Finite-Control-Set Model Predictive Control for DC-DC Converters

by

Zhengchen Guo

A dissertation submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Auburn, Alabama
May 2, 2026

Keywords: Power Electronics, DC-DC Converters, Model Predictive Control (MPC).

Copyright 2026 by Zhengchen Guo

Approved by

Robert Mark Nelms, Chair, Godbold Professor of Electrical and Computer Engineering
Eduard Muljadi, Danaher Professor of Electrical and Computer Engineering
Mark Halpin, Alabama Power Distinguished Professor of Electrical and Computer Engineering
John Y. Hung, Professor Emeritus of Electrical and Computer Engineering

Abstract

This dissertation develops novel solutions for Finite-Control-Set Model Predictive Control (FCS-MPC) applied to DC-DC power electronic converters. While MPC has shown promise in diverse power electronics applications, its wider adoption is often hindered by two critical challenges: high computational intensity and limited robustness against system variations. This research directly confronts these issues through a multi-faceted approach.

First, to address the computational bottleneck, a novel FPGA-based hardware acceleration framework is developed, enabling real-time, embedded FCS-MPC operation for power electronics control. This foundational work facilitates the implementation of advanced control algorithms. Subsequently, a unified MPC strategy is introduced to simultaneously regulate both output voltage and inductor current within a single control loop. To enhance the robustness of this controller towards variations, two distinct strategies are proposed: one employing an online adaptive weighting factor to dynamically adjust control priorities, and another utilizing a re-formulated cost function for implicit current regulation.

The effectiveness and practical viability of all proposed methods are rigorously confirmed through extensive simulation and experimental validation.

Artificial Intelligence (AI) Use Disclosure Statement

In the preparation of this dissertation, the following Artificial Intelligence (AI) tools were used: ChatGPT and Gemini. These tools were used primarily to literature search and review. The author acknowledges full responsibility for the intellectual content of this work and has ensured that all AI-assisted sections have been reviewed and revised for accuracy and appropriate academic style. All AI-generated content was reviewed and validated for relevance, appropriateness, and accuracy before incorporation into the final document to maintain scholarly integrity of this research.

Digital Accessibility Disclosure Statement

In the preparation of this dissertation, the following digital accessibility tools were used to ensure this document complies with federal requirements: Adobe Acrobat Pro Accessibility Checker. The author acknowledges full responsibility for the intellectual content of this work and has made a good faith effort to comply with digital accessibility requirements in publishing, wherein the nature of the content does not significantly change in order to do so. Furthermore, all content has been reviewed and revised to meet these requirements prior to final publication.

Acknowledgments

虽然 dissertation 早早就完成了，但 acknowledgment 这一部分打开了好几次却迟迟没有动笔。拖到最后要离开美国的这一天前还是做个小的总结。

从 2019 年满怀希冀的来到这里，到 2026 年选择离开，中间发生了太多没有预料到的事，对这个世界，对自己。命运安排的路足够曲折，确实要感谢这一路很多人的帮助，似乎少了哪一个人我都无法走到这里。

读博的这段生涯，是把我彻底碾碎的过程，最大的收获是破除了我执。《金刚经》里有言：佛告须菩提，诸菩萨摩诃萨，应如是降伏其心。所有一切众生之类，若卵生、若胎生、若湿生、若化生、若有色、若无色、若有想、若无想、若非有想、非无想，我皆令入无余涅槃而灭度之；如是灭度无量无数无边众生，实无众生得灭度者。何以故？须菩提！若菩萨有我相、人相、众生相、寿者相，即非菩萨。

这六年多的时光，或加上出生以来的所有人生，我都执着于这四相。生活需要支柱，我需要健康、金钱、学位/论文还有情感来作为我生命的支撑。我曾经认为自己是很有韧性的人，只要还留有一根柱子，只要不把我完全打倒，我都会继续生存下去。可惜天地不仁，以万物为刍狗。这些年我的这些自以为是的支柱相继倒塌，竟然一个不剩。我站在一片废墟之上，茫然看着四周倒塌的一切，那一刻的感受竟是非常奇妙的，甚至可以感觉自己能听到空旷的天地间荡漾的回声。是的，一切都完蛋了，一切都毁了，但我还活着，心脏还在跳动。那一刻我才意识到，生命不需要什么支柱。身体的健康与疼痛不随我的意志而控制，只能接纳和共处。我不需要金钱来提供安全感，住在大房子里和住在车里，睡在街头，没有本质区别。我不需要论文发表和博士学位来证明自己的头脑和意志，更不需要别人爱我来证明自己的价值。我根本不曾拥有和掌握过什么人和什么事，过去心不可得，现在心不可得，未来心不可得。安全感本身就是一个伪命题，指望外界的人事物来提供根本性的安全与舒适本就不可能，莫向外求似乎是一个很好的策略。

所谓山不转路转，路不转人转，人不转心转。因为外界的路基本都被堵死，这几年我被迫只能向内探索，或者说向内求。这仍然不是一条容易的路。我对自己要求非常严苛，每天作息要早睡早起，运动饮食精益求精，多刷会儿手机，或者工作的时候走神分心我都会责怪自己，这让我焦虑和痛苦，面目全非。我开始考虑放过自己，允许自己吃很多甜食，允许熬夜玩手机，放任自己发胖。向内求也是行不通的，人无法一味的对抗熵增。

不管是向外求还是向内求，只要我有了“求”的心态，即是执着于我、人、众生、寿者，痛苦纷至沓来。《心经》里说到：以无所得故，菩提萨埵，依般若波罗蜜多故，心无挂碍。无挂碍故，无有恐怖，远离颠倒梦想，究竟涅槃。这句话长期在我脑中回响。拿起放下，拥有失去，出生死亡，都是阴阳对应的产物。“求”就会产生“得”与“不得”。

我即天地，即是人、众生、寿者，空旷的废墟，或着说“空”才是永恒。

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Over the past few decades, the field of power electronic converters and motor drive control technologies has witnessed remarkable progress, significantly influencing a broad spectrum of modern applications. These applications span across various sectors such as renewable energy systems, communication infrastructures, and transportation networks, all of which increasingly depend on efficient and reliable power conversion and control systems. Within this expansive domain, converter control methodologies have emerged as one of the most actively researched areas in power electronics. This research encompasses a wide range of converter topologies designed to operate at low, medium, and high power levels, reflecting the diverse demands of modern electrical systems [1].

The design and development of control strategies for power converters have been comprehensive, involving the integration of classical linear control methods with modulation strategies, such as pulse width modulation (PWM). At the same time, researchers have explored advanced nonlinear control techniques to further improve system performance, robustness, and efficiency [2]. These efforts have resulted in a wide array of control approaches, tailored to meet the specific needs of various power conversion applications.

This evolution has been strongly supported by continuous advancements in digital signal processing (DSP) and embedded hardware technologies, which have opened the door to real-time implementation of sophisticated control algorithms. As a result, a variety of intelligent and adaptive control strategies—such as fuzzy logic control, adaptive control, sliding mode control, and model predictive control (MPC)—have been successfully applied in power electronics systems [3, 4, 5]. Among these, predictive control methods stand out for their forward-looking nature: they estimate future system behavior and select control

actions based on predicted outcomes. A specific type of predictive controller, known as deadbeat control, eliminates the need for conventional feedback loops by directly computing the required control inputs from a system model to bring the state to its reference in minimum time [6].

Model Predictive Control (MPC) offers another powerful framework, relying on a mathematical model of the system to predict its future states over a finite horizon [7]. The architecture of general MPC for power electronic converters is shown in the Fig.1.1.

Within this architecture, a predictive model is crucial for forecasting the system's future behavior based on collected data from the physical system. Various methodologies exist for developing this model, primarily falling into physics-based and data-driven approaches [8, 9]. To project further into the future, the predictive model can be executed iteratively over multiple steps. For instance, as illustrated in Fig.1.1, the model might run N times to predict system behavior across an N-step prediction horizon.

Subsequently, these predicted system behaviors are fed into a cost function module. This cost function serves to mathematically quantify control objectives, such as minimizing errors, control effort, or switching frequency. By evaluating different potential sequences of control actions over the prediction horizon N, the controller selects the sequence that yields the minimum cost. Model Predictive Control enhances robustness and flexibility through its receding horizon policy [10]. Instead of implementing the entire calculated control sequence, only the initial control action of the optimal sequence is applied to the system.

MPC was initially developed for use in the process control industry, where it proved to be highly effective in managing systems characterized by nonlinear behavior, multiple inputs and outputs, and complex operating constraints [11]. Given its natural compatibility with discrete-time operation and its capacity to manage constraints, MPC has since been widely adapted for power electronics. It has demonstrated its effectiveness in numerous applications, including electric drives, static synchronous compensators (STATCOMs), high-voltage direct

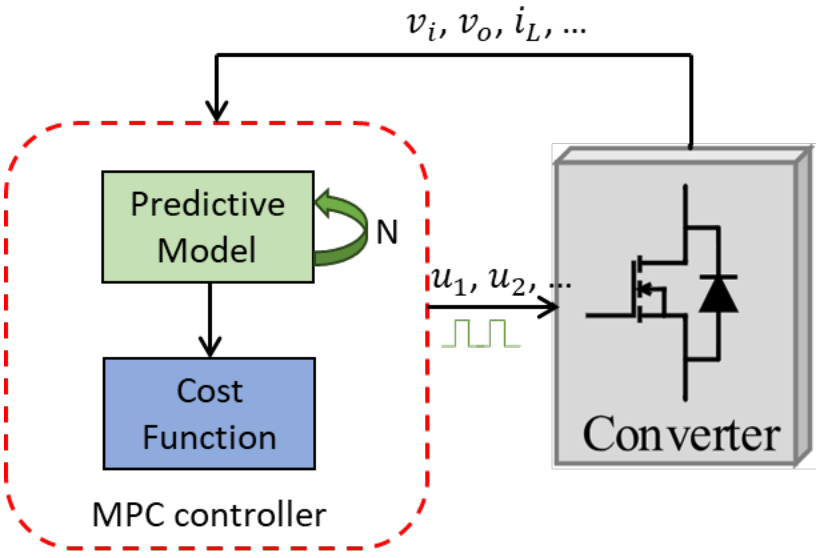


Figure 1.1: Block diagram of MPC for power electronic converters.

current (HVDC) transmission systems, and flexible alternating current transmission systems (FACTS), among others [12].

MPC has gained significant traction in power electronics over the past decade. Its ability to handle multiple variables and constraints intuitively, despite a high computational burden, makes it a powerful alternative to traditional linear controllers. Researchers have applied MPC across various power conversion systems –from grid-connected inverters and multilevel converters to motor drives and renewable energy interfaces.

In this introduction, key developments in MPC for power electronics is summarized and organized by major application areas. For each area, core MPC methods and contributions are outlined, and benefits and limitations are discussed in context, and notable trends over time are highlighted.

1.1 Power Converters and Inverters

Power converters (including DC-AC inverters and DC-DC converters) have been an active testbed for MPC techniques. Early works demonstrated that MPC could directly control converter switching to regulate currents and voltages with fast dynamic response

and constraint enforcement [13]. A seminal example is the predictive control of an Active Front End (AFE) rectifier, where MPC achieved tight DC voltage control and power factor correction by optimizing the converter' s switching state each sample instant [14]. The ability to handle multi-variable control objectives (e.g. current tracking and neutral-point balancing in multilevel converters) via a single cost function is a core advantage of MPC [15]. For instance, in Modular Multilevel Converters (MMCs), MPC naturally manages capacitor voltage balancing alongside output current control [15]. Researchers have noted that multiple control objectives are achieved easily through the MPC cost function, without requiring cascaded loops [15].

In converter applications, MPC offers fast transient response and intuitive multi-objective control. It can minimize error and transient overshoot in one control step, often outperforming PID regulators in dynamic scenarios [16]. MPC can also explicitly handle constraints such as current limits or voltage bounds, enhancing system protection. Notably, finite-control-set MPC (FCS-MPC) –where the controller directly selects among discrete switching states – has been favored in power converters for its simplicity (no modulator needed) [16]. This approach has shown excellent performance in tracking references and balancing internal states (e.g. capacitor voltages in multilevel inverters) even under fast-changing conditions.

Key challenges for MPC in converters include the need for an accurate model and the computational complexity of online optimization [17]. Evaluating a cost for all possible switching states (which grows combinatorially in multilevel converters) can tax digital controllers, especially at high switching frequencies [18]. Furthermore, basic FCS-MPC results in a variable switching frequency, since the optimal switch choice may change erratically, potentially causing uneven spectra and higher filtering requirements [19]. Another practical issue is tuning the cost function weights when multiple objectives (e.g. current error vs. voltage balancing) are present [20]. Over the decade, researchers addressed these issues by proposing modulated MPC and longer-horizon MPC. In modulated or continuous-control MPC, a PWM modulator is used with the MPC law to ensure a fixed switching frequency

[21]. For example, Moon et al. (2017) combined MPC with space-vector modulation to significantly reduce current ripple in a grid-tied inverter while lowering computational load [22]. Meanwhile, long-horizon MPC techniques were introduced to improve steady-state behavior: by optimizing over multiple future steps (using algorithms like sphere decoding to limit complexity), these approaches achieved lower harmonic distortion and more predictable switching patterns at the cost of higher computation [23]. Overall, the trend has been to make MPC more practical for industry by taming its computational demands and irregular switching; this is evidenced by many works on simplified algorithms, parallel processing, and weight auto-tuning in the late 2010s.

Early demonstrations around 2010 applied MPC to two-level converters and simple loads, proving feasibility. Over time, focus shifted to high-power and complex converters (e.g. multilevel NPC, flying capacitor, MMC) where MPC' s multi-variable strength is most needed. By the mid-2010s, numerous papers addressed MMC control via MPC, culminating in overview studies that cataloged challenges (model accuracy, computation, weighting) and proposed remedies [24]. There has also been a clear trend toward hybrid strategies –such as incorporating PWM, using multi-step prediction horizons, or adopting adaptive sampling –to improve the trade-off between dynamic performance and switching losses. For grid-connected inverters, recent research emphasizes ensuring grid code compliance (e.g. low THD, fast fault response) using MPC. For example, Alharbi et al. (2025) review how MPC-based PV inverters can be designed for stability and compliance, noting that continuous-control MPC can maintain a fixed switching frequency but requires careful design of the modulator and inner loops [14]. In summary, the last decade saw MPC evolve from an academic concept to a maturing control option for converters, with ongoing improvements targeting its remaining limitations.

1.2 Electric Drives (Motor Control)

Electric motor drives–including permanent magnet synchronous motors (PMSMs) and induction motors –represent another major area where MPC has been extensively explored. In drive control, the goal is typically to regulate torque and flux (or current) in an AC machine fed by a power electronic inverter. MPC has been applied as an alternative to traditional field-oriented control (FOC) and direct torque control (DTC), offering the promise of faster torque response and better multi-phase current regulation [25]. Notably, finite-control-set MPC can directly choose inverter switch states to drive stator currents, effectively replacing the PI current controllers and space-vector PWM of a conventional drive. This direct approach yields a very fast current loop (one control step delay) and naturally handles constraints like current saturation or inverter voltage limits.

Early work in the 2010s applied MPC for current control of induction machines and PMSMs, demonstrating dead-beat torque control performance. For example, Bolognani et al. showed that a simple FCS-MPC could control an induction motor' s torque and flux with a dynamic response superior to classic methods [26]. A key development for drives was the introduction of long-horizon MPC to improve steady-state current quality. Geyer and others presented multi-step predictive control for drives that optimizes a sequence of switch commands over several future intervals [27]. This drastically reduced torque ripple and harmonic distortion compared to one-step MPC or DTC, approaching the performance of optimized pulse patterns. Another stream of research looked at maintaining a constant switching frequency in drives: methods like optimized duty cycle control or incorporating a carrier were proposed to eliminate the variable switching rate inherent to basic FCS-MPC [28]. By 2015–2020, advanced schemes emerged such as model predictive pulse pattern control (for medium-voltage drives), and modulated predictive control for low-voltage drives, each aiming to harness MPC' s dynamic advantages while improving harmonic performance [29].

In motor drives, MPC enables precise, rapid control of torque and current, which is crucial for high-performance applications (e.g. traction, servo drives). Because MPC inherently manages multiple inputs/outputs, one controller can regulate torque, flux, and even other aspects like neutral point voltage or thermal limits simultaneously. Constraint handling is a strong advantage –MPC can respect inverter current limits or avoid overmodulation automatically, improving reliability. Additionally, MPC can be formulated to reduce common issues like torque ripples by penalizing switching or using a horizon that finds an optimal switching sequence that minimizes distortion. The flexibility of the cost function design allowed researchers to introduce custom control objectives (for example minimizing a motor' s copper losses or vibration) into the controller, something not easily done with fixed structure PI controllers.

Despite its promise, MPC in drives faces challenges. Accurate knowledge of motor parameters (inductances, EMF constants) is required for the predictive model and deviations (e.g. due to temperature or saturation) can degrade performance. Some works have addressed this via robust or adaptive MPC to tolerate parameter variation [30]. Computational demand is also a factor: drive control typically runs at high sampling rates (tens of kHz) for fast dynamics, so executing an optimization (even a simple one) every 50–100 μs is challenging. This has been partially mitigated by using FPGAs or efficient coding, and by reducing the candidate switching set (e.g. evaluating only a subset of voltage vectors) [31]. Another limitation is switching frequency control: basic MPC may lead to an uneven switching pattern that can excite mechanical resonances or complicate filtering in drives. Finally, the need to tune weighting factors in multi-objective drive MPC (e.g. torque error vs. switching penalty) has been a practical hurdle; recent approaches using auto-tuning (e.g. Bayesian optimization) have started to address this to make MPC more user-friendly [32].

Over the decade, there has been a clear trend from theoretical feasibility to practical implementation in MPC motor drive. Early studies (2010–2013) showed that even a one-step predictive controller could replicate the function of cascaded PI controllers and hysteresis

comparators, simplifying the control structure [25]. Mid-decade, focus shifted to refining performance: long-horizon MPC (2014–2016) demonstrated near-optimal current waveforms, and methods for constant switching and robustness came to the fore around 2017–2020 [27]. Multi-phase drives (e.g. five-phase motors) and multi-motor systems were also tackled with MPC, showing the method's scalability to more complex drive configurations (some achieving fault-tolerant control by predictive modulation of healthy phases) [33]. Towards the end of the decade, integration of estimation and prediction became important: e.g. sensorless MPC drives that include a state observer to estimate rotor position and speed, and model-free predictive control where the motor model is updated from data in real-time [34]. These developments indicate that MPC is moving closer to real-world drive applications, especially in high-performance domains where its benefits justify the complexity.

1.3 Renewable Energy Integration (PV and Wind Systems)

MPC has found fertile ground in renewable energy systems –notably in photovoltaic (PV) power converters and wind energy conversion systems –where the ability to handle variability and constraints is crucial. In grid-tied PV systems, MPC is used to control inverters (and sometimes DC/DC converters) to ensure maximum power extraction and compliance with grid requirements. In wind turbines, MPC has been applied for turbine blade pitch control and for managing the generator-side and grid-side converters to optimize power and reduce mechanical stress.

In PV integration, MPC controllers manage the DC-link and AC output to maximize power point tracking (MPPT) while regulating grid injection. A common setup is a DC-DC boost converter feeding a DC-AC inverter; MPC can coordinate both. For example, one study introduced an MPC scheme that simultaneously controlled the PV array voltage (for MPPT) and the grid inverter current, achieving fast MPPT and low grid current THD without any PI regulators [35]. An advantage of MPC here is the ability to incorporate forecast or estimation –e.g. anticipating irradiance changes or grid voltage sags. Recent works have

shown MPC-based PV inverters that support grid services: during normal operation they inject maximum active power, but if a grid fault (voltage sag) occurs, the MPC seamlessly shifts to provide reactive power support for low-voltage ride-through [36]. The controller does this by recognizing constraints (grid code mandates) and adjusting its cost function priorities in real time. Overall, MPC in PV systems has demonstrated fast response to transients (e.g. irradiance steps or grid disturbances) and the flexibility to handle multiple modes of operation [37].

MPC has been applied to wind turbines primarily in the turbine control layer (blade pitch and generator torque) and in the power electronics (converter control). A prominent application is individual blade pitch control of large wind turbines. By using MPC to collectively and individually adjust blade pitch angles, researchers achieved better load mitigation on the blades and tower while respecting actuator constraints [38]. Petrovic et al. (2019) developed an MPC framework that combined collective and individual pitch control with turbine operating constraints; it successfully reduced fatigue loads on the structure compared to classical controllers, illustrating MPC' s multi-input, multi-output strength in a highly nonlinear system [39]. On the electrical side, the generator-side converter of a wind turbine (often a full-scale or partial-scale converter) has been controlled with MPC to optimize power capture and smooth torque. MPC can handle the torque/speed constraints during high winds and enforce a soft cut-out by pitching or limiting power, all while maximizing energy capture in normal conditions. In floating offshore wind turbines, nonlinear MPC has been investigated to coordinate blade pitch and generator torque to maintain stability of the platform –a complex task due to coupling of mechanical and electrical dynamics [40].

Renewable energy applications introduce uncertainty (e.g. unknown future wind or solar input) which can challenge MPC performance. MPC controllers require some form of prediction of these disturbances; errors in forecasting wind or irradiance can degrade optimality. Robust or stochastic MPC approaches are being studied to cope with this. Another limitation is the potential complexity of the models: wind turbines are highly

nonlinear and have varying dynamics with operating point, so linear MPC designs (e.g. LPV models [41]) must carefully cover the range or risk instability. Computational effort is a concern mainly for fast inner-loop controls in power converters (similar to drives), but less so for slower outer-loop control (pitch control runs at 5-10 Hz where MPC optimization is easily handled). Still, implementing MPC in a turbine controller or PV inverter controller requires validation and failsafe measures, as these are safety-critical systems. So far, many MPC schemes for renewables remain in simulation or lab prototype stage. Also, economic MPC (optimizing not just control errors but also yield/cost) for renewables is a growing area –but it introduces multi-objective trade-offs that complicate tuning [42].

In PV systems, early research (2010-2015) focused on using MPC for basic current control and DC-link regulation, often demonstrating lower current distortion and faster MPPT than conventional methods. Around 2015-2020, as grid codes demanded more functionality (like reactive power support, fault ride-through), MPC strategies evolved to incorporate these: e.g. MPC that can seamlessly transition between maximum power and voltage support mode during grid disturbances [43]. There has also been a push towards two-stage PV MPC (DC/DC plus DC/AC) and coordinating them for optimal performance. In wind energy, MPC research initially (circa 2010) looked at generator-side control for maximizing power and limiting torque. Then attention shifted to advanced turbine control –individual pitch control, floating platform stabilization, etc, where MPC' s ability to juggle multiple constraints (e.g. maintaining platform angle, reducing blade loads, controlling power) is invaluable. A recent trend is incorporating fatigue prediction into MPC cost functions (so-called fatigue-oriented MPC [38]), effectively trading off short-term performance for long-term structural health of the turbine. Additionally, distributed MPC for wind farm control (coordinating multiple turbines to mitigate wake effects) and for PV farms (coordinating PV inverters for grid support) has emerged in the last few years, indicating a broadening scope of MPC from single device level to system-wide optimization in renewable energy integration.

1.4 Microgrids and Distributed Energy Systems

In microgrids –localized grids with distributed generation (PV, wind, batteries, etc.) that can operate independently or connected to the main grid –MPC has emerged as a promising control and energy management strategy. Microgrids involve hierarchical control: primary (voltage/frequency regulation), secondary (power sharing, economic dispatch), and tertiary (optimizing power flow with the main grid). MPC has been explored at all these levels due to its ability to handle the microgrid' s coupled, multi-timescale dynamics and constraints.

At the primary control level, MPC is used for the fast inner control of inverters that interface distributed energy resources. For example, in an islanded microgrid, an MPC-based voltage source inverter can regulate the load voltage and frequency while sharing load with other inverters. Unlike droop control, MPC can enforce voltage constraints and respond rapidly to load changes [44]. It can also handle nonlinear converter dynamics (like saturation or bandwidth limits) more effectively. Research has shown that MPC-controlled inverters can maintain stable microgrid operation under sudden load or generation changes, and easily incorporate features like virtual impedance or harmonic compensation by appropriate cost function design [45]. An MPC can also manage a hybrid system inverter (like a PV connected with battery inverter) by controlling both grid-forming and grid-feeding modes as conditions dictate.

At the secondary control layer, MPC has been employed for microgrid energy management –deciding how much power each source should generate and each storage should charge/discharge to meet demand optimally. Here, MPC typically runs on a slower timescale (seconds to minutes) using a prediction horizon of minutes to hours. It can take into account forecasts of load and renewable production, electricity prices, and battery state of charge, then compute an optimal dispatch that minimizes operating cost or maximizes efficiency, subject to constraints (e.g. battery charging limits, required spinning reserve). For instance, one study used MPC to optimally schedule a PV-battery microgrid, maintaining

stable operation under variable generation and load while extending battery life by preventing overcycling [37]. The fast transient response and constraint handling of MPC make it suitable to manage the often rapid fluctuations in a microgrid [46].

MPC in microgrids offers unified, optimal control across different assets. It excels at coordinating multiple distributed resources –for example, ensuring a battery smooths the PV output while a diesel generator only kicks in when absolutely needed, maintaining both power balance and fuel efficiency. MPC' s ability to look ahead using load/renewable forecasts can significantly improve microgrid economics by, say, pre-charging a battery in anticipation of a cloud passing over a PV array. This predictive dispatch leads to more stable voltage and frequency in the microgrid compared to reactive approaches [46]. Moreover, MPC readily handles the transition between grid-connected and islanded modes by enforcing synchronization constraints and adjusting reference targets during mode transfer, improving reliability of microgrid operation. Another advantage is the inclusion of network constraints: for microgrids with multiple buses, MPC can consider line thermal limits or voltage limits at different nodes, which is complex for decentralized controllers. Studies have shown MPC achieving good voltage regulation and power sharing without need for slow human operator adjustments, underlining its potential as a supervisory controller in microgrids.

The complexity of microgrids means MPC controllers can become computationally heavy, especially at the secondary/tertiary level when optimizing over many devices and a long horizon. Ensuring real-time solvability and convergence is a challenge; however, advances in solvers and the fact that slower control allows more computation time make it feasible in many cases. Another concern is model accuracy and uncertainty: microgrid models involve uncertainties in load, generation forecast errors, and sometimes switching events (like disconnects/faults) [47]. Robust MPC or adaptive MPC approaches are needed to maintain performance when reality deviates from the model. There is also an implementation challenge: deploying MPC in a microgrid requires a reliable communication and measurement infrastructure (for state estimation, sending setpoints, etc.). This introduces

latency and potential communication failures, which MPC must be designed to tolerate (e.g. using distributed or decentralized MPC frameworks that can operate with limited communication). Finally, while MPC can handle multiple objectives (economic vs. technical), setting the correct cost priorities (weights) for, cost-saving vs. battery degradation can be non-trivial and may require iterative tuning or higher-level decision logic [42].

Early uses of MPC in microgrids (2010-2015) were focused on proof-of-concept for inverter control and simple energy management. By the late 2010s, there was enough interest that comprehensive reviews appeared, indicating MPC was at the beginning of application in microgrids but showing huge potential as an alternative to conventional droop and PI-based schemes. In recent years, trends like distributed MPC where multiple controllers (for each inverter or microgrid cluster) solve smaller problems and coordinate via consensus have been considered. This addresses scalability and single point-of-failure issues [47]. Another trend is coupling economic optimization with control (sometimes called EMPC –economic MPC) for microgrids, to directly minimize fuel cost or emissions while keeping the grid stable [48]. Additionally, researchers are looking at resilience: MPC strategies that can handle island events, faults, and uncertainties (like sudden EV charging or plug-in) [49]. For example, Han et al. (2021) proposed an MPC approach for microgrids that explicitly considers the uncertainty of electric vehicle charging demand, maintaining stable operation despite the random charging events by including robust constraints [50]. The literature suggests that as computing becomes cheaper and renewable penetration rises, MPC-based microgrid control is transitioning from theory to practice, with some field demonstrations in advanced microgrid testbeds. The general trajectory is toward hierarchical MPC (different layers handling different timescales and objectives) and ensuring these layers work in harmony. In summary, MPC is poised to play a key role in managing the complexity of future microgrids, offering a systematic way to achieve stability, optimality, and adaptability in distributed energy systems.

1.5 Emerging Control Methods and Trends

Beyond classical MPC implementations, the past few years have seen emerging methods that enhance or radically change how MPC is applied in power electronics. Two notable trends are the use of Koopman operators for MPC and data-driven or learning-based MPC. These approaches aim to deal with nonlinearity and modeling challenges, or to automate the design of MPC controllers using data.

Traditional MPC relies on a model derived from first principles. Data-driven MPC approaches, by contrast, either identify the model from input-output data or even bypass explicit modeling by directly using data in the control law. A prominent development is the Data Enabled Predictive Control (DeePC) technique, which uses a Hankel matrix of past I/O data to predict future outputs without an explicit state-space model. This approach was applied to motor drives, where DeePC achieved predictive current control by utilizing just the measured data from the drive, effectively doing MPC in a model-free way [34]. In another vein, researchers have merged model-free adaptive control (MFAC) with MPC: by embedding MFAC algorithms (which estimate a pseudo-gradient from data) into the FCS-MPC framework, they developed data-driven predictive control (DDPC) schemes that eliminate reliance on knowing motor or converter parameters [51]. Such controllers adjust based on the I/O behavior, making them robust to model uncertainty. However, one challenge noted is handling unknown disturbances and uncertainties –purely data-driven schemes might struggle if the system operating conditions change outside the data they have seen. This controller achieved multi-objective control (currents and common-mode voltage reduction) and was completely independent of system parameters, illustrating the power of marrying data-driven modeling with classical observer design in MPC.

Another facet of learning-based MPC is using machine learning (ML) to assist MPC [52]. This includes training neural networks to approximate the MPC policy or cost-to-go (enabling longer horizons without heavy computation), and using reinforcement learning to tune cost function weights or discover optimal control sequences that MPC can then enforce.

While these approaches are still nascent in power electronics, initial studies (e.g. using deep RL for energy management, or neural nets to predict converter behavior) have shown that learning can enhance MPC performance and adaptability. The annual increase in computational power and available data in power electronics (through IoT and cloud monitoring) makes these data-driven approaches increasingly practical.The convergence of control theory with machine learning is likely to yield MPC controllers that can learn and adapt, providing both the performance of optimal control and the flexibility of data-driven approaches. However, ensuring theoretical guarantees (stability, constraint satisfaction) remains a priority as these methods mature.

1.6 Contributions and Organization of Dissertation

This dissertation advances the field of power electronics control by addressing the significant computational challenges of FCS-MPC and its application for dc-dc converters. The primary contributions, which have been validated through extensive simulation and hardware experiments, are organized as follows.

(a) A Novel FPGA-Based Hardware Acceleration Framework for MPC of Power Electronics Converters (Chapter 2) [53]: To overcome the primary limitation of MPC—its high computational demand—this work develops and validates a novel FPGA-based hardware acceleration framework. This architecture reduces the total MPC processing time to less than $1\mu s$, effectively eliminating the computational bottleneck and enabling real-time embedded implementation. The chapter provides a comprehensive design guideline that establishes a versatile and foundational framework for all subsequent contributions. The following works utilize this hardware acceleration method to realize their control algorithms.

(b) A Unified Single-Loop MPC Strategy for DC-DC Converters (Chapter 3) [54]: This work introduces a unified MPC method that simplifies the control architecture by

using a single control loop to simultaneously regulate both the output voltage and the inductor current. This approach enhances dynamic performance and is effective across all operating conditions, including start-up, steady-state, and both continuous (CCM) and discontinuous (DCM) conduction modes, without requiring additional modules.

(c) An Adaptive Multi-Objective MPC with Online Weighting Factor Tuning (Chapter 4) [55]: To improve robustness against variations (e.g. load changes), this research presents an adaptive MPC method that integrates an online, state-based adaptive weighting factor. This mechanism autonomously adjusts control priorities between voltage and current regulation in real-time. This innovation not only enhances dynamic performance but also simplifies hardware design by natively incorporating over-current protection and soft-start functionalities. The work also provides practical guidelines for weighting factor design applicable to various dc-dc converter topologies.

(d) An Enhanced Single-Loop MPC with a New Cost Function Formulation (Chapter 5) [56]: Building upon the work in Chapter 3, this contribution further refines the single-loop control strategy by introducing a newly formulated cost function. This enhancement improves system robustness by enabling more direct and precise control over both voltage and current dynamics within the single-loop framework, while still providing integrated over-current protection.

(e) Finally, Chapter 6 concludes the dissertation by summarizing the key findings and suggesting areas for future exploration.

Chapter 2

FPGA-based Hardware Acceleration for Model Predictive Control of Power Converters

2.1 Introduction

Model predictive control has emerged as a topic in the field of power electronics in recent years. MPC predicts the optimal behavior of a plant using its model, offering significant advantages over traditional control strategies, particularly for nonlinear, multiple-input multiple-output (MIMO), and constrained plants with complex dynamics [11].

Over the last two decades, the exploration of MPC-based control algorithms has been extensive, covering a broad array of power electronic systems, including electric drives, uninterruptible power supplies (UPS), micro-grids, and high-voltage DC systems [57].

MPC is well suited for managing power converter and drive operations due to its alignment with the inherent discrete characteristics of these systems [58]. Since power converters have a finite number of switching states, the MPC optimization problem can be simplified and reduced to the prediction of system behavior only for those possible switching states [59]. The optimization process seeks to identify the switching state that minimizes the cost function, thus dictating the most effective control action at each decision point [12].

Despite its theoretical and practical benefits, applications of MPC have been limited due to inherent and significant challenges [60]. The primary obstacle is the computational intensity required for predicting and selecting optimal behaviors, which becomes particularly prohibitive for complex converter systems and extended prediction horizons [61]. Each control cycle requires the evaluation of all possible states to determine the optimal one, which can be computationally demanding, especially as the number of switching states increases or when the system dynamics are particularly complex [7].

In recent work, field-programmable gate arrays (FPGAs) have become the preferred technology for implementing MPC, based on the capability for parallel processing and deep pipelining. This contrasts with traditional CPU-based architectures, highlighting the FPGA's distinction in handling the demands of MPC implementation [62]. In [63], an FPGA platform for a three-phase voltage source inverter (VSI) was built and its performance using MPC was analyzed. Furthermore, research in [11] emphasized the parallel processing capabilities of FPGA for MPC applied to a direct matrix converter. But it only has a simple 1-step prediction horizon and no pipeling between modules of FPGA implementation. And in [64], the author used high-level synthesis tools for FPGA development, enabling the use of more friendly C/C++ code for development. An FPGA-based real-time implementation of linear MPC schemes was proposed in [65]. However, to avoid a large computational burden, the authors used explicit MPC and presented only simulation results.

Prior research employed FPGAs for implementing MPC across various power electronic converters or systems. However, only implementation examples with small prediction horizons have been demonstrated yet and no attempts have been made for fast and embedded dynamic systems. Moreover, a notable gap in the literature is the absence of comprehensive guidelines or methodologies that detail the process of leveraging an FPGA to realize the potential of hardware acceleration for a long prediction power converter.

Despite the advantages of FPGAs, developing and organizing a deeply-pipelined and highly-parallel process for MPC implementation presents significant challenges. This chapter aims to bridge this gap by introducing a detailed guideline for implementing hardware acceleration for MPC on FPGAs. The approach is exemplified through the development of a 5-step prediction horizon MPC for buck converters, which demonstrates the feasibility and efficiency of our proposed guidelines with remarkable computation time reduction to 100 ns. The effectiveness of this methodology is further substantiated through both hardware-in-the-loop simulation and experimental verification.

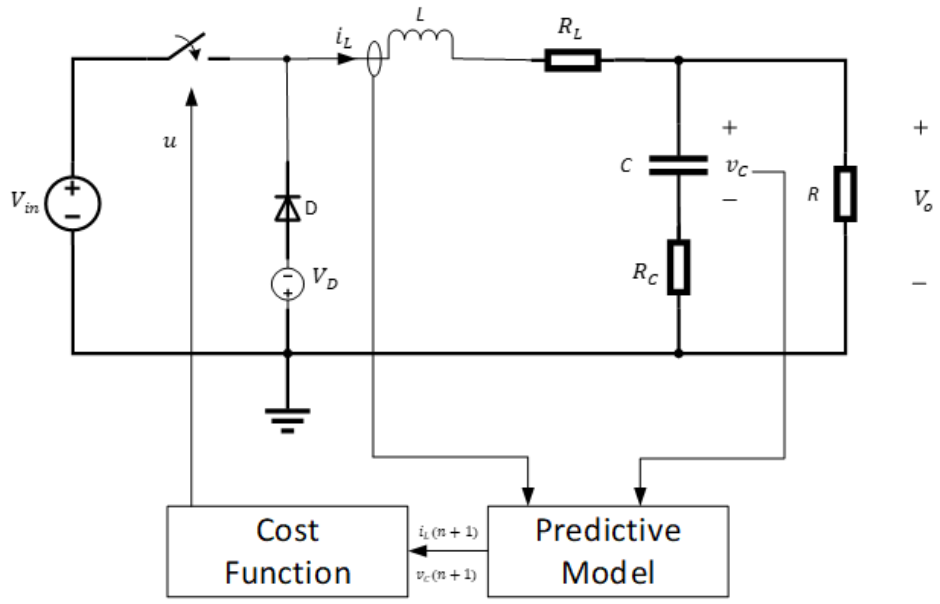


Figure 2.1: Block diagram of MPC on buck converters.

The chapter is structured as follows: Section 2.2 delves into the modeling and analysis of buck converters for Finite Control Set-MPC (FCS-MPC). Section 2.3 discusses the methodology for achieving pipelined and paralleled hardware acceleration in MPC implementations. Section 2.4 presents simulation and experimental results. Section 2.5 summarizes the results and future work.

2.2 Modeling and Analysis of MPC for a Buck Converter

This chapter uses a 5-step prediction horizon buck converter for illustrating FPGA-based hardware acceleration for MPC. The physical circuit diagram of buck converter with parasitic parameters is depicted in Fig.2.1, where L is the inductance and C is the capacitance, R_L and R_c are their parasitic resistances, respectively. R is the load resistance and V_D indicates the voltage drop of diode.The switching state, denoted by u, assumes a value of 1 for the switch-on state and 0 for the switch-off state. The MPC module consists of a predictive model and a cost function. The predictive model serves as a digital replica of the physical system,

designed to predict all future states. These predicted states are subsequently evaluated by the cost function module to determine the optimal states for the forthcoming step.

2.2.1 Predictive Model and Cost Function Design

The dynamic state equations of buck converter are shown in (2.1). To predict the future inductor current and capacitor voltage, forward Euler' s method is utilized to derive the discrete model presented in (2.2).

$$\begin{cases} \dot{i_L} = -\dfrac{R_L}{L} i_L - \dfrac{1}{L} v_C - \dfrac{R_C C}{L} \dot{v_C} + (u-1)V_D + uV_i \\ \dot{v_C} = \dfrac{R}{(R+R_C)C} i_L - \dfrac{1}{(R+R_C)C} v_C \end{cases} \tag{2.1}$$

$$\begin{cases} i_L(n+1) = k_1 i_L(n) + k_2 v_C(n) - \dfrac{h}{L} V_D + l_1 u \\ \qquad\qquad = f_1(i_L(n), v_C(n)) \\ v_C(n+1) = k_3 i_L(n) + k_4 v_C(n) \\ \qquad\qquad = f_2(i_L(n), v_C(n)) \end{cases} \tag{2.2}$$

Where, h is the time step size for calculation of forward Euler's method, $k_1 = (1 - \frac{hR_L}{L} - \frac{hRR_C}{L(R+R_C)})$, $k_2 = \frac{h}{L}(\frac{R_C R}{R_C+R} - 1)$, $l_1 = \frac{h}{L}(V_i + V_D)$, $k_3 = \frac{hR}{C(R+R_C)}$, $k_4 = (1 - \frac{h}{C(R+R_C)})$. The given digital model consists of two functions f_1 and f_2. Each function has two inputs $i_L(n)$ and $v_C(n)$, which are sampled state variables at the time instant n. The variable u can be treated as an internal state inside the model.

The cost function J, detailed in (2.3), facilitates the optimal selection of switching states or sequences for extended prediction horizons. Typically, the cost function encapsulates the controlled objectives and constraints. Increasing the number of control objectives and constraints enhances system performance but also augments system complexity.

In this specific case, the MPC targets three primary objectives: minimizing the inductor current error, regulating capacitor voltage error, and adhering to specified constraints, each

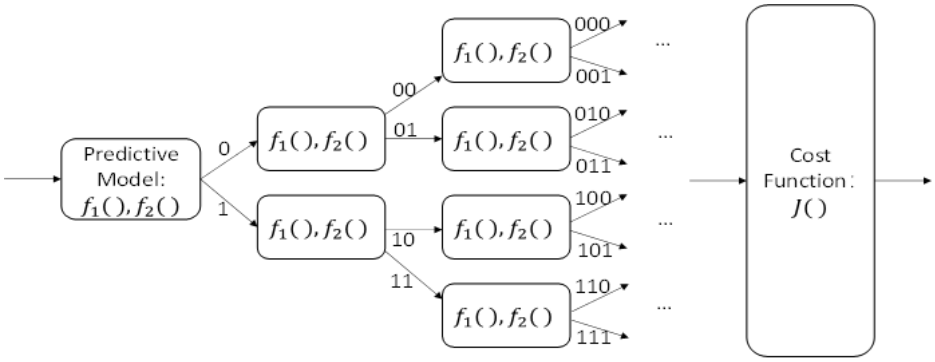


Figure 2.2: Computation flow of MPC on buck converters.

weighted by factors α, β, γ respectively. The voltage error ensures precise regulation of the output voltage. Addressing the current error enhances dynamic performance, serving as an auxiliary element in output voltage control. Additionally, constraints are imposed to limit maximum current, preventing inductor saturation. And the variable m represents the total prediction horizons. Consequently, J is a measure that balances the mismatch between predictions and references under the current limitation.

$$\begin{aligned} J &= \alpha \sum_{k=1}^{m} ||i_L^*(n+k) - i_L(n+k)||^2 \\ &+ \beta \sum_{k=1}^{m} ||v_C^*(n+k) - v_C(n+k)||^2 \\ &+ \gamma \sum_{k=1}^{m} ||i_{L\_max}(n+k) - i_L(n+k)||^2 \\ &= f_a(i_L(n+1), ..., i_L(n+m)) \\ &+ f_b(v_C(n+1), ..., v_C(n+m)) \\ &+ f_c(i_L(n+1), ..., i_L(n+m)) \end{aligned} \tag{2.3}$$

2.2.2 Computation Analysis

The computational load of MPC is predominantly influenced by three major factors: the complexity of prediction model and cost function, and the length of prediction horizons. The prediction module is tasked with calculating all future state variables throughout the

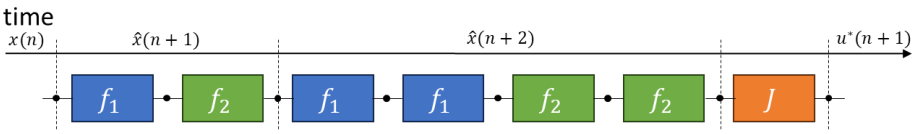


Figure 2.3: Sequential delay for a run.

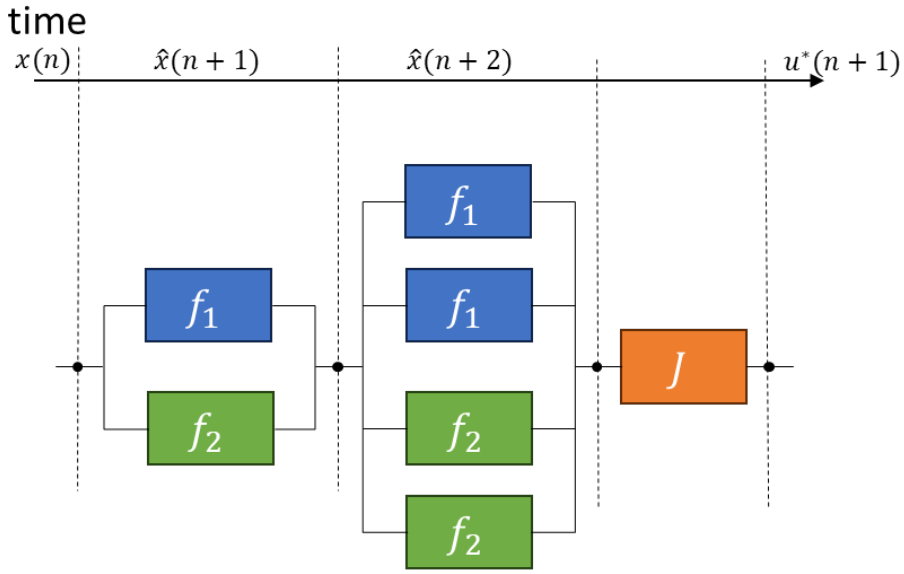


Figure 2.4: Task parallelism within a run.

prediction horizons. Then all available state variables are forward to the cost function module for selection. Specifically, for buck converter in this case, the prediction module entails two separate functions, as shown in (2.2), with each necessitating two iterations for an exhaustive search through switching states. The output from the predictive model module, based on prior module results and shown in Fig.2.2, denotes a "000" switching state sequence for a three-step horizon. Hence, the number of function calls grows exponentially as prediction horizons extend. For buck converter with 2 switching states, k-step prediction requires 2^k function calls. While FPGAs offer parallel computation capabilities, inefficiencies in pipeline programming can still result in delayed computations.

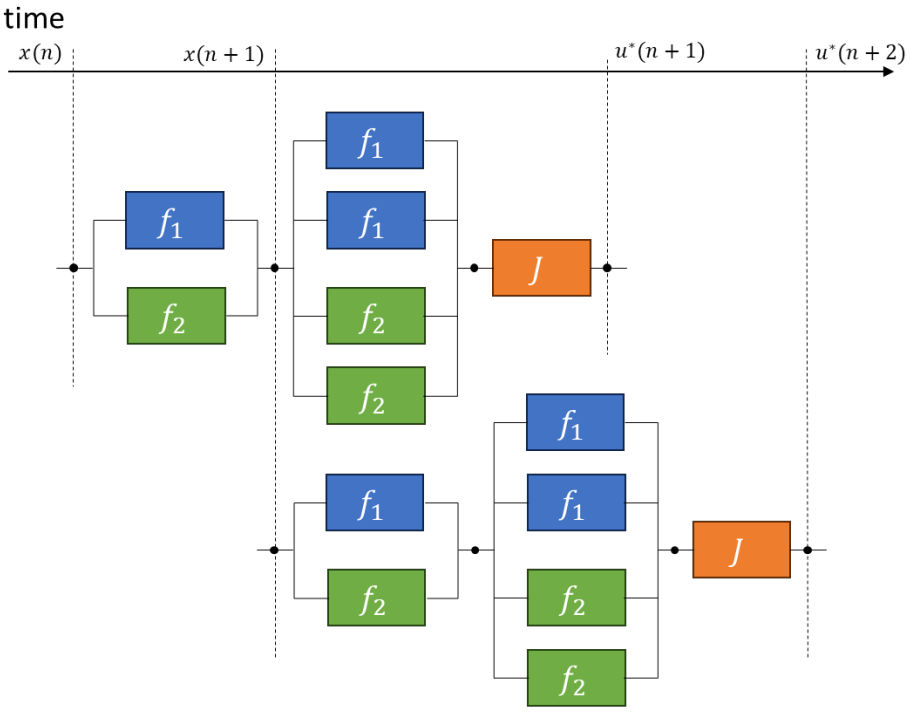


Figure 2.5: Task parallelism with pipelining.

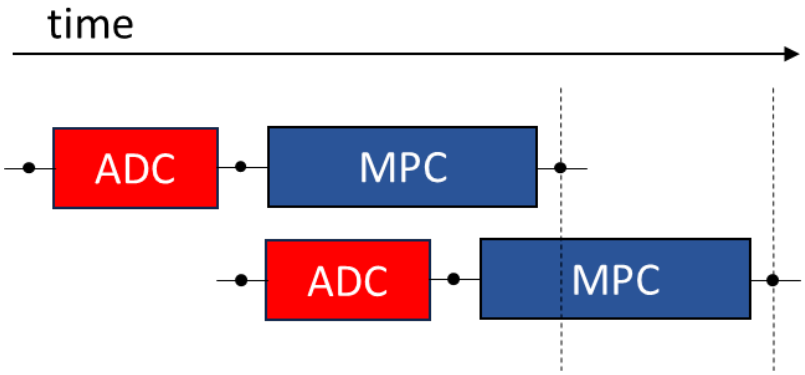


Figure 2.6: Parallelism and pipelining of tasks.

2.3 Hardware Acceleration on FPGA implementation for MPC

An FPGA is suited for MPC implementations, attributed to their all programmable nature. It allows for the customization of hardware architecture to match specific computational tasks instead of using general purpose tools. Therefore, FPGA development for hardware acceleration is about programming an architecture to implement the desired functionality. This methodology is comprised of two major phases: architecting the application and developing the computation kernels. The initial phase involves designing the macro-level architecture, which sets task parallelism within a run and task pipelining across runs. The subsequent phase dives into the micro-architecture, focusing on the acceleration of computation kernels. This involves detailed design elements such as loops, state machines, and datapaths, essential for enhancing task pipelining efficiency.

2.3.1 Marco-level Architecture

Optimization tools today typically operate at the function/procedure level. Each function can be converted into a specific hardware component. Each hardware component will in turn be composed of many smaller predefined components that typically implement basic functions such as add, subtract and multiply. These components, comprised of smaller predefined units, are tasked with executing fundamental operations.

For simplicity of illustration, the example in this section utilizes a 2-step prediction MPC. Fig. 2.3 shows the pre-pipelining sequential latency within one time running of a 2-step prediction MPC module. $x(n)$ denotes the sampled variables at time instant n. $\hat{x}(n+1)$ and $\hat{x}(n+2)$ are predicted variables and $u^*(n+1)$ is the optimal switching state for the time instant $n+1$.

To optimize an MPC program, the foundational step is to understand the program workflow to identify independent tasks or functions. A key pattern for identifying the independent tasks or functions on FPGAs is the producer-consumer paradigm. By applying the producer-consumer paradigm to the sequential program in Fig. 2.3 for extracting functionality that

can be executed in parallel, it is observed that functions f_1 and f_2 are independent and can be executed in parallel, consuming the sampled state variables. And the first-step prediction produces data for the second prediction. Similarly, the cost function is the consumer for the all predicted states produced by the these functions. As a result, the MPC module can be paralleled and pipelined with independent functions following producer-consumer paradigm as shown in Fig. 2.4. The whole latency is reduced a lot with parallelism and pipelining.

The previously discussed execution pattern exploited task-level parallelism within an invocation. But in successive runs, the second invocation can be running overlapped with the first one so that the same computation hardware is reused. As shown in Fig. 2.5, this strategy exemplifies task-level pipelining across invocations. Specifically, the hardware allocated for the first-step prediction can be reused instantly for the next sampled variables $x(n+1)$, even as the system processes the second-prediction for $x(n)$. This approach of pipelining reduced the latency, as it is limited now by the maximum latency among all tasks, rather than by the sum of latencies for all tasks. This pattern of pipelining can be applied to a higher-level operations, as demonstrated in Fig. 2.6, which gives an example of pipeling between an analogy-to-digital converter (ADC) module and MPC. The whole latency is limited by the slowest one instead of their sum.

2.3.2 Micro-level Architecture

Micro-level architecture is about achieved parallelism within tasks or functions. In the case of MPC, the compute functions need to be re-architected into load-compute-store sub-functions for more pipelining potential, as shown in the Fig. 2.7. The load and store functions encapsulate the data accesses and isolate the computation performed by the various compute functions. These sub-functions, typically interacting with different memory spaces and predefined hardware, operate independently, enabling their overlapped execution within invocations. FIFO is First-In-First-Out for data communication. Furthermore, examining functions f_1 and f_2 in (2.2) reveals the independence of computations relating input variables

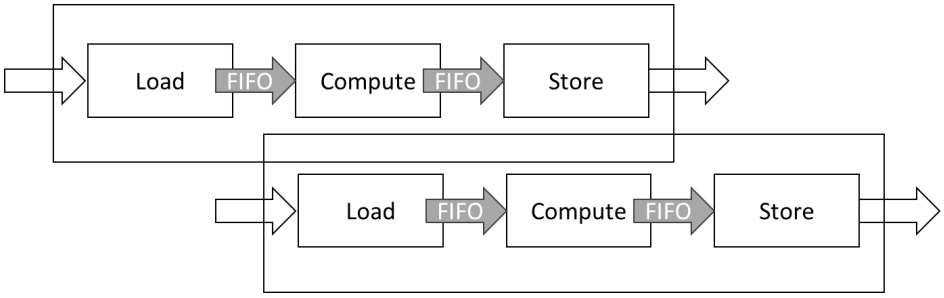


Figure 2.7: General load-compute-store pattern.

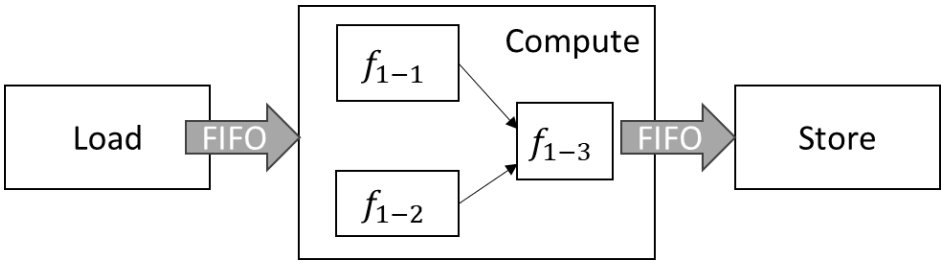


Figure 2.8: Load-compute-store pattern with sub-function for predictive model.

i_L and v_c, allowing for their computations to be parallelized, as demonstrated in Fig.2.8. Function f_{1-1} depicts that the computation related to inductor current and function f_{1-2} is for the capacitor voltage. Function f_{1-3} is the sum operation as a consumer for the previous functions. Fig. 2.9 illustrates the same pattern for the cost function.

In the computation of the cost function, identified by (2.3), it has inherent accumulation process within sub-functions f_a, f_b and f_c, which will be turned into loops for hardware implementation. Fig.2.10 shows the diagram of loop inside the function f_a. Loop is a typical sequential process with high latency, necessitating loop unrolling to decrease latency. Loop unrolling unwinds the loop, allowing multiple iterations of the loop to be executed concurrently as shown in Fig.2.11.

2.3.3 Overview of Method

Those illustrated architecture techniques are summarized in the Fig. 2.12. In the first phase, baseline performance and acceleration goals should be established: for example, the

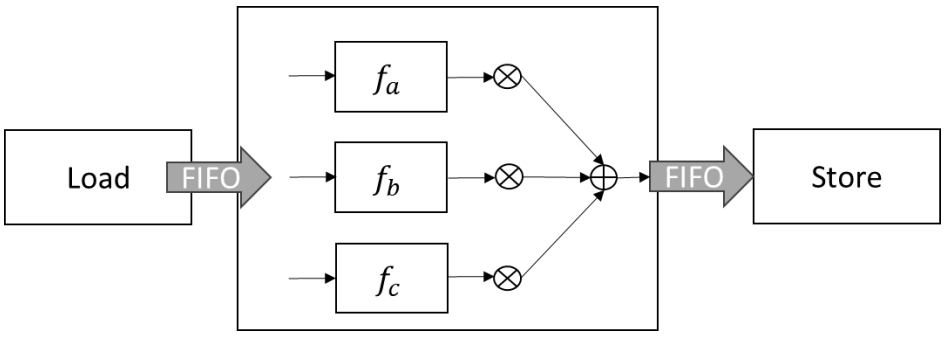


Figure 2.9: Load-compute-store pattern with sub-function for cost function.

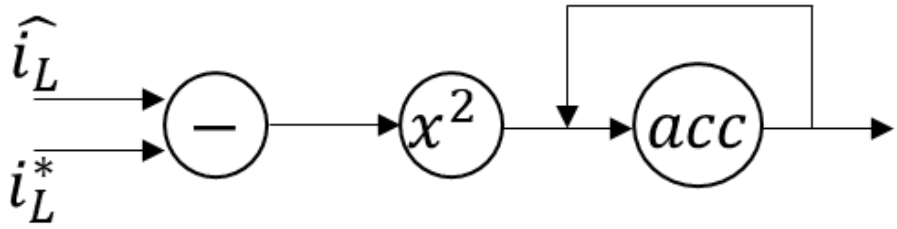


Figure 2.10: Loop inside cost function.

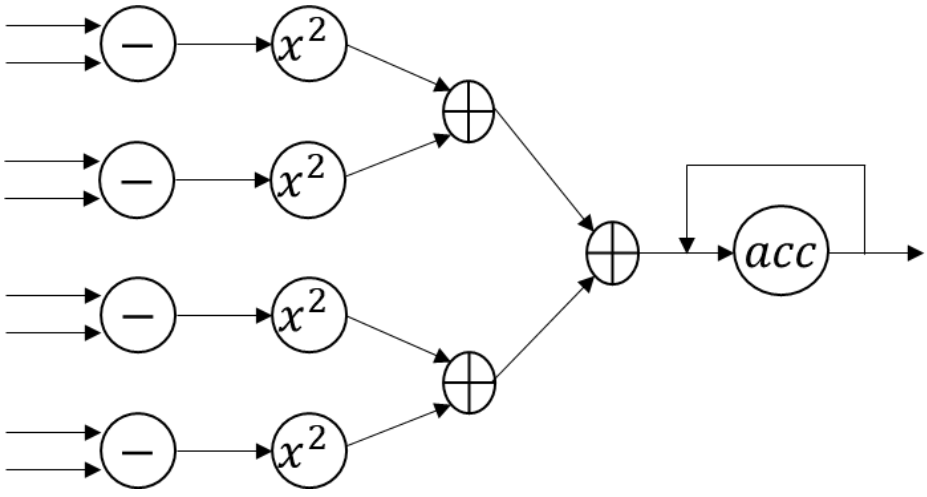


Figure 2.11: Unrolled loop inside cost function.

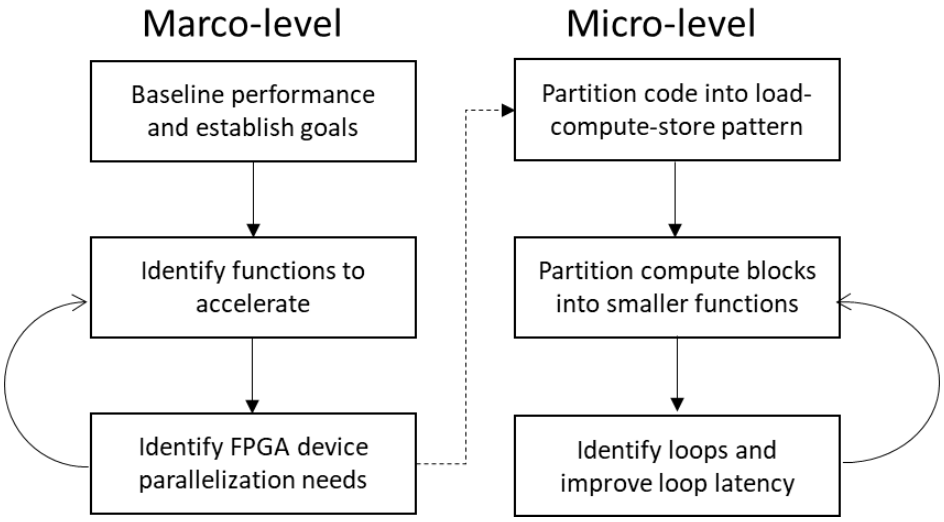


Figure 2.12: Overview of design methodology.

desired computing frequency and hardware resource. Subsequently, identifying critical functions for acceleration and the necessary level of parallelization forms the core of this iterative process. At the micro-level, a kernel is designed as a custom datapath aligned with specific functionalities, complemented by an associated data storage network. This involves crafting a load-compute-store architecture tailored to meet performance targets and unrolling loops to minimize latency.

To develop an FPGA-based hardware acceleration platform with the above techniques, in the first phase one should make key decisions about the baseline performance, for example the targeted runtime and bottlenecks of the MPC application. After establishing the performance baseline, the next step is to determine which functions should be accelerated in the device. As shown in Fig.2.3, it can be easily identified that the predictive model has the most potential for acceleration. And also when looking for acceleration candidates, consider the performance of the entire application not just of individual functions. Then the next step is to determine what level of parallelization is needed to meet the goals, specifically for the number of instantiated computation kernels, the width of datapath and so on. For the micro-level, a kernel is essentially a custom datapath for the desired functionality and an associated data storage network. Also referred to as the memory architecture or

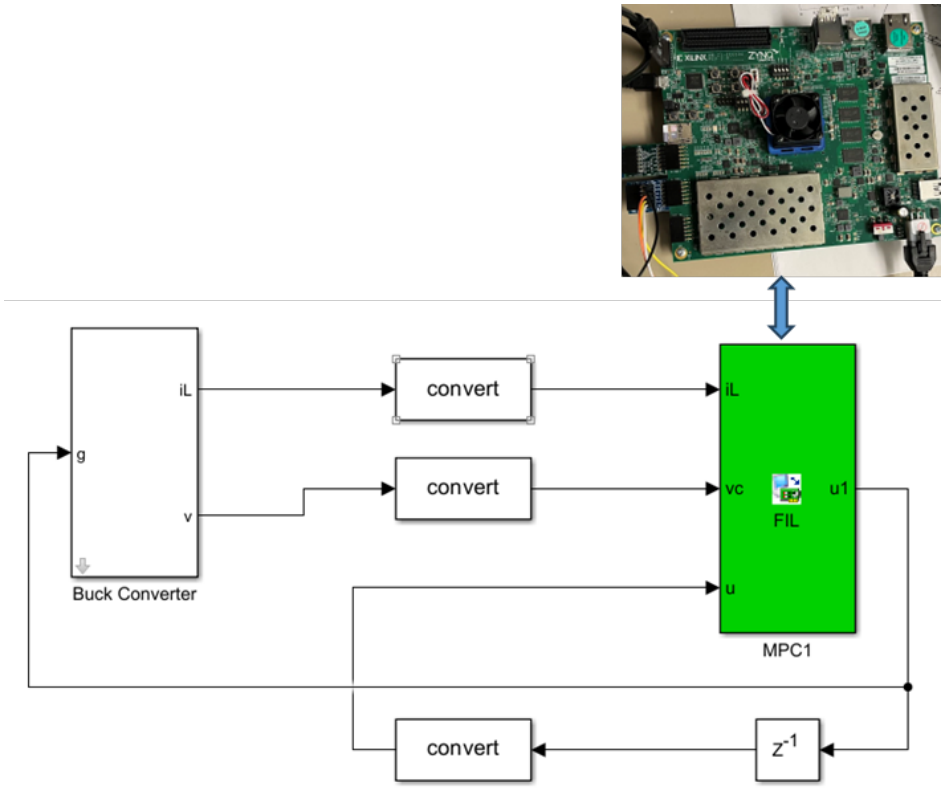


Figure 2.13: Simulink schematic diagram of FPGA-in-the-loop.

memory hierarchy of the kernel, this data storage network is responsible for moving data in and out of the kernel and through the custom datapath as efficiently as possible. Creating a load-compute-store structure that meets the performance goals could decrease the loop latency.

2.4 Simulation and Experimental Results

2.4.1 Hardware-in-the-loop Simulation

The MPC algorithm's efficacy is demonstrated via an FPGA-In-the-Loop (FIL) simulation utilizing a Zynq Ultrascale+ ZCU104 FPGA board. The algorithm is deployed on the FPGA, while the buck converter model operates within Simulink, as illustrated in Fig. 2.13. The Simulink model transmits output voltage and inductor current of the buck converter to the FPGA for calculation, which then controls the converter through gate signals. Fig. 2.14

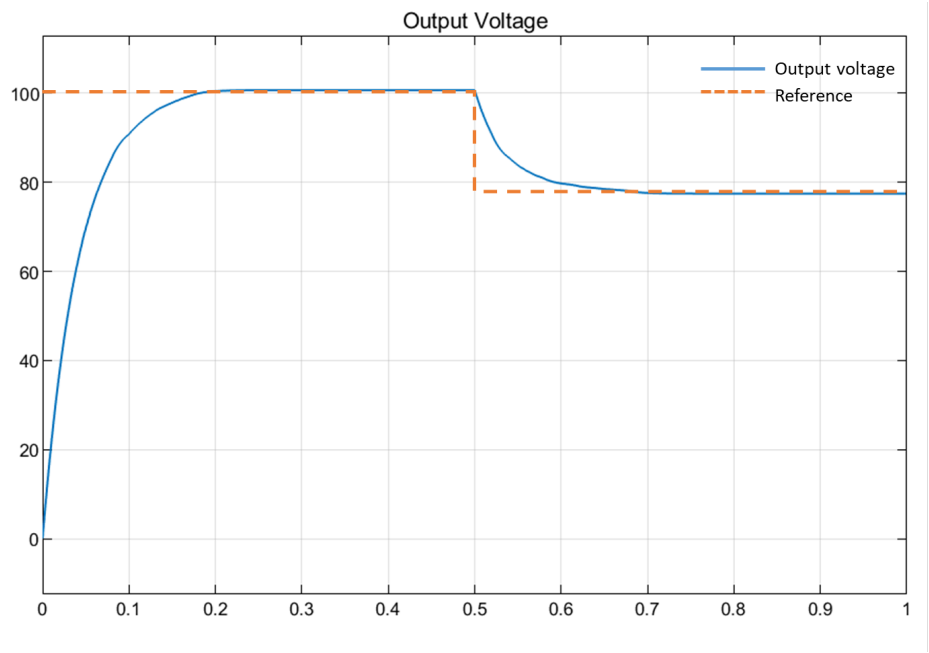


Figure 2.14: FPGA-in-the-loop simulation.

demonstrates how the output voltage tracks reference changes from the start-up to steady state, evidencing the FPGA-based MPC algorithm's operational efficacy.

2.4.2 Experimental results

A 48V synchronous buck converter prototype has been implemented for evaluation with FPGA-based 5-step MPC controller as shown in Fig. 2.15. The system clock is 100 MHz. The N-MOSFET is IRF540ZPBF and ADC is AD7476a. The measurement modules are LV25-NP and LA25-P.

Table 2.1 details the calculation cycles required for all processes. These cycles increase exponentially with the prediction horizon increment. For DSP or MCU architectures, which rely on serial computation, it typically takes 98 cycles to complete one control process. In contrast, a fully-pipelined FPGA allows most processes to run in parallel, reducing the time to complete one control process to 10 cycles, except for data-dependent paths.

As for the experiments of acceleration, the variable frequency nature of FCS-MPC makes direct measurement of actual calculation frequency challenging, necessitating the use of a

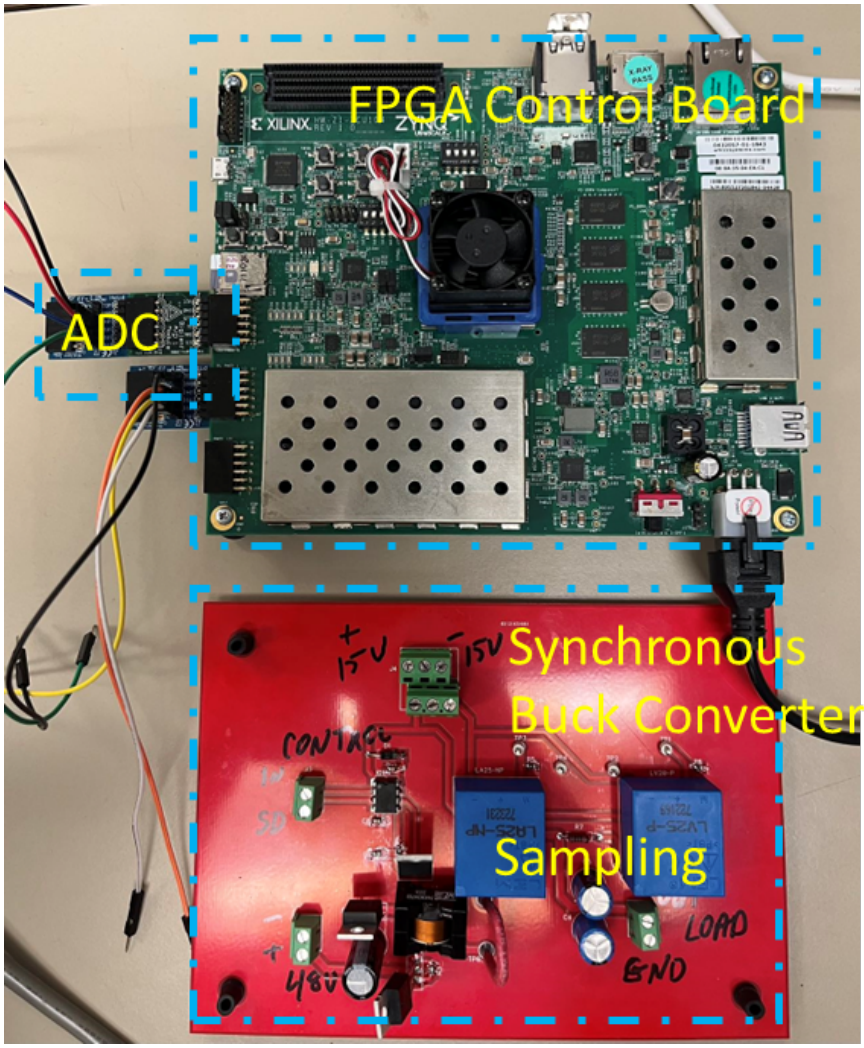


Figure 2.15: Experimental platform.

valid signal to signify the generation of gate driving signals, which are set to high. Fig. 2.16 depicts the output valid signal from the MPC. The time interval between two such valid signals is noted to be 100 ns, or 10 cycles at an 100 MHz clock frequency. Furthermore, Fig. 2.17 shows the results of the MPC's pipelining process with an ADC driver module, where the sampling rate is set as 1 MSps, leading to a sampling latency of 1 μs. The pipelining strategy, as outlined in Section 3, ensures that the total system latency remains at 1000 ns, dictated by the module with the highest latency, rather than an sum of their latencies 1100 ns.

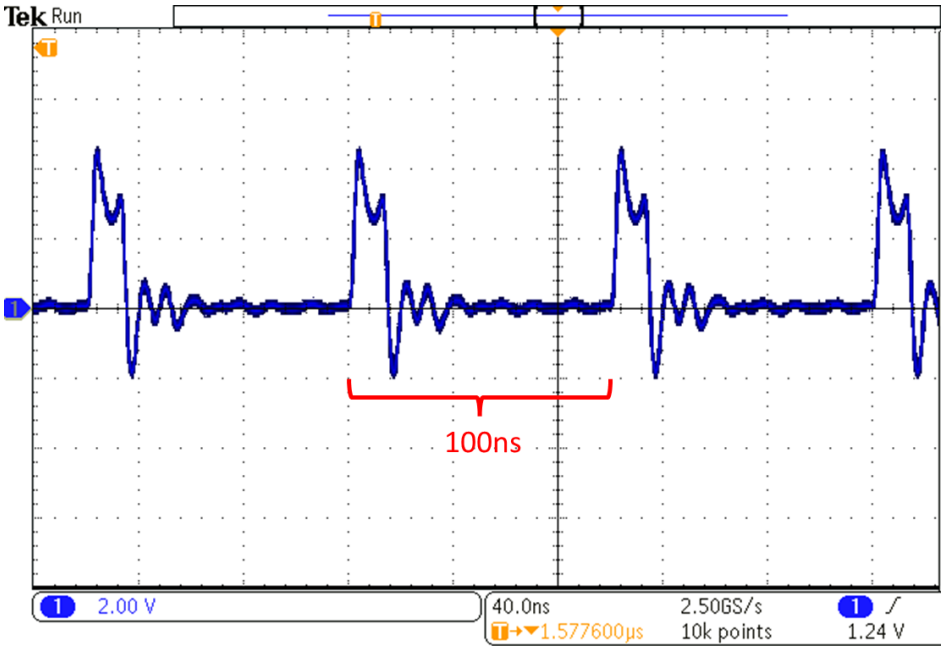


Figure 2.16: Time plot of MPC module.

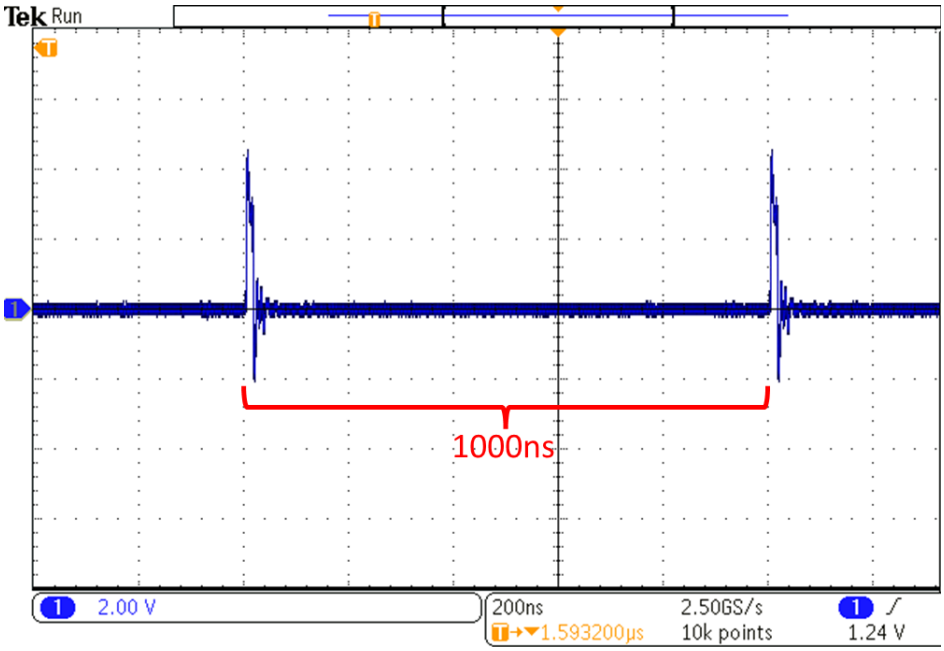


Figure 2.17: Time plot of the whole process.

Table 2.1: Execution Time Cycles

Task	**Clock cycles**
Initialization	1 cycle
Conversion of ADC values	1 cycle
Current prediction for $n+1$	1 cycle
Voltage prediction for $n+1$	1 cycle
Current prediction for $n+2$	2 cycles
Voltage prediction for $n+2$	2 cycles
Current prediction for $n+3$	4 cycles
Voltage prediction for $n+3$	4 cycles
Current prediction for $n+4$	8 cycles
Voltage prediction for $n+4$	8 cycles
Current prediction for $n+5$	16 cycles
Voltage prediction for $n+5$	16 cycles
Current cost calculation	16 cycles
Voltage cost calculation	16 cycles
Total cost calculation	1 cycle
Optimization and decision making	1 cycle
Total (serial)	98 cycles
Finalized (pipelined)	10 cycles
Final Execution Time (μs)	100

2.5 Conclusion

This chapter presented a method for FPGA-based hardware acceleration of MPC. It details architecting strategies spanning from macro to micro levels to accelerate MPC implementation. The validity of the proposed MPC algorithm is demonstrated through an FPGA-in-the-loop simulation, while experiments substantiate the acceleration results.

Chapter 3

Unified Model Predictive Control for DC-DC Buck Converters: From Start-up to Steady-State Operation

3.1 Introduction

The rapid advancement of renewable energy and electronic technology have made power electronic converters ubiquitous. As a result, performance requirements for these converters have increased significantly [66, 67, 68]. Modern applications demand broader stability ranges, higher output voltage accuracy, faster load transient response, and robust over-current protection [69].

Traditional voltage-mode control for DC-DC converters is often limited by poor input and load transient responses, especially when dealing with the non-linearities and different operation modes [1]. Current-mode control for a dc-dc converter typically utilizes the inductor current in an inner loop, with an outer loop generating its reference based on voltage regulation [70]. To simplify controller design, the inner and outer loops are typically decoupled by restricting the outer controller's bandwidth to a fraction of the inner loop's bandwidth. However, this approach limits the dynamic performance of higher control levels. Additionally, tuning multiple control parameters in multi-loop systems presents significant challenges [71].

Model Predictive Control (MPC) is well-suited to meet these demands and has been successfully implemented across various power electronic topologies, including DC-DC converters [57]. And Finite Control Set Model Predictive Control (FCS-MPC) has gained widespread adoption in power converters and drives in recent years [72]. Its capability to handle multiple objectives and nonlinear control makes it particularly suitable for power

electronic converters [20]. For instance, researchers have successfully applied FCS-MPC to control the output voltage of DC-DC converters [73].

Furthermore, current-mode MPC for a dc-dc converter has been developed to enhance their dynamic performance of the converter [74]. While researchers first proposed model predictive current control for boost converters [75], this implementation maintained a multi-loop architecture, merely replacing the PID module in the inner loop. This approach required a Kalman filter for the outer loop, increasing both system complexity and implementation challenges. Similarly, another proposed MPC strategy regulated inductor current but required two additional loops for indirect output voltage control [76]. A more recent NPI-MPC algorithm addressed the non-minimum phase behavior in DC-DC boost converters [77]. However, this approach utilized continuous-control set MPC and focused solely on steady-state performance.

Previous studies have not fully leveraged the multi-objective and nonlinear control capabilities of MPC. Instead, these implementations often merely substitute PI modules in the inner current control loop or require additional control modules. This multi-loop architecture increases the number of control parameters requiring tuning, thereby adding complexity to both design and implementation. Moreover, the inclusion of multiple control modules compromises the system's overall dynamic performance. Additionally, these controllers generally aim only at steady-state design and usually have a large current overshoot during the start-up.

This chapter proposes an MPC method for DC-DC buck converters that synchronously controls both voltage and current. By employing MPC as a single loop, the method enhances transient performance, directly controlling output voltage and inductor current from start-up to steady-state operation in both CCM and DCM without changing the control structure. Furthermore, the integrated current regulation functionality eliminates the need for separate over-current protection and soft-start processes, thereby simplifying the hardware design.

The use of weighting factors enables fully customizable control performance from start-up to steady-state.

This chapter is organized as follows: Section 3.2 presents the continuous and discrete-time models of the converter and proposes the cost function design. Section 3.3 analyzes control performance through simulation and compares results with other controllers. Section 3.4 presents the experimental results, and Section 3.5 concludes this chapter.

3.2 Working Principles of Proposed Control Strategy

3.2.1 Predictive Model for Buck Converter

Fig. 3.1 shows the circuit topology of a dc-dc synchronous buck converter, where L is the inductance and C is the capacitance. R is the load resistance. v_i and v_o are input and output voltage respectively. The switching state, denoted by u_1 and u_2, assumes a value of 1 for the switch-on state and 0 for the switch-off state. And u_2 is out of the phase with u_1.

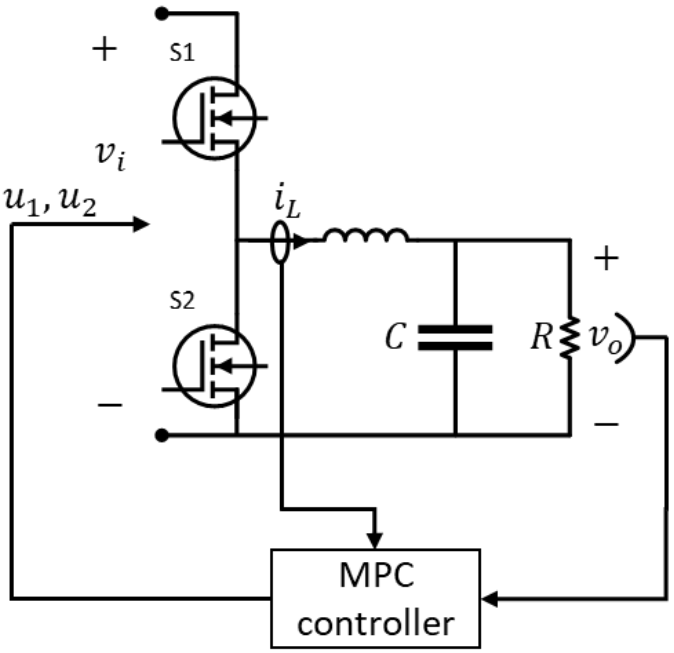


Figure 3.1: Block diagram of MPC on buck converters.

The independent states of the converter are the inductor current and output voltage. The state vector is defined as $x(t) = [i_L(t)\ v_o(t)]^T$. Suppose that the input voltage is constant and the controlled switching state $u(t)$ is a system input, the continuous system is described by the following equations:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{3.1}$$

where the matrices A and B are given by

$$A = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{RC} \end{bmatrix}, B = \begin{bmatrix} V_i \\ 0 \end{bmatrix}.$$

If taking parasitic parameters into consideration, R_L and R_C are their parasitic resistances, respectively, then A becomes

$$A = \begin{bmatrix} -\frac{R_L}{L} - (\frac{RR_C}{R+R_C}) & \frac{1}{L}(\frac{R_C}{R+R_C} - 1) \\ \frac{R}{C(R+R_C)} & -\frac{1}{C(R+R_C)} \end{bmatrix}.$$

The dc-dc buck converter can operate in CCM and DCM, depending on the value of inductor current $i_L(t)$. Specifically, CCM refers to the case where the current $i_L(t)$ is always positive and DCM means that the current $i_L(t)$ reaches zero during the switching cycles and at that time $u = 0$. Since only the value of inductor current changes operation in DCM, the model (3.1) can also be applied in DCM. Besides, (3.1) is a nonlinear model rather than the small-signal model. Hence it can be applied to all running processes from start-up to steady state.

A discrete-time model is required for MPC implementation as an internal prediction model. For simplicity, the forward Euler's method is used, resulting in the following discrete-time model:

$$x(k+1) = A_d x(k) + B_d u(k), \tag{3.2}$$

where, $A_d = I + T_s A$,$B_d = T_s B$, I is the identity matrix, and T_s is the sampling period.

3.2.2 Cost Function Design

The cost function $J(k)$, detailed in (3.3), facilitates the optimal selection of switching state sequences for extended prediction horizons.

$$J(k) = \alpha < i_{L,err} > (k) + v_{o,err}(k) + \beta \Delta u(k). \tag{3.3}$$

In this specific case, the cost function encapsulates three control objectives: the average inductor current error, output voltage error, and switching frequency constraint. The averaged current error objective is represented as the average value of the inductor current over the prediction horizon, given by

$$< i_{L,err} > (k) = \frac{1}{N} \sum_{l=k}^{k+N-1} ||i_L(l) - I_L||,$$

where N is the prediction horizon and I_L is the average current reference. The output voltage objective and switching constraint are defined as

$$v_{o,err}(k) = \sum_{l=k}^{k+N-1} ||v_o(l) - V_o||,$$

$$\Delta u(k) = ||u(k) - u(k-1)||,$$

in which V_o is output voltage reference and $V_o = I_L/R$, and R is the load resistance. The constraint term, the third term in (3.3), reduces the switching frequency to avoid excessive switching. The parameters α and β are weighting factors for inductor current and switching frequency constraints respectively.

3.3 Analysis and Simulation Results

3.3.1 Weighting Factors

Unlike conventional cost functions that typically have one or two control objectives, this system incorporates three objectives, making the design of the weighting factors critical to the system's control performance.

Different combinations of weighting factors will influence the control performance relative to these three objectives. Specifically, increasing the value of α enhances the emphasis on inductor current control, while the other objectives become less significant. For the DC-DC buck converter, the primary objective is to achieve smooth and stable output voltage with over-current protection. The secondary objective is to maintain good transient performance under varying conditions. The tertiary objective is to lower the switching frequency to reduce switching losses. Consequently, the voltage error is prioritized, the current error is secondary, and switching frequency is last. Since switching loss is not the primary concern, it should be adjusted after establishing a balance between current and voltage control.

Fig.3.2 presents the Matlab/Simulink simulation results, demonstrating the control effects on inductor current and output voltage using three different sets of weighting factors (all $\beta = 0.1$). The simulation parameters are shown in Table 3.1. All three sets successfully reach the desired reference values, with the voltage error normalized at steady state. As shown in Fig.3.2, when α_1 is set to a small value (0.1), the current control becomes limited, and the system prioritizes voltage control. This prioritization results in a large current spike—approximately 20 times the nominal current value—which could lead to inductor saturation. Such extreme current conditions would necessitate additional soft-start and over-current protection mechanisms.

However, as the current weighting increases with α_2 and α_3 (same and 10 times weighted as voltage control), the current overshoot is considerably reduced to a safe range. This

adjustment, however, leads to an increased voltage settling time, due to the lower emphasis on voltage control.

This control algorithm effectively regulates transient performance during start-up while maintaining stable steady-state operation and offering high customizability to meet various performance requirements. Designers can achieve trade-offs between voltage and current control based on specific hardware specifications, enabling faster voltage dynamic response while maintaining inductor current within safe operating limits. And these does not require additional control modules.

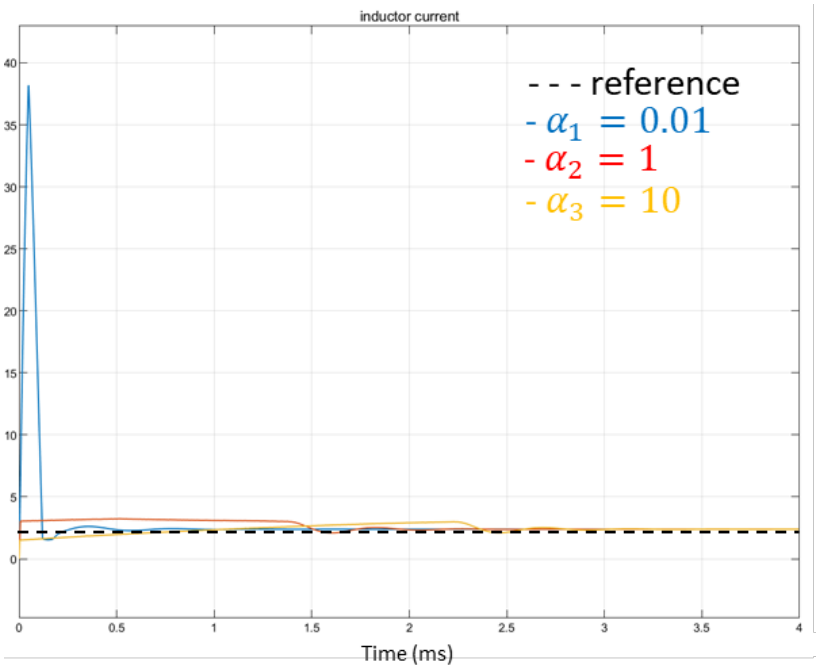
Table 3.1: Simulation and Experimental Setup Parameters

Parameters	**Values (CCM / DCM)**
Input voltage	48 V
Output voltage	24 V / 9 V
Load	10 Ω / 25 Ω
Inductance	47μH
Capacitance	94 μF
Switching frequency (PI controller)	200 kHz
PI gains	$K_p = 0.2$, $K_i = 10$
Prediction horizons	$N = 4$

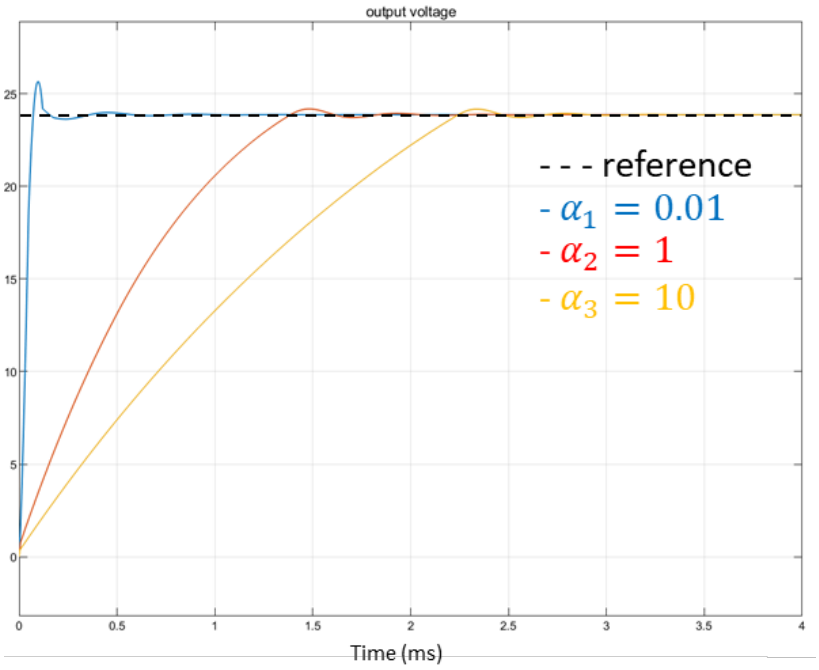
3.3.2 Comparison with Other Controllers

Fig.3.3 demonstrates the control performances of three different control: PID controller, conventional MPC and the proposed MPC.

As shown in Fig.3.3, the PID controller (blue lines) exhibits significant overshoot in both output voltage and inductor current. This larger overshoot necessitates longer settling time to reach the desired reference values. The conventional MPC approach (red lines), which considers only output voltage control in its cost function, achieves smaller voltage overshoot and reduced settling time. However, since its control objective is limited to voltage regulation, the inductor current rises excessively as the controller attempts to reach the reference voltage as quickly as possible. The proposed MPC strategy (yellow lines) incorporates both voltage and current control objectives. This dual-objective approach maintains the inductor

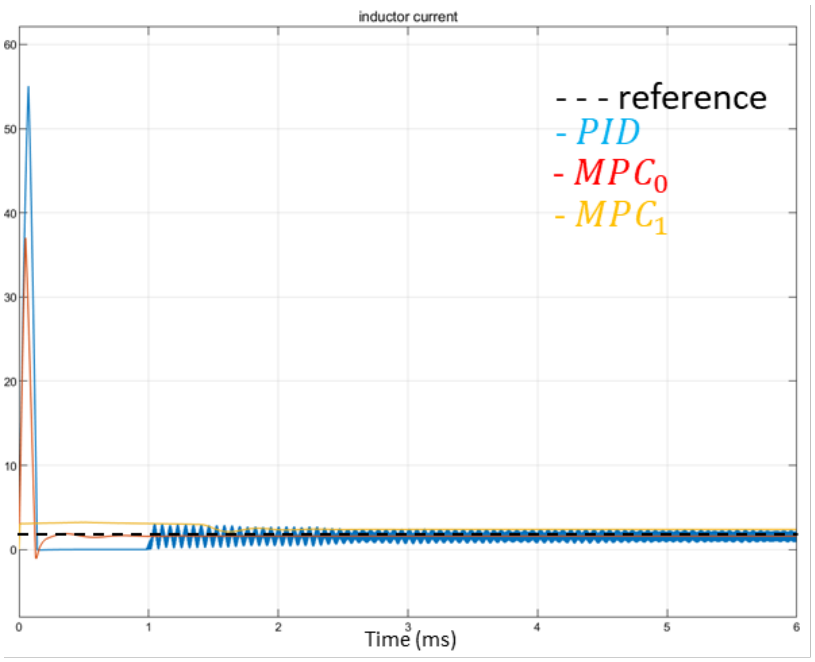


(a) Inductor current

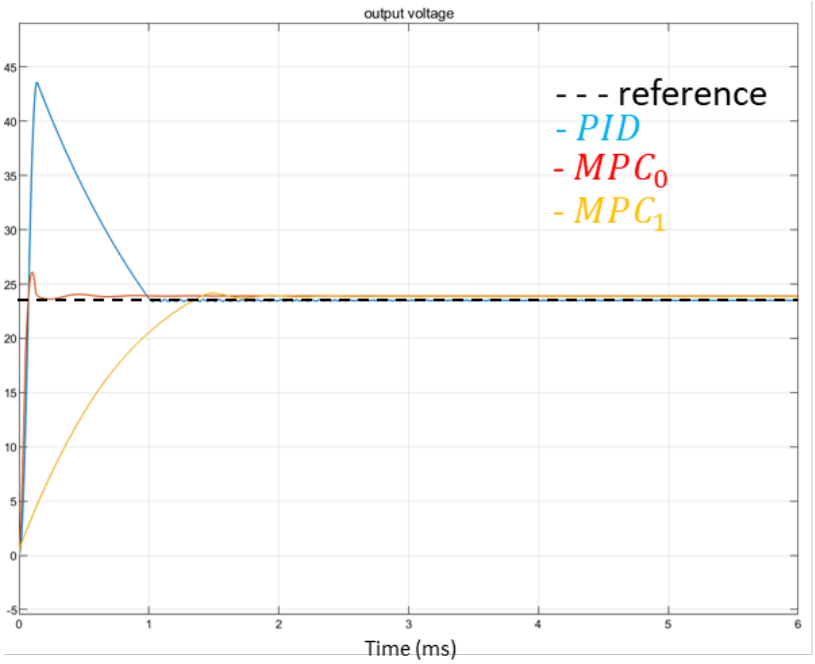


(b) Output voltage

Figure 3.2: Simulation results for start-up process with different weighting factors.



(a) Inductor current



(b) Output voltage

Figure 3.3: Simulation results for start-up process with different controllers.

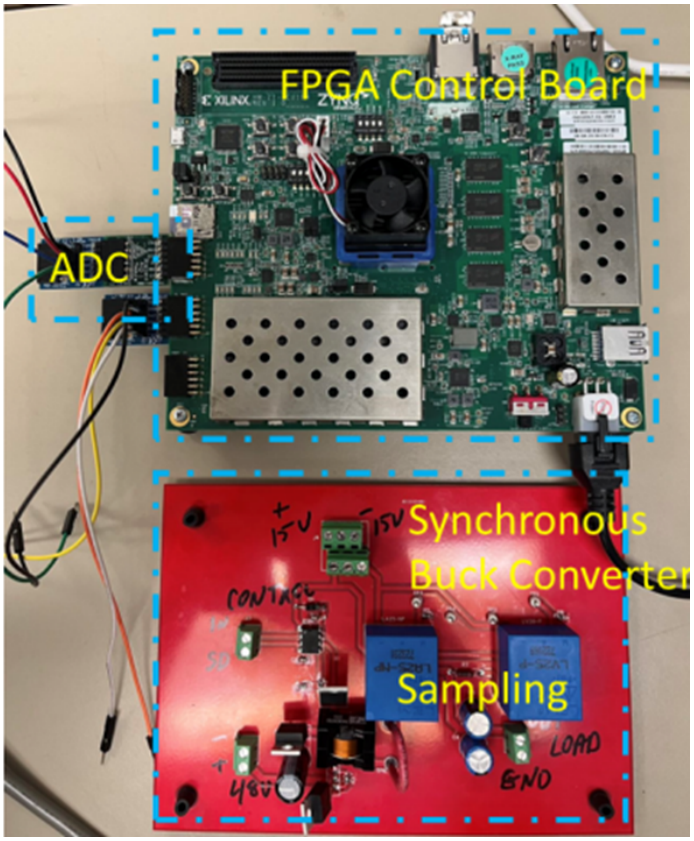
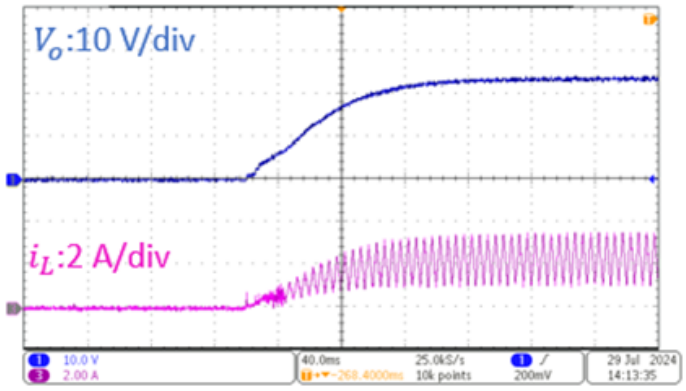


Figure 3.4: Experimental setup.

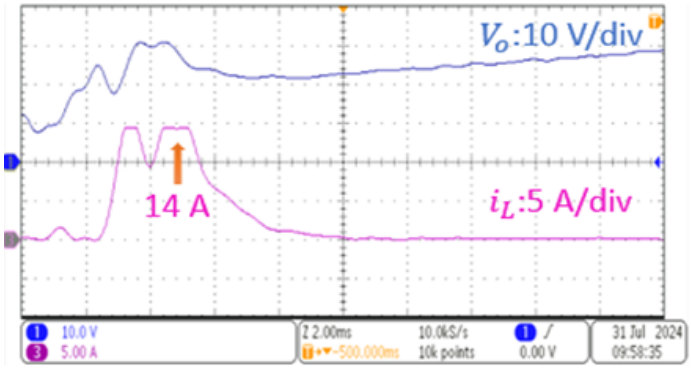
current within safe operating limits while effectively regulating output voltage. The settling time can be optimized by adjusting the weighting factor based on hardware-specific current overshoot tolerances.

3.4 Experimental Results

To verify the proposed MPC algorithm' s ability controlling voltage and current, a buck converter was built with MPC implemented on a field-programmable gate array (FPGA). The FPGA-based MPC controller uses a hardware acceleration method as described in [53]. The FPGA board is Xilinx MPSoC ZCU104, the ADC is AD7476a with 1 Msps sampling rate and the MOSFET is IRF540N. Fig.3.4 is the experimental setup. Table 3.1 shows the system' s parameters.

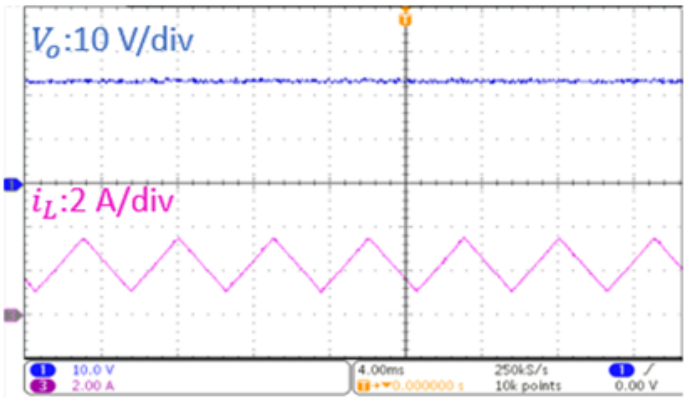


(a) proposed MPC

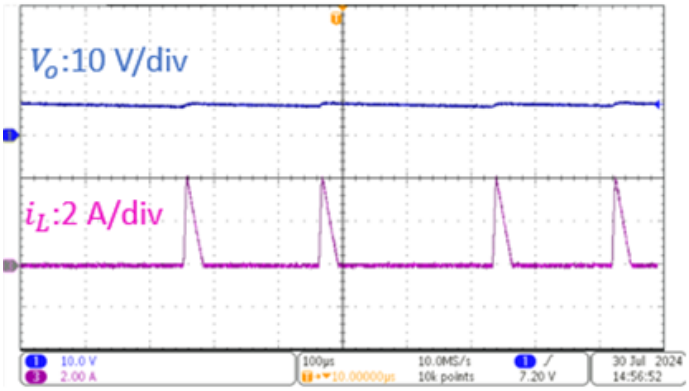


(b) PI

Figure 3.5: Experimental results for start-up process (a) proposed MPC; (b) PI.



(a) CCM



(b) DCM

Figure 3.6: Experimental results with proposed MPC in steady state (a) CCM; (b) DCM.

Fig.3.5a illustrates the start-up process using the proposed MPC method ($\alpha = 0.5$, $\beta = 0.1$), where the output voltage and inductor current gradually reach their reference values without any overshoot or current spike. In contrast, Fig.3.5b shows the start-up process with a PI controller ($K_p = 0.2$, $K_i = 10$), which exhibits a large current spike of around 14 A—about six times the current reference—along with a corresponding voltage overshoot. Consequently, additional processes or hardware are required to limit the over-current.

Fig.3.6a and 3.6b demonstrate the steady-state performance in CCM and DCM, respectively. The proposed control algorithm successfully regulates the DC-DC buck converter in both operating modes using mode-specific parameters. As shown in Fig.3.6, this MPC method achieves stable operation in both CCM and DCM without altering control modules.

3.5 Conclusion

This chapter presented a novel MPC strategy for DC-DC buck converters, utilizing a single control loop to regulate both voltage and current directly. This approach not only enhances transient performance but also simplifies the hardware with stable output voltage from start-up to steady-state operation in both CCM and DCM.

Chapter 4

Multi-objective FCS-Model Predictive Control with Adaptive Weighting Factors for DC-DC Converter

4.1 Introduction

The rapid advancement of electronic technology has placed increasing demands on power electronic converters, requiring broader stability, higher output voltage accuracy, faster transient response, and over-current protection [78]. Consequently, control systems must be capable of managing complex scenarios, including multi-variable control and robustness against different variations, while maintaining high dynamic performance [79, 80].

Finite Control Set Model Predictive Control (FCS-MPC) has gained widespread adoption in this area, owing to its inherent ability to handle multiple objectives and nonlinearities directly [81].

To enhance the performance of DC-DC converters, various MPC strategies have been developed. However, early approaches often retained a complex multi-loop architecture. For instance, the method in [82] merely replaced an inner-loop PID controller and required a Kalman filter for the outer loop. Similarly, another strategy relied on two additional loops for indirect voltage control [83], complicating implementation.

More recent work has utilized a single-loop, multi-objective cost function to directly control both current and voltage [54]. Although this simplifies the structure, it suffers from poor robustness and high sensitivity to parameter variations. While observers can improve its robustness [84], they add significant cost and complexity to the system [85].

A second significant challenge in multi-objective MPC is the design of weighting factors [80]. These factors are critical for balancing control objectives, yet there is no general guideline for their selection. The tuning process often relies on inefficient trial-and-error

or complex, computationally expensive methods like AI-based tuning [86], which are not cost-effective for most applications [87].

This chapter proposes a novel, adaptive multi-objective FCS-MPC for DC-DC converters that enhances robustness without sacrificing dynamic performance, building upon the framework established in Chapter 3, to enhance system robustness while maintaining high dynamic performance. The primary contributions of this work are:

1) A Unified, Multi-Objective FCS-MPC with Integrated Protection: it proposed a single-loop FCS-MPC strategy that simultaneously regulates output voltage and inductor current. This method ensures satisfying performance during both small-signal (steady-state) and large-signal conditions (e.g., start-up). Furthermore, by directly controlling the current, it eliminates the need for separate over-current protection and soft-start hardware, thereby simplifying the overall system design.

2) An Online Adaptive Weighting Factor for Enhanced Robustness: To solve the critical problem of sensitivity of variations, an adaptive weighting factor mechanism is introduced. This function operates online, automatically adjusting control priorities in real-time to maintain optimal performance and significantly improve the system's robustness.

3) A Novel and Generalized Weighting Factor Design Methodology: a straightforward and effective methodology for designing weighting factors based on scale analysis is established. This novel approach provides clear, practical guidelines that can be generalized to various DC-DC converter topologies, addressing a long-standing challenge in multi-objective MPC design.

This chapter is organized as follows. In section 4.2, the working principle of the dc-dc converter with MPC controller is summarized. Furthermore, the control problem is proposed and analyzed via simulation results based on Matlab/Simulink. And to address the poor robustness issue, an adaptive cost function is proposed and analyzed. Section 4.3 gives a

systematic method of weighting factors design based on numerical analysis. Section 4.5 offers experimental results to validate the control method's effectiveness. Finally, section 4.6 concludes this chapter.

4.2 Working Principles of Proposed Control Strategy

This section illustrates the working principle for MPC on DC-DC converter. It uses buck converter as example.

Fig. 2.1 shows the circuit topology of a dc-dc synchronous buck converter, where L is the inductance and C is the capacitance. R is the load resistance. V_i and v_o are the input and output voltage, respectively. The switching state for transistor S1, denoted by u_1, assumes a value of 1 for the switch-on state and 0 for the switch-off state. And the switching state u_2 of transistor S2 is out of the phase with S1.

The MPC contains two sub-modules: predictive model and cost function.

4.2.1 Predictive Model Design

The predictive model is the digital replica of the physical DC-DC converter. The independent states of the most second order DC-DC converters are the inductor current and output voltage. The state vector is defined as

$$x(t) = [i_L(t)\ v_o(t)]^T.$$

Suppose that the input voltage is constant and the controlled switching state $u(t)$ is a system input, the continuous system is described by the following equation:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{4.1}$$

where the matrices A and B are given by

$$A = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{RC} \end{bmatrix}, B = \begin{bmatrix} V_i \\ 0 \end{bmatrix}.$$

The dc-dc buck converter can operate in CCM or DCM, depending on the value of inductor current $i_L(t)$. Specifically, CCM refers to the case where the current $i_L(t)$ is always positive and DCM means that the current $i_L(t)$ reaches zero during the switching cycles and at that time $u_1 = 0$. Since only the value of the inductor current changes operation in DCM, the model (4.1) can also be applied in DCM. Besides, (4.1) is a nonlinear model rather than the small-signal model. Hence, it can be applied to all running processes from start-up to steady state.

A discrete-time model is required for MPC implementation as an internal prediction model. The forward Euler' s method is used, resulting in the following discrete-time model:

$$x(k+1) = A_d x(k) + B_d u(k), \tag{4.2}$$

where,

$$A_d = I + T_s A, B_d = T_s B,$$

I is the identity matrix, and T_s is the sampling period.

4.2.2 Multi-objective Cost Function Design

The cost function $J_1(k)$, detailed in (4.3), facilitates the optimal selection of switching state sequences for extended prediction horizons:

$$J(k) = \lambda i_{L,err}(k) + v_{o,err}(k) + \beta \Delta u(k). \tag{4.3}$$

For the most dc-dc converter, the cost function encapsulates three control objectives: the inductor current error, output voltage error, and other constraints. In this paper, the current error objective is represented as the average value of the inductor current over the prediction horizon, given by

$$i_{L,err}(k) = \frac{1}{N} \sum_{l=k}^{k+N-1} ||i_L(l) - I_L||,$$

where N is the prediction horizon, I_L is the average current reference, and $i_L(l)$ is the predicted average inductor current. The output voltage objective and switching constraint are defined as

$$v_{o,err} = \sum_{l=k}^{k+N-1} ||v_o(l) - V_o||,$$

$$\Delta u(k+1) = ||u(k+1) - u(k)||,$$

in which V_o is the output voltage reference and

$$V_o = I_L/R,$$

R is the load resistance. The constraint term, the third term in (4.3), reduces the switching frequency to avoid excessive switching. The output voltage error is normalized. And the parameters α and β are weighting factors for inductor current and switching frequency constraints, respectively.

4.2.3 Simulation Results

Fig. 4.1 presents the Matlab/Simulink simulation results, demonstrating the control effects on inductor current and output voltage using three different sets of weighting factors (all $\beta = 0.1$).

All three sets smoothly reach the desired voltage reference, with the inductor current regulated within the safe region. As shown in Fig. 4.1, when α_1 is set to a small value

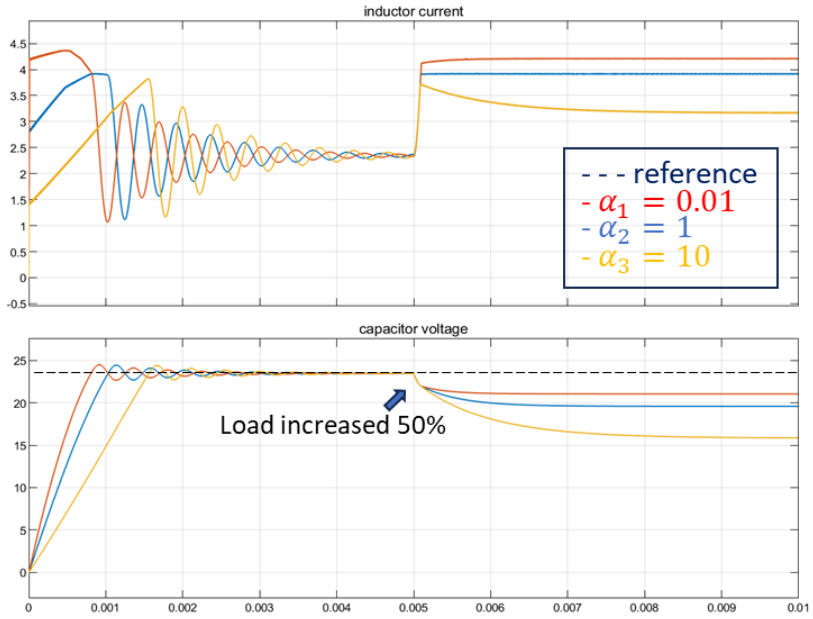


Figure 4.1: Simulation results with different weighting factors from start-up to steady-state.

(0.1), the current control becomes limited, and the system prioritizes voltage control. This prioritization results in a large current spike—approximately 20 times the nominal current value—which could lead to inductor saturation. Such extreme current conditions would necessitate additional soft-start and over-current protection mechanisms.

However, as the current weighting increases with α_2 and α_3 (same and 10 times weighted as voltage control), the current overshoot is considerably reduced to a safe range. This adjustment, nonetheless, leads to an increased voltage settling time, due to the lower emphasis on voltage control.

This control algorithm effectively regulates transient performance during start-up while maintaining stable steady-state operation and offering high flexibility to meet various performance requirements. Designers can achieve trade-offs between voltage and current control based on specific hardware specifications, enabling faster voltage dynamic response while maintaining inductor current within safe operating limits. And these does not require additional control modules.

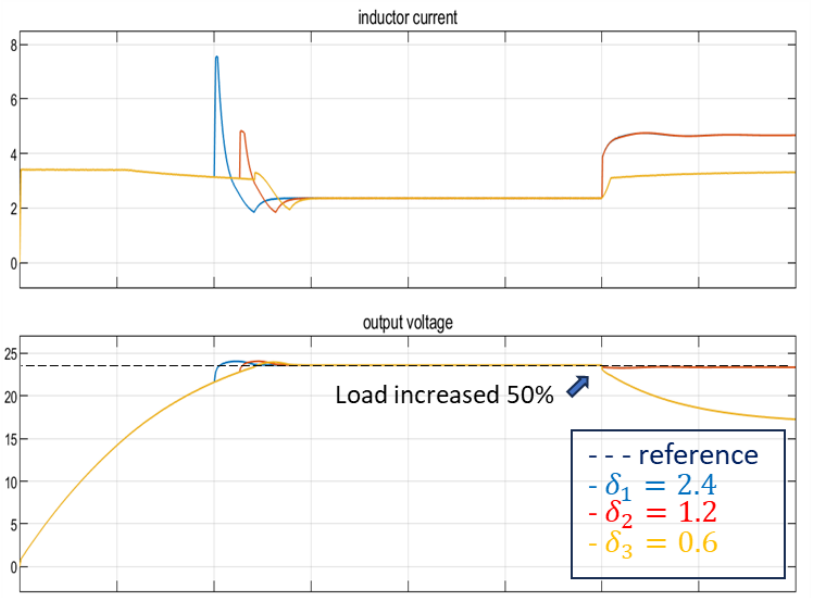


Figure 4.2: Simulation results with different δ from start-up to steady-state.

However, it introduces steady-state error under disturbances such as load variations. As shown in Fig. 4.1, a 50% load increment results in different steady-state errors depending on the cost function weight α. Since both current and voltage are regulated relative to their reference values, the system's operating quiescent point is predetermined. When the load varies, current and voltage control compete with each other, leading to a new steady state that deviates from the reference.

Unfortunately, varying loads are very common in practical applications. Therefore, it is necessary to solve this problem to improve the robustness of this method.

4.2.4 Adaptive Cost Function Design

Since the output voltage control is the final goal and the current control is only considered for improve transient performance and limit its maximum for safety, it is easy to propose an exit mechanism for current control. When the voltage error is less than a threshold value

or index (δ), we set the $\alpha = 0$ to let current control exit. Namely,

$$\alpha = \begin{cases} 0 & |v_o - v_{ref}| \leq \delta \\ \alpha_3 & |v_o - v_{ref}| > \delta \end{cases} \tag{4.4}$$

This α function is shown as dark step line in Fig. 4.3. However, abruptly deactivating the current control in this manner induces large inductor current spikes. As shown in Fig. 4.2, system performance is quite sensitive to the value of δ. A large $\delta = 2.4$ eliminates the static error but causes a large current spike, while a small $\delta = 0.6$ reduces the spike but preserves the static error. Although a compromise can be found through trial-and-error ($\delta = 1.2$), this hard-switching method is flawed because setting $\alpha = 0$ permanently disables the current-limiting safety feature.

Therefore, to achieve current control that attenuates gradually and automatically as the system approaches steady-state, a continuous function is proposed that adaptively adjusts the current control weight α, based on the output voltage error. In this way, the dc-dc converter is controlled with current and voltage during the start-up process and revoked under large-signal variations (faults) to guarantee the safety and fast transient response. And when it is reaching equilibrium (steady-state), the current control becomes dormant and voltage control is emphasized to realize precise output voltage control.

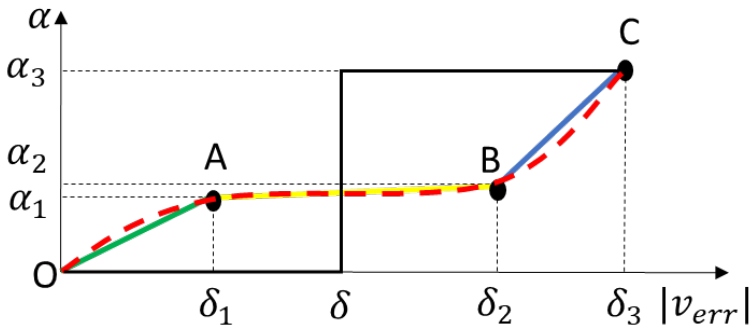


Figure 4.3: Adaptive α function of voltage error.

Figure 4.3 contrasts the conventional step function (dark line) in (4.4), with the proposed adaptive α function (red dashed line), which is defined by three distinct regions based on the voltage error.

In the first region (OA), when the voltage error is below a threshold δ_1, the system is under steady state. Hence the value of α should be small enough so that voltage control becomes dominant during steady-state, ensuring precise voltage regulation without interference, and current control is dormant.

Conversely, in the third region (BC), when the voltage error exceeds a threshold δ_2, α should be large enough to ensure the current control is fully active to manage large transients and prevent current spikes. Between these two states, the transition region (AB) provides a smooth, gradual increase in α from δ_1 to δ_2. This slope should be flat enough to prevent the overshoot voltage and current spike that an abrupt change in α would cause. After determining the critical points A and B, the final adaptive function for α can be implemented using a curve-fitting method.

4.3 Weighting Factors Design Method

For analytical and design simplicity, a single-step prediction horizon and an L_1 norm are used. The switching constraint is considered a secondary term that can be tuned at the last in the design process. Accordingly, the cost function (4.3) simplifies to:

$$J(k+1) = \alpha|i_L(k+1) - I_L| + |v_o(k+1) - V_o|. \tag{4.5}$$

Designing the critical points A, B, and C requires a quantitative scale analysis to determine the dominant term in the cost function and properly adjust the weighting factors. In the quantitative analysis, the order of magnitude of each term should be calculated to design the corresponding factors adjusting their weightings based on scale analysis.

4.3.1 Inductor Current Error

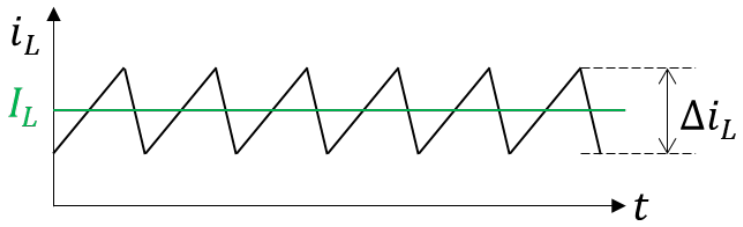


Figure 4.4: Inductor current waveform in the steady state.

When the dc-dc converter is in steady-state operation, the volt-seconds balance is satisfied for the inductor, as its current waveform is shown in Fig. 4.4, where I_L is the average value (current reference). Δi_L is the peak-peak value which depends on the inductance and switching frequency. Then, the order of magnitude of current error in cost function (4.5) (the first term) is given by

$$|i_L(k+1) - I_L| = O(\Delta i_L). \tag{4.6}$$

$$i_L = C\dot{v_C} + \frac{1}{R}v_C, \tag{4.7}$$

$$< i_L >= C < \dot{v_C} > + \frac{1}{R} < v_C > . \tag{4.8}$$

In steady state, (4.6) indicates that the current error, $|i_L(k+1) - I_L|$, is bounded by the peak-to-peak ripple, with an order of magnitude of $O(\Delta i_L)$. Therefore, the average current error is on the order of the ripple amplitude, $O(\Delta i_L)$.

4.3.2 Output Voltage Error

The discrete capacitor voltage can be deduced from the state equation in (4.1) as given by

$$v_o(k+1) = v_o(k) + \frac{T_s}{C} i_L(k) - \frac{T_s}{RC} v_o(k). \tag{4.9}$$

In steady state,

$$v_o(k) \approx V_o,$$

meaning their orders of magnitude are equivalent,

$$O(v_o(k)) = O(V_o).$$

The approximated voltage error from (4.9) is then given by

$$\begin{aligned} |v_o(k+1) - V_o| &\approx \frac{T_s}{C} |(i_L(k) - \frac{1}{R} V_o)| \\ &\approx \frac{T_s}{C} |(i_L(k) - I_L)| \\ &= O(\frac{T_s}{C} \Delta i_L). \end{aligned} \tag{4.10}$$

Substitute (4.6) into (4.10), so that

$$O(|v_o(k+1) - V_o|) = \varepsilon O(|i_L(k+1) - I_L|), \tag{4.11}$$

where, $\varepsilon = \frac{T_s}{C}$.

The coefficient ε reflects the numerical relationship between voltage error and current error for their orders of magnitude. Hence it is critical for weighting the voltage and current error terms. Table 4.1 shows some typical combinations of sampling period and capacitance in dc-dc converters.

Typically in dc-dc converters, $\varepsilon \ll 1$ because the sampling period is much smaller than the circuit's capacitance. Consequently, the voltage error is an order of magnitude smaller

Table 4.1: Different cases for ε

T_sC	94 μF	500 μF	1000 μF
$\frac{1}{10k}$	$\frac{50}{47}$	$\frac{1}{5}$	$\frac{1}{10}$
$\frac{1}{20k}$	$\frac{25}{47}$	$\frac{1}{10}$	$\frac{1}{20}$
$\frac{1}{100k}$	$\frac{5}{47}$	$\frac{1}{50}$	$\frac{1}{100}$
$\frac{1}{500k}$	$\frac{1}{47}$	$\frac{1}{250}$	$\frac{1}{500}$

than the current error

$$|v_o(k+1) - V_o| << |i_L(k+1) - I_L|.$$

This scale analysis allows for the straightforward design of the critical points A(δ_1, α_1), B(δ_2, α_2) and C(δ_3, α_3).

The preceding analysis is now applied to the specific dc-dc converter used in this work, with parameters listed in Table 4.2. Then,

$$\begin{aligned} O(\Delta i_L) &= \frac{V_i - V_o}{L} D T_s \approx 2.55, \\ \varepsilon &= 5/47 \approx 0.106, \\ \delta &= O(\Delta v_o) = O(\varepsilon \Delta i_L) = 0.272. \end{aligned}$$

in which $O(\Delta i_L)$ is the order of magnitude of current error, δ is the order of magnitude of voltage error. And when $v_{err} \leq \delta$, the system is treated as already being in steady-state.

4.3.3 Critical Points Selection Guideline

Based on the indices defined above, the weighting factor function can be constructed using specific critical points. The design guidelines for specifying these points are outlined below:

1. **Region OA (Small-signal region):** The system operates in steady-state; consequently, voltage control is dominant while current control is secondary. Thus,

$$\alpha_1 \leq \varepsilon, \delta_1 = \delta.$$

2. **Region AB (Transition):** The system is transitioning toward steady-state. The current should settle gradually without inducing significant spikes. Therefore, α_2 should be larger than α_1 but remain within a comparable range (typically within one order of magnitude). The value of δ_2 determines the duration of this transient process:

$$\varepsilon < \alpha_2 \leq 10\varepsilon,$$
$$\delta_2 = k\delta, \quad (k = 1, 2, ..., 10...).$$

3. **Region BC (Large-Signal Transient)**: The system undergoes a large-signal transient process. In this region, inductor current control must be active—or even dominant —to regulate the dynamic process and ensure system safety. Consequently, α_3 should exceed 10ε, while δ_3 can be selected based on specific design constraints:

$$10\varepsilon < \alpha_3,$$
$$\delta_2 \ll \delta_3.$$

In this application, three critical points are defined as

$$A = (\delta, \varepsilon) = (0.272, 0.106),$$
$$B = (10\delta, 4\varepsilon) = (2.72, 0.45),$$
$$C = (50\delta, 10\varepsilon) = (13.6, 1.06).$$

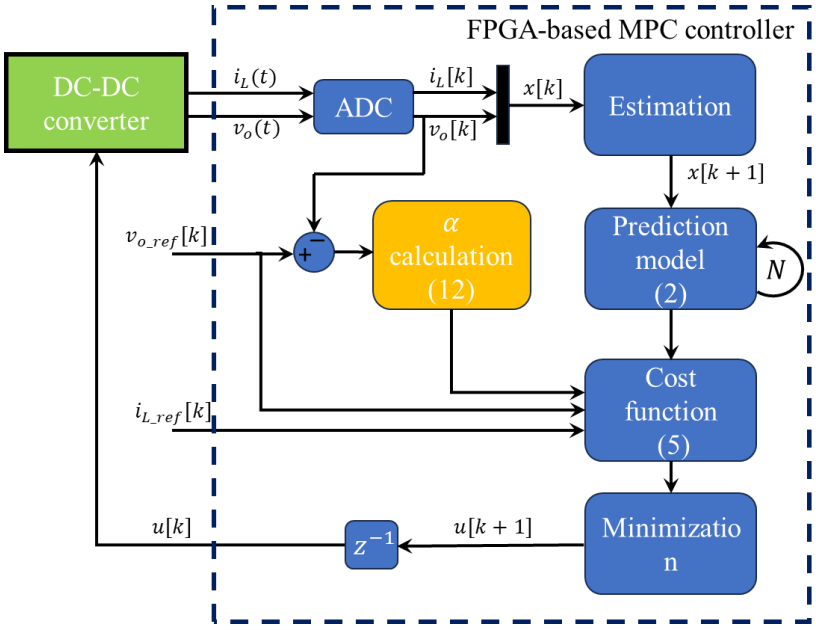


Figure 4.5: Control diagram of the proposed method.

Applying a curve-fitting method to these points results in the following function for the weight

$$\alpha = 0.0048v_{err}^2 + 0.0082v_{err}. \tag{4.12}$$

And the whole control diagram is shown in Fig. 4.5.

Table 4.2: Simulation and Experimental Setup Parameters

Parameters	**Values**
Input voltage	48 V
Output voltage	24 V
Nominal load	15 Ω
Inductance	47μH
Capacitance	94 μF
Prediction horizons	$N = 4$
Sampling frequency	200 kHz

4.4 Generalized Method for DC-DC Converter

This section demonstrates the generalization of the proposed design method to other DC-DC converters, using boost and flyback converters as examples. The application of this method relies on the following assumptions:

a) The converter is a second-order topology, either isolated or non-isolated;

b) The current control variable is the average inductor current;

c) The converter operates in Continuous Conduction Mode (CCM);

d) The sampling frequency is significantly higher than the switching frequency;

e) The circuit components are ideal, and parasitic parameters are negligible.

4.4.1 Boost Converters

Fig. 4.6 illustrates the DC-DC boost converter topology, with equivalent circuits for the switch-on and switch-off states shown in (b) and (c), respectively.

First, the order of magnitude of average inductor error, $O(\Delta i_L)$, is calculated using volt-second balance, consistent with Section 4.3.1.

Second, the relationship between the voltage error magnitude $O(v_{o,err})$ and $O(\Delta i_L)$ is derived. During the off-state, the inductor supplies the capacitor; applying KCL yields:

$$C\frac{dv_C}{dt} + \frac{v_C}{R} = i_L. \tag{4.13}$$

The discrete form corresponds to (4.9). While (4.13) strictly applies only during the off-state, the continuity of the inductor current allows this relationship to represent the system's numerical scaling for error estimation purposes.

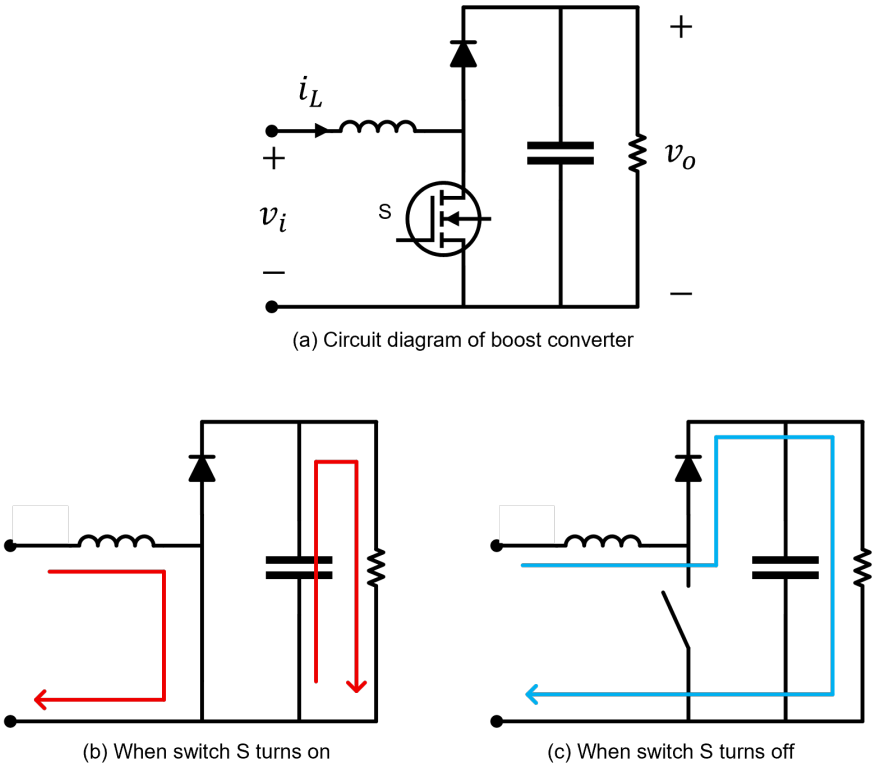


Figure 4.6: Circuit diagram of a DC-DC boost converter.

Then, applying the same approximations as in Section 4.3.2, the order of magnitude of the voltage error is estimated as:

$$O(v_{o,err}) = |v_o(k+1) - V_o| \approx \frac{T_s}{C}|(i_L(k) - \frac{1}{R}V_o)| \tag{4.14}$$

In the case of the buck converter, which represents the simplest topology,

$$\frac{1}{R}V_o = I_R = I_L,$$

because the inductor is continuously connected to the capacitor. Here, I_R represents the average load current. However, for other DC-DC converters, such as the boost converter,

$$I_R \neq I_L.$$

Therefore, it is necessary to establish the relationship between the average inductor current and the load current. For the boost converter,

$$P_{in} = P_{out} = V_{in}I_{in} = V_{in}I_L = V_oI_R. \tag{4.15}$$

Then,

$$I_R = (1 - D)I_L. \tag{4.16}$$

And substitute (4.16) into (4.14),

$$\begin{aligned} O(v_{o,err}) &= \frac{T_s}{C}|(i_L(k) - \frac{1}{R}V_o)| \\ &= \frac{T_s}{C}|(i_L(k) - I_R)| \\ &= \frac{T_s}{C}|(i_L(k) - (1 - D)I_L| \\ &= \frac{T_s}{C}|(i_L(k) - I_L| + \frac{T_s}{C}DI_L \\ &= \frac{T_s}{C}O(\Delta i_L) + m \end{aligned} \tag{4.17}$$

where, m is the modifier,

$$m = \frac{T_s}{C}DI_L.$$

Once the numerical relationship between $O(v_{o,err})$ and $O(\Delta i_L)$ is established, the weighting factors can be designed following the same procedure outlined in Section 4.3.3.

4.4.2 Flyback Converter

This section demonstrates the design method for isolated DC-DC converters, using the flyback converter as an example. The circuit diagram is presented in Fig. 4.7. The flyback converter is topologically equivalent to a buck-boost converter with the inductor replaced by a transformer.

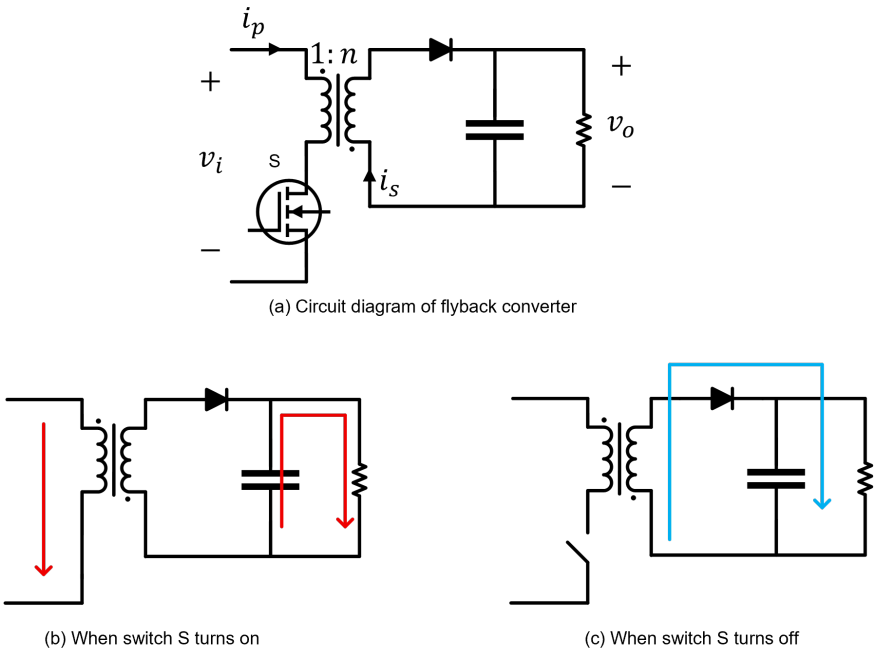


Figure 4.7: Circuit diagram of a DC-DC flyback converter.

In this configuration, the control variables are the average primary current, I_p, and the output voltage, V_o. Similar to the boost converter, the numerical relationship between $O(v_{o,err})$ and $O(\Delta i_L)$ is established based on the switch-off state. The resulting equations are identical to (4.14) and (4.17), with the exception that the modifier m differs. Specifically for the flyback converter, if the average primary current I_p is the controlled variable (where $I_L = I_p$), then:

$$P_{in} = P_{out} = V_{in} I_p = I_R R = \frac{V_o^2}{R} \tag{4.18}$$

Therefore,

$$I_R = \frac{V_{in}}{R} I_L = \frac{1-D}{nD} V_o, \tag{4.19}$$

where n is the winding ratio. And the modifier m becomes,

$$m = I_L - I_R = (1 - \frac{V_{in}}{R}) I_L = (1 - \frac{1-D}{nDR}) \frac{V_o}{R}. \tag{4.20}$$

On the other hand, if the average secondary current is controlled ($I_s = I_L$), the inductor is connected directly to the capacitor. This is similar to the buck converter case,

$$\frac{V_o}{R} = I_R = I_L. \tag{4.21}$$

Then

$$m = 0. \tag{4.22}$$

4.4.3 Overview

This section provides an overview of the proposed design method. First, the order of magnitude of the average inductor current error, $O(\Delta i_L)$, is obtained via the volt-second balance principle. Next, the numerical relationship between $O(\Delta i_L)$ and the output voltage error, $O(v_{o,err})$, is derived based on the specific circuit topology. Subsequently, the modifier m is calculated. Finally, the critical points A, B, and C are determined to establish the weighting factors. The complete workflow is illustrated in Fig. 4.8. Table 4.3 shows different modifiers for several dc-dc converters.

Table 4.3: Modifiers for different dc-dc converters

Circuits	**Buck**	**Boost**	**Buck-Boost**	**Flyback(Primary control)**
Δi_L	$(V_i - V_o)DT_s/L$	DT_sV_i/L	DT_sV_i/L	DT_sV_i/L
m	0	$\frac{T_s}{C}DI_L$	$(2-D)I_L$	$(1-\frac{1-D}{nDR})\frac{V_o}{R}$
$O(v_o)$	$O(\Delta i_L) + m$			

4.5 Experimental Results

To verify the proposed MPC algorithm, a dc-dc buck converter was built with MPC implemented on a field-programmable gate array (FPGA). The FPGA board used is the Xilinx Zedboard to implement the proposed MPC method with hardware acceleration [53]. The ADC chip is AD7476a ADC (200 ksps sampling rate) and MOSFETs type is IRF540N.The experimental setup is shown in the Fig.3.4. The system parameters are shown in Table 4.2.

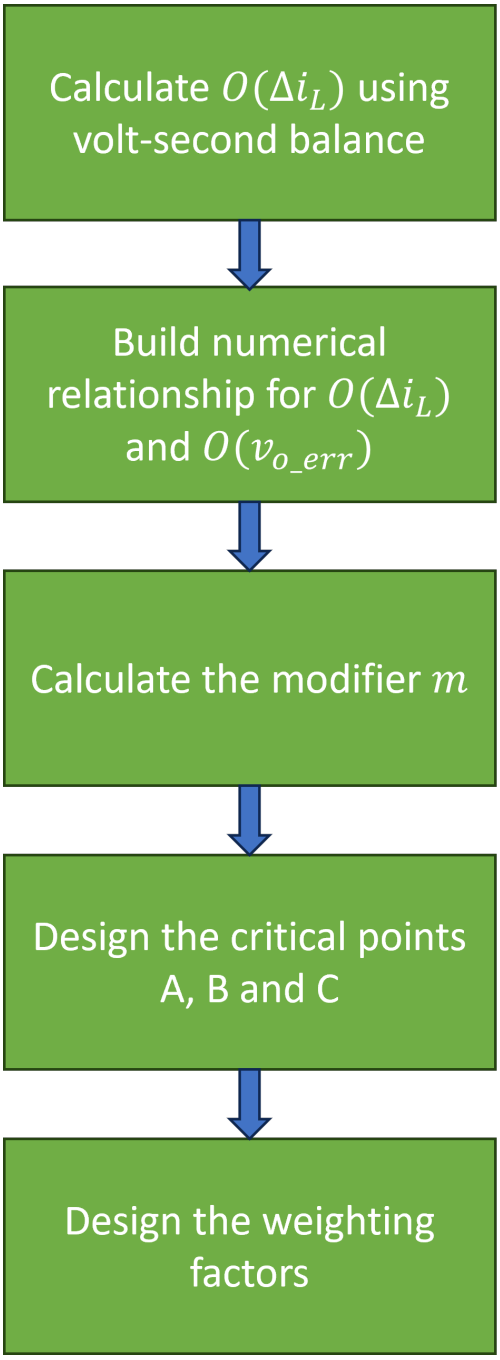


Figure 4.8: Workflow of the design method.

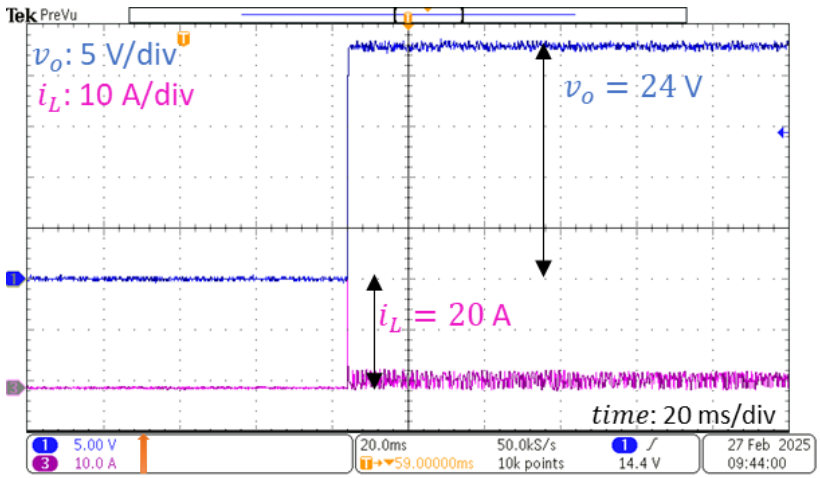


Figure 4.9: Experimental results for start-up process without adaptive cost function.

Figs. 4.9–4.12 present the experimental results of the proposed controller using the adaptive function defined in (4.12). Fig.4.9 illustrates the performance without the adaptive cost function; notably, a significant current spike occurs during the start-up process, reaching 10 times the reference current. In contrast, Fig.4.10 depicts the start-up process with the proposed controller, which facilitates a smooth transition out of the current-limiting phase. The peak inductor current is maintained below 6 A (within the safe operating region), and the resulting output voltage overshoot is limited to less than 2 V.

Furthermore, Fig.4.11 demonstrates the controller's robust response to disturbance, as the output voltage remains well-regulated during a load change from 15 Ω to 10 Ω (a variation of over 33%).

As comparison, Fig.4.12 shows that without the adaptive weighting factor, the system exhibits a substantial voltage drop during the load increment.

4.6 Conclusion

This chapter presented a validated adaptive MPC strategy for DC-DC converters that uses a single-loop, adaptive weighting function to co-regulate voltage and current. The approach enhances transient performance and stability and includes a systematic design

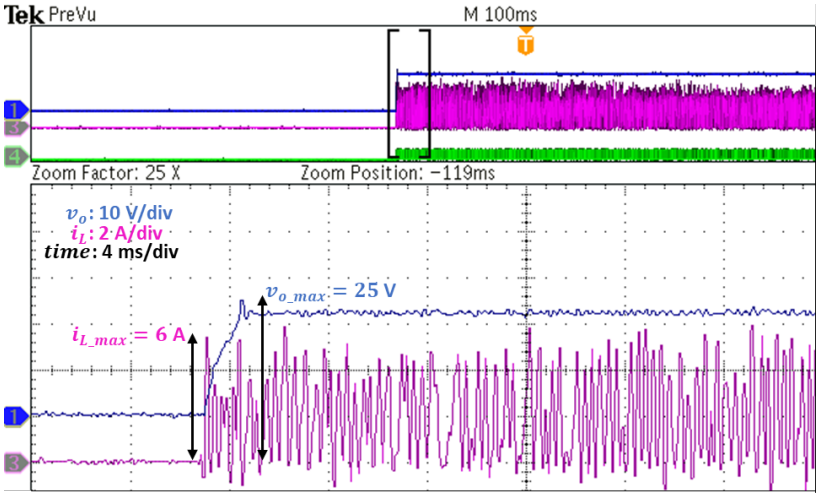


Figure 4.10: Experimental results for start-up process with the given adaptive cost function.

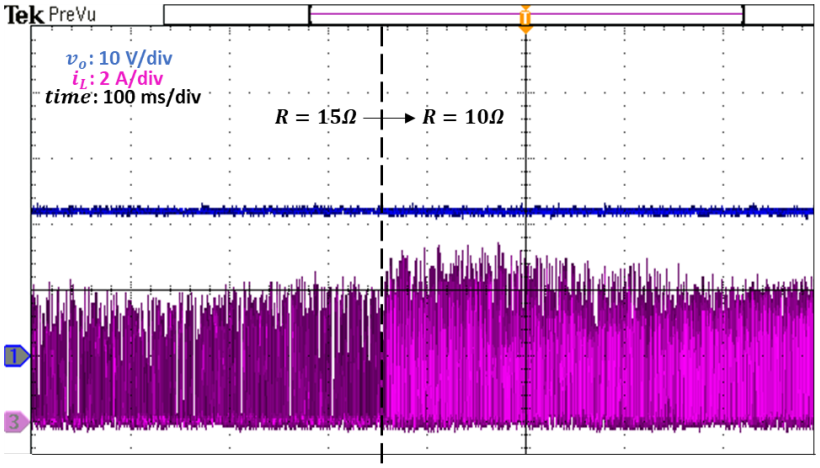


Figure 4.11: Experimental results for load change with the given adaptive function.

methodology for the weighting factor. A systematic design method for weighting factors is proposed based on numerical analysis.

In the future work, it will be shown how to extend this design method to other second order dc-dc converters.

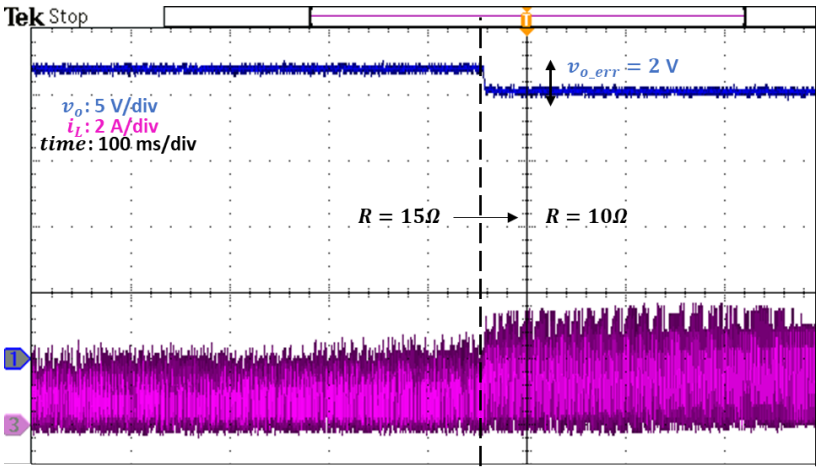


Figure 4.12: Experimental results for load change without the adaptive function.

Chapter 5

Robust Model Predictive Control for Synchronous Voltage and Current Control for DC-DC Buck Converters

5.1 Introduction

The rapid advancement of electronic technology, coupled with a continuously increasing demand for power, has led to the widespread adoption of power electronic converters. These converters are now essential across diverse applications, including industrial power systems, renewable energy integration, and data center power supplies [88]. Consequently, modern applications necessitate power electronic converters with a broader stability range, higher output voltage accuracy, quicker load transient response, and robust over-current protection [89].

Model Predictive Control (MPC) is a contemporary control method well-suited for complex systems, particularly those exhibiting high nonlinearity, multi-input-multi-output (MIMO) characteristics, and operational constraints. Within this domain, Finite Control Set Model Predictive Control (FCS-MPC) has gained widespread adoption in power converters due to its inherent ability to effectively manage multiple objectives and nonlinear control challenges [81].

The author in [90] proposed a generalized MPC for dc-dc converters and evaluated its performance for various dc-dc converters, including boost, buck and buck–boost converters. However, it is a continuous-control-set model-predictive control (CCS-MPC) and only focusing on output voltage regulation.

To enhance the dynamic performance of dc-dc converters, researchers have developed current-mode MPC. However, early attempts, such such as the model predictive current control for DC-DC boost converters, maintained a multi-loop structure [82]. This approach

involved replacing only the inner-loop PID module and still necessitated a Kalman filter for the outer loop, which added complexity to the implementation. Additionally, another MPC strategy regulated inductor current but relied on two extra loops for indirect output voltage control [83]. Although this was a current-controlled DC-DC converter, it was only effective for small-signal variations and exhibited a significant peak current during startup, which could potentially saturate the inductor. More recently, the NPI-MPC algorithm addressed the non-minimum phase behavior in dc-dc boost converters [91] but employed continuous-control set MPC, focusing solely on steady-state performance.

In [92], an Artificial Neural Network (ANN)-based MPC was proposed for DC-DC buck converters to improve robustness against parameter variations. Similarly, in [93], an offline data-driven approach was presented that works with MPC to optimally adjust its parameters. However, these modern data-driven methods demand substantial resources and time for training the offline models, which increases implementation costs and limits their real-time and embedded hardware capabilities.

Despite these advancements, prior studies have not fully exploited MPC' s of multi-objective and nonlinear control. Instead, most implementations simply replace PI modules in the inner current loop or introduce additional control modules, increasing system complexity. The multi-loop architecture also necessitates extensive parameter tuning, which introduces delays. Moreover, these designs primarily focus on small-signal voltage performance in the steady-state, often leading to significant current overshoot during start-up. Artificial intelligence (AI) methods could offer better performance, but the implementation cost is expensive and not easy to run in real-time.

Proposed in this chapter is an FCS-MPC approach for DC-DC buck converters that synchronously controls both voltage and current. By employing MPC as a single-loop controller, the method enhances transient performance and directly regulates output voltage and inductor current [54]. This method is able to regulate the output voltage and inductor

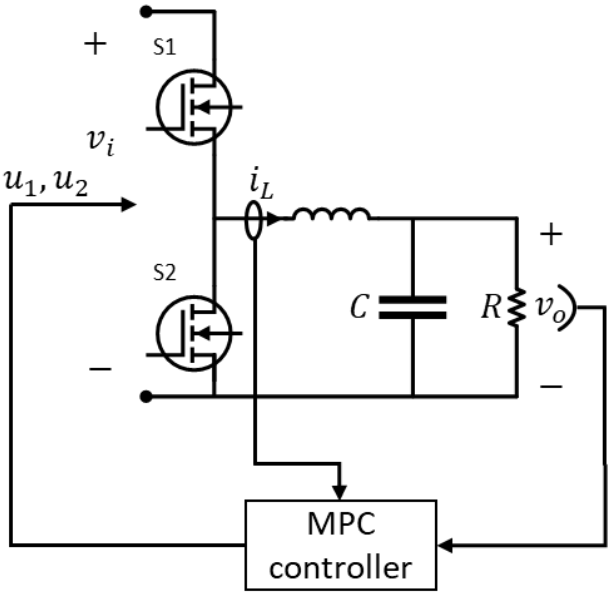


Figure 5.1: Block diagram of MPC for synchronous buck converters.

current for both small-signal and large-signal variations: from start-up process to steady-state. This eliminates the need for extra over-current protection and soft-start processes, simplifying hardware design. Furthermore, a novel cost function reformulates current control in terms of output voltage, making current regulation implicit. The proposed cost function enhances robustness against load disturbances. And it does not require a complex algorithm like machine learning, so that it is efficient for real-time and embedded operation.

This paper is organized as follows. In section 5.2, the hybrid continuous-time and discrete-time model of the dc-dc buck converter is summarized. Furthermore, the control problem is proposed and analyzed via simulation results based on Matlab/Simulink. And to address the issue, a new cost function design is proposed and analyzed. Section 5.3 offers experimental results to validate the control method's effectiveness. Finally, section 5.4 concludes this paper and future work.

5.2 Working Principles of Proposed Control Strategy

Fig. 5.1 shows the circuit topology of a dc-dc synchronous buck converter, where L is the inductance and C is the capacitance. R is the load resistance. V_i and v_o are the input and

output voltage, respectively. The switching state for transistor S1, denoted by u_1, assumes a value of 1 for the switch-on state and 0 for the switch-off state. And the switching state u_2 of transistor S2 is out of the phase with S1.

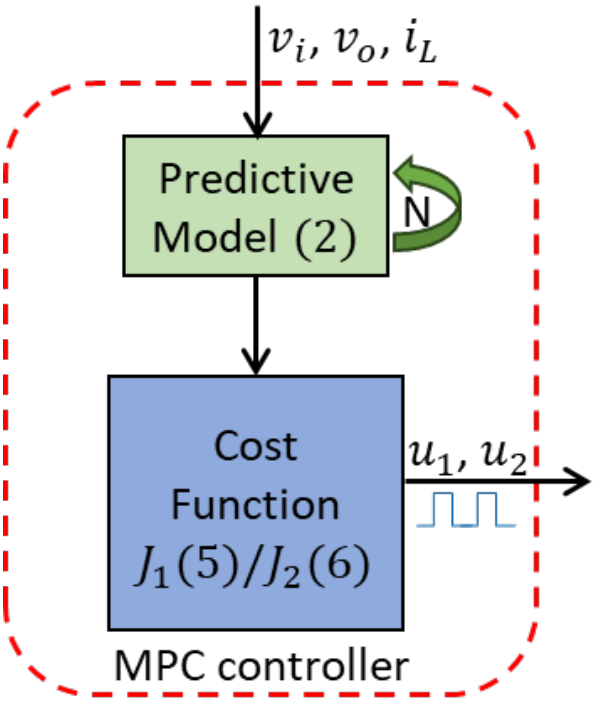


Figure 5.2: Block diagram of MPC controller

Fig. 5.2 is the block diagram of the MPC module as shown in Fig. 5.1. It contains two sub-modules: predictive model and cost function.

5.2.1 Predictive Model Design

The independent states of the converter are the inductor current and output voltage. The state vector is defined as $x(t) = [i_L(t)\ v_o(t)]^T$. Suppose that the input voltage is constant and the controlled switching state $u(t)$ is a system input, the continuous system is described by the following equations:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{5.1}$$

where the matrices A and B are given by

$$A = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{RC} \end{bmatrix}, B = \begin{bmatrix} V_i \\ 0 \end{bmatrix}.$$

The dc-dc buck converter can operate in CCM or DCM, depending on the value of inductor current $i_L(t)$. Specifically, CCM refers to the case where the current $i_L(t)$ is always positive and DCM means that the current $i_L(t)$ reaches zero during the switching cycles and at that time $u_1 = 0$. Since only the value of the inductor current changes operation in DCM, the model (5.1) can also be applied in DCM. Besides, (5.1) is a nonlinear model rather than the small-signal model. Hence, it can be applied to all running processes from start-up to steady state.

A discrete-time model is required for MPC implementation as an internal prediction model. For simplicity, the forward Euler's method is used, resulting in the following discrete-time model:

$$x(k+1) = A_d x(k) + B_d u(k), \tag{5.2}$$

where, $A_d = I + T_s A$, $B_d = T_s B$, I is the identity matrix, and T_s is the sampling period.

5.2.2 Cost Function Design

The cost function $J_1(k)$, detailed in (5.3), facilitates the optimal selection of switching state sequences for extended prediction horizons:

$$J_1(k) = \lambda < i_{L,err} > (k) + v_{o,err}(k). \tag{5.3}$$

In this specific case, the cost function encapsulates two control objectives: the average inductor current error $\lambda < i_{L,err} > (k)$ and the output voltage error $v_{o,err}(k)$.

The average current error objective is represented as the average value of the inductor current over the prediction horizon, given by

$$< i_{L,err} > (k) = \frac{1}{N} \sum_{l=k}^{k+N-1} ||i_L(l) - I_L||,$$

where N is the prediction horizon and I_L is the average current reference.

The output voltage objective and switching constraint are defined as

$$v_{o,err}(k) = \sum_{l=k}^{k+N-1} ||v_o(l) - V_o||,$$

in which V_o is the output voltage reference and $I_L = V_o/R$. The parameter λ is a weighting factor for the inductor current.

5.2.3 Simulation Results

Fig. 5.3 is the simulation result for the aforementioned MPC controlled buck converter based on Matlab/Simulink for different λ.

As shown in Fig. 5.3(a), when λ is set to a small value (0.1), the current control becomes limited, and the system prioritizes voltage control. This prioritization results in a large current spike—approximately 20 times the nominal current value—which could lead to inductor saturation. Such extreme current conditions would necessitate additional soft-start and over-current protection mechanisms. As the current weighting increases with λ (same and 10 times weighted as voltage control), the current overshoot is considerably reduced to a safe range. This adjustment also leads to an increased voltage settling time, due to the lower emphasis on voltage control.

The proposed control algorithm effectively manages transient performance during startup while maintaining steady-state stability and offering high flexibility for different performance requirements. However, it introduces inherent steady-state error under disturbances such as load and input voltage fluctuations. As shown in Fig. 5.3(b), a 50% load increment results

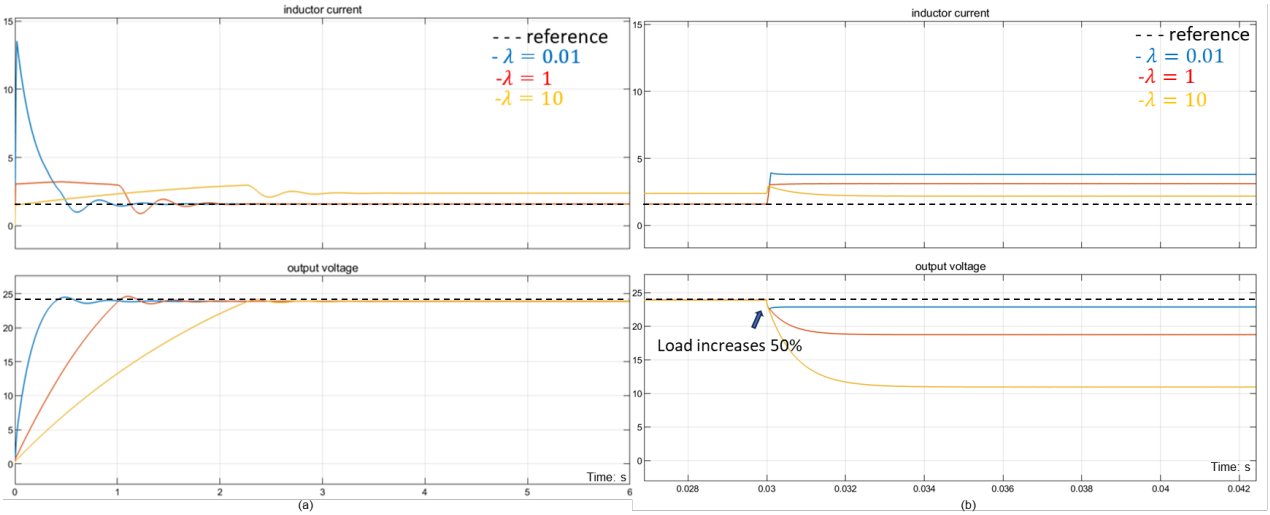


Figure 5.3: Simulation results of control effects with J_1: a) start-up process; b) load variation in the steady-state.

in distinct steady-state errors depending on the cost function weight λ. Since both current and voltage are regulated relative to their reference values, the system's operating quiescent point is predetermined. When the load varies, current and voltage control interact, leading to a new steady state that may deviate from the intended reference. As shown in Fig. 5.3(b), a 50% load increment results in distinct steady-state errors depending on the cost function weight λ.

5.2.4 New Cost Function Design

To address the above poor robustness issue, this paper introduces an improved cost function, J_2, based on (5.3). By analyzing the circuit diagram in Fig. 5.1 and its state-space model in (5.2), the state equations for inductor current and its average are derived in (5.4) and (5.5). For clarity of illustration, parasitic parameters are neglected, assuming $v_o = v_C$.

$$i_L = C\dot{v_C} + \frac{1}{R}v_C, \tag{5.4}$$

$$< i_L >= C < \dot{v_C} > + \frac{1}{R} < v_C > . \tag{5.5}$$

Substituting (5.5) into (5.3) yields:

$$\begin{aligned} J_1 &= \lambda \sum_{l=k}^{k+N-1} || < i_L(l) > -I_L|| + \sum_{l=k}^{k+N-1} ||v_C(l) - V_C|| \\ &= \lambda \sum_{l=k}^{k+N-1} ||C < \dot{v_C}(l) > + \frac{1}{R} < v_C(l) > - \frac{V_C}{R}|| \\ &\quad + \sum_{l=k}^{k+N-1} ||v_C(l) - V_C||. \end{aligned} \tag{5.6}$$

Expanding the first term in J_1, the new cost function J_2 is defined as:

$$\begin{aligned} J_2 = \lambda & \sum_{l=k}^{k+N-1} ||C < \dot{v_C}(l) > || \\ & + \lambda \sum_{l=k}^{k+N-1} ||\frac{1}{R} < v_C(l) > - \frac{V_C}{R}|| \\ & + \sum_{l=k}^{k+N-1} ||v_C(l) - V_C||. \end{aligned} \tag{5.7}$$

Since this formulation follows a normed vector space, the triangle inequality holds [94]:

$$J_1 \leq J_2. \tag{5.8}$$

Thus, J_2 serves as an upper bound of J_1. The first term in J_2 represents the average derivative of capacitor voltage. And its discrete form is given by

$$< \dot{v_C}(l) >= \frac{1}{lT_s} \int_0^{lT_s} \dot{v_C}(\tau) d\tau = \frac{v_C(T_s) - v_C(0)}{lT_s}. \tag{5.9}$$

Substituting (5.9) into (5.7) gives:

$$\begin{aligned} J_2 &= \frac{\lambda C}{lT_s} \sum_{l=k}^{k+N-1} ||v_C(l) - v_C(0)|| + \frac{\lambda}{R} \sum_{l=k}^{k+N-1} || < v_C(l) > -V_C|| \\ &\quad + \sum_{l=k}^{k+N-1} ||v_C(l) - V_C|| \\ &= \alpha \sum_{l=k}^{k+N-1} ||v_C(l) - v_C(0)|| + \beta \sum_{l=k}^{k+N-1} || < v_C(l) > -V_C|| \\ &\quad + \sum_{l=k}^{k+N-1} ||v_C(l) - V_C||, \end{aligned} \tag{5.10}$$

where α and β are new weighting factors given by

$$\alpha = \frac{\lambda C}{lT_s},\ \beta = \frac{\lambda}{R}.$$

Equation (5.10) represents the final form of the new cost function, comprising three terms dependent only on the capacitor voltage (output voltage). The first term captures voltage variations over the prediction horizon, the second term accounts for the average voltage error, and the third term represents the instantaneous voltage error. Since current control becomes implicit, eliminating competition between voltage and current control in the former cost function J_1.

Furthermore, J_2 is a conservative upper-bound of J_1 as mentioned in (5.8). J_2 expands the range of cost function penalty and decreases sensitivity to errors, changes, or uncertainties [95]. Consequently, this formulation enhances robustness towards disturbances. However, the trade-off is that J_1 is more optimal than J_2.

Table 5.1: Simulation and Experimental Setup Parameters

Parameters	**Values**
Input voltage	48 V
Output voltage	24 V
Nominal load	15 Ω
Inductance	47μH
Capacitance	94 μF
Prediction horizons	$N = 4$
Sampling frequency	200 kHz

5.3 Experimental Results

To validate the proposed MPC algorithm' s capability to regulate both voltage and current, a buck converter was implemented with MPC on a field-programmable gate array (FPGA). The FPGA board used is the Xilinx Zedboard to implement the proposed MPC method with hardware acceleration [53]. The ADC chip is AD7476a ADC (200 ksps sampling rate) and MOSFETs type is IRF540N. The experimental setup is shown in the Fig. 5.4. The system parameters are shown in Table 5.1. Additionally, in practice, weighting factors α and β are typically tuned empirically based on the desired trade-off between control performance.

5.3.1 Start-up Process

Table 5.2: Performance Comparison of Start-up Process

Weighting factors (α, β)	(15, 1)	(10, 1)	(8, 1)	(10, 2)
Peak current (A)	None	6.5	20	8.5
Voltage overshoot (V)	None	None	None	None
Voltage settling time (ms)	62	6	3	4
Voltage static error (%)	37.5	16.7	None	None

Fig.5.5 illustrates the startup behaviors using the proposed MPC cost function J_2 under four different cost function configurations. In Fig.5.5(a), where $\alpha = 15$ (high voltage change weighting, implicit current control), voltage and current overshoot are eliminated. However, the start-up process is delayed and the output voltage is only 13V so that the static error

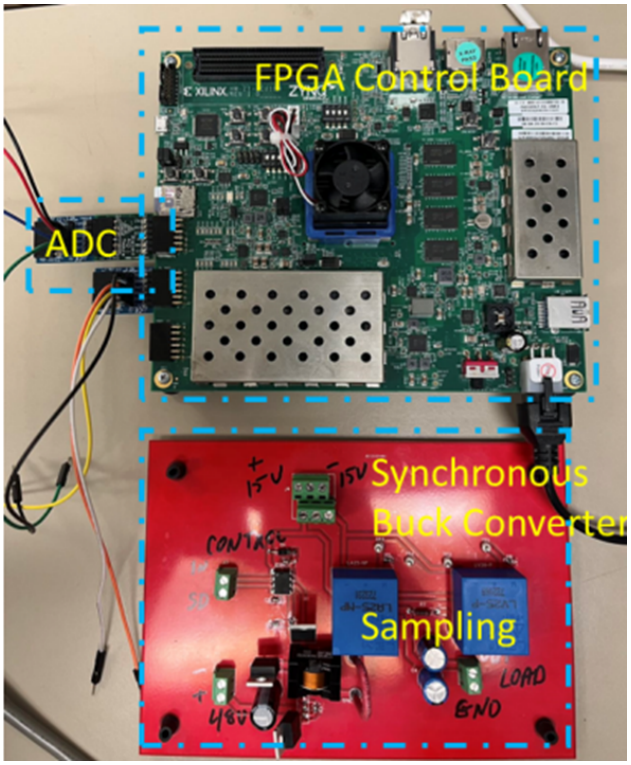
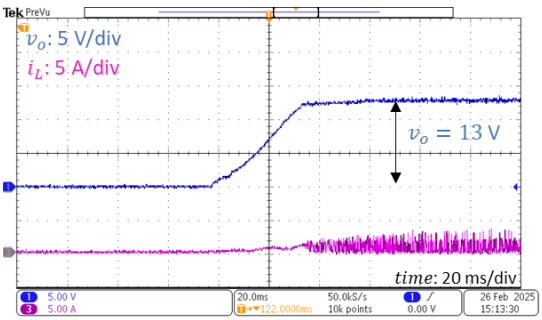


Figure 5.4: Experimental setup of synchronous buck converter with FPGA-based MPC controller

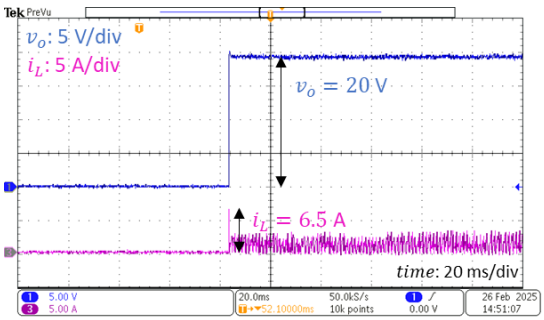
exists as 37.5% of the reference (24V). With the decrement of α to 10, the peak current is emerging as 6.5A and it still has 4V (8.5% of the reference) static error of output voltage .

In contrast, Fig. 5.5(c) with relatively small $\alpha = 8$, which is equivalent to conventional MPC voltage control only, exhibits a much faster dynamic response and no steady-state error, but produces a large current spike (20A, 12.5 times the nominal value), risking inductor saturation.

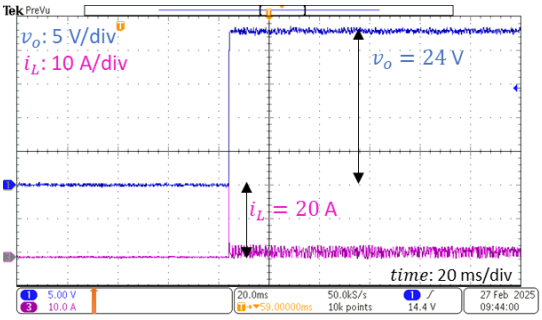
Fig. 5.5(d) demonstrates that with increased weighting of average voltage error ($\alpha = 10, \beta = 2$), a balance is achieved—no steady-state voltage error, and the current remains within the hardware' s safe limits. All performance data are included in the Table 5.2 for comparison. Weighting factors (10, 2) are the optimal for this case.



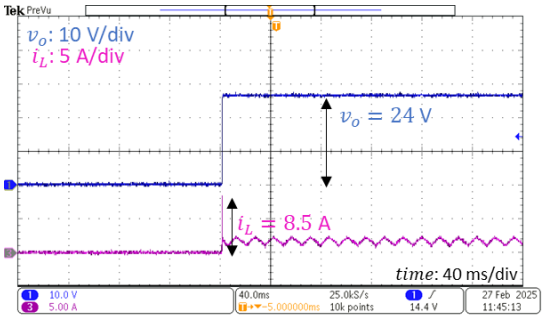
(a) $\alpha = 15, \beta = 1$



(b) $\alpha = 10, \beta = 1$



(c) $\alpha = 8, \beta = 1$



(d) $\alpha = 10, \beta = 2$

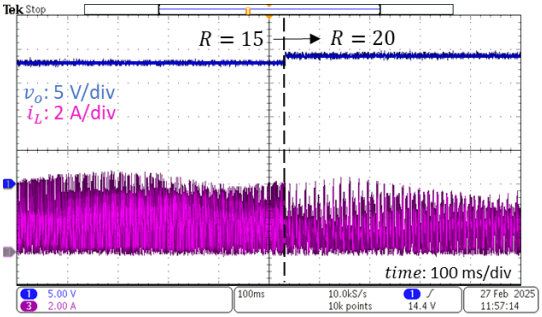
Figure 5.5: Experimental results for start-up process and steady-state with different weighting factors.

5.3.2 Load Variations

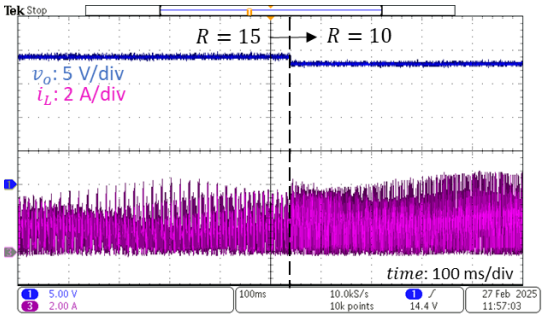
Fig.5.6 shows the experimental results of comparison with J_1 and J_2 in steady state when the load changes. Fig.5.6(a) and (b) show that J_1 ($\lambda = 10$) has obvious static error (2V) when the load was increased or decreased by more than 30%.

Fig.5.6(c) and (d) show that the proposed J_2 ($\alpha = 10, \beta = 2$) with the load resistance increased or decreased by more than 30%, the output voltage remains at the reference value.

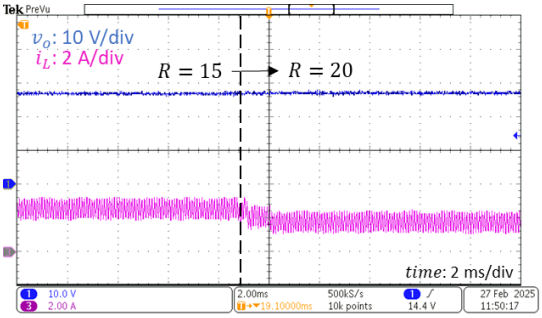
Therefore, the proposed cost function J_2 with proper weighting factors has better dynamic performance and robustness towards start-up process and load variations.



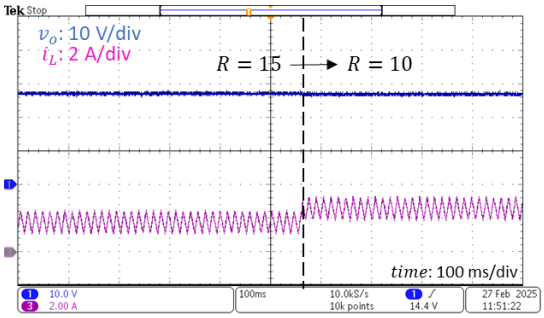
(a) J_1 with load resistance increased more than 30%.



(b) J_1 with load resistance decreased more than 30%.



(c) J_2 with load resistance increased more than 30%.



(d) J_2 with load resistance decreased more than 30%.

Figure 5.6: Experimental results for steady-state with load variations for cost functions J_1 and J_2.

5.4 Conclusion and Future Work

This chapter presented a novel MPC strategy for DC-DC buck converters, employing a single-loop control to regulate both voltage and current. This approach enhances transient performance while ensuring stable output voltage from startup to steady-state under disturbances, all while simplifying hardware implementation. The future work will discuss the weighting factors influence and theirs design as they are based on prediction horizon N, sampling period T_s, capacitance C and the load resistance R.

Chapter 6

Conclusion and Discussion of Findings

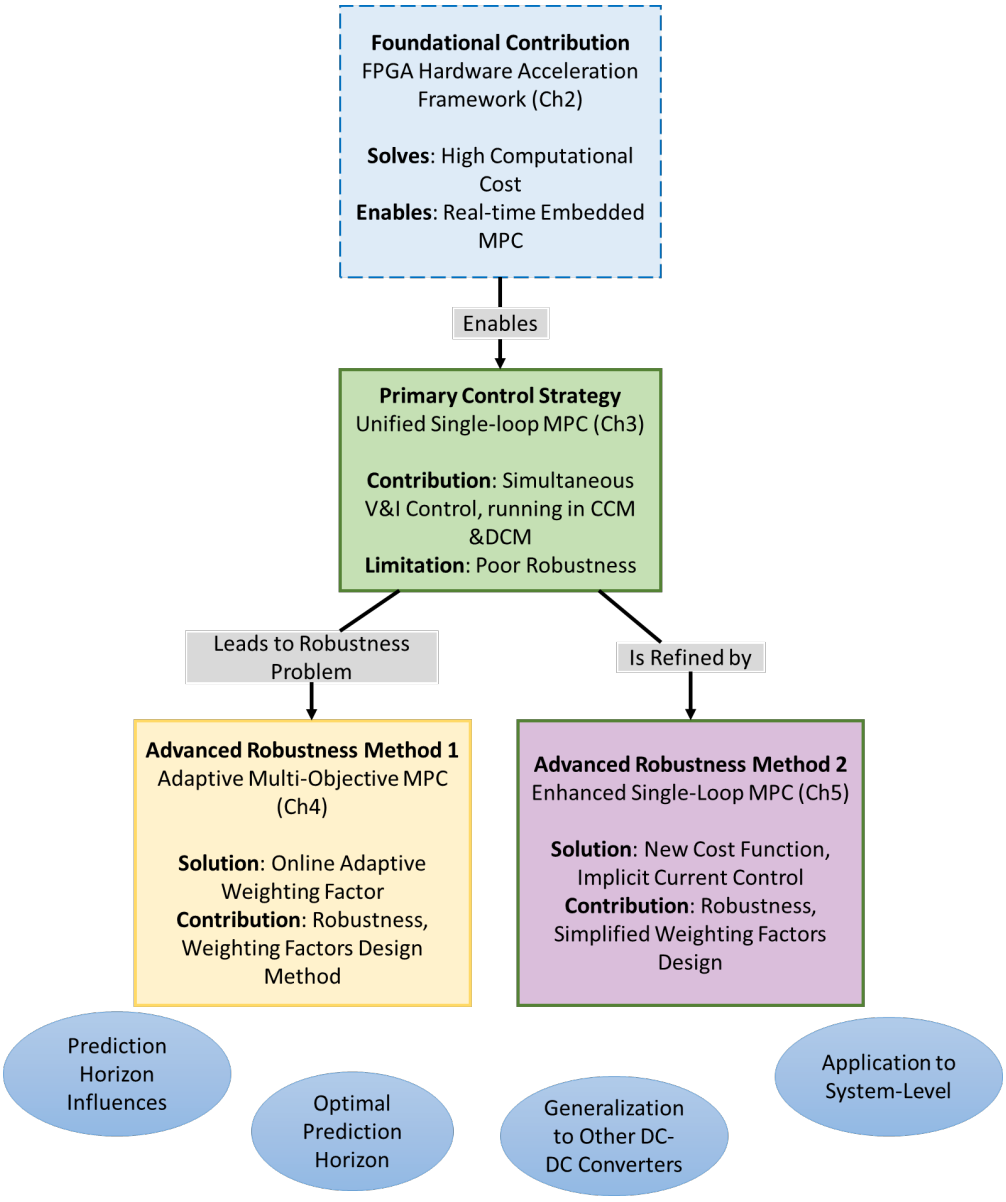


Figure 6.1: Workflow of this dissertation

This dissertation has successfully advanced the practical application of Finite-Control-Set Model Predictive Control (FCS-MPC) for DC-DC power converters. By establishing a

foundational hardware acceleration framework, this research has enabled the development and validation of advanced, robust control strategies, demonstrating a clear pathway from theoretical concept to practical realization. The workflow is shown in Fig. 6.1.

The primary contributions of this work form a cohesive research narrative. The investigation began by confronting the most significant barrier to MPC adoption—its computational expense. The development of a novel FPGA-based hardware acceleration framework successfully reduced processing times to under 1μs, creating a viable platform for real-time, embedded control. This foundational achievement enabled the subsequent design of a unified, single-loop MPC strategy that effectively co-regulates output voltage and inductor current across all operating modes (CCM and DCM). Recognizing the inherent robustness limitations of this initial strategy, the research then progressed to develop two parallel and complementary solutions. The first introduced an online adaptive weighting factor, providing dynamic and intelligent tuning to enhance performance against system variations. The second proposed a novel cost function formulation that achieves robust control through an elegant, implicit current regulation mechanism. The efficacy of each stage of this research has been rigorously confirmed through extensive simulation and experimental validation.

While the contributions presented herein are significant, they also illuminate promising avenues for future inquiry. The following areas represent the most critical next steps for building upon this work:

(a) Optimization of the Prediction Horizon: This dissertation utilized a fixed prediction horizon to ensure a consistent basis for comparison. However, the length of this horizon presents a fundamental trade-off between control performance and computational complexity. A systematic investigation into methods for optimizing the prediction horizon—or even adapting it in real-time—could yield substantial performance gains while managing implementation costs. This represents a critical research direction for refining the balance between stability and efficiency.

(b) Generalization of the Proposed Control Methods: The control strategies, particularly the novel cost function developed in Chapter 3, were validated using a buck converter topology. A valuable next step would be to generalize these methods to a wider range of DC-DC converters (e.g., boost, buck-boost) and other power electronic systems. This would significantly broaden the impact and applicability of the proposed controllers.

(c) Application to System-Level and Multi-Converter Architectures: The developed frameworks have demonstrated strong performance at the single-converter level. A logical and impactful extension of this research would be to apply these MPC strategies to more complex, system-level applications. This includes multi-phase DC-DC converters, series-parallel connected converter modules, and the coordinated control of multiple converters within a microgrid, opening new possibilities for high-performance, system-wide energy management.

Bibliography

[1] R. W. Erickson and D. Maksimovic, *Fundamentals of Power Electronics*. Springer Science & Business Media, 2007.

[2] J. D. Van Wyk and F. C. Lee, "On a future for power electronics," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 1, no. 2, pp. 59–72, 2013.

[3] B. K. Bose, "Expert system, fuzzy logic, and neural network applications in power electronics and motion control," *Proceedings of the IEEE*, vol. 82, no. 8, pp. 1303–1323, 1994.

[4] H. Komurcugil, S. Biricik, S. Bayhan, and Z. Zhang, "Sliding mode control: Overview of its applications in power converters," *IEEE Industrial Electronics Magazine*, vol. 15, no. 1, pp. 40–49, 2020.

[5] Y. Zhu, Z. Zhao, B. Shi, and Z. Yu, "Discrete state event-driven framework with a flexible adaptive algorithm for simulation of power electronic systems," *IEEE Transactions on Power Electronics*, vol. 34, no. 12, pp. 11 692–11 705, 2019.

[6] S. Bibian and H. Jin, "High performance predictive dead-beat digital controller for dc power supplies," *IEEE transactions on Power Electronics*, vol. 17, no. 3, pp. 420–427, 2002.

[7] A. Linder and R. Kennel, "Model predictive control for electrical drives," in *2005 IEEE 36th Power Electronics Specialists Conference*. IEEE, 2005, pp. 1793–1799.

[8] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Linear tracking mpc for nonlinear systems—part ii: The data-driven case," *IEEE Transactions on Automatic Control*, vol. 67, no. 9, pp. 4406–4421, 2022.

[9] ——, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2020.

[10] J. Mattingley, Y. Wang, and S. Boyd, "Receding horizon control," *IEEE Control Systems Magazine*, vol. 31, no. 3, pp. 52–65, 2011.

[11] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control," *Automatica (journal of IFAC)*, vol. 14, no. 5, pp. 413–428, 1978.

[12] S. Kouro, P. Cortés, R. Vargas, U. Ammann, and J. Rodríguez, "Model predictive control—a simple and powerful method to control power converters," *IEEE Transactions on industrial electronics*, vol. 56, no. 6, pp. 1826–1838, 2008.

[13] T. Dragičević and M. Novak, "Weighting factor design in model predictive control of power electronic converters: An artificial neural network approach," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8870–8880, 2018.

[14] Y. Alharbi, A. Darwish, and X. Ma, "A review of model predictive control for grid-connected pv applications," *Electronics*, vol. 14, no. 4, p. 667, 2025.

[15] A. Dekka, B. Wu, V. Yaramasu, R. L. Fuentes, and N. R. Zargari, "Model predictive control of high-power modular multilevel converters—an overview," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 7, no. 1, pp. 168–183, 2018.

[16] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, "Review on model predictive control: An engineering perspective," *The International Journal of Advanced Manufacturing Technology*, vol. 117, no. 5, pp. 1327–1349, 2021.

[17] B. Kouvaritakis and M. Cannon, "Model predictive control," *Switzerland: Springer International Publishing*, vol. 38, no. 13-56, p. 7, 2016.

[18] Z. Gong, X. Wu, P. Dai, and R. Zhu, "Modulated model predictive control for mmc-based active front-end rectifiers under unbalanced grid conditions," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 3, pp. 2398–2409, 2018.

[19] S. Vazquez, J. Leon, L. G. Franquelo, J. Carrasco, O. Martinez, J. Rodriguez, P. Cortes, and S. Kouro, "Model predictive control with constant switching frequency using a discrete space vector modulation with virtual state vectors," in *2009 IEEE international conference on industrial technology*. IEEE, 2009, pp. 1–6.

[20] E. Zerdali, M. Rivera, and P. Wheeler, "A review on weighting factor design of finite control set model predictive control strategies for ac electric drives," *IEEE Transactions on Power Electronics*, 2024.

[21] K. Holkar and L. M. Waghmare, "An overview of model predictive control," *International Journal of control and automation*, vol. 3, no. 4, pp. 47–63, 2010.

[22] H.-C. Moon, J.-S. Lee, J.-H. Lee, and K.-B. Lee, "Model predictive control of a grid-connected inverter to reduce current ripples and computation loads," in *2017 IEEE Applied Power Electronics Conference and Exposition (APEC)*. IEEE, 2017, pp. 1097–1102.

[23] S. Vazquez, C. Montero, C. Bordons, and L. G. Franquelo, "Model predictive control of a vsi with long prediction horizon," in *2011 IEEE International Symposium on Industrial Electronics*. IEEE, 2011, pp. 1805–1810.

[24] L. Ben-Brahim, A. Gastli, M. Trabelsi, K. A. Ghazi, M. Houchati, and H. Abu-Rub, "Modular multilevel converter circulating current reduction using model predictive control," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 6, pp. 3857–3866, 2016.

[25] S. Vazquez, J. I. Leon, L. G. Franquelo, J. Rodriguez, H. A. Young, A. Marquez, and P. Zanchetta, "Model predictive control: A review of its applications in power electronics," *IEEE industrial electronics magazine*, vol. 8, no. 1, pp. 16–31, 2014.

[26] S. Bolognani, S. Bolognani, L. Peretti, and M. Zigliotto, "Design and implementation of model predictive control for electrical motor drives," *IEEE Transactions on industrial electronics*, vol. 56, no. 6, pp. 1925–1936, 2008.

[27] 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.

[28] Q. Yang, P. Karamanakos, E. Liegmann, W. Tian, T. Geyer, R. Kennel, and M. L. Heldwein, "A fixed switching frequency direct model predictive control for neutral-point-clamped three-level inverters with induction machines," *IEEE Transactions on Power Electronics*, vol. 38, no. 11, pp. 13 703–13 716, 2023.

[29] T. Dorfling, H. du Toit Mouton, and T. Geyer, "Generalized model predictive pulse pattern control based on small-signal modeling—part 1: Algorithm," *IEEE Transactions on Power Electronics*, vol. 37, no. 9, pp. 10 476–10 487, 2022.

[30] A. I. Soliman, A. Farhan, M. Abdelrahem, and R. Kennel, "Enhanced sensorless model predictive control of induction motor based on extended kalman filter," in *2019 IEEE Conference on Power Electronics and Renewable Energy (CPERE)*. IEEE, 2019, pp. 309–313.

[31] D. Zhou, Z. Quan, Y. Li, and J. Zou, "A general constant-switching-frequency model-predictive control of multilevel converters with quasi-ps-pwm/ls-pwm output," *IEEE Transactions on Power Electronics*, vol. 35, no. 11, pp. 12 429–12 441, 2020.

[32] M. Kroneisl and V. Šmídl, "Bayesian optimization of fcs-mpc parameters for reduction of induction motor electromagnetic noise," in *2020 IEEE 29th International Symposium on Industrial Electronics (ISIE)*. IEEE, 2020, pp. 271–276.

[33] M. J. Duran, I. Gonzalez-Prieto, A. Gonzalez-Prieto, and J. J. Aciego, "The evolution of model predictive control in multiphase electric drives: A growing field of research," *IEEE Industrial Electronics Magazine*, vol. 16, no. 4, pp. 29–39, 2022.

[34] X. Liu, L. Qiu, Y. Fang, and J. Rodríguez, "Predictor-based data-driven model-free adaptive predictive control of power converters using machine learning," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 8, pp. 7591–7603, 2022.

[35] A. Lashab, D. Sera, J. M. Guerrero, L. Mathe, and A. Bouzid, "Discrete model-predictive-control-based maximum power point tracking for pv systems: Overview and evaluation," *IEEE Transactions on Power Electronics*, vol. 33, no. 8, pp. 7273–7287, 2017.

[36] M. B. Shadmand, R. S. Balog, and H. Abu-Rub, "Model predictive control of pv sources in a smart dc distribution system: Maximum power point tracking and droop control," *IEEE Transactions on Energy Conversion*, vol. 29, no. 4, pp. 913–921, 2014.

[37] J. Hu, Y. Xu, K. W. Cheng, and J. M. Guerrero, "A model predictive control strategy of pv-battery microgrid under variable power generations and load conditions," *Applied Energy*, vol. 221, pp. 195–203, 2018.

[38] D. Collet, M. Alamir, D. Di Domenico, and G. Sabiron, "Data-driven fatigue-oriented mpc applied to wind turbines individual pitch control," *Renewable Energy*, vol. 170, pp. 1008–1019, 2021.

[39] V. Petrović, M. Jelavić, and M. Baotić, "Mpc framework for constrained wind turbine individual pitch control," *Wind Energy*, vol. 24, no. 1, pp. 54–68, 2021.

[40] L. C. Henriksen, *Model predictive control of wind turbines*. Technical University of Denmark, 2011.

[41] A. Morsi, H. S. Abbas, and A. M. Mohamed, "Model predictive control of a wind turbine based on linear parameter-varying models," in *2015 IEEE Conference on Control Applications (CCA)*. IEEE, 2015, pp. 318–323.

[42] J. Hu, Y. Shan, Y. Yang, A. Parisio, Y. Li, N. Amjady, S. Islam, K. W. Cheng, J. M. Guerrero, and J. Rodríguez, "Economic model predictive control for microgrid optimization: A review," *IEEE Transactions on Smart Grid*, vol. 15, no. 1, pp. 472–484, 2023.

[43] C. Xue, J. Wang, and Y. Li, "Model predictive control for grid-tied multi-port system with integrated pv and battery storage," *IEEE Transactions on Smart Grid*, vol. 13, no. 6, pp. 4596–4609, 2022.

[44] K. Liu, L. Yang, T. Liu, and D. J. Hill, "Distributed model predictive frequency control of inverter-based networked microgrids," *IEEE Transactions on Energy Conversion*, vol. 36, no. 4, pp. 2623–2633, 2021.

[45] A. Ademola-Idowu and B. Zhang, "Frequency stability using mpc-based inverter power control in low-inertia power systems," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1628–1637, 2020.

[46] J. Hu, Y. Shan, J. M. Guerrero, A. Ioinovici, K. W. Chan, and J. Rodriguez, "Model predictive control of microgrids–an overview," *Renewable and Sustainable Energy Reviews*, vol. 136, p. 110422, 2021.

[47] Q. Yang, J. Zhou, X. Chen, and J. Wen, "Distributed mpc-based secondary control for energy storage systems in a dc microgrid," *IEEE Transactions on Power Systems*, vol. 36, no. 6, pp. 5633–5644, 2021.

[48] F. Garcia-Torres, C. Bordons, and M. A. Ridao, "Optimal economic schedule for a network of microgrids with hybrid energy storage system using distributed model predictive control," *IEEE transactions on industrial electronics*, vol. 66, no. 3, pp. 1919–1929, 2018.

[49] Z. Zhao, J. Guo, C. S. Lai, H. Xiao, K. Zhou, and L. L. Lai, "Distributed model predictive control strategy for islands multimicrogrids based on noncooperative game," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 3803–3814, 2020.

[50] S. Shahzad, M. A. Abbasi, M. A. Chaudhry, and M. M. Hussain, "Model predictive control strategies in microgrids: A concise revisit," *IEEE Access*, vol. 10, pp. 122 211–122 225, 2022.

[51] H. Tan, Y. Wang, M. Wu, Z. Huang, and Z. Miao, "Distributed group coordination of multiagent systems in cloud computing systems using a model-free adaptive predictive control strategy," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 8, pp. 3461–3473, 2021.

[52] Y. Wan, Q. Xu, and T. Dragičević, "Reinforcement learning-based predictive control for power electronic converters," *IEEE Transactions on Industrial Electronics*, 2024.

[53] Z. Guo and R. M. Nelms, "FPGA-based hardware acceleration for model predictive control of power electronic converters," in *IECON 2024-50th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2024, pp. 1–6.

[54] Z. Guo and R. Nelms, "Unified model predictive control for dc-dc buck converters: From start-up to steady-state operation," in *2025 IEEE Applied Power Electronics Conference and Exposition (APEC)*. IEEE, 2025, pp. 2703–2707.

[55] Z. Guo and R. M. Nelms, "Multi-objective fcs-mpc with adaptive weighting factors design for dc-dc converters," in *2026 IEEE Applied Power Electronics Conference and Exposition (APEC)*. IEEE, 2026.

[56] ——, "Robust model predictive control for synchronous voltage and current control for dc-dc buck converters," in *2025 IEEE Energy Conversion Conference Congress and Exposition (ECCE)*, 2025, pp. 1–6.

[57] P. Karamanakos, E. Liegmann, T. Geyer, and R. Kennel, "Model predictive control of power electronic systems: Methods, results, and challenges," *IEEE Open Journal of Industry Applications*, vol. 1, pp. 95–114, 2020.

[58] P. Karamanakos and T. Geyer, "Guidelines for the design of finite control set model predictive controllers," *IEEE Transactions on Power Electronics*, vol. 35, no. 7, pp. 7434–7450, 2019.

[59] 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, 2016.

[60] D. E. Quevedo, R. P. Aguilera, and T. Geyer, "Model predictive control for power electronics applications," *Handbook of Model Predictive Control*, pp. 551–580, 2019.

[61] V. K. Singh, R. N. Tripathi, and T. Hanamoto, "Implementation strategy for resource optimization of FPGA-based adaptive finite control set-mpc using xsg for a vsi system," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 9, no. 2, pp. 2066–2078, 2020.

[62] O. Gulbudak and E. Santi, "FPGA-based model predictive controller for direct matrix converter," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 7, pp. 4560–4570, 2016.

[63] S. Lucia, D. Navarro, O. Lucia, P. Zometa, and R. Findeisen, "Optimized FPGA implementation of model predictive control for embedded systems using high-level synthesis tool," *IEEE transactions on industrial informatics*, vol. 14, no. 1, pp. 137–145, 2017.

[64] T. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE translation journal on magnetics in Japan*, vol. 2, no. 8, pp. 740–741, 1987.

[65] M. Jeong, S. Fuchs, and J. Biela, "When FPGAs meet regionless explicit mpc: An implementation of long-horizon linear mpc for power electronic systems," in *IECON 2020 the 46th annual conference of the ieee industrial electronics society*. IEEE, 2020, pp. 3085–3092.

[66] J. M. Guerrero, M. Chandorkar, T.-L. Lee, and P. C. Loh, "Advanced control architectures for intelligent microgrids—part i: Decentralized and hierarchical control," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 4, pp. 1254–1262, 2012.

[67] M. Su, Z. Liu, Y. Sun, H. Han, and X. Hou, "Stability analysis and stabilization methods of dc microgrid with multiple parallel-connected dc–dc converters loaded by cpls," *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 132–142, 2016.

[68] Z. Guo, X. Zou, Y. Huang, Y. Kang, and K. Zou, "Full-state feedback based active damping control design for lcl-type grid-connected converter under weak grid," in *2018 IEEE International Power Electronics and Application Conference and Exposition (PEAC)*. IEEE, 2018, pp. 1–6.

[69] T. Dragičević, S. Vazquez, and P. Wheeler, "Advanced control methods for power converters in dg systems and microgrids," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 7, pp. 5847–5862, 2020.

[70] S. Chattopadhyay and S. Das, "A digital current-mode control technique for dc–dc converters," *IEEE Transactions on Power Electronics*, vol. 21, no. 6, pp. 1718–1726, 2006.

[71] R. Heydari, H. Young, F. Flores-Bahamonde, S. Vaez-Zadeh, C. Gonzalez-Castano, S. Sabzevari, and J. Rodriguez, "Model-free predictive control of grid-forming inverters

with *lcl* filters," *IEEE Transactions on Power Electronics*, vol. 37, no. 8, pp. 9200–9211, 2022.

[72] M. Narimani, B. Wu, V. Yaramasu, Z. Cheng, and N. R. Zargari, "Finite control-set model predictive control (fcs-mpc) of nested neutral point-clamped (nnpc) converter," *IEEE Transactions on Power Electronics*, vol. 30, no. 12, pp. 7262–7269, 2015.

[73] K.-Z. Liu and Y. Yokozawa, "An mpc-pi approach for buck dc-dc converters and its implementation," in *2012 IEEE International Symposium on Industrial Electronics*. IEEE, 2012, pp. 171–176.

[74] G. Bonanno and L. Corradini, "Digital predictive current-mode control of three-level flying capacitor buck converters," *IEEE Transactions on Power Electronics*, vol. 36, no. 4, pp. 4697–4710, 2020.

[75] P. Karamanakos, T. Geyer, and S. Manias, "Direct model predictive current control strategy of dc–dc boost converters," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 1, no. 4, pp. 337–346, 2013.

[76] L. Cheng, P. Acuna, R. P. Aguilera, J. Jiang, S. Wei, J. E. Fletcher, and D. D. Lu, "Model predictive control for dc–dc boost converters with reduced-prediction horizon and constant switching frequency," *IEEE Transactions on Power Electronics*, vol. 33, no. 10, pp. 9064–9075, 2017.

[77] Y. Li, S. Sahoo, T. Dragičević, Y. Zhang, and F. Blaabjerg, "Stability-oriented design of model predictive control for dc/dc boost converter," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 1, pp. 922–932, 2023.

[78] S. Peyghami, P. Palensky, and F. Blaabjerg, "An overview on the reliability of modern power electronic based power systems," *IEEE Open Journal of Power Electronics*, vol. 1, pp. 34–50, 2020.

[79] A. Afshari, M. Davari, M. Karrari, W. Gao, and F. Blaabjerg, "A multivariable, adaptive, robust, primary control enforcing predetermined dynamics of interest in islanded microgrids based on grid-forming inverter-based resources," *IEEE Transactions on Automation Science and Engineering*, vol. 21, no. 3, pp. 2494–2506, 2023.

[80] J. Rodriguez, C. Garcia, A. Mora, F. Flores-Bahamonde, P. Acuna, M. Novak, Y. Zhang, L. Tarisciotti, S. A. Davari, Z. Zhang *et al.*, "Latest advances of model predictive control in electrical drives—part i: Basic concepts and advanced strategies," *IEEE Transactions on Power Electronics*, vol. 37, no. 4, pp. 3927–3942, 2021.

[81] M. Narimani, B. Wu, V. Yaramasu, Z. Cheng, and N. R. Zargari, "Finite control-set model predictive control (fcs-mpc) of nested neutral point-clamped (nnpc) converter," *IEEE Transactions on Power Electronics*, vol. 30, no. 12, pp. 7262–7269, 2015.

[82] P. Karamanakos, T. Geyer, and S. Manias, "Direct model predictive current control strategy of dc–dc boost converters," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 1, no. 4, pp. 337–346, 2013.

[83] L. Cheng, P. Acuna, R. P. Aguilera, J. Jiang, S. Wei, J. E. Fletcher, and D. D. C. Lu, "Model predictive control for dc–dc boost converters with reduced-prediction horizon and constant switching frequency," *IEEE Transactions on Power Electronics*, vol. 33, no. 10, pp. 9064–9075, 2018.

[84] Q. Yang, J. Zhou, X. Chen, and J. Wen, "Distributed mpc-based secondary control for energy storage systems in a dc microgrid," *IEEE Transactions on Power Systems*, vol. 36, no. 6, pp. 5633–5644, 2021.

[85] I. Harbi, J. Rodriguez, E. Liegmann, H. Makhamreh, M. L. Heldwein, M. Novak, M. Rossi, M. Abdelrahem, M. Trabelsi, M. Ahmed *et al.*, "Model-predictive control of multilevel inverters: Challenges, recent advances, and trends," *IEEE Transactions on Power Electronics*, vol. 38, no. 9, pp. 10 845–10 868, 2023.

[86] S. Vazquez, D. Marino, E. Zafra, M. D. V. Peña, J. J. Rodríguez-Andina, L. G. Franquelo, and M. Manic, "An artificial intelligence approach for real-time tuning of weighting factors in fcs-mpc for power converters," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 12, pp. 11 987–11 998, 2022.

[87] M. Novak, T. Dragicevic, and F. Blaabjerg, "Weighting factor design based on artificial neural network for finite set mpc operated 3l-npc converter," in *2019 IEEE Applied Power Electronics Conference and Exposition (APEC)*. IEEE, 2019, pp. 77–82.

[88] Y. Chen, K. Shi, M. Chen, and D. Xu, "Data center power supply systems: From grid edge to point-of-load," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 11, no. 3, pp. 2441–2456, 2022.

[89] F. Blaabjerg, Y. Yang, K. A. Kim, and J. Rodriguez, "Power electronics technology for large-scale renewable energy generation," *Proceedings of the IEEE*, vol. 111, no. 4, pp. 335–355, 2023.

[90] A. Garces-Ruiz, S. Riffo, C. González-Castaño, and C. Restrepo, "Model predictive control with stability guarantee for second-order dc/dc converters," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 5, pp. 5157–5165, 2023.

[91] Y. Li, S. Sahoo, T. Dragičević, Y. Zhang, and F. Blaabjerg, "Stability-oriented design of model predictive control for dc/dc boost converter," *IEEE Transactions on Industrial Electronics*, vol. 71, no. 1, pp. 922–932, 2024.

[92] Y. Xiang, H. S.-H. Chung, R. Shen, and A. W.-L. Lo, "An ann-based output-error-driven incremental model predictive control for buck converter against parameter variations," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, 2023.

[93] K. Moradi, P. Zamani, and Q. Shafiee, "Data-driven predictive control of perturbed buck converters using a modified iterative feedback tuning algorithm," *IET Power Electronics*, vol. 17, no. 10, pp. 1314–1323, 2024.

[94] P. J. Antsaklis and A. N. Michel, *Linear systems*. Springer, 1997, vol. 8.

[95] K. Zhou and J. C. Doyle, *Essentials of robust control*. Prentice hall Upper Saddle River, NJ, 1998, vol. 104.