



Universidade Federal de Ouro Preto
Instituto de Ciências Exatas e Aplicadas
Departamento de Engenharia Elétrica



Trabalho de Conclusão de Curso

Estudo Comparativo entre Técnicas de Controle Aplicadas em um Conversor Buck

Felipe Rigotti Alves de Deus

João Monlevade, MG
2018

Felipe Rigotti Alves de Deus

Estudo Comparativo entre Técnicas de Controle Aplicadas em um Conversor Buck

Trabalho de Conclusão de curso apresentado à Universidade Federal de Ouro Preto como parte dos requisitos para obtenção do Título de Bacharel em Engenharia Elétrica pelo Instituto de Ciências Exatas e Aplicadas da Universidade Federal de Ouro Preto.

Orientador: Prof. Victor Costa da Silva Campos

Universidade Federal de Ouro Preto
João Monlevade
2018

D486e Deus, Felipe Rigotti Alves de.
Estudo comparativo entre técnicas de controle aplicadas em um Conversor Buck [manuscrito] / Felipe Rigotti Alves de Deus. - 2018.

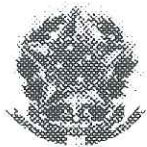
63f.: il.: color; grafs; tabs.

Orientador: Prof. Dr. Victor Costa da Silva Campos.

Monografia (Graduação). Universidade Federal de Ouro Preto. Instituto de Ciências Exatas e Aplicadas. Departamento de Engenharia Elétrica.

1. Engenharia elétrica. 2. Máquinas elétricas. 3. Controladores PID. I. Campos, Victor Costa da Silva. II. Universidade Federal de Ouro Preto. III. Título.

CDU: 621.313



ANEXO IV - ATA DE DEFESA

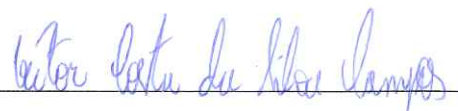
Aos 17 dias do mês de julho de 2018, às 17 horas e 10 minutos, no bloco H deste instituto, foi realizada a defesa de monografia pelo (a) formando (a) Felipe Rigotti Alves de Deus, sendo a comissão examinadora constituída pelos professores e convidados: Gabriel Azevedo Fogli, Márcio Feliciano Braga e Víctor Costa da Silva Campos.

O (a) candidato (a) apresentou a monografia intitulada: Estudo comparativo entre técnicas de controle aplicados em um conversor Buck. A comissão examinadora deliberou, por unanimidade, pela aprovação do(a) candidato(a), com a nota média 9,3, de acordo com a tabela 1. Na forma regulamentar foi lavrada a presente ata que é assinada pelos membros da comissão examinadora e pelo (a) formando(a).

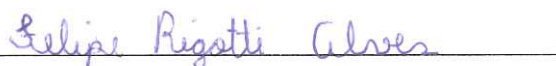
Tabela 1 – Notas de avaliação da banca examinadora

Banca Examinadora	Nota
Gabriel Azevedo Fogli	9,3
Márcio Feliciano Braga	9,3
Víctor Costa da Silva Campos	9,3
Média	9,3

João Monlevade, 17 de julho de 2018.


Víctor Costã da Silva Campos

Professor(a) Orientador(a)



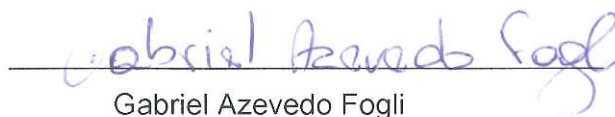
Felipe Rigotti Alves de Deus

Aluno (a)



Márcio Feliciano Braga

Professor(a) Convidado(a)



Gabriel Azevedo Fogli

Professor(a) Convidado(a)



ANEXO II - TERMO DE RESPONSABILIDADE

O texto do trabalho de conclusão de curso intitulado "Estudo Comparativo entre Técnicas de Controle Aplicadas em um Conversor Buck" é de minha inteira responsabilidade. Declaro que não há utilização indevida de texto, material fotográfico ou qualquer outro material pertencente a terceiros sem a devida citação ou consentimento dos referidos autores.

João Monlevade, 19 de Julho de 2018

Felipe Rigotti Alves de Deus
Felipe Rigotti Alves de Deus

Agradecimentos

Agradeço primeiramente a Deus por me proporcionar forças, esperança, saúde e oportunidade de cursar e concluir este curso.

Agradeço, a minha mãe Maria, e ao meu pai Laudemiro por todo apoio possível, me incentivando desde pequeno a estudar e me passando ensinamentos que serão lembrados para sempre. Também agradeço ao meu irmão, Gabriel.

Para não ser injusto não irei citar nomes, mas agradeço aos meu familiares por parte de pai e de mãe que em todo tempo me incentivaram para chegar até o final do curso.

Agradeço aos meus amigos de faculdade por todo aprendizado de amizade, companheirismo, respeito, compaixão e por todos bons momentos que vivenciamos. Principalmente aos irmãos da República Boca de Balde e da República Cativoiro.

Ao meu orientador Victor pelo apoio, dedicação, oportunidade e paciência por contribuir neste trabalho, por todo conhecimento que me passou onde foi essencial para a conclusão deste trabalho. Também não posso deixar de citar os professores Renan e Gabriel, sempre disposto a esclarecer dúvidas e dar todo suporte para a implementação prática do trabalho.

Aos amigos que formei em João Monlevade e também aos de São Mateus. Em especial, agradeço a Henrique, Cadu e Thiaguim que independente da distância mantivemos a amizade de longas datas.

Por fim, agradeço a Universidade Federal de Ouro Preto, Instituto de Ciências Exatas e Aplicadas - ICEA, Campus de João Monlevade, por me proporcionar essa oportunidade.

"Muitos são os planos no coração do homem; mas o desígnio do Senhor, esse prevalecerá."

– Salomão

Resumo

O uso de técnicas de controle preditivo na resolução de problemas de controle de grande complexidade tem crescido significadamente na última década. Isso se deve ao fato dessas abordagens apresentarem sucesso quando utilizadas em uma gama de aplicações, demonstrando sua robustez e eficiência. Quando comparado com técnica de controle clássico PID (proporcional integral derivativo), o seu uso é relativamente pequeno, mesmo que melhores resultados sejam obtidos. Diante disso, este trabalho visa estudar o desempenho de técnicas de controle clássico, moderno e preditivo. Mais especificamente, optou-se por utilizar o controlador PI clássico, o controlador por realimentação de estados e o controlador por MPC explícito para controle de tensão na saída de um conversor abaixador de tensão de topologia Buck e, então, avaliar se o controle foi efetivo ou não, comparando o sobressinal máximo, erro estacionário e tempo de acomodação. Inicialmente, em ambiente de simulação, o conversor foi projetado e os controladores apresentaram desempenho satisfatório. Por fim, os controladores foram implementados em um microcontrolador, a saber *Arduino Uno*, com o objetivo de controlar a tensão de saída em um conversor implementado em prática.

Palavras-chave: Conversor Buck, Controle MPC explícito, Controlador PID, Controle por Realimentação de Estados.

Abstract

The use of predictive control techniques in solving complex control problems has grown significantly in the last decade. This is due to the fact that these approaches are successful when used in a range of applications, demonstrating their robustness and efficiency. When compared with classical PID (proportional integral derivative) control technique, its use is relatively small, even though better results are obtained. Therefore, this work aims to study the performance of classical, modern and predictive control techniques. More specifically, we chose to use the classical PI controller, the state feedback controller and the explicit MPC controller to control the voltage at the output of a Buck topology voltage drop converter, and then evaluate whether the control was effective or not, in addition to comparing the maximum overshoot, steady error and accommodation time. Initially, in a simulation environment, the converter was designed and the controllers presented satisfactory performance. Finally, the controllers were implemented in a microcontroller, namely *Arduino*, in order to control the output voltage in a converter implemented in practice.

Keywords: Buck Converter, Controller Explicit MPC, PID Controller, State Feedback.

Lista de ilustrações

Figura 1 – Representação de um sistema malha aberta em diagrama de blocos. . .	7
Figura 2 – Representação de um sistema malha fechada em diagrama de blocos com realimentação de estado.	8
Figura 3 – Representação de um sistema malha fechada em diagrama de blocos com realimentação de estado e controle integral.	8
Figura 4 – Estratégia dos controladores MPC.	11
Figura 5 – Estrutura básica do MPC.	12
Figura 6 – Partição das regiões críticas. (a) Estado inicial da subdivisão de X em regiões; (b) Passo 1 da subdivisão; (c) Passo 2; (d) Estado final – regiões definidas.	14
Figura 7 – Princípio de Funcionamento dos Conversores CC-CC.	17
Figura 8 – Topologia conversor Buck com componentes não ideais.	19
Figura 9 – Formas de onda no conversor Buck. (a) Tensão no indutor; (b) Corrente no indutor; (c) Corrente no capacitor.	20
Figura 10 – Conversor Buck chave fechada (0, DT).	21
Figura 11 – Conversor Buck chave aberta (DT,T).	22
Figura 12 – Exemplo de sinais provenientes do circuito de modulação PWM. V_{erro} é a tensão de erro; $V_{portadora}$ é o sinal de tensão que é comparado com o sinal de erro; PWM é o sinal gerado pela comparação de V_{erro} e $V_{portadora}$	25
Figura 13 – Diagrama esquemático do sistema.	26
Figura 14 – Esquema de ligação do sensor de corrente modelo LTA 50P/SP1.	27
Figura 15 – Placa contendo conversor Buck e <i>driver</i> (destaque amarelo), condicionador do sinal de corrente (destaque vermelho), condicionador do sinal de tensão e carga (destaque verde) e <i>Arduino</i> (destaque azul).	28
Figura 16 – Diagrama de blocos para o controle da tensão.	30
Figura 17 – Bancada do experimento montada em laboratório.	35
Figura 18 – Sinal de controle PWM e sinal de tensão de saída do conversor regulado em 12V.	37
Figura 19 – Variação da carga de saída.. . . .	37
Figura 20 – Regime transitório do controlador por Realimentação de Estados.	38
Figura 21 – Regime estacionário do controlador por Realimentação de Estados.	39
Figura 22 – Regime transitório do controlador PI clássico.	39
Figura 23 – Regime estacionário do controlador PI clássico.	39
Figura 24 – Controlador por realimentação de estados ao ser submetido a variação de <i>Set-Point</i> quando em funcionamento.	40

Figura 25 – Controlador PI ao ser submetido a variação de <i>Set-Point</i> quando em funcionamento.	41
Figura 26 – Resposta do controlador por realimentação de estados a transientes de corrente. Inicialmente houve redução da corrente e posteriormente o aumento.	41
Figura 27 – Resposta do controlador PI a transientes de corrente. Inicialmente houve redução da corrente e posteriormente o aumento.	42
Figura 28 – Diagrama esquemático da simulação para o controlador MPC Explícito.	43
Figura 29 – Sinal de Tensão controlado na saída.	44
Figura 30 – Sinal de Corrente no indutor.	44
Figura 31 – Sinal a ser comparado com a triangular para gerar o PWM.	45

Sumário

1	INTRODUÇÃO	1
1.1	Motivação e Justificativa	1
1.2	Objetivos	2
1.2.1	Objetivo Geral	2
1.2.2	Objetivos Específicos	2
2	TÉCNICAS DE CONTROLE	3
2.1	Controlador PID	3
2.1.1	Estrutura	3
2.1.1.1	Ação Proporcional	4
2.1.1.2	Ação Integral	4
2.1.1.3	Ação Derivativa	4
2.1.2	PID Digital	5
2.1.3	<i>Sisotool</i>	6
2.2	Realimentação de Estados com Integrador	6
2.2.1	Alocação de Polos	10
2.3	Controle Preditivo Baseado em Modelo	11
2.3.1	MPC Explícito	13
2.3.1.1	Projeto do Algoritmo	15
3	CONVERSORES CC-CC	17
3.1	Conversor Buck	18
3.1.1	Modo de Operação Contínuo	19
3.1.2	Modelagem do Conversor	21
3.1.2.1	Chave Fechada	21
3.1.2.2	Chave Aberta	22
3.1.2.3	Modelo de Pequenos Sinais	22
3.1.3	Modulação PWM	24
4	MATERIAIS E MÉTODOS	26
4.1	Condicionamento dos Sinais	26
4.2	Circuito de Comando	27
4.3	Circuito de Potência	28
5	PROJETO DOS CONTROLADORES	29
5.1	Linearização em Ponto de Operação	29

5.2	Controlador PI	30
5.3	Controlador por Realimentação de Estados	31
5.4	Controlador MPC Explícito	32
6	RESULTADOS EXPERIMENTAIS	35
6.1	Ajuste da frequência do PWM do microcontrolador	35
6.2	Operação em Malha Aberta	36
6.3	Operação em Malha Fechada	37
6.3.1	Variando o <i>Set-Point</i>	38
6.3.2	Variando a carga	40
6.4	Simulação do Controlador por MPC Explícito	42
7	CONCLUSÃO E TRABALHOS FUTUROS	46
	REFERÊNCIAS	47
A	CONTROLADORES IMPLEMENTADOS EM <i>ARDUÍNO</i>	49

1 Introdução

1.1 Motivação e Justificativa

O controle automático desempenhou um importante avanço na área da engenharia e da ciência. Esta técnica é utilizada em diferentes sistemas, por exemplo mísseis espaciais, mísseis guiados e robôs. Além disso, o controle tornou-se parte integrante de diversos processos industriais (HERNÁNDEZ-ARROYO; DÍAZ-RODRÍGUEZ; PINZÓN-ARDILA, 2014).

O controlador PID é, de longe, a técnica de controle mais dominante em uso, mais de 90% dos laços de controle são do tipo PID. Na verdade a maioria dos laços são de fato PI porque a ação derivada não é usada com muita frequência. O controlador PID é baseado no erro de controle passado (I), presente (P) e futuro (D) e é surpreendente o quanto pode ser alcançado com uma estratégia tão simples. Nos diversos processos a serem controlados, é a primeira solução que deve ser testada para execução do controle em malha fechada (ÅSTRÖM; HÄGGLUND, 2001). Ele é usado para uma ampla gama de problemas: controle de processo, unidades de motor, memórias magnéticas e ópticas, automotivo, controle de vôo, instrumentação, etc (HERNÁNDEZ-ARROYO; DÍAZ-RODRÍGUEZ; PINZÓN-ARDILA, 2014).

Em toda graduação o controlador PID é o que recebe o maior enfoque. Mas a pergunta que fica são: "Quais são as outras técnicas de controle?". Além disso: "A resposta do PID é a melhor?". Afim de responder estas perguntas, este trabalho propõe estudar além do PID a técnica MPC explícito, que é um controlador que tem sido extensivamente estudado por pesquisadores da academia e engenheiros da indústria de processos, refinarias e petroquímicas. Um dos principais motivos para todo este enfoque no MPC é que ele pode lidar naturalmente com restrições de processo (JÚNIOR; ANDRADE, 2016).

Para tornar este trabalho multidisciplinar e verificar o funcionamento dos controladores além do nível de simulação, um conversor buck CC-CC será projetado e implementado para servir como planta para os controladores. Acredita-se que após a coleta dos resultados e comparação entre os mesmos o conteúdo deste trabalho pode ser utilizado como ferramenta didáticas a fim de complementar o ensino da Eletrônica de Potência e/ou das disciplinas que abordam técnicas de controle discreto.

1.2 Objetivos

1.2.1 Objetivo Geral

O objetivo geral deste trabalho é o projeto e implementação de três controladores baseados nas técnicas de controle PI clássico, Realimentação de estados e MPC explícito. O projeto dos mesmos foi realizado para efetuar o controle de tensão de saída em um conversor Buck.

1.2.2 Objetivos Específicos

Além do objetivo principal, uma série de outras atividades serão desenvolvidas durante o trabalho, são estas:

- Estudar o funcionamento do conversor Buck;
- Obter o modelo médio não linear que descreve o conversor Buck em espaço de estados;
- Estudar as técnicas de controle PI, Realimentação de Estados e MPC Explícito;
- Realizar a verificação do funcionamento das técnicas aplicadas através da simulação do sistema;
- Aplicar as técnicas de controle propostas no trabalho utilizando microcontrolador em um conversor implementado em prática.

2 Técnicas de Controle

Neste capítulo será apresentada uma conceituação teórica sobre as técnicas de controle utilizadas no trabalho, sendo elas, controle PID, controle por realimentação de estados e controle MPC explícito. Assim, é possível verificar a particularidade de cada técnica, bem como, entender o seu funcionamento e o processo realizado para projetar controladores.

2.1 Controlador PID

O controlador PID é um mecanismo de controle por realimentação que inicialmente calcula o erro entre a sua variável controlada (medida do processo) e o valor desejado (*set-point*), visando aplicar uma ação de controle para eliminar o desvio. A estrutura do controlador PID é composta pelas ações de controle proporcional, integral e derivativa, que podem ser combinadas originando diversos controladores (PINTO, 2014).

A gama de processos industriais que podem ser controlados usando controladores do tipo PID é grande, cuja popularidade pode ser atribuída parcialmente ao seu bom desempenho para uma faixa de operação e à sua simplicidade, que permite aos engenheiros operá-los de forma simples e direta (DORF; BISHOP, 2001). Åström e Hägglund (1995) citam uma pesquisa realizada em fábricas no Canadá onde foi constatado que em um total de 2000 malhas de controle estudadas apenas 3% não faziam uso de alguma das configurações do controlador PID. Mesmo que esses dados não sejam recentes, pesquisas apontam que tal realidade não mudou, pois mesmos que novas tecnologias relacionadas ao controle de processo sejam desenvolvidas o controlador PID se mantém apresentando um bom desempenho.

2.1.1 Estrutura

A lei de controle que descreve o controlador PID é dada por

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(\tau)}{dt} \right) \quad (2.1)$$

em que $e(t)$ é o sinal de erro. Nota-se que (2.1) apresenta três parâmetros de sintonia: o ganho proporcional K , o tempo integral T_i e o tempo derivativo T_d . A partir deles, os principais controladores encontrados na prática podem ser configurados (PINTO, 2014). Sendo eles:

- Controlador Proporcional (P) — Somente ação proporcional.

- Controlador Proporcional e Integral (PI) — Ação proporcional e ação integral.
- Controlador Proporcional e Derivativo (PD) — Ação proporcional e ação derivativa.
- Controlador Proporcional, Integral e Derivativo (PID) — Ação proporcional, ação integral e ação derivativa.

2.1.1.1 Ação Proporcional

A ação de controle gerada é diretamente proporcional a sua entrada, ou seja, ao sinal de erro em função do tempo, dada por

$$u(t) = Ke(t). \quad (2.2)$$

Na ação proporcional, o controlador pode ser entendido como um amplificador com ganho constante, de forma que, quanto maior o erro do processo, maior deve ser a ação de controle gerada. Assim o rápido ajuste da variável manipulada pode ser obtido, o que torna a dinâmica do processo mais rápida.

Quanto maior o valor do ganho K , menor será o erro em regime permanente, porém nunca será zero. Além disso, para valores muito grandes de ganho, o sistema se torna mais oscilatório podendo ficar instável (ÅSTRÖM; HÄGGLUND, 1995).

2.1.1.2 Ação Integral

A função principal é assegurar que a saída do processo tenha o valor determinado como sendo o *set-point*. O ganho proporcional geralmente provoca um sinal de erro, com uma ação integral, um erro positivo vai levar a um aumento do sinal de controle, e um erro negativo leva a uma diminuição do sinal de controle independente do tamanho do erro (ÅSTRÖM; HÄGGLUND, 1995).

Em tempo contínuo, a ação integral aumenta a ordem do sistema inserindo um polo adicional na origem. Entretanto, esse acréscimo torna a resposta do sistema mais lenta e instável. Em geral, essa ação de controle não é aplicada de maneira isolada, mas sim combinada com uma ação proporcional (PINTO, 2014).

2.1.1.3 Ação Derivativa

A função principal é melhorar a estabilidade do sistema em malha fechada (ÅSTRÖM; HÄGGLUND, 1995).

A derivada de uma função está relacionada com a tendência de variação da função em um determinado instante. Portanto, aplicar uma ação de controle proporcional à derivada do sinal de erro é o mesmo que aplicar uma ação baseada na tendência da evolução do erro. Pode-se dizer então que a ação derivativa tem característica antecipativa,

ou preditiva, e objetiva fazer com que o sistema reaja de forma mais rápida. Durante o regime permanente, a ação de controle derivativa é nula se o erro for constante.

Essa ação de controle é altamente vulnerável a ruídos, reagindo violentamente para controlá-los e introduzindo flutuações na resposta do controle. Em sinais ruidosos, recomenda-se que seja realizada a sua filtragem ou, até mesmo, deve-se evitar a sua utilização (ÅSTRÖM; HÄGGLUND, 1995).

2.1.2 PID Digital

Em sistemas digitais, o algoritmo de controle PID pode ser utilizado, porém é necessária a sua conversão para a forma discreta, em que uma nova ação de controle é executada após um intervalo regular de tempo, denominado por tempo de amostragem (T_s), e dentro do intervalo entre os tempos de amostragem a ação de controle é mantida constante (PINTO, 2014). No processo de discretização, as ações integral e derivativa devem ser aproximadas por operações equivalentes. Åström e Hägglund (1995) apresentam a aproximação realizada considerando três modos distintos, porém, nesta seção trata-se apenas o algoritmo discretizado do controlador PID.

As formas discretas dos controladores PID são conhecidas como posicionais ou de velocidade.

O algoritmo posicional é dado por

$$u(k) = u_o + K \left[e(k) + \frac{T_s}{T_i} \sum_{i=0}^k e(i) + T_d \left(\frac{e(k) - e(k-1)}{T_s} \right) \right]. \quad (2.3)$$

Atrasando (2.3) em uma amostra, ou seja, substituindo k por $k-1$, tem-se

$$u(k-1) = u_o + K \left[e(k-1) + \frac{T_s}{T_i} \sum_{i=0}^{k-1} e(i) + T_d \left(\frac{e(k-1) - e(k-2)}{T_s} \right) \right]. \quad (2.4)$$

Subtraindo a expressão encontrada da original, obtém-se o algoritmo PID de velocidade, dado por

$$u(k) = u(k-1) + K[e(k) - e(k-1)] + \frac{K_p T_s}{T_i} e(k-1) + \frac{K_p T_d}{T_s} [e(k) - 2e(k-1) + e(k-2)], \quad (2.5)$$

que também é conhecido como algoritmo incremental e faz o cálculo da diferença Δu entre as ações de controle $u(k)$ e $u(k-1)$ (ÅSTRÖM; HÄGGLUND, 1995).

Nota-se portanto que o algoritmo PID discreto depende do tempo de amostragem escolhido. Quanto menor for o tempo de amostragem, a resposta discreta fica mais próxima da ação contínua, porém o custo computacional aumenta. A escolha do tempo de amostragem adequado tem um compromisso entre esforço computacional e obtenção de informações que descrevam de forma desejada a dinâmica do sistema a ser controlado (PINTO, 2014).

2.1.3 *Sisotool*

O *Sisotool* é um ferramenta do *Matlab* que permite projetar controladores para sistemas *single-input, single output (SISO)*. Fazendo uso desta ferramenta, controladores podem ser projetados segundo:

- Ajuste automático de PID
- Ajuste automático de IMC;
- Lugar geométrico das raízes;

É válido ressaltar que o projeto dos controladores no *sisotool* é realizado com base na função de transferência (FT) do sistema.

Nesse trabalho, faz-se o uso da aplicação *PID tuning* para o projeto de um controlador PI. Para que o projeto do controlador seja realizado, inicialmente, a função de transferência do sistema deve ser importada para o *sisotool*, feito isso, a aba *Automated Tuning* deve ser selecionada e nela, o modo escolhido deve ser o interativo. Nesse modo é possível definir a largura de banda e a margem de fase do sistema controlado para que, só então, seja possível se obter a FT do controlador.

2.2 Realimentação de Estados com Integrador

A realimentação de estados é uma técnica de controle que permite fixar autovalores quaisquer para matrizes de estado do sistema em malha fechada, objetivando modificar a dinâmica do sistema. Para a sua aplicação em uma planta, deve-se assumir que todas as suas variáveis de estado são mensuráveis e estão disponíveis para serem utilizadas na realimentação (OGATA, 1995).

Como ponto de partida para seu uso os polos de malha fechada devem ser determinados com base nas especificações da resposta temporal e/ou da resposta em frequência. Supõe-se então que os polos estão alocados em $z = \mu_1, z = \mu_2, \dots, z = \mu_n$ e que, escolhendo uma matriz de ganhos apropriada para a realimentação de estados, é possível forçar o sistema a ter os polos de malha fechada nas posições desejadas, desde que o sistema original seja completamente controlável (OGATA, 1995).

O diagrama de simulação de um sistema de controle em espaço de estado pode ser apresentado como o modelo da Figura 1, cuja equação dinâmica é dada por

$$x[k + 1] = Ax[k] + Bu[k] \quad (2.6a)$$

$$y[k] = Cx[k] \quad (2.6b)$$

em que:

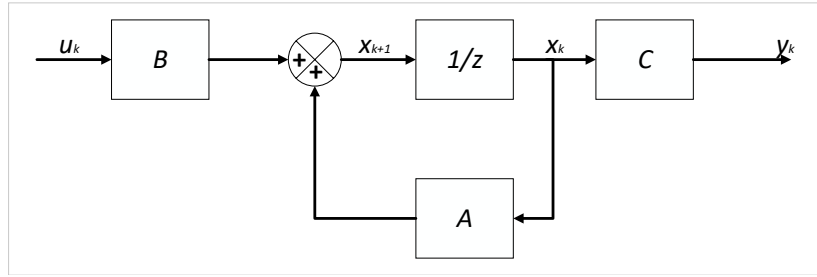


Figura 1 – Representação de um sistema malha aberta em diagrama de blocos.

Fonte: Adaptado de Ogata (1995).

- x é o vetor de estados;
- y é o sinal de saída;
- u é o sinal de controle;
- A , B , C são matrizes constantes de dimensões $n \times n$, $n \times 1$ e $1 \times n$, respectivamente.

Antes de prosseguir com a formulação da teoria de realimentação de estados é necessário conceituar a controlabilidade. Um sistema é dito controlável, se sempre for possível conduzi-lo, em um tempo finito, de um estado inicial, $x(0) = x_0$, para um estado final qualquer, x_f (CHEN, 1998). A matriz de controlabilidade é representada por

$$M = \begin{bmatrix} B & AB & A^2B & \dots & A^{n-1}B \end{bmatrix}. \quad (2.7)$$

O sistema é controlável se, e somente se, M têm n colunas linearmente independentes, isto é, se o posto da matriz de controlabilidade for igual a n ($\text{rank}(M) = n$). Portanto, se o número de colunas linearmente independentes da matriz de controlabilidade for igual a dimensão da matriz de estado, a técnica de realimentação de estados pode ser utilizada.

Retomando a formulação teórica da técnica, pode-se dizer que a sua ideia básica consiste em multiplicar cada variável de estado por um ganho fixo e, em seguida, realimentá-la para o terminal de entrada, a fim de se obter a lei de controle dada por

$$u[k] = -\mathbf{K}x[k], \quad (2.8)$$

em que K é um vetor linha denominado matriz de ganho de realimentação.

Substituindo (2.8) em (2.6a), tem-se

$$x[k + 1] = Ax[k] - BKx[k] \quad (2.9a)$$

$$= (A - BK)x[k] \quad (2.9b)$$

$$= \tilde{A}x[k] \quad (2.9c)$$

Portanto, desde que o sistema seja controlável, pode-se escolher todos os autovalores da matriz aumentada \tilde{A} de forma a tornar o sistema em malha fechada estável e fazer com que a resposta temporal do sistema siga as especificações de projeto. A Figura 2 representa graficamente o equacionamento exposto em (2.9c).

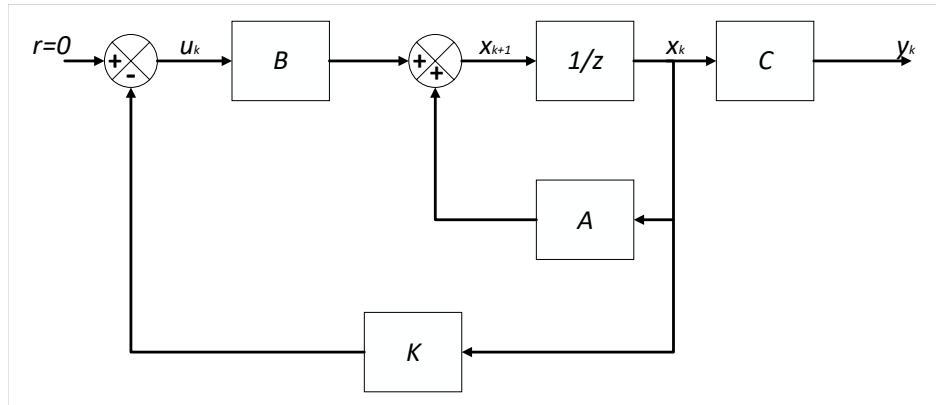


Figura 2 – Representação de um sistema malha fechada em diagrama de blocos com realimentação de estado.

Fonte: Adaptado de Ogata (1995).

Segundo Dorf e Bishop (2011), a escolha dos polos de malha fechada assegura que o sistema terá a dinâmica desejada, porém isto não é suficiente para que o sistema seja capaz de seguir uma determinada entrada, apresentando erro nulo. Uma das formas de garantir que o erro em estado estacionário seja nulo, é adicionar um integrador a ação de controle, como apresenta a Figura 3.

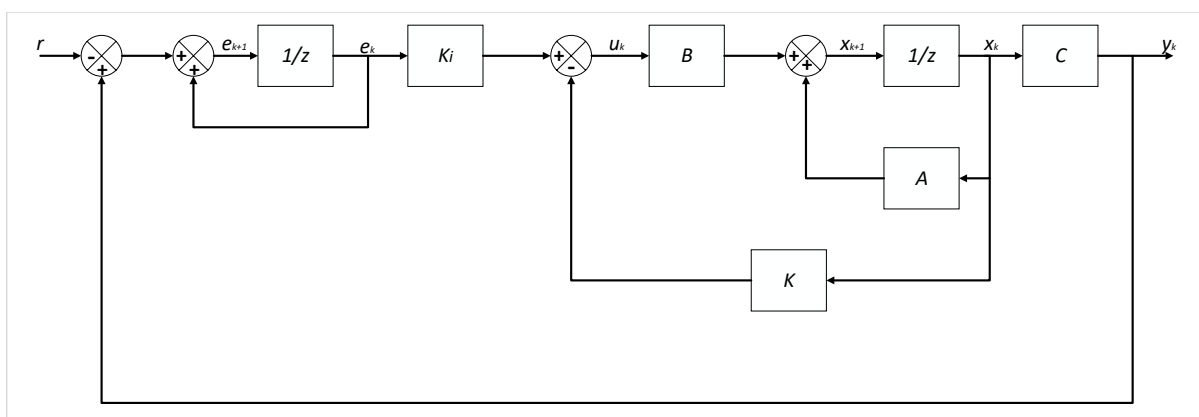


Figura 3 – Representação de um sistema malha fechada em diagrama de blocos com realimentação de estado e controle integral.

Fonte: Adaptado de Dorf e Bishop (2011).

Para tal sistema a lei de controle não será mais a apresentada em (2.8) e passa a ser representada por

$$u[k] = -\mathbf{K}x[k] - K_i e[k] \quad (2.10)$$

em que os parâmetros a serem encontrados são K e K_i , que podem ser aglutinados em um só vetor $\bar{K} = [\mathbf{K} \ K_i]$.

Uma possível solução para que a ação integral seja considerada na lei de controle por realimentação de estados, é considerar a integral do erro como um estado do sistema, de forma que suas matrizes sejam aumentadas. Do diagrama de blocos, pode-se definir o erro de rastreamento do sistema como

$$e[k+1] = e[k] + y[k] - r[k] \quad (2.11a)$$

$$= e[k] + Cx[k] - r[k] \quad (2.11b)$$

Portanto, o sistema aumentado é representado por

$$\begin{bmatrix} x[k+1] \\ e[k+1] \end{bmatrix} = \begin{bmatrix} A & 0 \\ C & I \end{bmatrix} \begin{bmatrix} x[k] \\ e[k] \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u[k] + \begin{bmatrix} 0 \\ -I \end{bmatrix} r[k] \quad (2.12a)$$

$$y = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x[k] \\ e[k] \end{bmatrix} \quad (2.12b)$$

que de forma alternativa é representado por (2.13).

$$\bar{x}[k+1] = \bar{A}\bar{x}[k] + \bar{B}u[k] - \bar{B}_r r[k] \quad (2.13a)$$

$$y = \bar{C}\bar{x}[k] \quad (2.13b)$$

O aumento do sistema com a integral do erro pode fazer com que o sistema se torne não controlável para os casos em que o número de saídas é maior que o de entradas. Isto torna necessária a realização do teste de controlabilidade para o sistema aumentado. Porém, para sistemas cujo o número de entradas é igual ou superior ao de saídas controladas, se o sistema original for controlável, o aumentado também será.

Assim como para o caso da realimentação sem integrador, se o par (\bar{A}, \bar{B}) for controlável, pode-se encontrar uma matriz de ganho \bar{K} que aloque os polos de malha fechada do sistema na posição desejada.

Charles, Nagle et al. (1995) apresentam ao menos duas formas de se encontrar os coeficientes da matriz de ganhos de realimentação. A primeira delas é igualando a equação dos autovalores da matriz aumentada aos polos desejados do sistema e a segunda é por meio da fórmula de Ackermann. Os ganhos também podem ser determinados por meio do *Matlab* fazendo uso do comando *place*, ou seja,

$$\mathbf{K} = \text{place}(A, B, p),$$

em que:

- A é a matriz de estados;
- B é a matriz de entradas;
- p é a matriz de polos desejados.

2.2.1 Alocação de Polos

A escolha dos polos em malha fechada de um sistema podem ser obtidos com base na sua resposta transitória. Dentre os parâmetros de resposta temporal, o tempo de acomodação (t_s) e o máximo sobressinal (M_p) são indispensáveis para a escolha dos polos de um sistema.

Charles, Nagle et al. (1995) apresenta a equação para o tempo de acomodação, em torno dos 2%, para um sistema de segunda ordem da seguinte forma

$$t_{s2\%} = \frac{4}{\zeta\omega_n} \quad (2.14)$$

$$(2.15)$$

e o máximo sobressinal percentual é dado por

$$M_p = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} \times 100. \quad (2.16)$$

Encontrando a frequência natural de oscilação (ω_n) e o coeficiente de amortecimento (ζ) do sistema, eles podem ser aplicados na equação (2.17) para descobrir a localização dos polos dominantes do sistema em malha fechada.

$$s = -\zeta\omega_n \pm j\omega_n\sqrt{1-\zeta^2} \quad (2.17)$$

Se a ordem do sistema a ser controlado for superior a segunda ordem, os outros polos podem ser alocados, no plano s , a esquerda dos que já foram determinados.

2.3 Controle Preditivo Baseado em Modelo

O controle preditivo baseado em modelo (MPC), em sua essência, não se refere a uma técnica de controle específica, mas sim aos métodos de controle que utilizam o modelo do processo para obtenção de um sinal de controle a partir da minimização de uma função objetivo (CAMACHO; BORDONS, 1999). Normalmente, essa função, também chamada de função de custo, é escrita como uma equação quadrática nos estados e no sinal de controle, e ao ser minimizada, a trajetória do sistema segue uma referência. Para determinação das ações de controle ótimo sobre um horizonte de tempo, utiliza-se um algoritmo de otimização, o que possibilita que restrições a ação de controle, ou aos estados do sistema possam ser consideradas (SOUZA, 2015). Pode-se dizer que os controladores projetados fazendo uso do MPC têm a seguinte estrutura:

- Uso explícito de um modelo para prever a saída do processo em instantes futuros de tempo;
- Cálculo de uma sequência de controle para minimizar a função objetivo;
- A cada instante de tempo, apenas o primeiro sinal de controle ótimo da sequência calculada é aplicado ao processo.

Segundo Camacho e Bordons (1999), os algoritmos MPC são caracterizados pela estratégia apresentada na Figura 4.

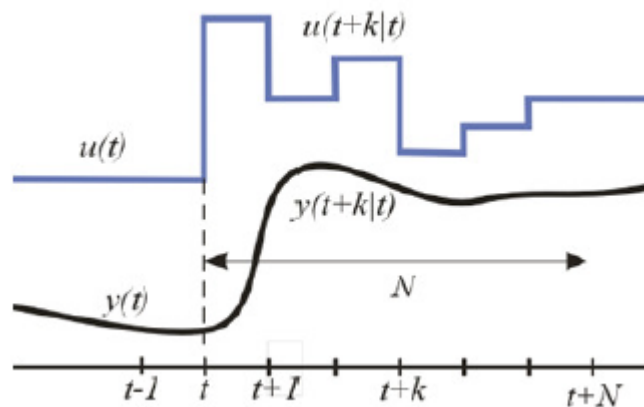


Figura 4 – Estratégia dos controladores MPC.

Fonte: Kido et al. (2011).

Por sua vez, tal estratégia pode ser descrita da seguinte forma:

1. As saídas futuras para um determinado horizonte N a partir do instante atual, chamado horizonte de predição, são previstas a cada instante t usando-se o modelo

do processo. Essas saídas previstas $y_{t+k|t}$ para $k = 1, \dots, N$ dependem dos valores de entrada e saídas até o instante t e dependem também dos sinais de controle futuros $u_{t+k|t}$, $k = 1, \dots, N - 1$, a serem determinados.

2. O conjunto de sinais de controle futuro, é calculado otimizando-se um determinado critério a fim de manter a saída do processo (e/ou seus estados) o mais próximo possível da trajetória de referência r_{t+k} , que pode ser o próprio *set-point* para a saída ou uma aproximação dele. O critério usado geralmente tem a forma de uma função quadrática dos erros entre as saídas previstas e a trajetória de referência pré-especificada.
3. O sinal de controle $u_{t|t}$ é enviado ao processo, enquanto que os próximos sinais de controle futuros são desconsiderados. No próximo instante de tempo, $u[t + 1]$ está disponível, o passo 1 é repetido com esse novo valor e todas as sequências são atualizadas.

A Figura 5 mostra um diagrama esquemático de um processo usando controle por MPC. Analisando-a pode-se fazer seguintes considerações, o modelo do sistema é usado para prever saídas futuras baseado em valores passados e presentes e nas ações de controle ótimo futuras. Estas ações são calculadas pelo otimizador levando em conta a função de custo, que por sua vez leva em conta o erro e as restrições do sistema.

Diante do que já foi descrito, pode-se dizer que um dos impeditivos para realizar a implementação de controladores MPC em alguns sistema, é o fato de se ter um modelo que descreve o processo a ser controlado de forma fiel.

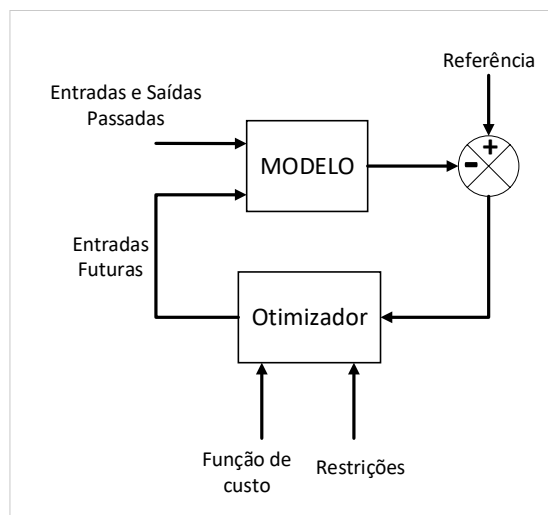


Figura 5 – Estrutura básica do MPC.

Fonte: Adaptado de Camacho e Bordons (1999).

Normalmente, o MPC é aplicado em sistemas discretos no tempo como o apresentado em (2.6). Considerando que sua função objetivo pode ser descrita por

$$J(k) = \sum_{j=0}^{N-1} x(k+j|k)^T Q x(k+j|k) + u(k+j|k)^T R u(k+j|k) \quad (2.18)$$

onde, N refere-se ao tamanho do horizonte de predição, que pode ser entendido como o número de vezes que o modelo da planta é utilizado para prever uma ação futura, e os parâmetros Q e R representam ponderações nos estados e na ação de controle, respectivamente (SOUZA, 2015). Desta forma o controlador por MPC deve encontrar uma ação de controle u dada por

$$\tilde{u} = \arg \min J(k), \quad (2.19)$$

em que \tilde{u} é um vetor com N valores da ação de controle u ,

$$\tilde{u} = (u_{k|k} \ u_{k+1|k} \ \dots \ u_{k+N-1|k}) \quad (2.20)$$

determinados pelo algoritmo de otimização, que, ao serem aplicados ao modelo (2.6) geram N valores preditos dos estados, tais como

$$\tilde{x} = (x_{k|k} \ x_{k+1|k} \ \dots \ x_{k+N-1|k}) \quad (2.21)$$

Dos N valores que foram preditos, apenas o primeiro elemento, a saber $(u_{k|k})$, é aplicado à planta.

A complexidade do problema de otimização resultante depende da quantidade de variáveis envolvidas, do tamanho do horizonte de predição utilizado e das restrições que são impostas ao sistema (KIDO et al., 2011). Como a função de custo deve ser otimizada a cada instante de amostragem com o objetivo de fornecer uma nova lei de controle para o sistema, pode-se dizer que, em sistemas de grande porte, um alto esforço de processamento é exigido para que o problema seja resolvido de forma eficiente. É nesse contexto que surge o algoritmo específico para controle preditivo conhecido por MPC explícito, tema de estudo deste trabalho.

2.3.1 MPC Explícito

O MPC explícito é uma abordagem de implementação que se baseia em programação multiparamétrica, que pode ser entendida como uma tecnologia de otimização que permite determinar a solução ótima de um problema de otimização como uma função explícita de certos parâmetros variáveis. Desta forma, ela evita que um novo problema de otimização tenha que ser resolvido na presença de alterações no parâmetro, uma vez que a solução ótima pode ser facilmente atualizada usando a função pré-computada (SCIBILIA, 2010).

No contexto do MPC, a programação multiparamétrica é utilizada para obter as entradas de controle ótimo como uma função explícita das medições dos estados, considerando-os como parâmetros do problema de otimização. As formulações explícitas de MPC fornecem a solução ótima como uma função por partes das variáveis de estado do sistema. O domínio típico de interesse é um subconjunto do espaço de estados, que é particionado em um número finito de regiões críticas. Para cada região, uma lei de controle produz o valor ideal da entrada de controle. Em conjunto, essas leis de controle formam a função por partes que representa a solução MPC ótima explícita (SCIBILIA, 2010).

A Figura 6 apresenta o procedimento pelo qual as regiões críticas são divididas. Inicialmente é definido um vetor de estados g , pertencente ao espaço X , que é delimitado pelas restrições do processo. Esse vetor é submetido ao problema de otimização e o resultado é uma região poliédrica de dimensão determinada pela quantidade de variáveis envolvidas. Determinada a região, o ponto de análise migra para dentro de uma outra região ainda inexplorada e a partir dela a programação é repetida e novas regiões são encontradas. Este processo é repetido até que o espaço X seja completamente preenchido.

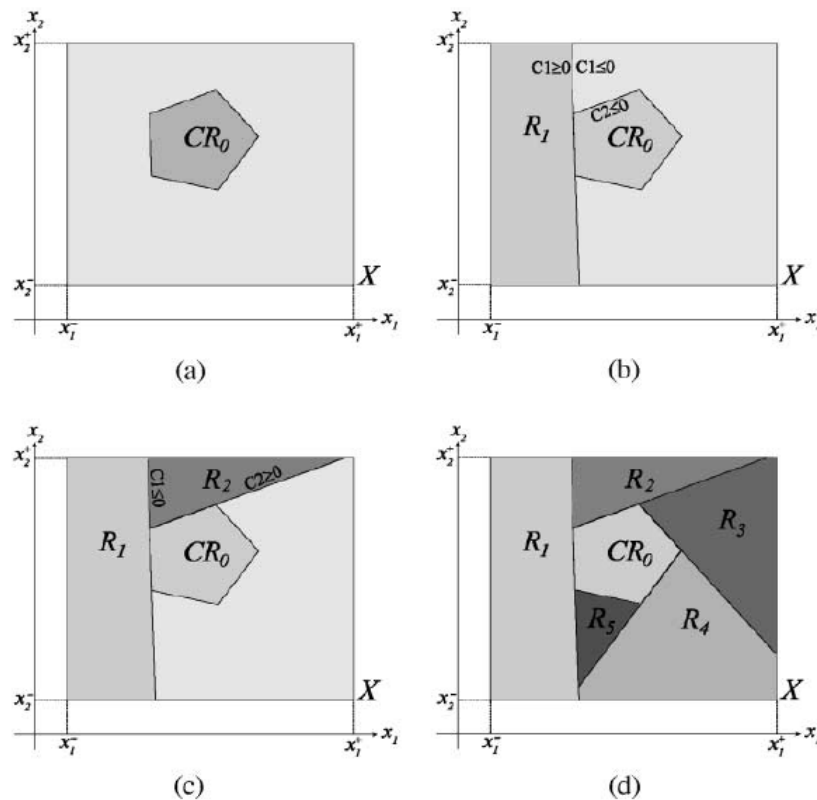


Figura 6 – Partição das regiões críticas. (a) Estado inicial da subdivisão de X em regiões; (b) Passo 1 da subdivisão; (c) Passo 2; (d) Estado final – regiões definidas.

Fonte: Bemporad et al. (2002).

Portanto, pode-se dizer que uma solução MPC explícita precisa apenas armazenar a função por partes e, iterativamente para cada nova medida, localizar qual região crítica

contém o vetor de estado atual e avaliar a lei de controle correspondente.

A abordagem do MPC de forma explícita, alterou a concepção da comunidade científica que afirmava que o MPC era uma tecnologia que só poderia ser aplicada a processos lentos devido a necessidade de se resolver um problema de otimização (em um horizonte finito) a cada instante de amostragem (ALESSIO; BEMPORAD, 2009).

O projeto dos controladores, feito neste trabalho, foi baseado em um *toolbox* proposto por Herceg et al. (2013), assim sendo, a monografia não descreve a formulação da programação multiparamétrica, mas apenas explica como a programação pode ser utilizada para obtenção da ação de controle. Em Alessio e Bemporad (2009), é apresentado um estudo sobre o MPC explícito e nele é feita a formulação geral da programação. Desta forma, caso o leitor tenha interesse em saber como é feita a formulação da programação multiparamétrica, ele pode consultar a literatura sugerida.

2.3.1.1 Projeto do Algoritmo

Assim como o controle por realimentação de estados, o controle segundo a técnica do MPC é realizado baseado na representação do sistema por meio do espaço de estados discretizado. Portanto, para descrição do projeto do algoritmo são consideradas (2.12a), que considera o erro e a entrada de referência, e (2.12b), que considera o erro.

Considerando a entrada de referência como uma variável de estado e supondo que a referência desejada não muda durante o intervalo de de predição, considera-se um modelo aumentado dado por

$$\begin{bmatrix} x[k+1] \\ r[k+1] \\ e[k+1] \end{bmatrix} = \begin{bmatrix} A & 0 & 0 \\ 0 & I & 0 \\ C & -I & I \end{bmatrix} \begin{bmatrix} x[k] \\ r[k] \\ e[k] \end{bmatrix} + \begin{bmatrix} B \\ 0 \\ 0 \end{bmatrix} u[k] \quad (2.22a)$$

$$\bar{y} = \begin{bmatrix} C & -I & \lambda I \end{bmatrix} \begin{bmatrix} x[k] \\ r[k] \\ e[k] \end{bmatrix}. \quad (2.22b)$$

Considerando que o erro entre a saída controlada, y , e o sinal de referência, r , como

$$s[k] = y[k] - r[k] \quad (2.23)$$

pode-se representar a saída do sistema aumentado por

$$\bar{y}[k] = \lambda e[k] + s[k] \quad (2.24)$$

Expandindo a última linha de (2.22a), e manipulando-a, pode-se representar $s[k]$ em termos do erro por meio de

$$e[k + 1] = Cx[k] - r[k] + e[k], \quad (2.25)$$

$$e[k + 1] = s[k] + e[k], \quad (2.26)$$

$$s[k] = e[k + 1] - e[k]. \quad (2.27)$$

Por fim, considerando que $\lambda e[k] + s[k] = 0, \forall k > \tau$, sendo τ , um grande período de tempo, a equação no instante a frente é igual a que descreve o instante anterior. Representando o que foi descrito de forma matemática

$$s[k] + \lambda e[k] - s[k + 1] - \lambda e[k + 1] = 0, \quad (2.28)$$

$$s[k + 1] - s[k] + \lambda(e[k + 1] - e[k]) = 0. \quad (2.29)$$

Relacionando (2.27) e (2.29) obtém-se uma equação que representa a dinâmica do erro em termos de λ

$$s[k + 1] = (1 - \lambda)s[k], \quad (2.30)$$

assim sendo, λ é um parâmetro variável, e pode ser ajustado para que o erro tenha um comportamento dinâmico desejado.

3 Conversores CC-CC

Este capítulo tem por finalidade apresentar o princípio de funcionamento dos conversores *choppers*, dando uma atenção especial para o conversor Buck. Para este, são abordadas as equações necessárias para fazer o correto dimensionamento dos componentes para a sua implementação prática. Também será realizada uma breve descrição do controle de fonte chaveadas por meio da modulação por largura de pulso ou PWM (do inglês, *Pulse Width Modulation*).

O funcionamento dos conversores CC-CC pode ser representado de forma satisfatória pelo esquema apresentado na Figura 7, que consiste em uma fonte de tensão contínua, E_1 , responsável por alimentar um bloco denominado de conversor CC-CC. Este, por sua vez, tem a função de fornecer energia elétrica para fonte, E_2 , através de semicondutores de potência, funcionando como interruptores, e por elementos passivos, geralmente indutores e capacitores, responsáveis por controlar o fluxo de potência entre as fontes (BARBI; MARTINS, 2008).

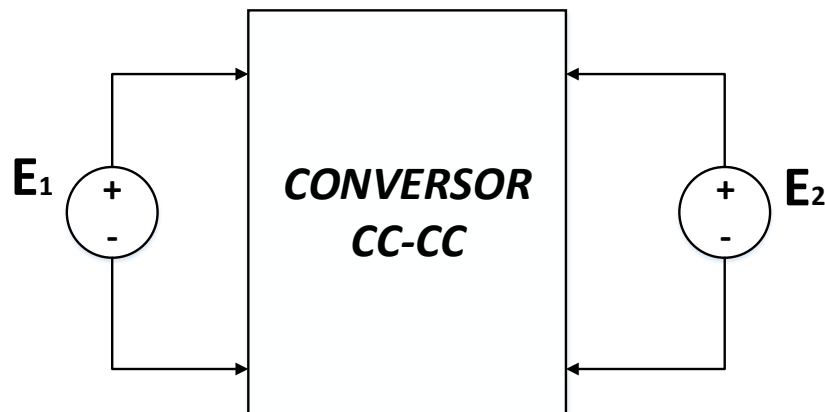


Figura 7 – Princípio de Funcionamento dos Conversores CC-CC.

Fonte: Adaptado de Barbi e Martins (2008).

Sendo assim, E_1 funciona como fonte energia e E_2 como carga, por exemplo: resistor, motor de corrente contínua, banco de baterias ou outro conversor *chopper*.

Para a engenharia, um sistema é dito ideal se apresentar rendimento de 100%, ou seja, quando toda a potência de entrada do sistema é entregue à saída. Na prática, a melhor forma de se regular uma tensão CC é fazendo uso dos conversores chaveados, uma vez que sua eficiência pode variar entre 70% e 98%, pois as perdas devido ao chaveamento são menores que as perdas em reguladores lineares, em que os reguladores lineares emulam uma resistência (BARBI; MARTINS, 2008).

A variável de controle responsável por manter a tensão de saída em um nível desejado é o *duty cycle* (D), que representa a fração do período em que o interruptor está fechado. Portanto,

$$D = \frac{t_{on}}{T} \quad (3.1)$$

onde:

t_{on} é o tempo em que a chave permanece conduzindo;

T é período de chaveamento.

Em linhas gerais, os conversores CC-CC operam em dois modos, o contínuo ou descontínuo. Quem determina o modo operando é a corrente que circula no indutor. Se durante todo o período de comutação da chave ela não vai a zero, diz-se que ele está operando no modo de condução contínuo (MCC), caso contrário o modo de operação é denominado de descontínuo (MCD). A Tabela 1 apresenta a finalidade dos principais conversores e o ganho estático em função da razão cíclica no MCC.

Tabela 1 – Principais conversores CC-CC básicos.

Conversor	Ganho Estático	Aplicação
Buck	D	abaixador
Boost	$\frac{1}{1-D}$	elevador
Buck-Boost	$\frac{D}{1-D}$	abaixador para $D < 0.5$ elevador para $D > 0.5$

Fonte: Adaptado Barbi e Martins (2008).

Dentre os conversores apresentados na Tabela 1, apenas o Buck será objeto do estudo, pois foi escolhido para ser a planta na qual os controladores irão atuar.

3.1 Conversor Buck

O conversor Buck (*step-down*) regula a tensão de saída média em um nível inferior ao da de entrada. Isto é possível através da comutação controlada de uma chave (MOSFET's, IGBT's, etc) que liga e desliga periodicamente a fonte de alimentação do circuito resultando em uma tensão mais baixa na saída. Este tipo de conversor geralmente é usado em fontes de alimentação DC reguladas, como as utilizadas em computadores, e em dispositivos portáteis compostos por diversos sub-circuitos que necessitam de diferentes níveis de tensão em relação ao fornecido pela fonte de alimentação (SKVARENINA, 2002). A Figura 8 apresenta o circuito básico do conversor Buck.

A chave S , ao ser comutada, faz com que o diodo D esteja em corte. Com isso, a energia da fonte é transferida para o indutor provocando o aumento de i_L até o ponto que

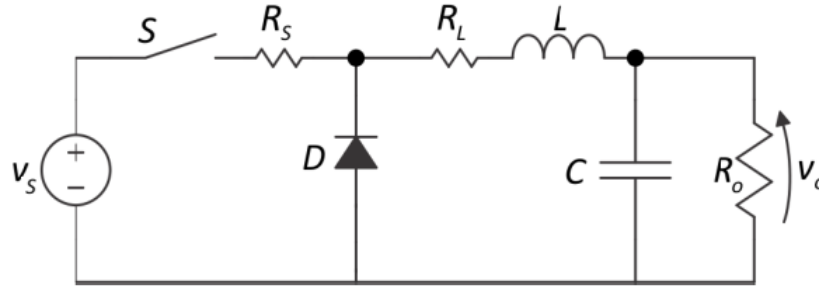


Figura 8 – Topologia conversor Buck com componentes não ideais.

Fonte: Barbi (2015).

essa corrente fique maior que a que circula na carga ($i_L > \frac{V_o}{R_o}$), nesse instante a energia passa a ser transferida para o capacitor.

Quando a chave é desligada, o diodo passa para o seu estado de condução, fazendo que a energia armazenada no indutor seja entregue a carga e ao capacitor. Enquanto o valor instantâneo da corrente no indutor for maior que a corrente na carga ($\frac{V_o}{R_o}$), o capacitor é carregado com a diferença entre as duas correntes. Quando ela for menor, o capacitor se descarrega a fim de manter tensão constante na saída do conversor. Nesse processo, é válido ressaltar que, a tensão suportada pelo diodo e pela chave deve ser no mínimo igual a tensão de entrada (POMILIO, 2007).

As formas de onda da corrente e tensão nos elementos passivos do conversor são apresentadas na Figura 9.

3.1.1 Modo de Operação Contínuo

À medida que a condução de corrente ocorre durante todo o período de comutação, a tensão média na saída é um produto do ciclo de trabalho com a tensão de entrada CC. Para o caso ideal, a equação 3.2 representa o que foi descrito (SKVARENINA, 2002).

$$V_o = D \cdot V_s \quad (3.2)$$

De acordo com (3.2), nota-se que a tensão de saída só depende do ciclo de trabalho D . Desta forma, regulando apenas o ciclo de trabalho, é possível manter uma tensão fixa na saída para diferentes valores de tensão na entrada (HART, 2010).

Segundo HART (2010), a variação de corrente no indutor durante o período em que a chave está fechada, pode ser encontrada por meio de

$$\Delta I_{L_{fechado}} = \left(\frac{V_s - V_o}{L} \right) \frac{D}{f} \quad (3.3)$$

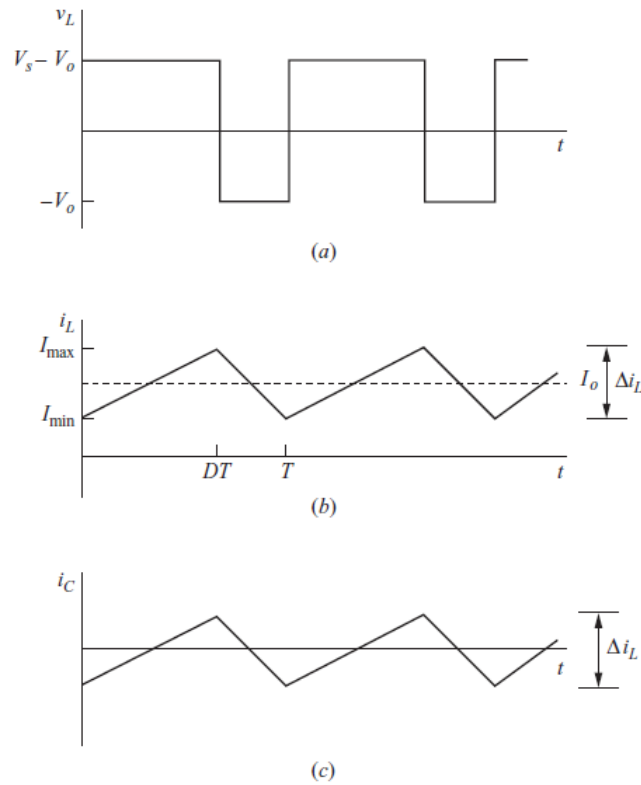


Figura 9 – Formas de onda no conversor Buck. (a) Tensão no indutor; (b) Corrente no indutor; (c) Corrente no capacitor.

Fonte: HART (2010).

E a corrente mínima que flui é dada por

$$I_{min} = V_o \left[\frac{1}{R} - \frac{(1-D)}{2Lf} \right] \frac{D}{f} \quad (3.4)$$

Manipulando (3.4), considerando que I_{min} é nula e que a frequência de chaveamento é bem definida, encontra-se uma equação que fornece o valor da indutância crítica, ou seja, que garante a existência de corrente positiva em todo período

$$L_{min} = \frac{(1-D)R_o}{2f}. \quad (3.5)$$

No âmbito da idealidade do circuito é possível que a tensão de saída seja perfeitamente constante. Para tal, basta considerar que a capacitância em paralelo com a resistência de carga têm valor infinito. Como na prática isso não é possível, deve-se encontrar um valor de capacitância que assegure a variação da tensão de saída entre os limites especificados no projeto. Em HART (2010), a seguinte relação é definida para obtenção da variação da tensão

$$\Delta V_o = \frac{V_o(1-D)}{8LCf^2}, \quad (3.6)$$

isolando C , encontra-se que a capacitância pode calculada por meio de

$$C = \frac{V_o(1 - D)}{8\Delta V_o L f^2}. \quad (3.7)$$

3.1.2 Modelagem do Conversor

Dentre as técnicas já propostas para a obtenção de modelos matemáticos para conversores estáticos CC, a mais aceita na comunidade internacional de especialistas em eletrônica de potência é a que utiliza o modelo médio do espaço de estado (BARBI, 2015).

Para que o modelo obtido seja o mais fiel possível ao sistema real, as resistências da chave e do indutor são levadas em consideração. Além disso, a modelagem realizada considera um conversor operando no MCC, cuja frequência de chaveamento é constante.

3.1.2.1 Chave Fechada

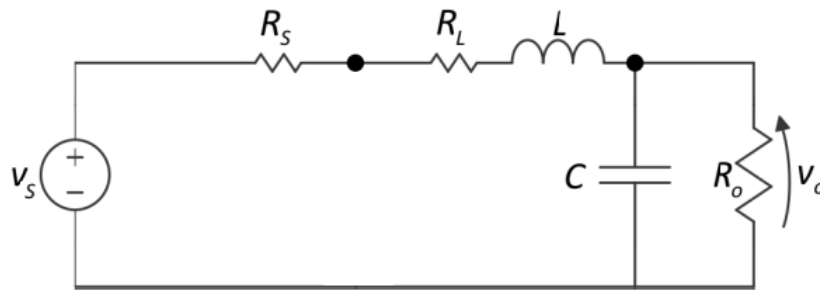


Figura 10 – Conversor Buck chave fechada (0, DT).

Fonte: Barbi (2015).

Aplicando a lei de Kirchhoff das tensões no conversor apresentado na Figura 10, são obtidas as seguintes equações

$$L \frac{di_L}{dt} = V_s - v_C - R_L \cdot i_L - R_s \cdot i_L, \quad (3.8)$$

$$C \frac{dv_C}{dt} = i_L - \frac{v_C}{R_o}. \quad (3.9)$$

A representação matricial de (3.8) e (3.9) devem ser feitas na forma de $\dot{x} = A_1x + B_1u$ e $y = C_1x$. Sendo assim

$$\begin{bmatrix} \frac{di_L}{dt} \\ \frac{dv_C}{dt} \end{bmatrix} = \begin{bmatrix} \frac{-(R_L+R_s)}{L} & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{R_o \cdot C} \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix} + \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} V_s, \quad (3.10)$$

$$y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix}. \quad (3.11)$$

Esta representação faz referência ao intervalo dT .

3.1.2.2 Chave Aberta

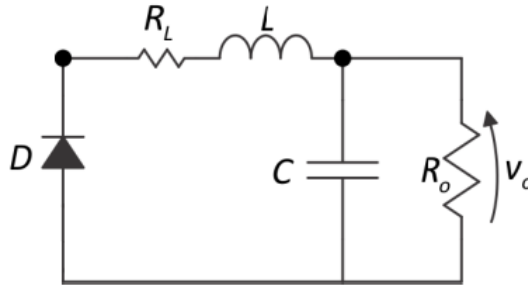


Figura 11 – Conversor Buck chave aberta (DT,T).

Fonte: Barbi (2015).

Seguindo os mesmos procedimentos adotados para o período em que a chave esta fechada, as seguintes equações são obtidas,

$$L \frac{di_L}{dt} = -v_C - R_L \cdot i_L, \quad (3.12)$$

$$C \frac{dv_C}{dt} = i_L - \frac{v_C}{R_o}. \quad (3.13)$$

Representado-as na forma de $\dot{x} = A_2x + B_2u$ e $y = C_2x$

$$\begin{bmatrix} \frac{di_L}{dt} \\ \frac{dv_C}{dt} \end{bmatrix} = \begin{bmatrix} \frac{-(R_L)}{L} & \frac{-1}{L} \\ \frac{1}{C} & \frac{-1}{R_o \cdot C} \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} V_s \quad (3.14)$$

$$y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} i_L \\ v_C \end{bmatrix} \quad (3.15)$$

Esta representação faz referência ao intervalo $(1-d)T$

3.1.2.3 Modelo de Pequenos Sinais

Para obter funções de transferência que relacionam as variáveis de estado com o *duty cycle* do circuito, é necessário ponderar as matrizes de estado no tempo, segundo as equações de 3.16 (POMILIO, 2007).

$$\dot{x} = [A_1 \cdot d + A_2 \cdot (1-d)]x + [B_1 \cdot d + B_2 \cdot (1-d)]V_s \quad (3.16a)$$

$$\dot{y} = [C_1 \cdot d + C_2 \cdot (1-d)]x \quad (3.16b)$$

Introduzindo uma pequena perturbação e fazendo a separação das componentes CC e CA, as variáveis se decompõem em:

$$\begin{aligned}x &= X + \hat{x} \\i_L &= I_L + \hat{i}_L \\v_o &= V_o + \hat{v}_o \\d &= D + \hat{d}\end{aligned}\tag{3.17}$$

Na primeira equação de (3.17) o termo X representa a componente contínua (valor médio) e o termo \hat{x} a alternada (BASTOS, 2013). Desconsiderando a perturbação no sinal de entrada, obtêm-se a representação apresentada em (3.18).

$$\dot{\hat{x}} = AX + BV + A\hat{x} + [(A_1 - A_2)X + (B_1 - B_2)V_s]d\tag{3.18}$$

Em (3.18), os termos contendo produtos envolvendo \hat{x} e d podem ser desprezados, pois, por definição, são variações pequenas. Portanto o comportamento do sistema em regime permanente é

$$\begin{aligned}AX + BV &= 0 \\X &= -A^{-1}BV\end{aligned}\tag{3.19}$$

Onde:

$$\begin{aligned}A &= A_1D + A_2(1 - D) \\B &= B_1D + B_2(1 - D)\end{aligned}\tag{3.20}$$

Considerando apenas a componente alternada em (3.18), ou seja, condição em que o sistema está oscilando em torno do ponto médio de funcionamento encontra-se 3.21.

$$\dot{\hat{x}} = A\hat{x} + [(A_1 - A_2)X + (B_1 - B_2)V_s]d\tag{3.21}$$

De forma análoga, o procedimento é realizado para a saída:

$$V_o + \hat{v}_o = CX + C\hat{x} + [(C_1 - C_2)X]d\tag{3.22}$$

Onde:

$$C = C_1D + C_2(1 - D)\tag{3.23}$$

Do processo feito anteriormente, em regime permanente, tem-se:

$$V_o = C \cdot X\tag{3.24}$$

$$\hat{v}_o = C \cdot \hat{x} + [(C_1 - C_2) \cdot X]d\tag{3.25}$$

A função de transferência pode ser obtida aplicando a transformada de Laplace em (3.19) e (3.24), conforme é mostrado a seguir.

$$\hat{x}(s) = [sI - A]^{-1}[(A_1 - A_2)X + (B_1 - B_2)V_s]\hat{d}(s) \quad (3.26)$$

$$\hat{v}_o(s) = C \cdot \hat{x}(s) + [(C_1 - C_2) \cdot X]\hat{d}(s) \quad (3.27)$$

Substituindo 3.26 em 3.27 tem-se a função de transferência da tensão de saída em relação ao *duty cycle*.

$$G(s) = \frac{\hat{v}_o(s)}{\hat{d}(s)} = C[sI - A]^{-1}[(A_1 - A_2)X + (B_1 - B_2)V_s] + (C_1 - C_2) \cdot X \quad (3.28)$$

É válido ressaltar que a função de transferência encontrada representa variações em torno do ponto médio de funcionamento, e, portanto, não deve ser usada para simular o funcionamento do circuito. A sua principal utilidade é fornecer informações para o projeto dos controladores (BASTOS, 2013). O ponto médio corresponde aos valores médios obtidos em regime permanente que é definido pelo vetor X . Este deve ser escrito em função dos valores CC das entradas e da largura de pulso média que definida no projeto do conversor (POMILIO, 2007). Para obtenção da função de transferência da corrente em relação ao *duty cycle*, basta seguir o procedimento que foi apresentado.

3.1.3 Modulação PWM

São inúmeras as formas existentes para efetuar o controle da tensão de saída, a fim de mantê-la em um nível desejado. Porém, a mais utilizada e escolhida para o projeto proposto é a modulação PWM.

Neste tipo de modulação a frequência de chaveamento é constante, de forma que, apenas a largura do pulso de acionamento da chave que efetua o controle da tensão de saída é variado. O controle deve ser realizado em malha fechada, pois há a necessidade de se ter amostras do nível da tensão de saída em tempo real para obtenção do sinal de erro e readequação da largura de pulso (LOPES, 2012).

A tensão de saída amostrada é comparada com um sinal de referência (*set-point*), e o resultado desta comparação é chamado de tensão de erro. Por sua vez, a tensão de erro é modulada através de um sinal periódico, do tipo dente de serra ou triangular, e a saída desta modulação é uma onda retangular que define o tempo de condução da chave do conversor. Quando a amplitude da tensão de erro é maior que a da portadora, então o sinal PWM assumirá nível alto, e a chave irá conduzir, caso contrário a chave estará aberta. Assim, o tempo de condução da chave é proporcional a amplitude da tensão de erro, e a frequência de chaveamento do conversor é estabelecida pela frequência do sinal

da portadora (BARBI; MARTINS, 2008). O processo descrito é apresentado graficamente na Figura 12.

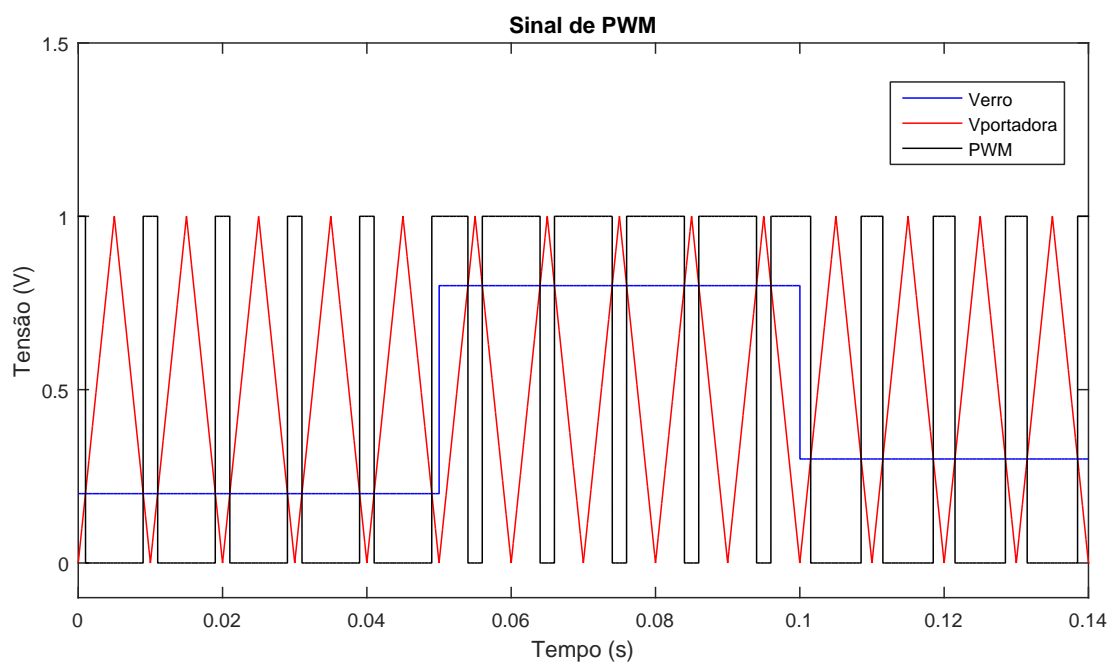


Figura 12 – Exemplo de sinais provenientes do circuito de modulação PWM. V_{erro} é a tensão de erro; $V_{portadora}$ é o sinal de tensão que é comparado com o sinal de erro; PWM é o sinal gerado pela comparação de V_{erro} e $V_{portadora}$.

4 Materiais e Métodos

Neste capítulo são descritos os materiais e métodos utilizados na execução deste trabalho. A Figura 13 apresenta o diagrama do sistema com as etapas necessárias para se promover o controle efetivo da tensão do conversor.

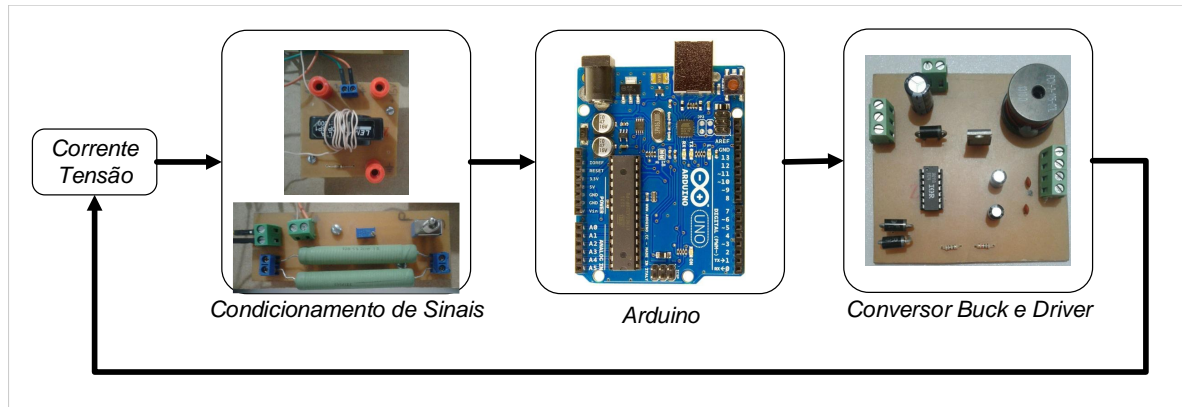


Figura 13 – Diagrama esquemático do sistema.

4.1 Condicionamento dos Sinais

É necessário fazer o condicionamento dos sinais de tensão e corrente para que, quando amostrados, possam ser lidos pelo microcontrolador.

Para a leitura da tensão, foi projetado um simples divisor de forma a limitar o valor de tensão a ser lido pelo microcontrolador em $5V$, pois o conversor A/D do *Arduino Uno* tem capacidade de fazer leitura de valores de tensão que estão compreendidos no intervalo de $0V$ e $5V$. Um trimmer de $10k\Omega$ foi conectado em paralelo com a carga, e a sua resistência foi regulada da seguinte forma: quando a tensão de saída do conversor for $15V$, a tensão de saída do divisor, que é lida pelo microcontrolador, é $5V$.

Para leitura da corrente no indutor foi utilizado um sensor modelo *LTA 50P/SP1*, que possui largura de banda máxima de $100kHz$, capacidade de leitura de $50A_{rms}$ e relação de transformação $1mA/1A$. O seu uso nesta aplicação é para fazer leituras de corrente inferiores a $3A$, portanto foi enrolado um fio de cobre totalizando nove voltas, para reduzir a sua capacidade máxima de leitura para $5.55A$.

A Figura 14 apresenta o diagrama esquemático para medição da corrente, onde o resistor de saída (R_{meas}) é indispensável. Devido ao efeito Hall, em que se baseia o princípio de funcionamento do sensor, uma tensão proporcional é gerada sob o resistor quando circula corrente no sensor. Assim, o microcontrolador irá ler a tensão gerada e a interpretará como sendo uma corrente. Com isso, pode-se regular a resistência para que ao passar $1A$ se tenha uma queda de $1V$ no resistor.

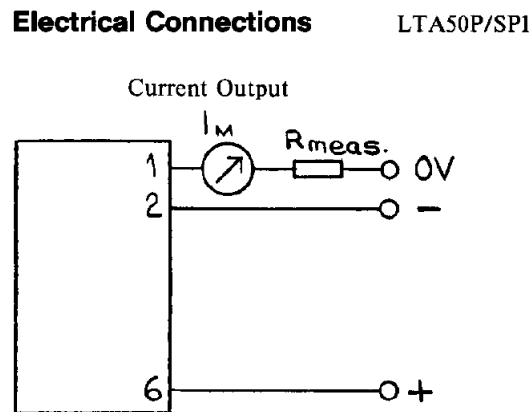


Figura 14 – Esquema de ligação do sensor de corrente modelo LTA 50P/SP1.

Fonte: Datasheet do componente.

O circuito condicionador da corrente pode ser visto na Figura 15. Nela pode ser visto o sensor utilizado, juntamente com um resistor de 100Ω responsável por gerar o sinal de tensão a ser lido pelo Arduíno, cuja magnitude é a mesma da corrente que flui pelos enrolamentos do sensor. Em situações nas quais a corrente de carga seja superior à $5A$, um circuito externo deve ser projetado para adequar a tensão no resistor aos níveis do conversor A/D.

4.2 Circuito de Comando

O circuito de comando é responsável por realizar o controle da chave semicondutora do circuito. Tal chave é essencial para o funcionamento do conversor, pois é o seu controle que determina quanto de energia será transmitida da entrada para a saída. No trabalho em questão um MOSFET modelo *IRF830*, foi utilizado para fazer o chaveamento.

Para que o MOSFET passe para o seu estado de condução, uma tensão entre $10V$ e $20V$ deve ser aplicado no *gate*. Como o sinal PWM do *Arduino*, atinge magnitude de $5V$ quando em nível alto é necessário fazer o uso de um *drive* para adequação dos níveis de tensão. O CI *IR2110* é um *drive* de alta velocidade e alta tensão com duas saídas independentes podendo operar em *High Side* ou *Low Side*. *High Side* é um termo utilizado quando o *drive* é ligado a um transistor MOSFET de junção NPN em que seu terminal de fonte não é conectado ao sinal de terra e sim à carga. *Low Side* é quando o MOSFET de junção NPN tem seu terminal de fonte conectado ao sinal de terra e a carga é conectada à fonte do transistor

Na aplicação o MOSFET operou na configuração *High Side*, desta forma apenas uma das saídas do *IR2110* foi utilizada.

4.3 Circuito de Potência

A Tabela 2 apresenta as especificações de projeto do sistema a ser controlado.

Tabela 2 – Parâmetros de projeto do Conversor Buck.

Parâmetro	Símbolo	Valor
Tensão de Entrada	V_s	15V
Tensão de Saída	V_o	12V
Frequência de Chaveamento	f_s	3.921kHz
Indutância	L	1mH
Capacitância	C	470 μ F
Resistência de Carga	R	22 Ω
Diodo	—	FUR460
Semicondutor MOSFET	—	IRF830
Driver	—	IR2110

Considerando o circuito como ideal, para que condição de operação apresentada seja atingida é necessário que o *duty cycle* do sinal de PWM seja igual a 0.8. Em regime permanente, a corrente média drenada pela fonte tem magnitude de aproximadamente 0.5454A. Conhecendo os parâmetros de projeto, o ciclo de trabalho e a corrente média drenada pela fonte, encontra-se, segundo (3.3) e (3.6), a ondulação da corrente no indutor (ΔI_L) de 0.6A e a ondulação de tensão na saída (ΔV_o) de 0.04V, respectivamente.

O conversor juntamente com o *Arduino* e os condicionadores de sinal podem ser vistos na Figura 15, na qual é destacada a função de cada circuito.

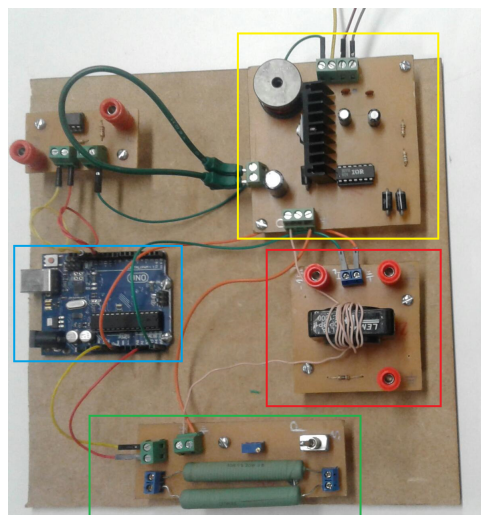


Figura 15 – Placa contendo conversor Buck e *driver* (destaque amarelo), condicionador do sinal de corrente (destaque vermelho), condicionador do sinal de tensão e carga (destaque verde) e *Arduino* (destaque azul).

5 Projeto dos Controladores

O capítulo em questão apresenta a linearização do sistema em torno de um ponto de operação visando tratá-lo como se fosse linear. Após a linearização é realizada a discretização das matrizes obtidas fim de projetar os controladores em tempo discreto. A discretização foi realizada usando um tempo de amostragem $T_s = 0.00025s$.

De posse da representação do sistema em espaço de estados é feita a descrição do processo de projeto dos controladores.

5.1 Linearização em Ponto de Operação

Os controladores são projetados segundo o modelo linearizado do sistema em torno de um ponto de operação. Em (3.28) é apresentada FT do sistema que relaciona tensão de saída e o *duty cycle*. Através dela é possível definir as matrizes do espaço de estados que representam o sistema, de forma que

$$\begin{aligned} A &= A \\ ; B &= (A_1 - A_2)X + (B_1 - B_2)V_s \\ ; C &= C \\ ; D &= (C_1 - C_2). \end{aligned} \tag{5.1}$$

A variável X representa o ponto de operação do sistema, que para o caso em questão é

$$X = \begin{bmatrix} i_L \\ v_C \end{bmatrix} = \begin{bmatrix} 0.5454 \\ 12 \end{bmatrix}.$$

Conhecendo o ponto de operação e o valor dos componentes do conversor, as matrizes A , B e C do sistema contínuo do modelo em pequenos sinais que foram encontradas são

$$A = \begin{bmatrix} 0 & -1000 \\ 2127.1 & 212.8 \end{bmatrix}, \tag{5.2a}$$

$$B = \begin{bmatrix} 15000 \\ 0 \end{bmatrix}, \tag{5.2b}$$

$$C = \begin{bmatrix} 0 & 1 \end{bmatrix}. \tag{5.2c}$$

Discretizando-as com o comando *c2d* do *Matlab*, tem-se

$$A_d = \begin{bmatrix} 0.9348 & -0.2416 \\ 0.51401 & 0.9114 \end{bmatrix}, \quad (5.3a)$$

$$B_d = \begin{bmatrix} 3.6679 \\ 0.9785 \end{bmatrix}, \quad (5.3b)$$

$$C_d = \begin{bmatrix} 0 & 1 \end{bmatrix}. \quad (5.3c)$$

Para o projeto do controlador PI, as matrizes contínuas são utilizadas para obtenção da função de transferência do sistema e as matrizes discretizadas para o projeto dos controladores segundo as técnicas de realimentação de estados e MPC Explícito.

5.2 Controlador PI

Villalva (2010) sugere que, para um resultado satisfatório no projeto de controladores PI pode-se definir uma frequência de corte limitada a um décimo da frequência de chaveamento do sistema, e Pomilio (2014) propõe que, escolhendo uma margem de fase entre 60° e 80° é possível se manter um bom compromisso entre velocidade e reduzida oscilação no regime transitório.

Em geral, para que seja feito o controle da tensão de saída de conversores é empregado o controle em casata, Figura 16 (MARTINZ, 2007). Para o projeto da malha em cascata, é necessário se ter uma malha mais estável com dinâmica rápida, que é definida como malha interna, sendo neste caso a malha de corrente. Enquanto que a malha mais externa, que é a mais lenta, deve enxergar o conjunto Malha Interna + Planta como parte de um processo único a ser controlado.

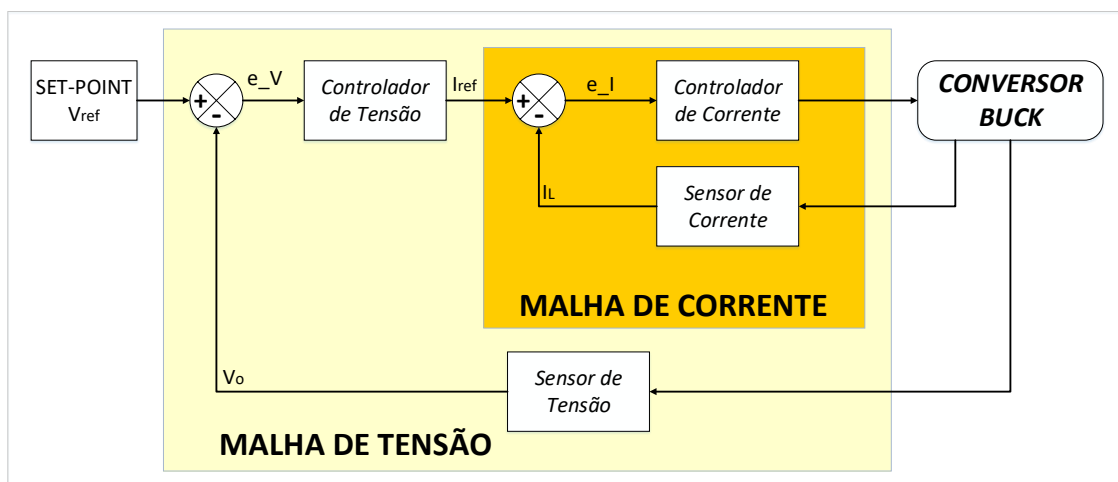


Figura 16 – Diagrama de blocos para o controle da tensão.

Fonte: Adaptado de Martinz (2007).

O projeto da malha de corrente foi realizado considerando a frequência de corte sendo um décimo da frequência de chaveamento e uma margem de fase de 70° . Como descrito, é necessário se ter a FT do processo, e esta é dada por

$$G_{iL}(s) = \frac{i_L(s)}{d(s)} = \frac{1500(s + 96.71)}{s^2 + 96.71s + 2.128 \cdot 10^6} \quad (5.4)$$

Para que seja garantida a condição da malha externa enxergar a malha interna como uma constante, o seu controlador é projetado em uma frequência de corte de $\frac{1}{10}$ da malha interna. A FT usada para o projeto pode ser obtida multiplicando as plantas de pequenos sinais

$$G(s) = \frac{vC(s)}{d(s)} \cdot \frac{d(s)}{i_L(s)} = \frac{2127.7}{s + 96.71} \quad (5.5)$$

Importando para o *Sisotool* as funções de transferência referente a cada uma das malhas e considerando os parâmetros de projeto que foram definidos obtêm-se os controladores de corrente e tensão, cuja FT contínua é dada por

$$C_{iL}(s) = \frac{0.10067 \cdot s + 100.67}{s}, \quad (5.6)$$

$$C_{vC}(s) = \frac{0.0959 \cdot s + 20.848}{s}, \quad (5.7)$$

sendo $C_{iL}(s)$ e $C_{vC}(s)$ as FT's do controladores de corrente e tensão, respectivamente.

Fazendo a discretização de ambas equações no tempo de amostragem escolhido segundo o comando *c2d* e representando as FT's obtidas através de equações a diferença para serem implementadas em microcontrolador, tem-se

$$i_L[k] = i_L[k - 1] + 0.1007e[k] - 0.0755e[k - 1] \quad (5.8)$$

$$v_C[k] = v_C[k - 1] + 0.0959e[k] - 0.09069e[k - 1] \quad (5.9)$$

5.3 Controlador por Realimentação de Estados

O projeto deste controlador foi realizado com base na resposta temporal do sistema quando submetido a operação em malha fechada. Os parâmetros de desempenho especificados e utilizados para o cálculo da frequência natural de oscilação e do coeficiente de amortecimento foram, o tempo de assentamento igual à $5ms$ e máximo sobressinal de 5%.

Fazendo uso de (2.14), (2.16) e (2.17), descobre-se que os polos dominantes dos sistema devem ser alocados em

$$s_{1,2} = -800 \pm j \cdot 838.97. \quad (5.10)$$

Como o sistema tem três variáveis de estado, é necessário que um terceiro polo, referente ao erro, seja alocado no plano s . Para que não influencie significativamente a dinâmica do sistema ele deve ser alocado a esquerda dos polos dominantes. Consoante ao que foi descrito o terceiro polo, s_3 , foi alocado em -1200 .

Como estamos tratando de controle discreto, para se determinar os ganhos da matriz de realimentação, foi feito o mapeamento de s_1 , s_2 e s_3 no plano z segundo propõe a teoria da transformação bilinear exposta em Charles, Nagle et al. (1995). Portanto, no plano z os polos estão alocados em

$$z_1 = 0.8018 + j \cdot 0.1718 \quad (5.11a)$$

$$z_2 = 0.8018 - j \cdot 0.1718 \quad (5.11b)$$

$$z_3 = 0.7391 \quad (5.11c)$$

Por fim, os ganhos da matriz K foram determinados fazendo uso do comando *place*. Portanto

$$K = \text{place}(A, B, z),$$

$$K = \begin{bmatrix} 0.1343 & 0.0110 & 0.0092 \end{bmatrix}. \quad (5.12)$$

Tendo a matriz de ganhos do sistema, a lei de controle a ser implementada no microcontrolador pode ser equacionada considerando os pontos utilizados para fazer a linearização do sistema. Matematicamente tem-se que a lei de controle é dada por

$$u[k] = 0.8 - (0.1343 \cdot (iL[k] - 0.5455) + 0.011 \cdot (vC[k] - 12.0) + 0.092 \cdot e[k]). \quad (5.13)$$

5.4 Controlador MPC Explícito

Como a planta do sistema é um conversor, a restrições de operação impostas pelo projetista não deve exceder aos níveis de tensão e corrente que os componentes suportam. No conversor utilizado para estudo de caso, os níveis máximos de tensão e corrente exigidos são $15V$ e $2A$, respectivamente. Isto se deve ao fato de que a tensão de entrada é limitada aos $15V$, e ao fato de que o núcleo do indutor utilizado satura quando circula por ele corrente igual à $2.3A$.

Da mesma forma que técnicas de controle podem se mostrar ineficientes quando tem que controlar alguma variável com o conversor operando no MDC, elas podem ser ineficientes para o caso em que o núcleo do indutor está saturado. Conhecendo o ponto de operação no qual a linearização do circuito foi realizada, tem-se as seguintes restrições

$$\begin{aligned} -0.5454A &\leq i_L \leq 1.4545A; \\ -12.0V &\leq v_C \leq 3.0V. \end{aligned}$$

Para os outros dois estados do sistema aumentado, referência e erro, os valores das restrições impostos a eles foram

$$\begin{aligned} -12.0V &\leq r_k \leq 3.0V, \\ -\infty &\leq e_k \leq \infty, \end{aligned}$$

a referência segue as mesmas restrições impostas para a tensão de saída. Para que erro estacionário possa variar para quaisquer valores, optou-se por deixá-lo com alto grau de liberdade.

O último parâmetro ao qual restrições devem ser impostas é o *duty-cycle* do PWM, que é a entrada do sistema, e é responsável por manter o valor de tensão no nível desejado. Sabe-se que ele só assume valores compreendidos entre 0 e 1, e o valor usado como ponto de operação para linearização foi igual 0.8. Assim sendo, as restrições impostas são

$$-0.8 \leq \text{duty} - \text{cycle} \leq 0.2.$$

É válido ressaltar que nas restrições, utiliza-se valores negativos, pois o ponto de operação utilizado para a linearização do sistema é considerado como referência, mas a faixa de valores que o parâmetro pode variar deve se manter o mesmo. Por exemplo, na operação descrita o *duty-cycle* foi 0.8, sabe-se porém que o *duty-cycle* pode variar entre 0 e 1. Para que o intervalo se mantenha o mesmo nas restrições ele é definido de $-0.8 \leq \text{duty} - \text{cycle} \leq 0.2$.

Impondo as restrições para os parâmetros da planta, deve-se definir um tamanho para o horizonte de predição e os valores para os parâmetros Q e R . Além deles, o projetista deve escolher um valor para λ , parâmetro que multiplica o erro visando tornar a resposta do sistema menos oscilatória. Baseado em sucessivos testes e analisando a resposta do sistema simulado com a lei de controle gerada, foram definidos os valores, que fizeram com que a resposta do controlador fosse satisfatória. A Tabela 3, apresenta os valores utilizados.

Tabela 3 – Parâmetro definidos para gerar o controlador.

Parâmetros	Valores Utilziados
Q	5
R	1
λ	0.001
N	7

6 Resultados Experimentais

Neste capítulo são apresentados os resultados experimentais da implementação do conversor Buck. Inicialmente, para verificação do funcionamento, o sistema operou em malha aberta. Posteriormente, os controladores projetados foram implementados e se tornaram parte atuante no sistema fazendo com o que ele fosse operado em malha fechada. Devido a limitação de memória do microcontrolador utilizado para executar a ação de controle da tensão do conversor, não foi possível implementar o controlador projetado pela teoria do MPC explícito. Portanto, em implementação prática será feita a análise apenas do controlador PI e por realimentação de estados com integrador.

A Figura 17 mostra uma foto da bancada do projeto para a realização dos testes e coleta dos resultados experimentais.

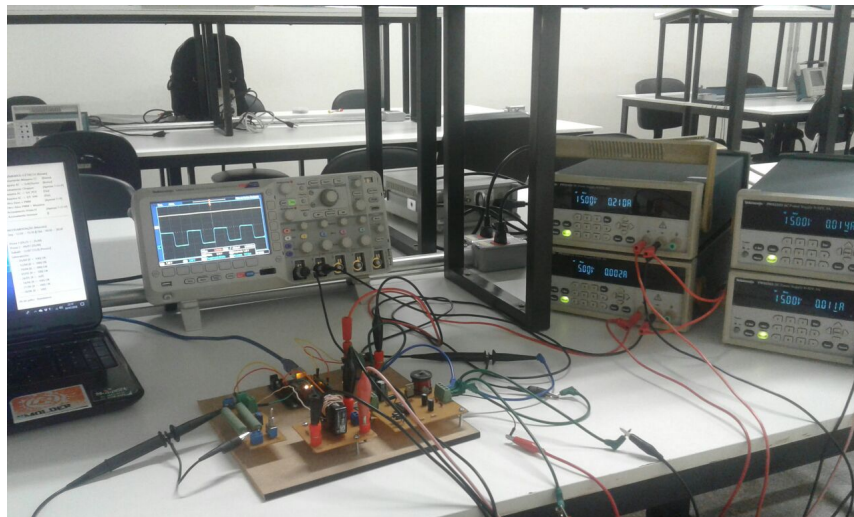


Figura 17 – Bancada do experimento montada em laboratório.

6.1 Ajuste da frequência do PWM do microcontrolador

O controle de abertura do MOSFET foi realizado através do comando *analog.Write* que gera pulsos de PWM a uma frequência fixa e largura de pulso variável. O *Arduino Uno* possui frequência padrão de 980Hz nos pinos 5 e 6 e de 490Hz para os demais pinos. Porém é possível ajustar a frequência para valores pré-determinados ajustando os *timers* do microcontrolador. O *Arduino Uno* possui três *timers* e cada um deles é responsável pelo controle da frequência de dois pinos de saída PWM.

- *Timer 0*: controla os pinos 5 e 6;

- *Timer 1*: controla os pinos 9 e 10;
- *Timer 2*: controla os pinos 3 e 11;

Como deseja trabalhar com frequência próxima a $4kHz$, este valor pode ser obtido ajustando o *timer 2*. Os valores de frequência que podem ser gerados ajustando o *timer* em questão podem ser vistos na Tabela 4.

Tabela 4 – Configuração do *Timer 2* do *Arduino*.

Configuração	Divisor	Frequência (Hz)
0x01	1	31250
0x02	8	3921.00
0x03	32	976.56
0x04	64	488.28
0x05	128	244.14
0x06	256	122.07
0x07	1024	30.52

Através do comando $TCCR2B = (TCCR2B \& 0b11111000) | 0x02$ na IDE do *Arduino*, o divisor do *timer 2* é alterado para 8, fazendo com que a frequência do PWM dos pinos 3 e 11 sejam alterados para $3921.00Hz$. No projeto, o pino utilizado para a saída PWM foi o pino 11.

6.2 Operação em Malha Aberta

Inicialmente foi testado o funcionamento do sistema em malha aberta. A tensão de entrada aplicada foi $15V$ e um sinal de PWM foi regulado objetivando obter uma tensão de saída em torno de $12V$, ponto de operação no qual a linearização do sistema foi realizada. A Figura 18, mostra o sinal de tensão de saída (amarelo) com valor médio igual à $12.01V$ e o sinal de PWM (azul), com frequência de chaveamento de $3.921kHz$.

Sabe-se porém que, conversores *chopper* operando em malha aberta são úteis em situações onde não há possibilidade de perturbação ou distúrbio no circuito. Em aplicação real é extremamente improvável se obter tal condição, pois interferências e variações podem ocorrer a qualquer instante na carga ou tensão de entrada. Portanto, a fim de comprovar a necessidade de implementação de um controlador para regular a tensão de saída em um valor pré determinado quando o sistema sofre interferência, a resistência da carga de saída foi alterada quatro vezes através da associação e desassociação de uma carga de 10Ω , em paralelo com a carga inicial de 22Ω . A Figura 19 apresenta a resposta do circuito para tais perturbações.

Analisando a Figura 19, verifica-se que após a ocorrência da diminuição da resistência de saída do circuito o conversor sofre uma queda de tensão, pois o circuito passa a

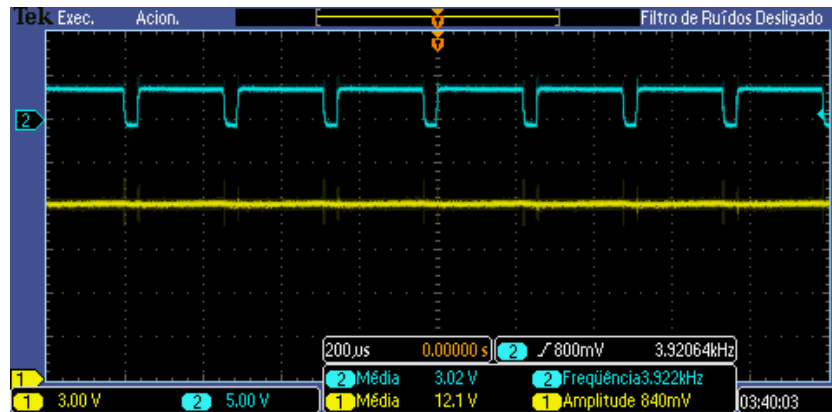


Figura 18 – Sinal de controle PWM e sinal de tensão de saída do conversor regulado em 12V.

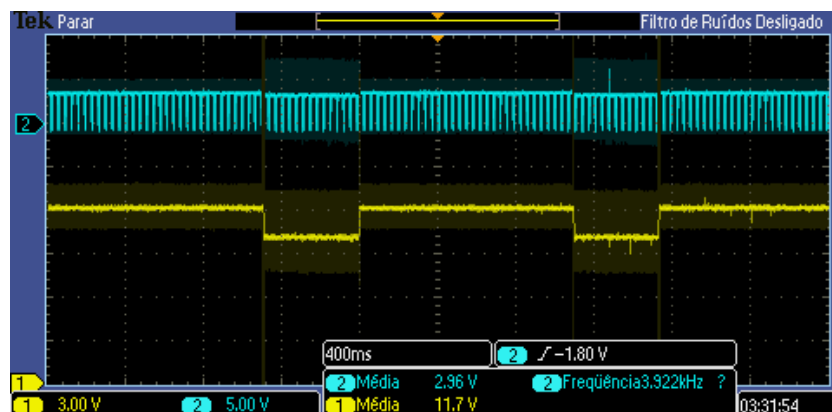


Figura 19 – Variação da carga de saída..

drenar mais corrente, e conseqüentemente aumentam-se as quedas de tensão em resistências parasitas e na chave do circuito. Ao ser desassociada o circuito volta a operar com nível de tensão desejado inicialmente. Desta forma, na ausência de um controlador, o sinal PWM se mantém o mesmo independente da existência ou não de uma perturbação, isto não é desejado, pois as cargas que estão conectadas na saída do circuito podem não funcionar de forma adequada em níveis de tensão inferiores ao seu valor nominal.

6.3 Operação em Malha Fechada

Na operação em malha fechada, novamente foi aplicada uma tensão de 15V tensão na entrada do conversor. O comportamento dos controladores foi avaliado para as seguintes situações:

1. *Set-Point* definido em um valor distinto dos 12V utilizado para a linearização;
2. Variação da carga de saída com o conversor em operação.

6.3.1 Variando o *Set-Point*

A primeira análise em malha fechada observou o comportamento dos controladores quando o sistema opera com carga de 22Ω e a tensão de saída desejada igual à $5.0V$.

As Figuras 20 e 22, apresentam, respectivamente, o regime transitório e estacionário da resposta do controlador por realimentação de estados para levar a tensão de saída ao nível desejado. Através da resposta transitória observa-se que o controlador apresentou um sobressinal elevadíssimo, ordem de 90%, e levou aproximadamente $53,0ms$ para que o sinal de tensão atingisse valores em torno de 2% do valor de referência. E, pela resposta em regime estacionário, percebe-se que o controlador apresentou desempenho satisfatório, pois erro em regime é de apenas 0.2% em relação ao nível desejado.

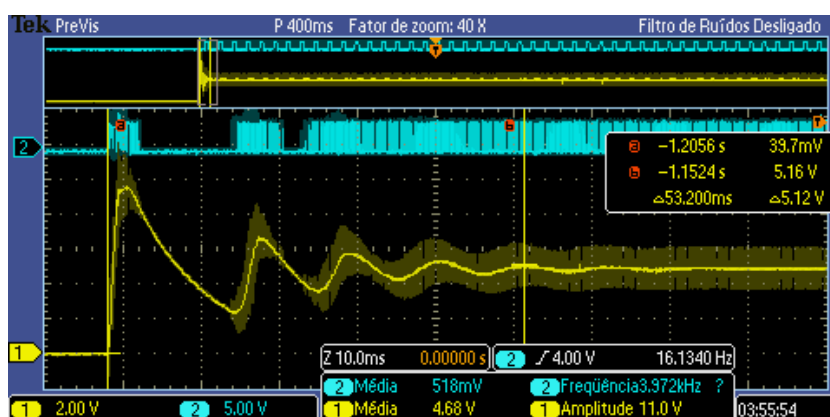


Figura 20 – Regime transitório do controlador por Realimentação de Estados.

Em contrapartida, as Figuras 22 e 23 ilustram o desempenho do controlador PI clássico para que o valor definido como *Set-Point* fosse alcançado. Nota-se que o sobressinal máximo foi de aproximadamente 20% e tempo para o assentamento da resposta deste controlador foi de aproximadamente $64,3ms$. Analisando a resposta em regime permanente observa-se um erro em torno de 2.8% em relação ao valor de referência.

Portanto, em análise comparativa, conclui-se que, caso o sistema não seja prejudicado devido sobressinais elevados de tensão a deseja-se que a resposta se assente rapidamente em torno do valor definido para referência o controlador por realimentação de estados apresenta melhor desempenho. Porém, se sobressinais prejudicarem os componentes do sistema o controlador PI é o que deve ser utilizado, levando em conta que com a sua utilização não é possível obter um tempo de resposta rápido.

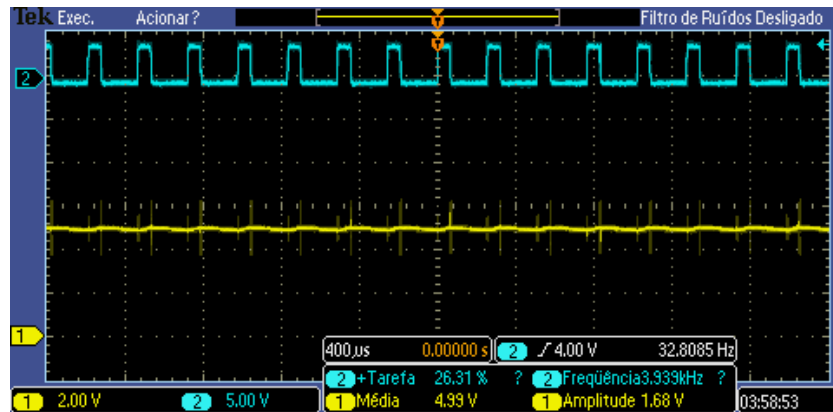


Figura 21 – Regime estacionário do controlador por Realimentação de Estados.

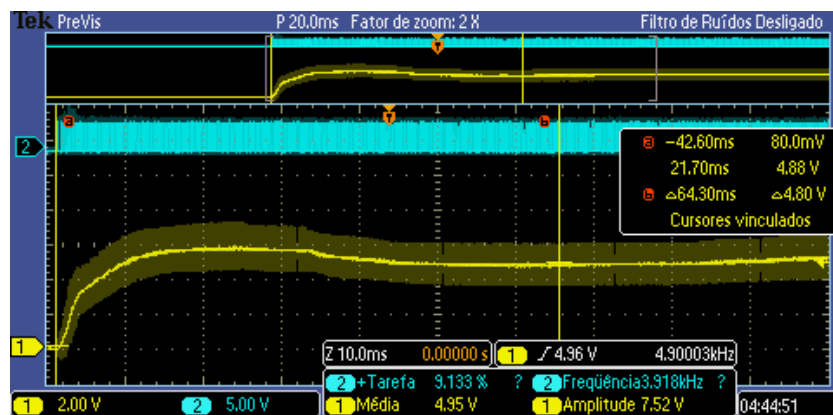


Figura 22 – Regime transitório do controlador PI clássico.

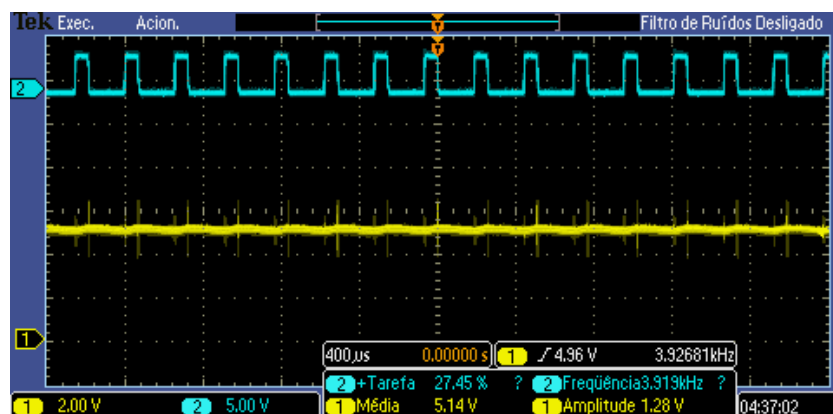


Figura 23 – Regime estacionário do controlador PI clássico.

Para verificar o comportamento dos controladores quando há variação do valor de referência com o conversor já em funcionamento, foi desenvolvida uma rotina que executa o seguinte *loop* para determinado período de tempo.

1. Referência de tensão definida em 4V;
2. Referência alterada para 8V;
3. Referência alterada para 9V;
4. Volta ao passo 1.

As Figuras 24 e 25 apresentam a resposta dos controladores para a situação descrita. Analisando as Figuras com base na escala de tensão que o osciloscópio foi configurado, percebe-se que ambos controladores apresentam resposta em regime permanente próxima dos valores especificados para *Set-Point*. Em concordância com a operação analisada anteriormente, o controlador por realimentação de estados apresentou um sobressinal maior que o PI clássico, porém o tempo de assentamento da resposta é relativamente menor.

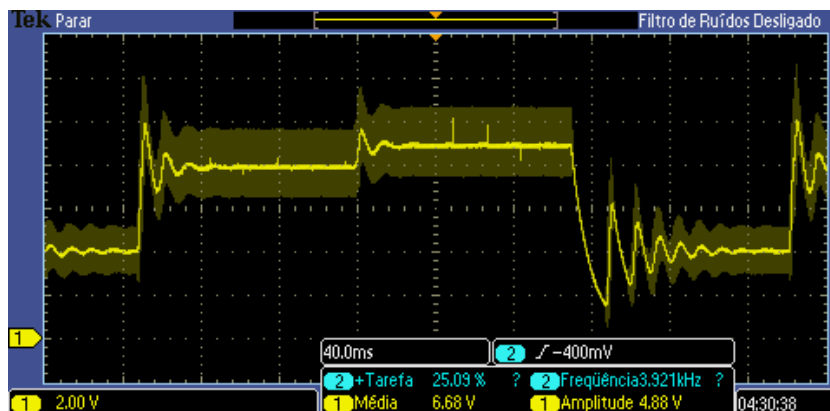


Figura 24 – Controlador por realimentação de estados ao ser submetido a variação de *Set-Point* quando em funcionamento.

Em primeira análise, tratando-se da operação em regime permanente, ambos os controladores apresentaram um desempenho satisfatório, pois o valor médio da tensão de saída foi bem próximo ao definido como *Set-Point*. Porém, em implementação prática, o regime transitório foi relativamente lento e não se assemelhou com as resposta obtidas em simulação, que não foram apresentadas nesta monografia.

6.3.2 Variando a carga

Com descrito na subseção 3.2 em conversores operando em malha aberta, a redução da carga ocasiona a diminuição da tensão de saída, e o aumento da carga ocasiona uma

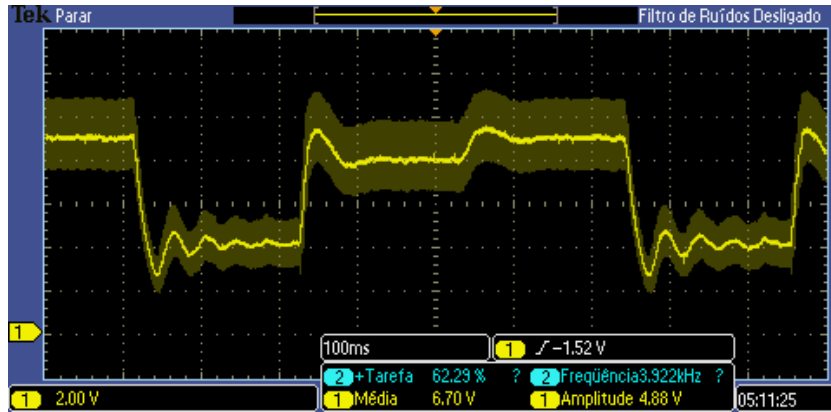


Figura 25 – Controlador PI ao ser submetido a variação de *Set-Point* quando em funcionamento.

elevação da mesma. Portanto, para comprovar que o sistema operando em malha fechada corrige o sinal de PWM de forma a manter a tensão em um valor desejado, independente da variação da carga, foi realizado um teste em que, o valor de referência da tensão foi definido em $6V$ e, inicialmente, a carga de saída foi variada de aproximadamente 7Ω para 22Ω , referente a desassociação do paralelo entre um resistor de 22Ω e um de 10Ω . Posteriormente a associação foi refeita e a carga de saída voltou a ser os 7Ω .

A Figura 26 apresenta o regime transitório e estacionário da resposta do controlador por realimentação de estados na presença das variações de carga que fora descrita. E a Figura 27 mostra o regime transitório e estacionário da resposta do sistema quando é aplicado o controlador PI clássico na malha de controle.

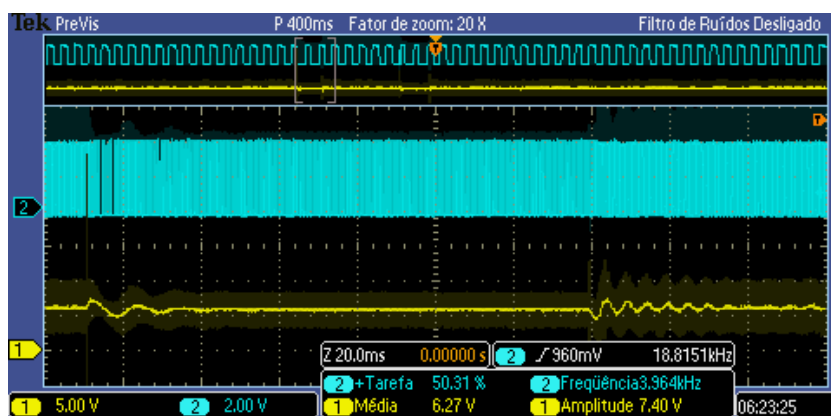


Figura 26 – Resposta do controlador por realimentação de estados a transientes de corrente. Inicialmente houve redução da corrente e posteriormente o aumento.

Em relação a operação que objetiva levar o sinal de tensão a um nível desejado partindo de $0V$, o controlador por realimentação de estados apresentou máximo sobressinal

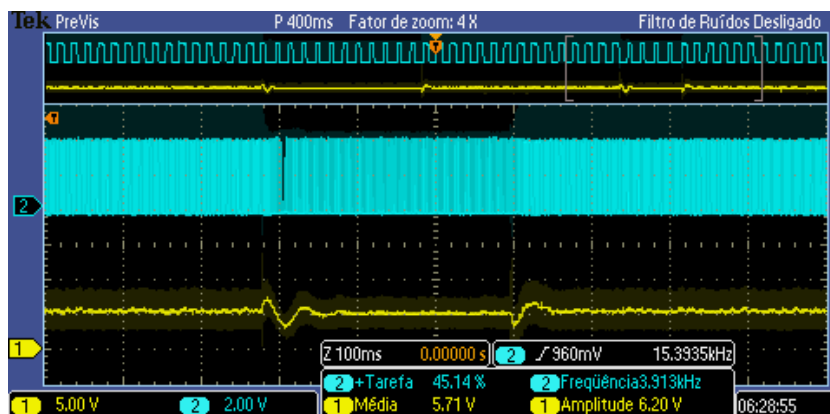


Figura 27 – Resposta do controlador PI a transientes de corrente. Inicialmente houve redução da corrente e posteriormente o aumento.

relativamente pequeno na presença de variação de carga. A resposta do controlador devido a aumento da carga apresentou um sobressinal em torno de 16% e tempo de assentamento do sinal próximo de 20ms. E para a redução da carga o controlador apresentou máximo sobressinal de aproximadamente 25% e tempo de assentamento em torno de 28ms.

Comparando o desempenho dos controladores, conclui-se que o projetado segundo a teoria de realimentação por estados apresentou uma resposta melhor que o obtido por meio do *Sisotool*. A sua resposta transitória foi consideravelmente mais rápida e no regime permanente, ele conseguiu levar o sinal de tensão para um valor próximo mais próximo dos 6V.

6.4 Simulação do Controlador por MPC Explícito

Como não se obteve êxito na implementação do controlador MPC explícito, somente os resultados para ele foram obtidos por meio da simulação no *Simulink*.

As Figura 28 mostra o esquemático do sistema que foi simulado. Na simulação, foram escolhidas as mesmas condições de operação apresentadas na subseção 6.3.1 visando a análise do regime transitório da resposta do controlador. Inicialmente o sistema operou com carga de 22Ω. Porém, para fazer a análise do comportamento do controlador quando há variação na carga de saída, foi realizada uma associação em paralelo do resistor utilizado com um resistor de 10Ω quando a simulação atingiu 0.1s.

As Figuras 29, 30 e 31 mostram, respectivamente, o comportamento da tensão na carga, a corrente no indutor e o sinal com magnitude proporcional ao tempo que o PWM permanecerá em nível alto. Através delas pode-se analisar a efetividade da ação de controle proposta pelo MPC explícito.

Analisando o comportamento do sinal de tensão, Figura 29, nota-se que o controlador

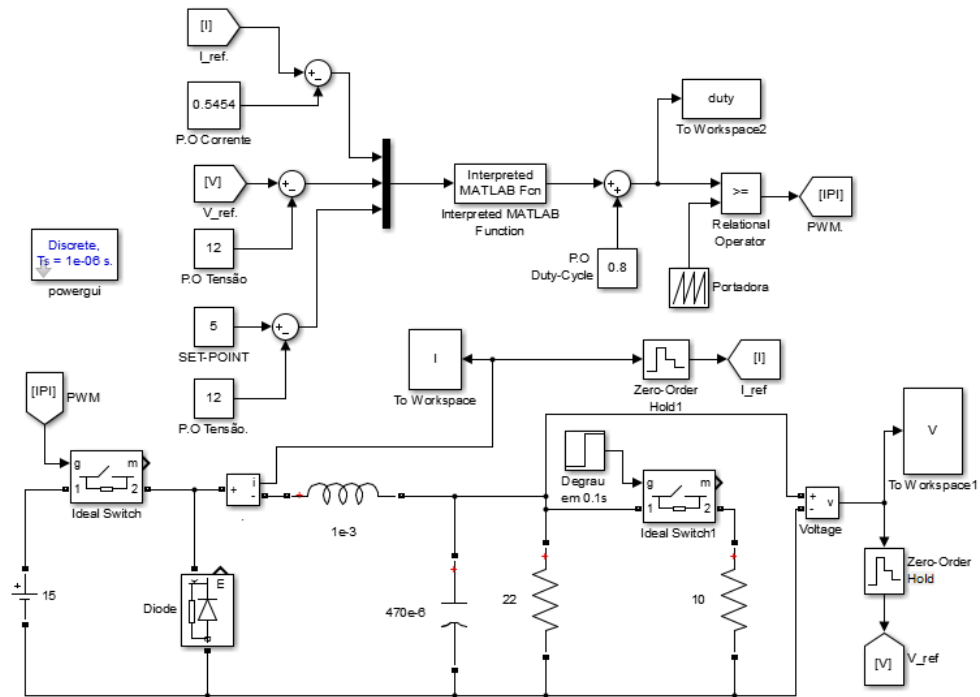


Figura 28 – Diagrama esquemático da simulação para o cotrolador MPC Explícito.

levou cerca de $60ms$ para conduzir o sinal ao assentamento e, na presença de variação de carga, o tempo de assentamento foi em torno de $20ms$, considerando para ambos casos o critério de 2%. Analisando ainda a resposta transitória, observa-se que o sistema apresenta sobressinal, porém ele não viola as restrições de projeto, ou seja, não alcança e nem se aproxima dos $15V$ referente a tensão de entrada. Quanto ao regime estacionário, o controle apresentou bons resultados, pois o erro em regime foi praticamente nulo.

Através da análise dos sinais de corrente e *duty-cycle*, pode-se observar que ambos limites de restrição não foram violados, comprovando que o controlador obtido através do MPC explícito consegue lidar bem com as restrições de projeto.

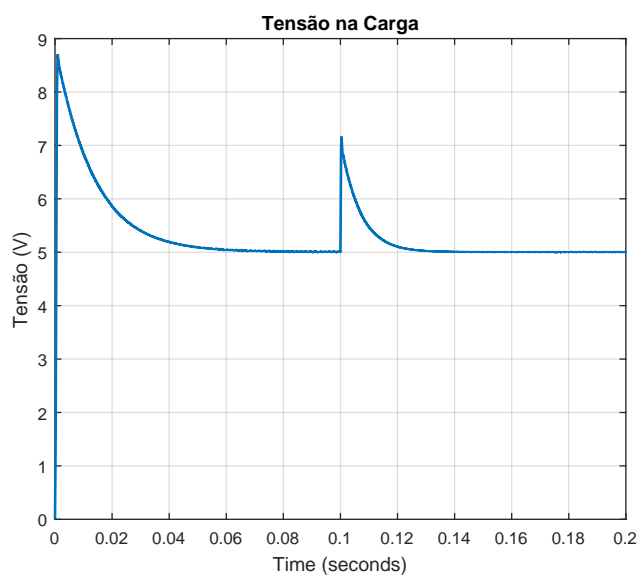


Figura 29 – Sinal de Tensão controlado na saída.

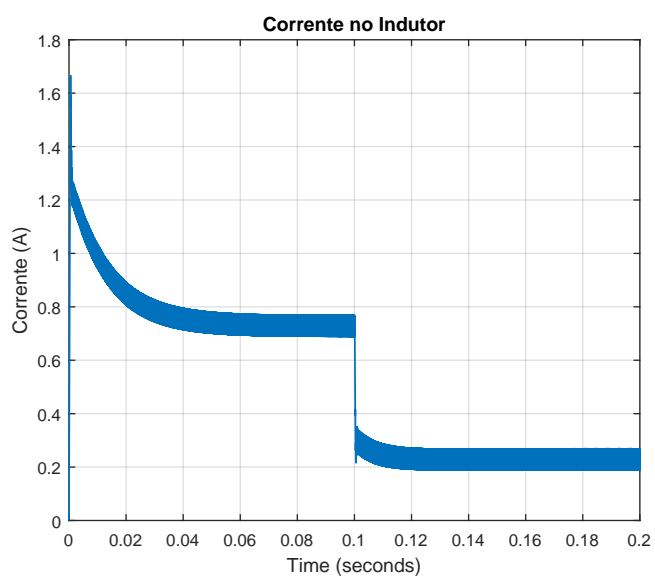


Figura 30 – Sinal de Corrente no indutor.

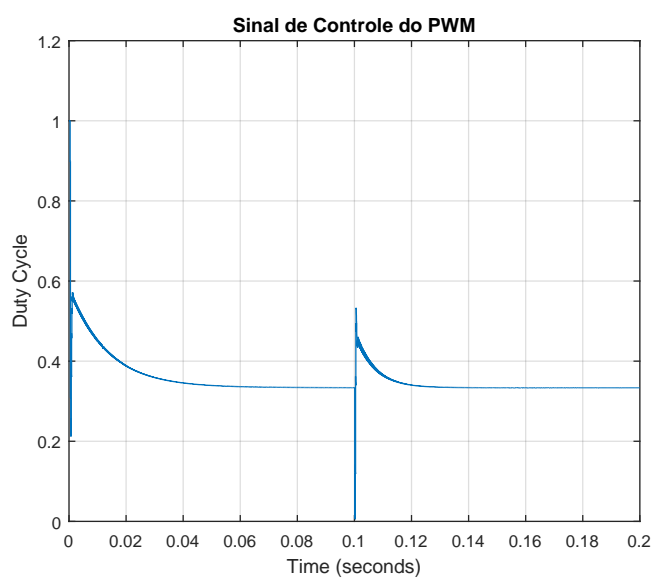


Figura 31 – Sinal a ser comparado com a triangular para gerar o PWM.

7 Conclusão e Trabalhos Futuros

Este trabalho apresentou um estudo comparativo entre técnicas de controle clássico, moderno e preditivo, segundo as técnicas de controle PID, realimentação de estados e MPC explícito. Devido à necessidade de se obter um controle preciso de tensão em conversores de potência, o conversor Buck escolhido como sistema no qual os controladores operaram.

Inicialmente foi realizada uma descrição teórica sobre o funcionamento de cada uma das técnicas de controle, haja visto que, exceto para o caso do controlador PID, frequentemente utilizado para controle de processos, os conhecimentos teóricos sobre as técnicas de controle propostas era extremamente limitado. Seguindo a mesma linha, foi realizada a descrição teórica do conversor Buck onde foi abordado o seu princípio de funcionamento, apresentadas as equações que são utilizadas para o seu projeto e uma técnica que pode ser utilizada para efetuar o controle da corrente ou tensão, a saber Modulação PWM.

Para que os controladores fossem implementados em laboratório, inicialmente foi realizada a simulação dos mesmos. Em simulação todos os controladores foram efetivos, pois conseguiram manter a tensão de saída em um determinado *Set-Point* independente das variações que foram impostas no circuito, porém analisando o comportamento transitório da resposta de cada um deles, o controlador MPC explícito apresentou o melhor desempenho, seguido do controlador por realimentação de estados. Ainda que na implementação prática o tempo de resposta do controlador PI e realimentação de estados foram mais elevados quando comparados ao valor projetado, eles apresentaram um comportamento satisfatório. Desta forma pode-se dizer que o trabalho apresentou êxito em seu desenvolvimento.

No escopo inicial do trabalho estava proposto realizar a implementação prática dos controladores projetados segundo a teoria de cada uma das técnicas. Porém com o microcontrolador *Arduíno Uno* só foi possível fazer a implementação do controlador PI e do controlador por realimentação de estados, pois este não tem memória suficiente para executar a lei de controle obtida no MPC explícito.

Como trabalho futuro é sugerido que o microcontrolador seja substituído por um *Arduíno Due*, que possui maior memória e mais recursos quando comparado ao que foi utilizado, seja feita a implementação da lei de controle gerada pelo MPC explícito. O *Arduíno Due* tem capacidade de definir a frequência de chaveamento e de amostragem para valores superiores aos $4kHz$ utilizados no trabalho, ocasionando uma redução nas perdas e menor variação percentual da tensão de saída e da corrente do indutor. Em seu trabalho, Leal (2018) apresenta detalhadamente uma forma de configurar portas do *Arduíno Due* para gerar um sinal de PWM. Além disso, também é explicado como se realiza interrupções na rotina do programa para que a frequência de amostragem e de chaveamento sejam a mesma desta, forma recomenda-se que o material seja analisado.

Referências

- ALESSIO, A.; BEMPORAD, A. A survey on explicit model predictive control. Springer, p. 345–369, 2009. 15
- ÅSTRÖM, K. J.; HÄGGLUND, T. *PID controllers: theory, design, and tuning*. [S.l.]: Isa Research Triangle Park, NC, 1995. v. 2. 3, 4, 5
- ÅSTRÖM, K. J.; HÄGGLUND, T. The future of pid control. *Control engineering practice*, Elsevier, v. 9, n. 11, p. 1163–1175, 2001. 1
- BARBI, I. *Modelagem de Conversores CC-CC Empregando Modelo Médio em Espaço de Estados*. [S.l.: s.n.], 2015. 218 p. ISBN 9788590104698. 19, 21, 22
- BARBI, I.; MARTINS, D. C. *Conversores CC-CC Básicos Não Isolados*. [S.l.: s.n.], 2008. 377 p. ISBN 8590520323. 17, 18, 25
- BASTOS, R. F. *Sistema de Gerenciamento para Carga e Descarga de Baterias (Chumbo-Ácido) e para Busca do Ponto de Máxima Potência Gerada em Painéis Fotovoltaicos Empregados em Sistemas de Geração Distribuída*. Dissertação (Mestrado), 2013. 23, 24
- BEMPORAD, A. et al. The explicit linear quadratic regulator for constrained systems. *Automatica*, Elsevier, v. 38, n. 1, p. 3–20, 2002. 14
- CAMACHO, E. F.; BORDONS, C. Model predictive control, berlin, 1999, isbn 3540762418, 280 pages. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, Wiley Online Library, v. 13, n. 11, p. 1091–1093, 1999. 11, 12
- CHARLES, L. P.; NAGLE, H. T. et al. Digital control system analysis and design. *Prentice Hall*, 1995. 9, 10, 32
- CHEN, C.-T. *Linear system theory and design*. [S.l.]: Oxford University Press, Inc., 1998. 7
- DORF, R.; BISHOP, R. *Sistemas de Controle Modernos, oitava edição edn*. [S.l.: s.n.], 2001. 3
- DORF, R. C.; BISHOP, R. H. *Modern control systems*. [S.l.]: Pearson, 2011. 8
- HART, D. W. *Power Eletronics*. [S.l.: s.n.], 2010. ISBN 0849373360. 19, 20
- HERCEG, M. et al. Multi-Parametric Toolbox 3.0. In: *Proc. of the European Control Conference*. Zürich, Switzerland: [s.n.], 2013. p. 502–510. <<http://control.ee.ethz.ch/~mpt>>. 15
- HERNÁNDEZ-ARROYO, E.; DÍAZ-RODRÍGUEZ, J. L.; PINZÓN-ARDILA, O. Estudio del comportamiento de un control mpc [control predictivo basado en el modelo] comparado con un control pid en una planta de temperatura. *Facultad de Ingeniería*, Universidad Pedagógica y Tecnológica de Colombia, v. 23, n. 37, 2014. 1
- JÚNIOR, N.; ANDRADE, G. Sintonia ótima de controladores mpc considerando incertezas de modelagem. Universidade Federal da Bahia. Escola Politécnica, 2016. 1

- KIDO, R. H. Z. et al. Uma aplicação de controle preditivo multiparamétrico em planejamento da produção. [sn], 2011. 11, 13
- LEAL, W. C. Sistema de controle para compartilhamento de corrente de baterias em microrredes cc. 2018. 46
- LOPES, K. dos S. *Sistema de fontes reguladas/isoladas/sincronizadas para aplicação em UPS*. Tese (Doutorado) — Universidade Estadual de Londrina, 2012. 24
- MARTINZ, F. O. *Estudo de estratégias de rastreamento da corrente e da tensão de saída CA de um conversor do tipo fonte de tensão*. Dissertação (Mestrado) — USP, 2007. 30
- OGATA, K. *Discrete-time control systems*. [S.l.]: Prentice Hall Englewood Cliffs, NJ, 1995. v. 2. 6, 7, 8
- PINTO, J. E. M. G. *Aplicação prática do método de sintonia de controladores PID utilizando o método do relé com histerese*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2014. 3, 4, 5
- POMILIO, J. *Modelagem de Fontes Chaveadas: Método da inspeção*. [S.l.: s.n.], 2007. 19, 22, 24
- POMILIO, J. Projeto de sistemas de controle linear para fontes chaveadas. *UNICAMP, São Paulo*, 2014. 30
- SCIBILIA, F. Explicit model predictive control: Solutions via computational geometry. Norges teknisk-naturvitenskapelige universitet, Fakultet for informasjonsteknologi, matematikk og elektroteknikk, Institutt for teknisk kybernetikk, 2010. 13, 14
- SKVARENINA, T. L. *The Power Eletronics*. [S.l.: s.n.], 2002. ISBN 0849373360. 18, 19
- SOUZA, E. M. de. *Controle Preditivo Robusto Baseado em Modelo Aplicado a Sistemas Não-Lineares Incertos Linearizados por Realimentação de Estados*. Dissertação (Mestrado), 2015. 11, 13
- VILLALVA, M. G. *Conversor Eletrônico de Potência Trifásico para Sistema Fotovoltaico Conectado a Rede Elétrica. 2010. 292 f.* Tese (Doutorado) — Tese (Doutorado em Engenharia Elétrica)—Faculdade de Engenharia, Universidade Estadual de Campinas, Campinas, 2010.

A Controladores Implementados em *Arduíno*

Controlador PI

```

1 //Entradas
2 float SET_TENSAO = 5.0;
3 float SensorTensao = 0.0;
4 float SensorCorrente = 0.0;
5 int cont = 0;
6
7 //Saida
8 int Pin_PWM = 3;
9 float duty = 0.0;
10
11 //Malha_Tensao
12 float erro_Vo_Atual = 0.0;
13 float erro_Vo_Ante = 0.0;
14 float PI_Vo_Atual = 0.0;
15 float PI_Vo_Ante = 0.0;
16
17 //Malha_Corrente
18 float erro_IL_Atual = 0.0;
19 float erro_IL_Ante = 0.0;
20 float PI_IL_Atual = 0.0;
21 float PI_IL_Ante = 0.0;
22
23 void setup() {
24
25   Serial.begin(9600);
26   analogReference(DEFAULT);
27
28   pinMode(A4, INPUT); //Entrada de corrente
29   pinMode(A3, INPUT); //Entrada de tensao
30
31   pinMode(Pin_PWM, OUTPUT);
32   // Configura a frequencia do PWM
33   TCCR2B = TCCR2B & 0b11111000 | 0X02; //Frequencia de 4KHz
34   //Frequencia do PWM para 4KHz (0x01 = 31,5 KHz | 0X02 = 4 KHz | 0
       x03 = 976 Hz) nas portas 3 e 11

```

```
35 }
36
37 void loop() {
38
39
40 //   if(cont >= 50 && cont<= 100){
41 //       SET_TENSAO = 4.0;
42 //   }
43 //
44 //   if(cont>100&&cont<=150){
45 //       SET_TENSAO = 8.0;
46 //   }
47 //
48 //   if(cont>150){
49 //       SET_TENSAO = 9.0;
50 //       cont = 0.0;
51 //   }
52 //
53 //   cont=cont+1;
54
55
56 SensorTensao = analogRead(3)*0.0048875855; //analogRead faz
    leituras de 0 a 1023 -- Fazendo o mapeamento
57 //Leitura_0 -> Tensao = 0V || Leitura_511 -> Tensao = 2.5V
58 //Leitura_1023 -> Tensao = 5V
59
60 SensorCorrente = analogRead(4)*0.0048875855; //analogRead faz
    leituras de 0 a 1023 -- Fazendo o mapeamento
61 //Leitura_0 -> Tensao = 0V || Leitura_511 -> Tensao = 2.5V
62 //Leitura_1023 -> Tensao = 5V
63
64 Serial.print(SET_TENSAO);
65
66
67 //Controlador_Malha_Tensao_EXTERNA
68
69 erro_Vo_Atual = (0.334*SET_TENSAO - SensorTensao);
70 PI_Vo_Atual = (PI_Vo_Ante + (0.0959*erro_Vo_Atual) - (0.09069*
    erro_Vo_Ante));
71
72 if (PI_Vo_Atual >= 5.0){
73 PI_Vo_Atual = 5.0;
```

```
74 }
75
76 if (PI_Vo_Atual <= 0.0){
77 PI_Vo_Atual = 0.0;
78 }
79
80 PI_Vo_Ante = PI_Vo_Atual;
81 erro_Vo_Ante = erro_Vo_Atual;
82
83 //Controlador_Malha_Corrente_Interna
84
85 erro_IL_Atual = (PI_Vo_Atual - SensorCorrente);
86 PI_IL_Atual = PI_IL_Ante + (0.1007*erro_IL_Atual) - (0.0755*
      erro_IL_Ante);
87
88 if (PI_IL_Atual >= 1.0){
89 PI_IL_Atual = 0.99;
90 }
91
92 if (PI_IL_Atual <= 0.0){
93 PI_IL_Atual = 0.0;
94 }
95
96 PI_IL_Ante = PI_IL_Atual;
97 erro_IL_Ante = erro_IL_Atual;
98
99 //Duty-Cycle
100
101 duty = PI_IL_Atual*255; //Mapeando o valor do duty-cycle de 0 a
      255
102
103 //Gerando o PWM
104
105 analogWrite(Pin_PWM,duty);
106
107 }
```

Controlador Realimentação de Estados

```
1 //Entradas
2     float SensorTensao = 0.0;
3     float SensorCorrente = 0.0;
```



```
4         float SET_TENSAO = 6.0;
5         int cont = 0;
6
7 //Saida
8         int Pin_PWM = 3;
9         float duty = 0.0;
10
11 //Auxiliares
12         float integral = 0.0;
13         float PWM = 0.0;
14
15 void setup() {
16
17         Serial.begin(9600);
18         analogReference(DEFAULT);
19
20         pinMode(A4,INPUT); //Entrada de corrente
21         pinMode(A3,INPUT); //Entrada de tensao
22
23         pinMode(Pin_PWM,OUTPUT);
24 // Configura a frequencia do PWM
25         TCCR2B = TCCR2B & 0b11111000 | 0X02; //Frequencia de 4
                KHz
26 //Frequencia do PWM para 4KHz (0x01 = 31,5 KHz | 0X02 = 4 KHz | 0
                x03 = 976 Hz) nas portas 3 e 11
27 }
28
29 void loop() {
30
31         // if(cont >= 300 && cont <= 600){
32         //     SET_TENSAO = 4.0;
33         // }
34         //
35         // if(cont>600&&cont<=900){
36         //     SET_TENSAO = 8.0;
37         // }
38         //
39         // if(cont>900){
40         //     SET_TENSAO = 9.0;
41         //     cont = 0.0;
42         // }
43         //
```

```
44     //  cont=cont+1;
45
46
47     SensorTensao = analogRead(3)*0.0048875855; //analogRead
        faz leituras de 0 a 1023 -- Fazendo o mapeamento
48         //Leitura_0 -> Tensao = 0V || Leitura_511 ->
            Tensao = 2.5V
49         //Leitura_1023 -> Tensao = 5V
50
51     SensorCorrente = analogRead(4)*0.0048875855; //analogRead
        faz leituras de 0 a 1023 -- Fazendo o mapeamento
52         //Leitura_0 -> Tensao = 0V || Leitura_511 ->
            Tensao = 2.5V
53         //Leitura_1023 -> Tensao = 5V
54
55 //Controlador Malha Tensao
56     PWM = 0.8 - (0.1343*(SensorCorrente - 0.5455) + 0.011*(
        SensorTensao - 12.0) + 0.0092*integral);
57
58     if (PWM >= 1.0){
59         PWM = 0.95;
60     }
61
62     if (PWM <= 0.0){
63         PWM = 0.0;
64     }
65
66     integral = integral + SensorTensao - SET_TENSAO*0.3334;
67
68     if (integral >= 10){
69         integral = 10;
70     }
71
72     //Duty-Cycle
73
74     duty = PWM*255.0; //Mapeando o valor do duty-cycle de 0 a
        255
75
76     //Gerando o PWM
77         analogWrite(Pin_PWM,duty);
78     }
```