zhao, L. Chang, R. Shao, and K. Spence, "Power system support functions provided by smart inverters—A review," *CPSS Transactions on Power Electronics and Applications,* vol. 3, no. 1, pp. 25-35, 2018.

[8] S. Vazquez, J. Rodriguez, M. Rivera, L. G. Franquelo, and M. Norambuena, "Model Predictive Control for Power Converters and Drives: Advances and Trends," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 2, pp. 935-947, 2017, doi: 10.1109/TIE.2016.2625238.

[9] M. G. Judewicz, S. A. González, N. I. Echeverría, J. R. Fischer, and D. O. Carrica, "Generalized predictive current control (GPCC) for grid-tie three-phase inverters," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 7, pp. 4475-4484, 2015.

[10] J. Holtz, "Pulsewidth modulation for electronic power conversion," *Proceedings of the IEEE,* vol. 82, no. 8, pp. 1194-1214, 1994.

[11] T. Geyer and D. E. Quevedo, "Multistep finite control set model predictive control for power electronics," *IEEE Transactions on Power electronics,* vol. 29, no. 12, pp. 6836-6846, 2014.

[12] S. Vazquez, A. Marquez, R. Aguilera, D. Quevedo, J. I. Leon, and L. G. Franquelo, "Predictive optimal switching sequence direct power control for grid-connected power converters," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 4, pp. 2010-2020, 2014.

[13] S. A. Larrinaga, M. A. R. Vidal, E. Oyarbide, and J. R. T. Apraiz, "Predictive control strategy for DC/AC converters based on direct power control," *IEEE Transactions on Industrial Electronics,* vol. 54, no. 3, pp. 1261-1271, 2007.

[14] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[15] T. Geyer, *Model predictive control of high power converters and industrial drives*. John Wiley & Sons, 2016.

[16] J. Rodriguez and P. Cortés, "Weighting Factor Design," in *Predictive Control of Power Converters and Electrical Drives*. Chichester, UK: John Wiley & Sons, Ltd., 2012, ch. 11, pp. 163-168.

[17] M. Morari, C. E. Garcia, and D. M. Prett, "Model predictive control: theory and practice," *IFAC Proceedings Volumes,* vol. 21, no. 4, pp. 1-12, 1988.

[18] J.-S. Lai and F. Z. Peng, "Multilevel converters-a new breed of power converters," *IEEE Transactions on Industry Applications,* vol. 32, no. 3, pp. 509-517, 1996.

[19] P. M. Bhagwat and V. Stefanovic, "Generalized structure of a multilevel PWM inverter," *IEEE Transactions on Industry Applications,* no. 6, pp. 1057-1069, 1983.

[20] J. Rodriguez *et al.*, "Multilevel converters: An enabling technology for high-power applications," *Proceedings of the IEEE,* vol. 97, no. 11, pp. 1786-1817, 2009.

[21] M. Marchesoni, M. Mazzucchelli, and S. Tenconi, "A nonconventional power converter for plasma stabilization," *IEEE Transactions on Power Electronics,* vol. 5, no. 2, pp. 212-219, 1990.

[22] Y. Yu, G. Konstantinou, B. Hredzak, and V. G. Agelidis, "Operation of cascaded H-bridge multilevel converters for large-scale photovoltaic power plants under bridge failures," *IEEE Transactions on Industrial Electronics,* vol. 62, no. 11, pp. 7228-7236, 2015.

[23] L. M. Tolbert, F. Z. Peng, and T. G. Habetler, "Multilevel converters for large electric drives," *IEEE Transactions on Industry Applications,* vol. 35, no. 1, pp. 36-44, 1999.

[24] F. Z. Peng, J.-S. Lai, J. W. McKeever, and J. VanCoevering, "A multilevel voltage-source inverter with separate DC sources for static var generation," *IEEE Transactions on Industry Applications,* vol. 32, no. 5, pp. 1130-1138, 1996.

[25] J. Rodriguez, J.-S. Lai, and F. Z. Peng, "Multilevel inverters: a survey of topologies, controls, and applications," *IEEE Transactions on Industrial Electronics,* vol. 49, no. 4, pp. 724-738, 2002.

[26] L. M. Tolbert, F. Z. Peng, and T. G. Habetler, "Multilevel PWM methods at low modulation indices," *IEEE Transactions on Power Electronics,* vol. 15, no. 4, pp. 719-725, 2000.

[27] ABB, "ABB's Model Predictive Torque Control (MPTC) protects against gas supply interruptions," pp. [Online]. Available: https://new.abb.com/drives/media/abbs-model-predictive-torque-control-mptc-protects-against-gas-supply-interruptions, 2016.

[28] P. Cortés, M. P. Kazmierkowski, R. Kennel, D. E. Quevedo, and J. R. Rodriguez, "Predictive Control in Power Electronics and Drives," *IEEE Trans. Industrial Electronics,* vol. 55, no. 12, pp. 4312-4324, 2008.

[29] H. A. Young, M. A. Perez, and J. Rodriguez, "Analysis of finite-control-set model predictive current control with model parameter mismatch in a three-phase inverter," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 5, pp. 3100-3107, 2016.

[30] T. Geyer, G. Papafotiou, and M. Morari, "Model predictive direct torque control—Part I: Concept, algorithm, and analysis," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 6, pp. 1894-1905, 2008.

[31] H. A. Young, M. A. Perez, J. Rodriguez, and H. Abu-Rub, "Assessing finite-control-set model predictive control: A comparison with a linear current controller in two-level voltage source inverters," *IEEE Industrial Electronics Magazine,* vol. 8, no. 1, pp. 44-52, 2014.

[32] J. Rodríguez, R. M. Kennel, J. R. Espinoza, M. Trincado, C. A. Silva, and C. A. Rojas, "High-performance control strategies for electrical drives: An experimental assessment," *IEEE Transactions on Industrial Electronics,* vol. 59, no. 2, pp. 812-820, 2011.

[33] H. Abu-Rub, J. Guzinski, Z. Krzeminski, and H. A. Toliyat, "Predictive current control of voltage-source inverters," *IEEE Transactions on Industrial Electronics,* vol. 51, no. 3, pp. 585-593, 2004.

[34] J. Rodriguez *et al.,* "State of the art of finite control set model predictive control in power electronics," *IEEE Transactions on Industrial Informatics,* vol. 9, no. 2, pp. 1003-1016, 2012.

[35] R. P. Aguilera *et al.*, "Selective harmonic elimination model predictive control for multilevel power converters," *IEEE Transactions on Power Electronics,* vol. 32, no. 3, pp. 2416-2426, 2016.

[36] P. Acuna, R. P. Aguilera, A. M. Ghias, M. Rivera, C. R. Baier, and V. G. Agelidis, "Cascade-free model predictive control for single-phase grid-connected power converters," *IEEE Transactions on Industrial Electronics,* vol. 64, no. 1, pp. 285-294, 2016.

[37] A. A. Ahmed, B. K. Koh, and Y. I. Lee, "A comparison of finite control set and continuous control set model predictive control schemes for speed control of induction motors," *IEEE Transactions on Industrial Informatics,* vol. 14, no. 4, pp. 1334-1346, 2017.

[38] P. Cortés *et al.*, "Guidelines for weighting factors design in model predictive control of power converters and drives," in *IEEE ICIT*, Churchill, Victoria, Australia, pp. 1-7, 2009.

[39] S. A. Davari, D. A. Khaburi, and R. Kennel, "An improved FCS–MPC algorithm for an induction motor with an imposed optimized weighting factor," *IEEE Transactions on Power Electronics,* vol. 27, no. 3, pp. 1540-1551, 2011.

[40] M. B. Shadmand, S. Jain, and R. S. Balog, "Autotuning technique for the cost function weight factors in model predictive control for power electronic interfaces," *IEEE Journal of Emerging and Selected Topics in Power Electronics,* vol. 7, no. 2, pp. 1408-1420, 2018.

[41] L. Comparatore, A. Renault, J. Pacher, R. Gregor, J. Rodas, and M. Rivera, "Model based predictive control with switcher of redundant vectors for a cascade h-bridge multilevel statcom," in *IEEE ANDESCON*, 2016, pp. 1-4.

[42] W. Jiang, W. Song, J. Ma, and B. Liu, "Multiple voltage vectors model predictive current control for single-phase cascaded H-bridge rectifiers," in *IEEE International Power Electronics and Application Conference and Exposition (PEAC)*, 2018, pp. 1-5.

[43] M. Ciobotaru, R. Teodorescu, and F. Blaabjerg, "A new single-phase PLL structure based on second order generalized integrator," in *2006 37th IEEE Power Electronics Specialists Conference*, 18-22 June 2006 2006, pp. 1-6, doi: 10.1109/pesc.2006.1711988.

[44] Y. Han, M. Luo, X. Zhao, J. M. Guerrero, and L. Xu, "Comparative performance evaluation of orthogonal-signal-generators-based single-phase PLL algorithms—A survey," *IEEE Transactions on Power Electronics,* vol. 31, no. 5, pp. 3932-3944, 2015.

[45] P. Cortés, A. Wilson, S. Kouro, J. Rodriguez, and H. Abu-Rub, "Model predictive control of multilevel cascaded H-bridge inverters," *IEEE Transactions on Industrial Electronics,* vol. 57, no. 8, pp. 2691-2699, 2010.

[46] H. Abu-Rub, M. Malinowski, and K. Al-Haddad, *Power electronics for renewable energy systems, transportation and industrial applications*. John Wiley & Sons, 2014.

[47] M. Easley, A. Y. Fard, F. Fateh, M. B. Shadmand, and H. Abu-Rub, "Auto-tuned Model Parameters in Predictive Control of Power Electronics Converters," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, pp. 3703-3709.

[48] G. Wang *et al.*, "A review of power electronics for grid connection of utility-scale battery energy storage systems," *IEEE Transactions on Sustainable Energy,* vol. 7, no. 4, pp. 1778-1790, 2016.

[49] S. Vazquez, S. M. Lukic, E. Galvan, L. G. Franquelo, and J. M. Carrasco, "Energy storage systems for transport and grid applications," *IEEE Transactions on Industrial Electronics,* vol. 57, no. 12, pp. 3881-3895, 2010.

[50] L. Maharjan, T. Yamagishi, and H. Akagi, "Active-power control of individual converter cells for a battery energy storage system based on a multilevel cascade PWM converter," *IEEE Transactions on Power Electronics,* vol. 27, no. 3, pp. 1099-1107, 2010.

[51] A. Gholizad and M. Farsadi, "A novel state-of-charge balancing method using improved staircase modulation of multilevel inverters," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 10, pp. 6107-6114, 2016.

[52] L. Maharjan, S. Inoue, and H. Akagi, "A transformerless energy storage system based on a cascade multilevel PWM converter with star configuration," *IEEE Transactions on Industry Applications,* vol. 44, no. 5, pp. 1621-1630, 2008.

[53] F. Z. Peng, Y. Liu, S. Yang, S. Zhang, D. Gunasekaran, and U. Karki, "Transformer-less unified power-flow controller using the cascade multilevel inverter," *IEEE Transactions on Power Electronics,* vol. 31, no. 8, pp. 5461-5472, 2016.

[54] J. Rodriguez *et al.*, "Predictive Current Control of a Voltage Source Inverter," *IEEE Transactions on Industrial Electronics,* vol. 54, no. 1, pp. 495-503, 2007.

[55] H. T. Nguyen, E.-K. Kim, I.-P. Kim, H. H. Choi, and J.-W. Jung, "Model predictive control with modulated optimal vector for a three-phase inverter with an LC filter," *IEEE Transactions on Power Electronics,* vol. 33, no. 3, pp. 2690-2703, 2017.

[56] M. Easley, S. Jain, M. B. Shadmand, and H. Abu-Rub, "Computationally Efficient Distributed Predictive Controller for Cascaded Multilevel Impedance Source Inverter With LVRT Capability," *IEEE Access,* vol. 7, pp. 35731-35742, 2019.

[57] C. Xia, T. Liu, T. Shi, and Z. Song, "A simplified finite-control-set model-predictive control for power converters," *IEEE Transactions on Industrial Informatics,* vol. 10, no. 2, pp. 991-1002, 2014.

[58] S. Kwak and J.-C. Park, "Switching strategy based on model predictive control of VSI to obtain high efficiency and balanced loss distribution," *IEEE Transactions on Power Electronics,* vol. 29, no. 9, pp. 4551-4567, 2013.

[59] R. Chan, J. Baek, and S. Kwak, "Simple algorithm with fast dynamics for cascaded H-bridge multilevel inverter based on model predictive control method," in *Applied Power Electronics Conference and Exposition (APEC), 2017 IEEE*, 2017: IEEE, pp. 696-702.

[60] M. Easley, S. Jain, M. B. Shadmand, F. Fateh, and B. Mirafzal, "Hierarchical Model Predictive Control for Cascaded Multilevel Inverters," in *Applied Power Electronics Conference and Exposition (APEC)*, Anaheim, CA, 2019, pp. 614-619.

[61] X. Liang, "Emerging power quality challenges due to integration of renewable energy sources," *IEEE Transactions on Industry Applications,* vol. 53, no. 2, pp. 855-866, 2016.

[62] A. Sangwongwanich, Y. Yang, F. Blaabjerg, and H. Wang, "Benchmarking of constant power generation strategies for single-phase grid-connected photovoltaic systems," *IEEE Transactions on Industry Applications,* vol. 54, no. 1, pp. 447-457, 2017.

[63] E. On, "Grid code high and extra high voltage," *E. On Netz Gmbh, Bayreuth, Germany,* 2006.

[64] S. Mukherjee, V. R. Chowdhury, P. Shamsi, and M. Ferdowsi, "Power-Angle Synchronization for Grid-Connected Converter With Fault Ride-Through Capability for Low-Voltage Grids," *IEEE Transactions on Energy Conversion,* vol. 33, no. 3, pp. 970-979, 2018.

[65] M. Dietmannsberger, F. Grumm, and D. Schulz, "Simultaneous implementation of LVRT capability and anti-islanding detection in three-phase inverters connected to low-voltage grids," *IEEE Transactions on Energy Conversion,* vol. 32, no. 2, pp. 505-515, 2017.

[66] Y. Liu, B. Ge, H. Abu-Rub, and H. Sun, "Hybrid pulsewidth modulated single-phase quasi-Z-source grid-tie photovoltaic power system," *IEEE Trans. Ind. Informat.,* vol. 12, no. 2, pp. 621-632, 2016.

[67] S. B. Kjaer, J. K. Pedersen, and F. Blaabjerg, "A review of single-phase grid-connected inverters for photovoltaic modules," *IEEE Trans. Ind. Appl.,* vol. 41, no. 5, pp. 1292-1306, 2005.

[68] D. Cao, S. Jiang, X. Yu, and F. Z. Peng, "Low-cost semi-Z-source inverter for single-phase photovoltaic systems," *IEEE Transactions on Power Electronics,* vol. 26, no. 12, pp. 3514-3523, 2011.

[69] Y. Huang, M. Shen, F. Z. Peng, and J. Wang, "Z-Source Inverter for Residential Photovoltaic Systems," *IEEE Transactions on Power Electronics,* vol. 21, no. 6, pp. 1776-1782, 2006.

[70] Y. Tang, J. Wei, and S. Xie, "Grid-tied photovoltaic system with series Z-source inverter," *IET Renewable Power Generation,* vol. 7, no. 3, pp. 275-283, 2013.

[71] Y. Liu, H. Abu-Rub, and B. Ge, "Z-Source/Quasi-Z-Source Inverters: Derived Networks, Modulations, Controls, and Emerging Applications to Photovoltaic Conversion," *IEEE Ind. Electron. Mag.,* vol. 8, no. 4, pp. 32-44, 2014.

[72] J. Anderson and F. Z. Peng, "Four quasi-Z-source inverters," in *39th Annual IEEE PESC*, Rhodes, Greece, pp. 2743-2749, 2008.

[73] Y. Liu, H. Abu-Rub, B. Ge, F. Blaabjerg, O. Ellabban, and P. C. Loh, *Impedance source power electronic converters*. John Wiley & Sons, 2016.

[74] Y. Li, S. Jiang, J. G. Cintron-Rivera, and F. Z. Peng, "Modeling and control of quasi-Z-source inverter for distributed generation applications," *IEEE Transactions on Industrial Electronics,* vol. 60, no. 4, pp. 1532-1541, 2012.

[75] C. Garcia, J. Rodriguez, C. Silva, C. Rojas, P. Zanchetta, and H. Abu-Rub, "Full predictive cascaded speed and current control of an induction machine," *IEEE Transactions on Energy Conversion,* vol. 31, no. 3, pp. 1059-1067, 2016.

[76] T. Jin, X. Shen, T. Su, and R. C. Flesch, "Model Predictive Voltage Control Based on Finite Control Set with Computation Time Delay Compensation for PV Systems," *IEEE Transactions on Energy Conversion,* vol. 34, no. 1, pp. 330-338, 2018.

[77] M. Mosa, R. S. Balog, and H. Abu-Rub, "High-performance predictive control of quasi-impedance source inverter," *IEEE Transactions on Power Electronics,* vol. 32, no. 4, pp. 3251-3262, 2017.

[78] S. Jain, M. Easley, F. Fateh, and M. B. Shadmand, "Distributed Predictive Control Scheme for Grid-Tied Cascaded Multilevel Impedance Source Inverter with LVRT Capability," in *IEEE ECCE*, 2018, pp. 6236-6242.

[79] S. Sajadian and R. Ahmadi, "ZSI for PV systems with LVRT capability," *IET Renewable Power Generation,* vol. 12, no. 11, pp. 1286-1294, 2018.

[80] S. Bayhan, H. Abu-Rub, and R. S. Balog, "Model predictive control of quasi-z-source four-leg inverter," *IEEE Transactions on Industrial Electronics,* vol. 63, no. 7, pp. 4506-4516, 2016.

[81] A. Bakeer, M. A. Ismeil, and M. Orabi, "A powerful finite control set-model predictive control algorithm for quasi Z-source inverter," *IEEE Transactions on Industrial Informatics,* vol. 12, no. 4, pp. 1371-1379, 2016.

[82] A. Lashab, D. Sera, and J. M. Guerrero, "A low-computational high-performance model predictive control of single phase battery assisted Quasi Z-source PV inverters," in *10th International Conference on Power Electronics and ECCE Asia (ICPE 2019-ECCE Asia)*, 2019, pp. 1873-1878.

[83] A. Lashab, D. Sera, and J. M. Guerrero, "Model Predictive Control of Cascaded Multilevel Battery Assisted Quasi Z-Source PV Inverter with Reduced Computational Effort," in *IEEE Energy Conversion Congress and Exposition (ECCE)*, 2019, pp. 6501-6507.

[84] Y. Liu, H. Abu-Rub, Y. Xue, and F. Tao, "A discrete-time average model-based predictive control for a quasi-Z-source inverter," *IEEE Transactions on Industrial Electronics,* vol. 65, no. 8, pp. 6044-6054, 2017.

[85] Y. Yang, H. Wang, and F. Blaabjerg, "Reactive power injection strategies for single-phase photovoltaic systems considering grid requirements," *IEEE Trans. Ind. Appl.*, vol. 50, no. 6, pp. 4065-4076, 2014.

[86] S. Jain, M. Easley, M. B. Shadmand, and B. Mirafzal, "Decoupled active and reactive power predictive control of impedance source microinverter with LVRT capability," in *2018 IEEE Power and Energy Conference at Illinois (PECI)*, 2018: IEEE, pp. 1-6.

[87] M. Ciobotaru, R. Teodorescu, and F. Blaabjerg, "A new single-phase PLL structure based on second order generalized integrator," in *37th IEEE PESC*, Jeju, South Korea, pp. 1-6, 2006.

[88] M. Mirhosseini, J. Pou, and V. G. Agelidis, "Single-and two-stage inverter-based grid-connected photovoltaic power plants with ride-through capability under grid faults," *IEEE Transactions on Sustainable Energy*, vol. 6, no. 3, pp. 1150-1159, 2015.

[89] Y. Zhu and J. Fei, "Disturbance observer based fuzzy sliding mode control of PV grid connected inverter," *IEEE Access*, vol. 6, pp. 21202-21211, 2018.

[90] Y. Zhu and J. Fei, "Adaptive global fast terminal sliding mode control of grid-connected photovoltaic system using fuzzy neural network approach," *IEEE Access*, vol. 5, pp. 9476-9484, 2017.

[91] B. Xu and X. Ran, "Sliding Mode Control for Three-Phase Quasi-Z-Source Inverter," *IEEE Access*, vol. 6, pp. 60318-60328, 2018.

[92] S. Sabyasachi, V. B. Borghate, R. R. Karasani, S. K. Maddugari, and H. M. Suryawanshi, "Hybrid Control Technique-Based Three-Phase Cascaded Multilevel Inverter Topology," *IEEE Access*, vol. 5, pp. 26912-26921, 2017.

[93] Y. Liu, B. Ge, H. Abu-Rub, and F. Z. Peng, "An effective control method for three-phase quasi-Z-source cascaded multilevel inverter based grid-tie photovoltaic power system," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6794-6802, 2014.

[94] X. Yuan, J. Chai, and Y. Li, "A transformer-less high-power converter for large permanent magnet wind generator systems," *IEEE Transactions on Sustainable Energy*, vol. 3, no. 3, pp. 318-329, 2012.

[95] Y. Liu, B. Ge, H. Abu-Rub, and F. Z. Peng, "Control system design of battery-assisted quasi-Z-source inverter for grid-tie photovoltaic power generation," *IEEE Trans. Sustain. Energy*, vol. 4, no. 4, pp. 994-1001, 2013.

[96] H. Abu-Rub, A. Iqbal, S. M. Ahmed, F. Z. Peng, Y. Li, and G. Baoming, "Quasi-Z-source inverter-based photovoltaic generation system with maximum power tracking control using ANFIS," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 11-20, 2013.

[97] B. Ge, F. Z. Peng, H. Abu-Rub, F. J. Ferreira, and A. T. de Almeida, "Novel energy stored single-stage photovoltaic power system with constant DC-link peak voltage," *IEEE Transactions on Sustainable Energy*, vol. 5, no. 1, pp. 28-36, 2014.

[98] U. K. Shinde, S. G. Kadwane, S. Gawande, M. J. B. Reddy, and D. Mohanta, "Sliding mode control of single-phase grid-connected quasi-Z-source inverter," *IEEE Access*, vol. 5, pp. 10232-10240, 2017.

[99] B. Ge, Y. Liu, H. Abu-Rub, and F. Z. Peng, "State-of-Charge Balancing Control for a Battery-Energy-Stored Quasi-Z-Source Cascaded-Multilevel-Inverter-Based Photovoltaic Power System," *IEEE Trans. Ind. Electron.,* vol. 65, no. 3, pp. 2268-2279, 2018.

[100] A. Ayad, P. Karamanakos, and R. Kennel, "Direct model predictive current control strategy of quasi-Z-source inverters," *IEEE Transactions on Power Electronics,* vol. 32, no. 7, pp. 5786-5801, 2017.

[101] Y. Shan, J. Hu, K. W. Chan, Q. Fu, and J. M. Guerrero, "Model Predictive Control of Bidirectional DC-DC Converters and AC/DC Interlinking Converters - A New Control Method for PV-Wind-Battery Microgrids," *IEEE Trans. Sustain. Energy,* pp. 1-1, 2018.

[102] D. Sun, B. Ge, W. Liang, H. Abu-Rub, and F. Z. Peng, "An energy stored quasi-Z-source cascade multilevel inverter-based photovoltaic power generation system," *IEEE Trans. Ind. Electron.,* vol. 62, no. 9, pp. 5458-5467, 2015.

[103] D. Sun *et al.*, "Modeling, Impedance Design, and Efficiency Analysis of Quasi-Z Source Module in Cascaded Multilevel Photovoltaic Power System," *IEEE Trans. Ind. Electron.,* vol. 61, no. 11, pp. 6108-6117, 2014.

[104] M. Aleenejad and R. Ahmadi, "Fault-tolerant multilevel cascaded H-bridge inverter using impedance-sourced network," *IET Power Electronics,* vol. 9, no. 11, pp. 2186-2195, 2016.

[105] M. Trabelsi, S. Bayhan, H. Abu-Rub, L. Ben-Brahim, and P. Zanchetta, "Finite control set model predictive control for grid-tied quasi-Z-source based multilevel inverter," in *IEEE ICIT*, Taipei, Taiwan, pp. 299-304, 2016.

[106] C.-C. Hua, C.-W. Wu, and C.-W. Chuang, "A digital predictive current control with improved sampled inductor current for cascaded inverters," *IEEE Transactions on Industrial Electronics,* vol. 56, no. 5, pp. 1718-1726, 2009.

[107] M. A. Perez, P. Cortés, and J. Rodríguez, "Predictive control algorithm technique for multilevel asymmetric cascaded H-bridge inverters," *IEEE Transactions on Industrial Electronics,* vol. 55, no. 12, pp. 4354-4361, 2008.

[108] C.-Y. Tang, Y.-T. Chen, and Y.-M. Chen, "PV power system with multi-mode operation and low-voltage ride-through capability," *IEEE Trans. Ind. Electron.,* vol. 62, no. 12, pp. 7524-7533, 2015.

[109] I. F II, "IEEE recommended practices and requirements for harmonic control in electrical power systems," *New York, NY, USA,* 1993.

[110] A. Y. Fard, M. Easley, G. T. Amariucai, M. B. Shadmand, and H. Abu-Rub, "Cybersecurity Analytics using Smart Inverters in Power Distribution System: Proactive Intrusion Detection and Corrective Control Framework," in *IEEE International Symposium on Technologies for Homeland Security (HST),* 2019, pp. 1-6.

Appendix A - Generalized Script to Produce Lookup Matrix for Proposed Optimal Control of Cascaded H-Bridge Inverter

In Chapter 3 of this thesis, it was merely explained how the proposed lookup matrix could be used for online optimization. However, no guidance was given in how such a lookup matrix can be constructed. I will now offer this. A generalized algorithm will for producing the matrix will not be provided; there are countless methods to produce this matrix, and to provide an algorithm ambiguous enough to encapsulate all possible methods will not be enlightening. Additionally, providing a more detailed algorithm that assumes structural decisions for the programmer will undermine the fact that countless methodologies exist, some of which can improve on the speed, simplicity, and/or memory usage of my method. For these reasons, I have decided to show exclusively the way I have constructed the lookup matrix. In doing so, it will provide guidance into how such a matrix *can* be created, without suggesting that the reader bound themselves to my structural decisions.

When implementing the experimental tests for Chapter 3, the provided script was ran in order to generate the lookup matrix in my workspace. Then, the matrix was manually defined within a Simulink function. The script was ran offline to prevent overrunning of the dSPACE CP1103 in real time implementation at the first iteration. Recall that the lookup matrix contains a list of redundant, optimal switching sequences at each matrix address. A matrix address is specified by the optimal output voltage level M_k, found from (3.3), as well as the previously implemented switching sequence s_k. The row number of the matrix address is found from M_k, and s_k is equal to the column number. The switching sequences within the list will apply M_k in minimal switching events, with respect to s_k.

The Matlab script is provided below:

```
% Script to produce lookup matrix for proposed optimal controller using
% Matlab 2018a. The following script is written generally for any number of
% H-bridges where "cells" is the variable denoting the number of H-bridges
% in the CMI
%
%%%%%%%%%%%%%%%%%%%%%%%%%% Mitchell Easley %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%% Originally produced January 4, 2019 %%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%% Comments added for thesis July 21, 2020 %%%%%%%%%%%%%%%%%

clear %clear workspace (recommended but not always necessary)
```

```
cells = 4; % define number of cells or H-bridges in CMI

% define control set "s"
% The desired control set lists the logic value of the upper switches in
% the CMI. The most significant bit is S_a, then S_b, etc.

for i = 1:4^cells % for each switching sequence (4^cells)

    v = de2bi(i-1,2*cells); % converts a decimal value into its binary
                            % equivalent to specify the gate signals
    % The input of de2bi is the decimal value (i-1) and the desired number
    % of elements in the output vector. For a N-cell CMI, there will be 2N
    % upper switches. Thus, the output vector must be 2N elements long

for k = 1:2*cells %second, inner for loop, reshaping the de2bi vector

    s(2*cells-k+1,i) = v(k); %this line merely transposes and reverses the
    % order of the values, and collects them in the control set "s"
    % Now, the value of S_a is in row one, S_b in row 2, etc. The column
    % specifies the switching sequence. Switching sequence one is in column
    % one, switching sequence two is in column two, etc.

    % It is not necessary to adhere to this control set structure, but the
    % script below would need to be adjusted according to the desired
    % structure
end % end of inner for loop
end % end of outer for loop

% Now, the entire control set has been collected and stored in "s"
% The next step is to build the first two layers of the lookup matrix
% The third dimension of the lookup matrix is referred to as the "layer"
% Recall from Chapter 3, the value in the first layer indicates the length
% of the list at its matrix address (number of layers). The second layer
% indicates the layer at which the control should select at the next
% address call. Thus, layer two is referred to as "the pointer layer"

% It should be noted that, during construction of the lookup matrix, an
% additional layer is added temporarily. This was done to document the
% voltage level that each switching sequence in the list implements. This
% layer will be removed once the matrix is constructed entirely. This layer
% is temporarily placed as layer one. Layer one from chapter 3 is
% temporarily layer two here, and layer two from chapter 3 is temporarily
% layer three here. This temporary layer could be eliminated entirely by
% creating an array which has the voltage level for each switching sequence
% and using it instead of the temporary first layer(this would in fact be
% more memory-efficient)

for i = 1: 4^cells % for each switching sequence
    for ii = 1:(2*cells+1) % for each possible M_k

        % the script will determine the voltage level of the associated
        % switching sequence s_k. The control goes through all rows for a
        % particular switching sequence in "s". If an odd-numbered row
        % contains a logic-high, a voltage level is added. If an
        % even-numbered row is logic-high, a voltage level is decremented.
        % The reader is encouraged to check for themselves that this is a
        % valid way to determine the output voltage level of a particular
```

```
        % switching sequence.

        level = 0;
        for k = 1:(2*cells)
            level = level + (-1)^(k+1)*s(k,i);
        end

        Mat(ii,i,1) = level; % Layer one is set to the output voltage level
        % of switching sequence i. Recall that ii is a variable denoting
        % the desired output voltage level M_k at the next state

        Mat(ii,i,3) = 4; % Each pointer starts at four, as the lower layers
        % do not contain switching sequences. Thus, the pointer layer
        % points to layer four, which currently does not exist

        Mat(ii,i,2) = 3; % Layer two (which will be layer one eventually)
        % is set as three, as all matrix addresses currently only contain
        % three layers
    end
end

% The following lines of code fill the lookup matrix with optimal switching
% sequences. All switching sequences contained within a matrix address are
% optimal and redundant with respect to output current tracking and
% switching events. That is, all switching sequences in a list will
% implement M_k in minimal switching events with respect to s_k.
% The following incrementing variables are defined below:
%
% ii: denotes the row of the matrix address where switching sequences are
% being placed
%
% i: denotes the column number (or switching sequence s_k) of the matrix
% address where switching sequences are being placed
%
% k: denotes a "candidate switching sequence" which may get placed as an
% optimal switching sequence at matrix address (ii,i)
%
% m: an incrementing variable to determine the output voltage level of the
% candidate switching sequence k
%
% Below, the script goes through all of the two-dimensional matrix
% addresses. It iterates through all switching sequences for a particular
% row of the matrix address, then moves to the next matrix address. This
% means, it considers the optimal output voltage level at -1*cells for each
% switching sequence, then considers the optimal output voltage level
% (-1*cell + 1), and continues to (1*cells). This defines the first for
% loop and its first nested for loop. In the third for loop, a candidate
% switching sequence is defined, and ultimately evaluated. In the fourth and
% final for loop, the output voltage of the candidate switching sequence is
% determined (using the technique from earlier).
%
% To be included in the list of optimal switching sequences at matrix
% address (ii,i), the candidate switching sequence k must meet two
% criteria:
%
% 1) Does switching sequence k produce the desired output voltage? This is
% checked by seeing if level+cells+1 = ii. This is because M_k cannot equal
```

```
% ii, as M_k includes negative integers, whereas ii (denoting row number)
% must consist of natural numbers in Matlab.
%
% 2) Does candidate switching sequence k produce the desired output voltage
% level in minimal switching events with respect to switching sequence i?
% This is written concisely in the script as:
%   if(abs(level - Mat(ii,i,1)) == sum(abs(s(:,k) - s(:,i))))
% This line checks the difference in output voltage level between switching
% sequence i and candidate switching sequence k. The minimum number of
% H-bridge leg inversions is equal to this difference. That is, to
% increment or decrement the output voltage level by one, this can be done
% in minimal switching events by inverting one value in control set "s"
%
% If both these criteria are met, candidate switching sequence k is
% considered optimal, and is included in the list at matrix address (ii,i).
% It is then added directly above the highest layer at the matrix address,
% and layer two is incremented to account for the additional layer


for ii = 1:(2*cells+1) % row number of matrix address
    for i = 1:4^cells % column number (switching sequence) of matrix
        % adress
        for k = 1:4^cells % considering all switching sequences for list of
            % optimality at matrix address
            level = 0; % set level to zero, then determine using final
            % nested for loop using previously described technique
            for m = 1:(2*cells)
                level = level + ((-1)^(m+1))*s(m,k);
            end
            if(level+(cells+1) == ii) % checking criterion 1
                if(abs(level - Mat(ii,i,1)) == sum(abs(s(:,k) - s(:,i))))
                    % if criterion 1 is met, check criterion 2
                    Mat(ii,i,Mat(ii,i,2) + 1) = k; % if both criteria are
                    % met, add candidate switching sequence k to list of
                    % optimal switching sequences at matrix address
                    Mat(ii,i,2) = Mat(ii,i,2) + 1; % increment size of list
                end % end of criterion 2 if
            end % end of criterion 1 if
        end % end of third for loop (establishes candidate switch. seq. k)
    end % end of second for loop (establishes matrix address column)
end % end of first for loop (establishes matrix address row)


Mat = Mat(:,:,2:length(Mat(1,1,:))); % Eliminating the temporary first
% layer
Mat(:,:,1) = Mat(:,:,1) - 1; % decrementing the new layer one, which
% specifies the number of layers at the associated matrix address

Mat(:,:,2) = Mat(:,:,2) - 1; % decrementing the new layer two (the pointer
% layer, which specifies the layer which contains the value of the next-
% up switching sequence
```