UNIVERSIDADE FEDERAL DE OURO PRETO
ESCOLA DE MINAS
COLEGIADO DO CURSO DE ENGENHARIA DE CONTROLE
E AUTOMAÇÃO - CECAU

PEDRO HENRIQUE EMERICK CALDEIRA

CONTROLE DE VELOCIDADE DE UM MOTOR DE CORRENTE CONTÍNUA UTILIZANDO O MÉTODO VRFT

MONOGRAFIA DE GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Ouro Preto, 2020

PEDRO HENRIQUE EMERICK CALDEIRA

CONTROLE DE VELOCIDADE DE UM MOTOR DE CORRENTE CONTÍNUA UTILIZANDO O MÉTODO VRFT

Monografia apresentada ao Curso de Engenharia de Controle e Automação da Universidade Federal de Ouro Preto como parte dos requisitos para a obtenção do Grau de Engenheiro de Controle e Automação.

Orientador: prof. João Carlos Vilela de Castro

Ouro Preto
Escola de Minas – UFOP
2020

MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
ESCOLA DE MINAS
DEPARTAMENTO DE ENGENHARIA CONTROLE E AUTOMACAO

FOLHA DE APROVAÇÃO

Pedro Henrique Emerick Caldeira

Controle de Velocidade de um Motor de Corrente Contínua Utilizando o Método VRFT

Membros da banca

Adrielle de Carvalho Santana - Ms. - UFOP
Danny Augusto Vieira Tonidandel - Ms. - UFOP
João Carlos Vilela de Castro - Ms. - UFOP

Versão final
Aprovado em 17 de julho de 2020

De acordo

João Carlos Vilela de Castro - Professor Orientador

Documento assinado eletronicamente por **Joao Carlos Vilela de Castro**, **PROFESSOR 3 GRAU**, em 26/08/2020, às 15:11, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do Decreto nº 8.539, de 8 de outubro de 2015.

A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0 , informando o código verificador **0077650** e o código CRC **0659C4B4**.

Referência: Caso responda este documento, indicar expressamente o Processo nº 23109.006040/2020-12 SEI nº 0077650

R. Diogo de Vasconcelos, 122, - Bairro Pilar Ouro Preto/MG, CEP 35400-000
Telefone: 3135591533 - www.ufop.br

RESUMO

O seguinte trabalho tem como propósito o estudo do método *Virtual Reference Feedback Tunning* (VRFT) aplicado ao projeto de controladores com estrutura PI e PID para controlde de velocidade de giro um motor de corrente contínua. O método VRFT não requer o uso de modelo matemático no projeto do controlador, o que o classifica como um método livre de modelo, ou orientado a dados, sendo necessário que se colha dados a partir de ensaios, para sua implementação. Com esses dados e a partir de um modelo de referência, um sinal de referência virtual é calculado e utilizado para identificação de um controlador com estrutura pré definida. Neste trabalho, dados reais de um motor de corrente contínua de pequeno porte são utilizados no procedimento VRFT a fim de sintonizar parâmetros de controladores com estrutura PI e PID. O controlador final é implementado em um microcontrolador da família Arduíno. Depois de analisados, os resultados dos controladores se mostram coerentes, apontando a versalidade do método e mostrando ser uma boa alternativa no projeto de controle, principalmente para casos em que os parâmetros de motores não estão disponíveis ou são de difícil obtenção.

Palavras-chaves: PI, PID, Dados, controle de velocidade, VRFT.

ABSTRACT

The following work has the purpose of studying the virtual reference feedback tuning (VRFT) method in the control a DC motor using PI and PID methods that are standardly applied in industry. The VRFT method designes controllers using data, so is necessary to generate it through trials. It is a model free method and it does not require the mathematical modeling of the process. With this data and a desired reference a controller signal is computed and used for parameter identification to a predefined control. In this work, real data from a DC motor is used in the VRFT procedure to tune the parameters of a PI and PID controller for angular speed control. The controllers were implemented with the help of a microcontroller of the arduino family. After analysis, the resulting controllers are shown to be cohesive, indicating that the method is versatile and a possible alternative, mainly in cases that the plant's model is not available or are hard to obtain.

Key-words: PI, PID, Data, velocity control, VRFT.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de blocos representando motor. 22
Figura 2 – Diagrama de blocos da planta 22
Figura 3 – Diagrama de blocos representando modelo. 22
Figura 4 – Circuito representando a armadura do motor CC. Fonte(v.researchgate.net) . 26
Figura 5 – Diagrama de corpo livre de um motor CC. Fonte(v.researchgate.net) 26
Figura 6 – Arquitetura Von Neumann.
Fonte: sciencedirect 28
Figura 7 – Arquitetura Harvard.
Fonte: sciencedirect 29
Figura 8 – Comparação entre modelos de placa de Arduino.
Fonte: filipeflop.com.br 30
Figura 9 – Motor utilizado no trabalho 31
Figura 10 – Foto da placa montada com o arduino (a esquerda) e o conversor buck . . . 31
Figura 11 – Resposta de velocidade angular em malhar aberta (azul), considerando um sinal de entrada na forma de onda quadrada (tracejada em preto) 32
Figura 12 – Resposta de velocidade angular em malhar fechada (azul), considerando um sinal de entrada na forma de onda quadrada (tracejada em preto) e o erro virtual (em vermelho). 32
Figura 13 – Resposta temporal da velocidade angular em malha fechada (azul) seguindo uma referência de onda na forma quadrada (tracejada em preto). 34
Figura 14 – Resposta temporal da velocidade angular em malha fechada (azul) seguindo uma referência de onda na forma quadrada (tracejada em preto), e o modelo de referência (pontilhado em vermelho). 35
Figura 15 – Resposta temporal da velocidade angular em malha fechada (azul) agora com uma carga aplicada em seu eixo, e seguindo uma referência de onda na forma quadrada (tracejada em preto), e o modelo de referência (pontilhado em vermelho). 36

LISTA DE ABREVIATURAS E SIGLAS

VRFT	*Virtual Reference Feedback Tuning*
LQR	*Linear Quadratic Regulator*
LQG	*Linear Quadratic Gaussian*
MBC	*Model Based Control*
MIMO	*Multiple Imput Multiple Output*
IFT	*Iterative Feedback Tuning*
SPSA	*Simultaneous Perturbation Stochastic Approximation*
MFAC	*Model free Adaptative Control*
UF	*Unfalsified Control*
IFT	*Iterative Feedback Tunning*
CBT	*Correlation Based Tuning*
APD	*Approximate Dynamic Programming*

SUMÁRIO

1 INTRODUÇÃO 8
1.1 Objetivo Geral 9
1.2 Objetivos 10

2 REVISÃO DA LITERATURA 11
2.1 Controle de Sistemas Realimentados 11
2.2 Teoria de Controle 11
2.3 Controle baseado em modelo 12
2.4 Modelagem e Identificação 13
2.5 Controle Baseada em Dados 14
2.5.1 Definindo um Controlador Baseado em dados 14
2.6 Alguns métodos de controle baseados em dados existentes: 14
2.6.1 Simultaneous Perturbation Stochastic Approximation (SPSA) 14
2.6.2 Model-free Adaptative Control (MFAC) 15
2.6.3 Unfalsified Control (UF) methodology 15
2.6.4 Iterative feedback tuning (IFT) 15
2.6.5 Correlation based tuning (CBT) 15
2.6.6 Noniterative data-driven model reference control 16
2.6.7 Subspace Approach 16
2.6.8 Approximate Dynamic Programming (ADP) 16

3 METODOLOGIA 17
3.1 VRFT 18
3.2 O método VRFT 21
3.2.1 A ideia de modelo de referência 21
3.2.2 Definindo o modelo 22
3.2.3 A equação do erro 23
3.2.4 O controlador 24
3.2.5 Pseudo Inversa 25
3.3 Motores de Corrente Contínua 25
3.3.1 Modelagem de Motores C.C. 25
3.3.2 Acionamento de Motores C.C. 27
3.4 Controladores Embarcados 27
3.4.1 Microcontroladores 27
3.4.2 Arquitetura dos Microcontroladores 28
3.5 Arduino 29

3.6 Desenvolvimento da plataforma de Controle 30
3.6.1 Motor utilizado 30
3.6.2 Circuito de acionamento de motor 31
3.7 Projeto do Controlador VRFT para Controle de Velocidade do Motor 31
3.8 Avaliação dos Controladores 33

4 RESULTADOS 34

5 CONCLUSÕES E RECOMENDAÇÕES 37
5.1 Recomendações para trabalhos futuros 37

REFERÊNCIAS 38

1 INTRODUÇÃO

Antes mesmo de chegar à forma moderna que têm hoje os humanos primitivos começaram a fazer para si ferramentas. Apesar de não parecer exorbitante o ato de desenvolver uma ferramenta foi um enorme salto cognitivo, sinalizando tanto ciência que pensavam no futuro pertinente ao presente quanto que a matéria trabalhada da forma correta poderia produzir melhores resultados. Não muito diferente, apesar da imensa distância temporal desde as primeiras ferramentas produzidas pelo homem, o papel da engenharia continua sendo usar a matéria disponível em beneficio da humanidade, imaginando um eficiente uso no futuro e trabalhando para a alcançar uma melhoria. Porém uma ferramenta só pode ser tão útil quanto a maneira que é usada.

Juntamente com a complexidade de problemas e soluções abordados pela engenharia, também foram se tornando cada vez mais complexos os métodos de manipulação e do uso da matéria. Consequentemente a engenhosidade humana, sempre persistente na sistematização trouxe à tona e começou a fazer uso da matemática. Com ela é possível a descrição minuciosa de causa e efeito, fazendo com que não mais seja necessário basear-se somente em intuição ou mesmo tentativa e erro ao trabalhar com a natureza. Ao se descrever matematicamente o comportamento de um sistema este comportamento se torna previsível no âmbito do modelo. Como exemplo, com o ângulo de lançamento e a força aplicada é possível construir um modelo adequado para descobrir o destino de um projétil. Todavia, quando se trata de sistemas mais complexos, um modelo nunca conseguir prever todos os pormenores físicos, por exemplo.

Assim como na biologia podia-se somente especular sobre o funcionamento de tecidos vivos antes do desenvolvimento de lentes com as quais pôde-se construir um microscópio, no qual seria possível observar células, também a modelagem de processos pode ser insuficiente para descrição fiel do mesmo pelo simples fato de não se ter ciência de certos fenômenos, ou mesmo pela dificuldade de detectá-los.

Analogamente ao projétil, um motor elétrico pode ter seu comportamento deduzido a partir das leis da física. Essa abordagem, de se obter modelos matemáticos a partir de leis conhecidas, é muitas vezes chamada de abordagem fenomenológica, ou de abordagem pelos primeiros princípios. Se considerarmos um motor comum disponível no mercado existem um número enorme de variáveis como por exemplo temperatura, e fenômenos físicos como a histerese. Ao analisar essas e outros tipos variações que podem ser incógnitas, a modelagem matemática por vezes pode não ser atrativa quando se é necessário um controle minucioso da posição, ou mesmo da velocidade. Este trabalho visa contornar essa deficiência da modelagem com o uso de uma técnica de controle baseada em dados, que toma proveito da capacidade computacional que é disponível hoje, usando dados de um sistema a modelagem do mesmo não depende totalmente do conhecimento do funcionamento do sistema e é mais apto a reconhecer

os pormenores que escapariam ao conhecimento humano.

Motores de corrente contínua têm excelentes características no quesito controle de velocidade, podendo se mostrar vantajosos para serem usados em indústrias mesmo que seus custos de manutenção muitas vezes possam ser maiores do que motores de indução. Devido a este extenso uso na indústria, frequentemente o controle de velocidade, posição e torque nas pesquisas científicas.

Muitas vezes estas indústrias possuem produções e equipamentos que operam em larga escala e, inclusive, processos que apresentam a característica de interdependência entre suas variáveis, ou seja, processos multivariáveis (ou MIMO: *Multiple-Input Multiple-Output*). Por este motivo, a modelagem e/ou identificação destes processos tem se tornado uma tarefa cada vez mais complexa e custosa, além de muitas vezes resultar em modelos imprecisos. Dessa forma, aplicar as técnicas tradicionais de controle baseado em modelo nestes ambientes tem se tornado uma escolha pouco prática (HOU; WANG, 2013).

Atualmente os motores são embutidos em cadeias cada vez mais complexas nos processos industriais. Em contrapartida , cada vez mais esse motores têm seus dados coletados durante toda sua operação por sistemas supervisórios que os monitoram e armazenam dados de suas operações. Esses dados contém informações importantes e são de fácil acesso e manuseio, o que torna vantajoso sintonizar os controladores a partir dos mesmos. Métodos de controle baseado em dados (*Data Driven Control*) estão presente na literatura desde o trabalho dec(ZIEGLER; NICHOLS, 1995), principalmente a partir dos anos 1990 esse tipo de metodologia tem sido aprimorada e expandida e tem ganhado crescente notoriedade.

Dentre os métodos baseados em dados temos o *Virtual Reference Feedback Tuning* (VRFT), que apresenta a vantagem de, a partir de um ou dois experimento(s) realiza a sintonia do controlador, sem necessidade do conhecimento ou identificação do modelo, o que o classifica em metodologia *data-driven*. Várias extensões para esta metodologia já foram propostas na literatura, em que se destaca a apresentada no trabalho (CAMPESTRINI et al., 2016), que trata da formulação MIMO do problema. Por outro lado, existem as técnicas iterativas de controle baseado em dados, onde é necessária uma sequência de experimentos para refinar os parâmetros do controlador de forma iterativa. Entre estas, a mais popular e estudada na literatura é o *Iterative Feedback Tuning* (IFT) /(HJALMARSSON; BIRKELAND, 1998).

1.1 Objetivo Geral

Projetar e desenvolver um controlador de velocidade para motores de corrente contínua usando o método baseado em dados conhecido como Método da Referência Virtual, simular o processo e averiguar os alcances do controlador no que se refere ao tempo de assentamento e sobresalto e comparar os resultados do experimento com a simulação.

1.2 Objetivos

- Coleta de dados do motor juntamente com seu circuito de controle de tensão.
- Levantamento do modelo da planta a ser controlada.
- Desenvolver um controle usando os dados coletados.
- Comparar resultados.

2 REVISÃO DA LITERATURA

Nesse capítulo é apresentado uma breve síntese da ciência reponsável pela engenharia de controle, começando com a ideia de realimentação de um sistema que é precursora da teoria de controle. Aqui é exposto um resumo da teoria de controle e de sua evolução, métodos que começaram na tentativa e erro foram progredindo para abordagens baseadas no uso de modelos fenomenológicos, modelos identificados, até o desenvolvimento de técnicas que são projetadas diretamente de dados colhidos do processo, sem o uso de modelos para este último, conhecidas como métodosd de controles baseados em dados, ou *data-driven control*. Ao final uma pequena amostra do estado da arte desses métodos de controle baseados em dados é apresentada.

2.1 Controle de Sistemas Realimentados

Um dos primeiros usos de controle de sistemas realimentados que se tem registro é um relógio de água utilizado no egito a mais de dois mil anos atrás. O mesmo princípio no século 19 permitiu o controle de velocidade de máquinas a vapor. Mas somente na década de 30 que uma teoria de controle de sistemas realimentados foi desenvolvida por Nyquist (1928). Desde então outros métodos foram se desenvolvendo como o método do lugar das raízes que é uma técnica gráfica que permite visualizar de que forma os pólos de um sistema em malha fechada variam quando se altera o valor de um parâmetro, que geralmente é o ganho. O método de espaço de estados, por outro lado, é uma representação por meio de um modelo matemático dos espaços de estados de um sistema físico com um conjunto de variáveis de entrada, de saída e de estado relacionadas entre si por meio de equações diferenciais de primeira ordem. Já o controle ótimo usa um conjunto de equações diferenciais lineares com um custo a ser minimizado que é descrito por uma função quadrática. Temos também a criação de métodos de controle como o robusto e o adaptativo, que lidam com incerteza e com a variação de parâmetros respectivamente.

2.2 Teoria de Controle

Teoria de controle é um ramo da matemática que lida com o controle de sistemas dinâmicos em processos e máquinas. O objetivo, em suma, é desenvolver um controlador, é comum que isso seja feito com a modelagem matemática do sistema a ser controlado, que comande o sistema preferêncialmente sem *delay* ou *overshoot*, e que garanta estabilidade. Esse método de arquitetar controladores mesmo sendo empregado por muito tempo, continua sendo usado nos dias de hoje, porém, com o avanço da indústria e da ciência, novos processos e novas pesquisas vêm criando desafios que exigem o melhoramento dos métodos de controle e o possível desenvolvimento de novos. Podemos distinguir as diferentes técnicas da teoria de controle de acordo com os diferentes métodos usados, baseando-se em modelos ou em dados.

Essas técnicas são altamente dependentes do tipo da processo a ser controlado e a presença ou não de conhecimentos prévios sobre suas particularidades.

2.3 Controle baseado em modelo

O controle de sistemas pode ser representado por equações matemáticas denominadas modelos matemáticos. Esses modelos são muito úteis para análise e construção de sistemas de controle. O modelo de equações diferenciais é do tipo que opera no domínio do tempo. Para chegar nesse modelo faz-se os seguinte: aplicando as leis básicas que governam o funcionamento do sistema e obtêm-se delas as equações diferenciais em termos de entrada e saída por meio da eliminação das variáveis intermediarias. Já na modelagem no domínio da frequência os sinais dos sistemas são examinados com respeito a frequência, podendo também conter informações sobre o deslocamento de fase. O uso deste domínio pode ser vantajoso por simplificar a análise matemática uma vez que converte equações diferenciais para algébricas que são mais fáceis de se resolver.

O controle ótimo e a introdução de parâmetros em espaços de estados por (KALMAN, 1960) deram início ao controle moderno, que abrange sistemas lineares e sistemas não lineares. Para os casos lineares esses sistemas são costumeiramente projetados com alocação de polos e zeros, como no controle linear quadrático (LQR - Linear Quadratic Regulator) ou controle linear quadrático Gaussiano (LQG - Linear Quadratic Gaussian) e no controle robusto. Todas estas diferentes metodologias são consideradas técnicas de controle baseado em modelo, ou MBC (Model-Based Control). Sistemas não lineares por outro lado são usualmente projetados com controles baseados em Lyapunov ou sistemas de linearização por realimentação. Essas formas de se projetar controladores são baseadas em modelo e sua elaboração começa com a modelagem da planta, ou identificação do modelo, seguido do projeto do controle baseado no modelo obtido, esse controle leva em consideração o princípio da certeza de equivalência, tendo assim uma prerrogativa que o modelo representa o sistema real. Para um planta ser modelada baseada em princípios fundamentais é preciso que os parâmetros sejam calibrados *on* e *offline* usando dados medidos, esse modelo deve seguir uma configuração que cubra o sistema real ou ao menos seja próximo o suficiente em termos de viés de variância de erro no modelo constatado. Um modelo nunca será uma representação perfeita da realidade, então a aparição de erros é inevitável, dinâmicas não modeláveis sempre estarão presentes no processo. Isso faz dos controladores baseados em modelos inerentemente menos seguros.

A fim de contornar as deficiências de controladores baseados em modelos foram desenvolvidas várias formas de se escrever os erros do modelo em sistema de malha fechada, como por exemplo descrições aditivas, multiplicativas, a suposição à priori dos limites dos erros ou na incerteza da modelagem. Porém esses contornos são limitados, a modelagem por princípios fundamentais não tem muito a oferecer no quesito da quantificação explícita de erros. O principal obstáculo na aplicação de projeto de modelos baseado em controle robusto é a falta de descrições

de incertezas que sejam adequadas e práticas. Existem obstáculos práticos e teóricos para se estabelecer uma teoria de controle que se aplique a todos os casos, uma vez que dinâmicas não modeladas e robustez são partes opostas que se recusam a serem resolvidas simultaneamente numa estrutura de um controlador baseado em modelo convencional, e quanto mais preciso o modelo for, maior será o esforço gasto no design do sistema de controle. Não existe um procedimento que seja capaz de elaborar um modelo eficiente e preciso da planta, como também existem tipos de complexidades, como as de ordem muito alta, que ainda não se sabe endereçar, e que portanto não podemos fazer uso delas como modelo de sistemas controle. Conjuntamente deve-se levar em conta a condição de persistência de excitação da entrada, sem impulsos persistentemente excitantes não é possível construir um modelo preciso, sem um modelo preciso a maioria dos resultados teóricos baseados em modelos resultam num sistema de malha fechada cuja estabilidade e convergência não podem ser garantidas.

Controladores baseados em modelos são fundamentados pelo princípio da certeza de equivalência, se o modelo em que for baseado não possuir semelhança a planta, o controlador não terá um bom funcionamento (HOU; WANG, 2013). Mesmo quando o modelo é preciso o suficiente, os resultados das análises teóricas, como as de estabilidade, convergência e robustez de um sistema de malha fechado, com rigorosas provas matemáticas, nem sempre serão válidos se as suposições feitas sobre o sistema não estiverem corretas. O controle adaptativo, por exemplo é feito sob uma série de suposições, usando um algoritmo, todos os sinais são restritos quando o tempo tende a infinito, daí alguns resultados sucedem. Tudo isso pode ser feito e a conclusão resultante ser valiosa, no entanto não é o suficiente para dar segurança quanto ao uso do controlador. O teorema não pode proteger a planta de um controle adaptativo conectado, isso se dá porque a solução dada exclui a possibilidade que em algum momento do tempo ao ir ao infinito, o controle conectado a planta vai render o *loop* planta-controlador instável, como foi demonstrado por (ROHRS et al., 1982), uma vez que podem surgir comportamentos erráticos frutos de uma dinâmica não prevista na criação do modelo. Os métodos recursivos, incluindo o de Lyapunov, são também comuns para construção de sistemas de controle não lineares, porem são dependentes que o modelo da planta esteja correto. No entanto fatores impossíveis de ser considerados são inevitáveis durante a modelagem, o que pode levar os controladores à inutilidade se o modelo não estiver correto. Como equações cinéticas e modelos estruturais são incluídos nos controladores, sua precisão pode influenciar na desempenho do sistema. A teoria de controle que utiliza modelos funciona muito bem enquanto permanece teoria, sua aplicação prática pode encontrar problemas variados.

2.4 Modelagem e Identificação

Existem duas maneiras de criação de modelos, a fenomenológica que descreve empiricamente a relação de um fenômeno à outro, de forma que seja consistente com a fundamentação teórica, também é conhecida como modelagem a partir de princípios fundamentais. A outra

forma de modelagem é por identificação, que usa dados coletados para a partir dos mesmos extrair o comportamento do sistema com certo grau de confiabilidade.

2.5 Controle Baseada em Dados

A indústria possui tecnologias de processo, equipamentos e sistemas de produção cada vez mais complexos, o que torna a modelagem dos mesmos a partir de princípios fundamentais progressivamente mais difícil. A quantidade de dados gerados a todo momento, todos os dias é gigantesca. Os processos industriais dependem desses dados para aferir o estado das operações e equipamentos. Usar esses dados, *on* e *offline*, para projetar diretamente controladores, prever e avaliar o estado do sistema, avaliar desempenho, tomar decisões e diagnosticar falhas, seria muito significativo. Portanto foi possível que a teoria de controle desenvolvesse novos métodos baseados em dados. Até o momento existem poucos métodos de controle baseados em dados, dentre eles: *model free control*, controle adaptativo livre de modelo, ajuste iterativo por realimentação, ajuste de realimentação por referência virtual (VRFT - do inglês *Virtual Reference Feedback Tunning*), controle com aprendizagem iterativa.

2.5.1 Definindo um Controlador Baseado em dados

Um controlador baseado em dados é projetado dependendo somente de entradas e saídas medidas na planta a ser controlada. Essa abordagem é caracterizada pelo fato da construção do controlador e da análise do sistema de controle serem feitos usando somente os sinais de entrada e saída lidos do sistema de controle em malha fechada, sem o uso de modelo para a planta. Assim, a planta não é mais quem domina o processo de construção do controlador. Os dados lido no sistema de malha fechada são o ponto de partida para os projetos, como também são o critério final para análize de desempenho do sistema. Algumas métodologias de projeto de controladores baseados em dados podem incluir informações implícitas da estrutura da planta controlada, assim como podem usar uma estrutura de controlador prédeterminado, de forma que o processo de sintonia é feito *offline*. Com intuito de englobar todos os métodos podemos usar a definição que controle *data-drinven* inclui todas as metodologias de controle nas quais um controlador é projetado usando diretamente dados de entrada ou saída, *on* e *offline*, do sistema a ser controlado, mas nenhuma informação explícita do modelo matemático do controlador. A estabilidade, convergência e robustez podem ser garantidos por uma análise matemática razoável sob certas condições (HOU, 1994).

2.6 Alguns métodos de controle baseados em dados existentes:

2.6.1 *Simultaneous Perturbation Stochastic Approximation* (SPSA)

Proposto por (SPALL, 1992), esse método usa somente dados lido em malha fechada, descartando a modelagem matemática da parte controlada da planta, para sintonizar os parâmetros

do controlador. O método baseado em SPSA assume que as dinâmicas não lineares do controlador são desconhecidas. O controlador serve como um aproximador da função cuja estrutura é fixa. Os parâmetros são sintonizáveis. O aproximador pode ser uma rede neural, polinomial ou qualquer outro tipo de aproximador. Se uma rede neural por exemplo seja escolhida como o controlador, o número de camadas e nós são determinadas, então o coeficiente sintonizável é o parâmetro do controlador.

2.6.2 *Model-free Adaptative Control* (MFAC)

Proposta inicialmente por Hou (1994), em que a ideia principal é que, usando um modelo de linearização dinâmico com um conceito original chamado pseudo derivação parcial em cada ponto da operação para substituir o sistema não linear discreto no tempo, daí estimar a pseudo derivação parcial online usando somente a entrada e saída da planta controlada, e finalmente construir a estratégia de controle sem modelo para uma classe de sistemas não lineares.

2.6.3 *Unfalsified Control* (UF) *methodology*

Metodologia proposta por Safonov e Tsao (1995), ela recursivamente falsifica os parâmetros de controle que falham as especificações de desempenho. O processo é todo baseado nos dados de entrada e saída e não usa modelo matemático da planta. UC é um método de controle regressivo que se diferencia dos métodos regressivos tradicionais. UC pode falsificar o controlador, o qual não pode estabilizar o sistema de controle antes de ser inserido no loop de realimentação, então o desempenho transiente é relativamente bom. Os principais elementos do controle não falsificado são: um conjunto invertível de controladores candidatos, especificações de desempenho detectável pelo custo, e o mecanismo de regressão.

2.6.4 *Iterative feedback tuning* (IFT)

É um esquema típico de controle baseado em dados que envolve a otimização iterativa do parâmetro do controlador fixo para um gradiente de deempenho de controle. Foi proposto por Hjalmarsson, Gunnarsson e Gevers (1994). Em cada iteração a estimativa é construída a partir de um conjunto finito de dados obtidos parcialmente pela operação do sistema de malha fechada em condições normais e parcialmente por um experimento no qual a saída do sistema é realimentada em malha fechada.

2.6.5 *Correlation based tuning* (CBT)

Proposto por Karimi, Mišković e Bonvin (2002), é um método iterativo de sintonização do controlador que se baseia em dados. A ideia por trás desse método é inspirada na conhecida abordagem de correlação na identificação do sistema. Os parâmetros do controlador são sintonizados iterativamente seja para retirar a correlação do erro da saída em malha fechada

entre o sistema designado e o encontrado com o sinal de referência externa, ou para reduzir essa correlação.

2.6.6 *Noniterative data-driven model reference control*

Proposto por Karimi, Heusden e Bonvin (2007). Essa abordagem de sintonização do controlador leva a um problema de identificação onde a entrada é afetada somente pelo ruído mas a saída não. O objetivo desse método é chegar a um controlador linear que se aproxime do modelo de referência. O critério usado é não convexo em relação aos parâmetros do controlador. Uma aproximação que seja convexa para um controlador linearmente parametrizado pode ser definido usando o modelo de referência.

2.6.7 *Subspace Approach*

Na literatura as abordagem subspace Huang e Kadali (2008), data space Fujisaki, Duan e Ikeda (2004) e a data-driven simulation Markovsky et al. (2005) se mostraram compartilhar a mesma ideia, que as dinâmicas do sistema são representadas como um subespaço vetorial de dimensões finitas, consistindo de séries no tempo de dados de entrada, estado e saída ou entrada e saída. O pilar destes métodos é que a base desse subespaço vetorial de dimensões finitas, chamado de matriz dinâmica, envolve todas as informações dinâmicas do sistema. Existem várias técnicas de identificação disponíveis para a estimação desse espaço de estado.

2.6.8 *Approximate Dynamic Programming* (ADP)

Foi proposto (WERBOS; MILLER; SUTTON, 1990) como solução para problemas do controle ótimo que prossegue no tempo. Esse método combina aprendizagem reenforçada usando estruturas críticas adaptativas, com programação dinâmica. ADP inclui quatro esquemas principais: programação dinâmica heurística, programação dinâmica dual-heurística, programação dinâmica heurística baseada em ação (Conhecida como *Q-learning*), programação dinâmica dual-heurística baseada em ação. *Q-learning* é introduzido aqui porque o método não requer conhecimento do modelo da planta, *Q-learning* é um método que foi proposto como solução para processos discretos de decisões de Markov, em que o número de estados e ações são finitos e o modelo não está disponível.

3 METODOLOGIA

Este capítulo tem o objetivo de apresentar as etapas realizadas no projeto de um controlador para controle de velocidade de um motor de corrente contínua. Um circuito de controle de corrente foi usado com o fim de ajustar a tensão aplicada no motor: um conversor buck. Esse conversor mantém a tensão de acordo com uma escala prédefinida. A leitura da tensão e o sinal de referência foram feitos pelo arduino. A velocidade do motor é medida em radianos por segundos em um encoder ótico que usa um disco com frestas acoplado ao eixo do motor. Essa velocidade é a saída do processo y(t). A tensão aplicada ao motor é a variável manipulada u(t).

Um motor de corrente corrente contínua é um equipamento já bastante conhecido. Quando o propósito é controlar sua velocidade, geralmente o que se faz é modelar uma função de transferência de segunda ordem que representa suas duas principais dinâmicas: parte mecânica e elétrica. A dinâmica elétrica tende a ser muito mais rápida que a mecânica, o que frequentemente motiva o uso de um modelo simplificado de primeira ordem. Mas essa simplificação pode não ser adequada para aplicações que exigem maior desempenho. Se trabalharmos sob a suposição que o processo de controle da velocidade é propriamente representado por um modelo na forma discreta de segunda ordem, ou seja, que a função de transferência do processo é da forma

$$G(z) = K_g \frac{z}{(z-b)(z-c)}. \tag{3.1}$$

Com números reais positivos K_g e com $b < 1$ e $c < 1$. Escolheu se projetar um controlador proporcional e integral (PI), que é um dos tipo mais usados para esse tipo de processos. Um controle PI tem a seguinte estrutura,

$$C(z,\rho) = \begin{bmatrix} K_p & K_i \end{bmatrix} \begin{bmatrix} 1 \\ \frac{z}{z-1} \end{bmatrix}. \tag{3.2}$$

Onde K_p é o ganho proporcional e K_i o ganho integral. Definimos $K_c = K_p + K_i$, e temos que $f = \frac{K_p}{K_c}$, assim o sistema de malha fechada é descrito pela função de transferência:

$$T(z,\rho) = \frac{K_c z(z-f)}{(z-b)(z-c)(z-1) + Kz(z-f)}. \tag{3.3}$$

A fim de facilitar o desenvolvimento matemático escolhe-se um modelo de referência também de segunda ordem que corresponda à anulação de um dos polos do processo com um zero do controlador, por exemplo fazendo $f = c$. Com isso a função de transferência pode ser

escrita como

$$T(z,\rho) = \frac{k_g k_c z}{(z-1)(z-b) + k_g k_c z} = \frac{k_g k_c z}{z^2 + (k_g k_c - b - 1)\, z + b}. \tag{3.4}$$

O modelo da referência $T(z)$ deve ser escolhido de forma que a equação (3.4) se iguale à $T_(z)$ para algum valor de K_c. Isso ocorrerá para qualquer número real K na seguinte função de transferência

$$T(z,\rho) = \frac{k_g k_c z}{z^2 + (k_g k_c - b - 1)\, z + b}. \tag{3.5}$$

A escolha do modelo de referência requer conhecimento de um dos polos do processo, no caso o parâmetro b, uma vez que é necessário para que se faça cumprir hipótese B_y (Bazanella (2012) página 13). Como podemos facilmente identificar o polo dominante de forma aproximada com a resposta do sistema em malha aberta é possível obter uma representação razoável do sistema e assim conseguir criar um modelo de referência de segunda ordem, fazendo com que a condição correspondente seja satisfeita. Por últimos temos que escolher um parâmetro K de forma que defina o modelo de referência $T_d(z)$. Qualquer valor de K é possível de ser usado, mas é claro que valores diferentes resultam em performances diferentes.

3.1 VRFT

De acordo com Bazanella (2012) *Virtual reference Feedback Tuning* (VRFT) é um método de de otimizar o rastreamento da referência virtual a partir de um índice de custo $J_y(\rho)$. Uma das característica mais chamativa é o fato do método não ser iterativo. A ideia principal do método é criar o projeto do problema como uma previsão de identificação do erro de um modelo para um controlador ideal, de forma auto regressiva. Para fazer uma predição de identificação em um modelo procura-se minimizar uma função quadrática $J^{VR}(\rho)$ em função dos parâmetros do modelo. Idealmente o mínimo global dessa função será igual ao mínimo global de $J_y(\rho)$, mas sua otimização seria menos trabalhosa por ser uma função quadrática. A otimização da função custo $J_y(\rho)$ do rastreamento da referência virtual pode ser problemático, uma vez que máximos e mínimos locais podem ocorrer nas mesmas. Já na função $J^{VR}(\rho)$ não existe essa inconveniência.

Seja em uma malha aberta ou fechada, coletando dados de entrada $u(t)$ e de saída $y(t)$ de um sistema. De modo a se obter medidas $y(t)$, pode-se definir uma referência virtual $r(t)$ tal que

$$T_d(z)r(t) = y(z). \tag{3.6}$$

onde T_d é o modelo de referência desejado para a resposta em malha fechada.

Se o sistema estiver operando em malha fechada com um controlador ideal $C_d(z)$, e a referência virtual $r(t)$ for aplicada seus dados de saída seriam os mesmo que os dados $u(t)$ e

$y(t)$ que foram coletados. O VRFT funciona como se esse experimento falso realmente tivesse sido feito. Assumindo hipoteticamente, que os dados são coletados. Dessa forma, o erro virtual será dado por

$$\bar{e}(t) = r(t) - y(z). \tag{3.7}$$

No experimento virtual o controlador tem como entrada o sinal de erro. Portanto, temos os sinais de entrada e saída do controlador, $e(t)$ e $u(t)$ respectivamente. Usando os dados que temos desse falso experimentos podemos identificar o controlador $C_d(z)$. Sendo $e(t)$ a entrada, $C(z,\rho)$ o modelo para esse controlador, temos que $C(z,\rho)e(t)$ será a saída deste controlador. Um critério de predição de erro $J^{VR}(\rho)$ pode ser formado como:

$$J^{VR}(\rho) \triangleq \bar{E}[u(t) - C(z,\rho)\bar{e}(t)]^2. \tag{3.8}$$

Para um controlador de parâmetros lineares, o critério $J^{VR}(\rho)$ pode ser escrito como

$$\begin{aligned} J^{VR}(\rho) &= \bar{E}\left[u(t) - \rho^T \bar{C}(z)\bar{e}(t)\right]^2 \\ &= \bar{E}\left[u(t) - \rho^T \varphi(t)\right]^2. \end{aligned} \tag{3.9}$$

Sendo o vetor de regressão φ definido como

$$\varphi(t) = \bar{C}(z)\bar{e}(t) = \bar{C}(z)\frac{1 - T_d(z)}{T_d(z)}y(t). \tag{3.10}$$

Essa é uma formulação padrão para a identificação dos parâmetros do sistema, onde o sistema a ser identificado é o controlador ideal, $u(t)$ é a saída que é mensurada e $e(t)$ é a entrada também mensurada. Os dados são então gerados de

$$u(t) = \bar{C}(z)\bar{e}(t). \tag{3.11}$$

A função de custo de identificação de erro $J^{VR}(\rho)$ pode ser desenvolvida para

$$\begin{aligned} J^{VR}(\rho) &= \bar{E}\left[\left(\rho_d - \rho^T\right)\varphi(t)\right]^2 \\ &= E\left[\left(\rho_d - \rho^T\right)\tilde{C}(z)\frac{1 - T_d(z)}{T_d(z)}G(z)u(t)\right]^2, \end{aligned}$$

ou alternativamente

$$J^{VR}(\rho) = \bar{E}\left[\left(\rho_d - \rho^T\right)\bar{C}(z)\frac{1}{C_d(z)}u(t)\right]^2. \tag{3.12}$$

Que pelo Teorema de Parseval[0] pode ser escrito como

$$J^{VR}(\rho) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{\left|C_d\left(e^{J\omega}\right)\right|^2} \left|(\rho-\rho_d)^T \bar{C}\left(e^{J\omega}\right)\right|^2 \Phi_u\left(e^{J\omega}\right) d\omega. \tag{3.13}$$

Para garantirmos que $C_d(z)$ não possui zeros no círculo unitário ou que esses zeros não correspondam a frequências no mínimo de $u(t)$; de outra forma a integral não existe. Considerando (3.12) notamos que $J^{VR}(\rho) = 0$ e portanto ρ_d é um mínimo global de $J^{VR}(\rho)$. Além disso, este índice se apresenta numa forma quadrática:

$$J^{VR}(\rho) = (\rho-\rho_d)^T A^{VR}(\rho-\rho_d),$$

onde

$$A^{VR} = \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{1}{\left|C_d\left(e^{j\omega}\right)\right|^2} \bar{C}\left(e^{j\omega}\right) \bar{C}^*\left(e^{j\omega}\right) \Phi_u\left(e^{j\omega}\right) d\omega. \tag{3.14}$$

é uma matriz semidefinida positiva por construção. Se torna claro que ρ_d é um mínimo global dessa função, como desejado, desde que providenciado que A^{VR} seja uma matriz definida positiva. Se assumimos que a entrada tem um espectro discreto, ou seja,

$$\Phi_u\left(e^{j\omega}\right) = \sum_{k=1}^{q} \lambda_k \delta\left(\omega-\omega_k\right), \tag{3.15}$$

onde $\delta(.)$ é a função delta de Dirac, e λ_k são números positivos. Com isso

$$A^{VR} = \frac{1}{2\pi}\sum_{k=1}^{q} \frac{1}{\left|C_d\left(e^{j\omega_k}\right)\right|^2} \bar{C}\left(e^{j\omega_k}\right) \bar{C}^*\left(e^{j\omega_k}\right) \lambda_k. \tag{3.16}$$

Cada uma das matrizes $C(e^{j\omega_k})^* C(e^{j\omega_k})$ é semi positiva e definida, e a matriz A^{VR} é uma função peso q dessas matrizes. A classificação dessa soma é $min(q, \rho)$ como uma propriedade genérica, ou seja, para quase todo espectro com q elementos diferentes de zero.

Um processo quasi-estacionário é dito ter uma ordem suficientemente rica q se seu espectro tem ao menos q componentes não nulos. Com isso podemos dizer que o parâmetro ótimo do controlador ρ_d é o mínimo global único de $J^{VR}(\rho)$ se o sinal de entrada for suficientemente rico em ordem ρ.

[0] https://pt.wikipedia.org/wiki/Teorema-de-Parseval

Voltando para a equação padrão (3.12), o problema de encontrar o mínimo global de uma função dessa forma pode ser resolvido pelo método dos mínimos quadrados, cuja solução é dada pelas soluções $\rho*$ do seguinte sistema de equações lineares, conhecida como equação normal:

$$\bar{E}\left[\varphi(t)\varphi^T(t)\right]\rho_* = \bar{E}[\varphi(t)u(t)]. \tag{3.17}$$

A solução da equação normal existe e é única se a matriz $\bar{E}\left[\varphi(t)\varphi^T(t)\right]\rho_*$ for de posto completo.

Um vetor quasi-estacionário $\phi(t)$ é dito ser persistentemente excitante se

$\bar{E}\left[\varphi(t)\varphi^T(t)\right]\rho_* > 0.$

Podemos obter o vetor regressor filtrando a entrada $u(t)$, para a condição de excitação persistente, que é necessário para a solução da equação normal seja única, para traduzir para condições em $u(t)$. Esse problema também chamado de 'transferência de excitação', acontece que, apesar de alguns detalhes técnicos, podemos dizer que o vetor regressor é persistentemente excitante se $u(t)$ for de ordem suficientemente rica. Se fizermos o mínimo de $J^{VR}(\rho)$ sob a suposição que o controlador encontrado esta na gama de controladores possíveis, produzimos o resultado desejado, que é o mínimo global do critério de rastreamento de desempenho $J_y(\rho)$. Esse é o ponto principal do método VRFT. Porém a equação (3.17) não é uma quantia diretamente computável, uma vez que envolve expectativas. Dado um conjunto de N entradas e saídas coletadas do sistema, a solução da equação normal é obtida pelos cálculos a seguir:

$$\hat{\rho} = \left[\sum_{t=1}^{N}\varphi(t)\varphi^T(t)\right]^{-1}\sum_{t=1}^{N}\varphi(t)u(t) = \rho_*. \tag{3.18}$$

Esses cálculos resultarão exatamente na solução assintótica somente nesse caso ideal, onde não há ruído. E se tratando de sinais com ruídos, então o resultado da equação (3.17) será o parâmetro ρ e o mesmo será uma variável aleatória. De qualquer forma, essa característica não iterativa é a vantagem chave de se trabalhar com o critério VRFT no lugar do critério de referência do modelo $J_y(\rho)$.

Em suma, considerando que o sinal medido é livre de ruído e que o sinal de entrada é de ordem rica o suficiente, e sobre a suposição de ser linearmente parametrizado, o processo VRFT identifica o controlador.

3.2 O método VRFT

3.2.1 A ideia de modelo de referência

Temos uma planta $G(s)$, que $u(t)$ é o sinal de entrada e o sinal de saída que chamamos de $y(t)$, essa é uma planta no modelo contínuo, como apresentado na Figura 1, a princípio

não precisamos saber seu modelo, o que estamos buscando conhecer é qual sinal de controle que aplicamos à ela para que sua saída siga um modelo de referência que é o comportamento desejado.

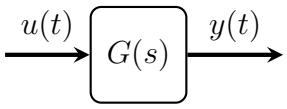

Figura 1 – Diagrama de blocos representando motor.

Primeiramente é preciso definir-se um modelo de referência, ou seja , como o sistema deve comportar em malha fechada, e isso é feito lendo o sinal de saída e usando o mesmo para realimentar a planta, esse sinal será comparado com o sinal de referência, no esquema que usamos o sinal já se encontra na forma discreta pois é fruto de um encoder que lê velocidade angular, e a referência já tem seu valores discretos no tempo, uma vez que foi gerada virtualmente. O sinal resultante da soma do sinal de saída da planta com a referência desejada que chamamos de $e(k)$, é a entrada do controlador, como mostrado na Figura 2.

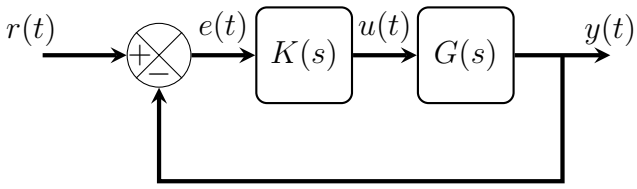

Figura 2 – Diagrama de blocos da planta .

3.2.2 Definindo o modelo

Para identificar o controlador pelo modelo de referência é preciso que seja definido o comportamento do modelo que vamos chamar de $T'(z)$, como indicado na Figura 3

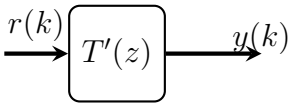

Figura 3 – Diagrama de blocos representando modelo.

Para nossa aplicação definimos que esse modelo tenha ordem unitária, ou seja grau relativo igual a um. Esse modelo terá a seguinte fórmula

$$T'(s) = \frac{1}{\tau s + 1}. \tag{3.19}$$

O termo τ representa a constante de tempo do modelo, ou seja, o tempo que o sistema $T'(s)$ leva para chegar a aproximadamente 63% do valor final. Ajustando o sistema em malha

fechada desejado $T_d(z)$ para que ele se comporte da mesma forma $T(s)$, porém no domínio discreto.

Ao transformar o modelo para forma discreta temos

$$T_d(z) = \frac{(1-A)z^{-1}}{1-Az^{-1}}. \tag{3.20}$$

A relação entre A e s é $A = e^{-\frac{1}{\gamma}T_s}$

Uma vez que temos $T(z) = \frac{Y(z)}{R(z)}$ se multiplicarmos cruzado com (3.20) temos que

$$(1-A)z^{-1}R(z) = Y(z) - AzY(z). \tag{3.21}$$

Esse modelo é usado de forma inversa para achar a referência virtual, que é de onde o método tira seu nome. O sinal não existe fisicamente, na prática ele é encontrado ao calcular-se $R(z)$ desse modelo.

3.2.3 A equação do erro

A partir dos dados amostrados em malha aberta da planta, dados pela a tensão aplicada no motor $u(t)$, e pela sua velocidade angular $y(t)$ e para que nosso controlador produza o efeito pretendido temos que o resultado do nosso modelo será os mesmo desses dados, ou seja as saídas serão idênticas, para isso $r(t)$ será igual a $u(t)$.

Se fizermos a transformada inversa da equação (3.21) tem-se,

$$(1-A)r(k-1) = y(k) - Ay(k-1), \tag{3.22}$$

$$r(k-1) = \frac{1}{1-A}y(k) - \frac{A}{A-1}y(k-1). \tag{3.23}$$

Transladando no tempo, a partir da soma de uma unidade de tempo em todos os argumentos temporais da equação (3.23), chega-se na equação

$$r'(k) = \frac{1}{1-A}y(k+1) - \frac{A}{A-1}y(k). \tag{3.24}$$

Como visto na fig. 2, imagina-se esse sistema como tendo r' de entrada, e como já possuímos $u(t)$ podemos calcular $K(z)$ usando métodos de regressão para identificar o modelo, sendo necessário somente calcular o erro virtual, que como de costume, nomeia-se $e'(k)$, usando os dados colhidos, e esse erro será dado por

$$e'(k) = r'(k) - y(k). \tag{3.25}$$

$$e'(t) = r'(t) - y'(k). \tag{3.26}$$

3.2.4 O controlador

A partir do erro virtual e o sinal de entrada aplicado à planta, pode-se encontrar os parâmetros do controlador. Para tanto, defini-se uma estrutura para a função de transferência do controlador. Adotando-se como exemplo um controlador P.I.D. que tem o seguinte formato:

$$K(z) = K_p + \frac{K_i}{1 - z^{-1}} + K_d\left(1 - z^{-1}\right) = \frac{U(z)}{E(z)}. \tag{3.27}$$

E como temos que $y(z) = G(z)\tilde{e}(z)$ fica

$$G(z) = \left(\frac{1 - z^{-1}}{(1 - A)z^{-1}}\right) y(z), \tag{3.28}$$

temos que $[\tilde{r}(z) - y(z)]$ pode ser desenvolvido

$$\begin{aligned}
[\tilde{r}(z) - y(z)] &= \left[\left(\frac{1 - Az^{-1}}{(1 - A)z^{-1}}\right) y(z) - y(z)\right] \\
&= \left[\left(\frac{1 - Az^{-1}}{(1 - A)z^{-1}} - 1\right) y(z)\right] \\
&= \left(\frac{1 - Az^{-1} - (1 - A)z^{-1}}{(1 - A)z^{-1}}\right) y(z) \\
&= \left(\frac{1 - z^{-1}}{(1 - A)z^{-1}}\right) y(z).
\end{aligned} \tag{3.29}$$

Portanto ficamos com

$$\left[K_p + \frac{k_i}{1 - z^{-1}} + k_d\left(1 - z^{-1}\right)\right]\left(\frac{1 - z^{-1}}{(1 - A)z^{-1}}\right) y(z)$$

$$\left[\frac{k_p(z - 1)}{1 - A} + \frac{k_i z}{1 - A} + \frac{k_d\left(1 - z^{-1} - z^{-1} + z^{-1}\right)}{(1 - A)z^{-1})}\right] y(z)$$

$$\begin{aligned}
&\left[\frac{k_p(z - 1)}{1 - A} + \frac{k_i z}{1 - A} + \frac{K_d\left(z - 2 - z^{-1}\right)}{1 - A}\right] y(z) \\
&\left[\frac{z - 1}{1 - A} \quad \frac{z}{1 - A} \quad \frac{z - 2 - z^{-1}}{1 - A}\right] [y(z)]\, [k_p \;\; k_i \;\; k_d]^{\intercal}.
\end{aligned} \tag{3.30}$$

Vamos definir $\left[\frac{z-1}{1-A} \quad \frac{z}{1-A} \quad \frac{z-2-z^{-1}}{1-A}\right] = \phi$

E temos $[K_p \;\; K_i \;\; K_d] = \theta$

Como podemos ver chegamos a uma equação matricial $U(z) = \phi(z)\theta$, onde θ é um vetor de parâmetros. E esses são os que buscamos identificar, em que chamamos ϕ de matriz de regressores na identificação de sistemas.

3.2.5 Pseudo Inversa

Como procuramos encontrar os valores de θ, é bem intuitivo que peguemos a equação acima e multipliquemos ambos os lados por $\phi^{-1}(z)$, na direita ficamos com a matriz identidade multiplicada por θ, e na esquerda $\phi^{-1}(z)U(z)$ para que isso seja possível é necessário que a matriz de regressores não tenha linhas ou colunas linearmente independentes, sendo essa a razão pela qual usamos sinais persistentemente excitantes na entrada. Teríamos uma solução trivial para equação se ϕ fosse uma matriz quadrada, contudo nessas proporções essa matriz não conteria informações o suficiente para obtermos uma boa estimativa, nesse caso como já mencionado escolhemos um número n de dados. E em vista disso prosseguimos multiplicando a equação pela transposta de ϕ chegamos a uma matriz quadrada

$$\phi^T(z)U(z) = \phi^T(z)\phi(z)\theta. \tag{3.31}$$

Como pretendemos encontrar os valores de θ iremos isola-lo multiplicando toda a equação por $(\phi^T(z)U(z))^{-1}$ chegando a:

$$(\phi^T(z)\phi(z))^{-1}(\phi^T(z)U(z)) = I\theta \tag{3.32}$$

.

Essa forma de equação que chegamos é a pseudo inversa, que é a ideia fundamental da identificação por mínimos quadrados, nesse experimento como já foi dito usamos o MATLAB para realização desses cálculos.

3.3 Motores de Corrente Contínua

3.3.1 Modelagem de Motores C.C.

Um dos dispositivos comumente utilizados como atuadores para controle de sistemas eletromecânicos o motor de corrente contínua, podendo prover movimento rotacional ou torque a partir da eletricidades. Moviemtneos lineares também são possíveis a partir de dispositivos como engrenagens cremalheiras, polias, cabos e correias. Os componentes elétricos do circuito da armadura de um motor de corrente contínua podem ser representados pelo circuito da Figura 4.

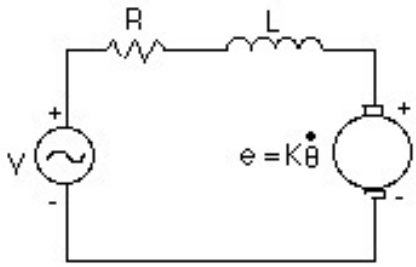

Figura 4 – Circuito representando a armadura do motor CC. Fonte(v.researchgate.net)

Os componentes mecânicos pelo diagrama de corpor livre da Figura 5

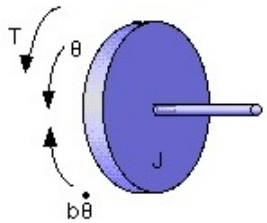

Figura 5 – Diagrama de corpo livre de um motor CC. Fonte(v.researchgate.net)

O torque $T(t)$ do motor está relacionado a corrente na armadura i por uma constante K_t e a força contra eletromotriz $e(t)$ está relacionada a velocidade de rotação $\omega(t)$ por uma constante K_e de acordo com as seguintes equações

$$\begin{aligned} T &= K_t i(t) \\ e &= K_e \dot{\omega(t)}. \end{aligned} \tag{3.33}$$

Com essas equações, e assumindo que $\omega(t) = \dot{\theta}(t)$, podemos escrever as seguintes relações baseando-se nas leis de Newton e combinando com as de Kirchhoff

$$\begin{aligned} J\ddot{\theta}(t) + b\dot{\theta}(t) &= K_i \\ L\frac{di(t)}{dt} + Ri(t) &= V - k\dot{\theta}(t) \end{aligned} . \tag{3.34}$$

Com o auxílio da transformada de Laplace podemos deixar as equações acima e função de s,

$$\begin{array}{l} s\left(Js+b\right)\Theta(s)=KI(s) \\ \left(Ls+R\right)I(s)=V-K_n\Theta(s) \end{array}. \tag{3.35}$$

Assim podemos eliminar o termo $I(s)$ e chegar na função de transferência em malha aberta onde a velocidade de rotação é a saída e a voltagem é a entrada.

A modelagem de motores é um processo relativamente simples, e já é suficiente para suprir muitos casos de uso dos mesmos, porém existem limites na eficácia e confiabilidade desse tipo de abordagem uma vez que é necessário que se assuma a constância de certos parâmetros. O atrito, por exemplo pode mudar consideravelmente com a velocidade. Além do mais, o conhecimento exato destes parâmetros podem ser um problema. Por essa razão métodos baseados em dados, e métodos *on-line*, podem se mostrar vantajosos frente aos baseado em modelos.

3.3.2 Acionamento de Motores C.C.

Motores de corrente contínua são facilmente acionados ao se aplicar tensão, porém para se garantir uma boa performance quanto a velocidade e torque é necessario uma boa calibração da voltagem e da corrente aplicada. Vários dispositivos já foram criados com circuitos próprios para esse tipo de calibração. Dentre eles podemos destacar o conversor buck conhecido por ser eficientes mesmo send orelativamente simples. O Buck é um conversor CC/CC que diminui a tensão enquanto aumenta a corrente de sua entrada para sua saída. É um tipo de fonte chaveada que costumeiramente possui um diodo e um transistor, sendo que em algumas versões mais modernas o diodo seja substituído por um segundo transistor usado para retificação síncrona, um elemento de armazenamento de energia, um capacitor, um ou mais indutores.

3.4 Controladores Embarcados

3.4.1 Microcontroladores

Um microcontrolador é um pequeno dispositivo com capacidade computacional feito de circuitos integrados, possuindo memória e pinos de entrada e saída. O microcontrolador é programável pois toda sua lógica de operação é estruturada na forma de um programa que é gravado dentro do componente e permite controlar o estado das suas saídas, de acordo com o que é desejado (SOUZA; SOUSA, 2012). Costumeiramente um microcontrolador tem uma memoria FLASH, que armazena as instruções, uma memória de variáveis RAM, que armazena os dados durante a execução do programa mas é perdida quando o dispositivo é desligado, e uma terceira que armazena informações como variáveis a longo prazo. O microcontrolador possui um microprocessador que interprete os dados e as instruções do programa, além de variadas portas

de entrada e saídas que são a forma com que o microcontrolador interage com o meio externo. Microcontroladores são também acompanhados de vários outros instrumentos como Timers, controladores de interrupção, comunicação serial, geradores de modulação por largura de pulso (PWM), conversores analógico/ digital, etc.

3.4.2 Arquitetura dos Microcontroladores

Os microcontroladores podem ser divididos em dois principais grupos: Os que utilizam a arquitetura Harvard e os que utilizam a arquitetura von-Neumann, sendo a principal diferença entre elas a forma de trocar informações entre a Unidade Central de Processamento (CPU do inglês "*Central Processing Unit*") e a memória. Os microcontroladores que utilizam a arquitetura von-Neumann possuem somente uma zona de memória disponível e um barramento. Devido ao fato de todos os dados serem trocados utilizando estas 8 linhas, este barramento é sobrecarregado e a comunicação acaba se tornando extremamente lenta e ineficiente. Dessa forma o CPU pode ler uma instrução ou ler/escrever dados na memória, não sendo possível realizar ambas as operações de forma simultânea, pois as instruções compartilham o barramento com os dados. Na figura 6 é mostrado o esquema de barramentos da arquitetura Von Neumann.

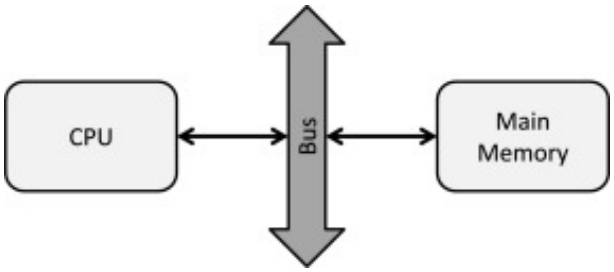

Figura 6 – Arquitetura Von Neumann.
Fonte: sciencedirect

Os microcontroladores que utilizam a arquitetura Harvard possuem dois barramentos internos, como mostrado na figura 7, um barramento de dados de 8 bits e um barramento de instruções, que pode ser de 12, 14 ou 16 bits. Essa arquitetura possibilita que, enquanto uma instrução está sendo executada, outra seja buscada na memória, aumentando assim a velocidade de processamento desejado (SOUZA; SOUSA, 2012).

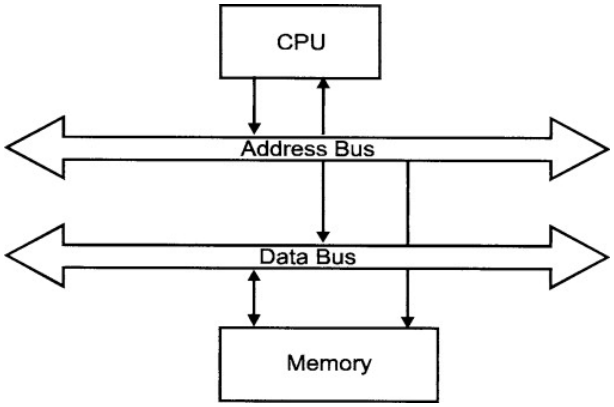

Figura 7 – Arquitetura Harvard.
Fonte: sciencedirect

3.5 Arduino

Arduino é uma plataforma eletrônica open-source construida com o intuito de ser de fácil utilização quanto ambos software e hardware. Os módulos Arduino são capazes de ler entradas, seja um sensor, botão ou uma mensagem, e transforma-los em uma saída como por exemplo ligar um monitor, acender uma luz ou mandar uma mensagem online. Dessa forma foi criada uma placa composta por um microcontrolador Atmel que pode ser facilmente conectada à um computador e programada via IDE (do inglês “Integrated Development Environment”) usando uma linguagem baseada em C/C++, dispensando a necessidade de outros equipamentos além de um cabo USB (THOMSEN, 2020). Atualmente é possível encontrar uma vasta gama de modelos de placas Arduino a fim de atender à necessidades diferenciadas, as placas se diferenciam em tamanho, forma, número de entradas e saídas, memória.

	Arduino Uno	Arduino Mega2560	Arduino Leonardo	Arduino Due	Arduino ADK	Arduino Nano	Arduino Pro Mini	Arduino Esplora
Microcontrolador	ATmega328	ATmega2560	ATmega32u4	AT91SAM3X8E	ATmega2560	ATmega168 (versão 2.x) ou ATmega328 (versão 3.x)	ATmega168	ATmega32u4
Portas digitais	14	54	20	54	54	14	14	-
Portas PWM	6	15	7	12	15	6	6	-
Portas analógicas	6	16	12	12	16	8	8	-
Memória	32 K (0,5 K usado pelo bootloader)	256 K (8 K usados pelo bootloader)	32 K (4 K usados pelo bootloader)	512 K disponível para aplicações	256 K (8 K usados pelo bootloader)	16 K (ATmega168) ou 32K (ATmega328), 2 K usados pelo bootloader	16 K (2k usados pelo bootloader)	32 K (4 K usados pelo bootloader)
Clock	16 Mhz	16 Mhz	16 Mhz	84 Mhz	16 Mhz	16 Mhz	8 Mhz (modelo 3.3v) ou 16 Mhz (modelo 5v)	16 Mhz
Conexão	USB	USB	Micro USB	Micro USB	USB	USB Mini-B	Serial / Módulo USB externo	Micro USB
Conector para alimentação externa	Sim	Sim	Sim	Sim	Sim	Não	Não	Não
Tensão de operação	5v	5v	5v	3.3v	5v	5v	3.3v ou 5v, dependendo do modelo	5v
Corrente máxima portas E/S	40 mA	40 mA	40 mA	130 mA	40 mA	40 mA	40 mA	-
Alimentação	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	7 - 12 Vdc	3.35 - 12 V (modelo 3.3v), ou 5 - 12 V (modelo 5v)	5v

Figura 8 – Comparação entre modelos de placa de Arduino.
Fonte: filipeflop.com.br

Nesse trabalho, por uma questão de disponibilidade, foi utilizado o modelo Arduino Nano, com um controlador ATmega328 e com nível lógico operando em cinco volts, possui quatorze pinos de entrada e saída dos quais seis provêm saída em PWM.

3.6 Desenvolvimento da plataforma de Controle

O cerne deste trabalho é a avaliação do método de controle, portanto a fim de reduzir as incertezas que podem surgir na montagem um um circuito e no uso de componentes diversos, foi optado pelo uso de dispositivos que já estavam à disposição.

3.6.1 Motor utilizado

Foi empregado neste trabalho um pequeno motor de corrente contínua, como visto na Fig 9, projetado para funcionar entre as faixas de -12 e 12V, nele já havia acoplado um encoder com uma resolução de 334 pulsos por revolução.

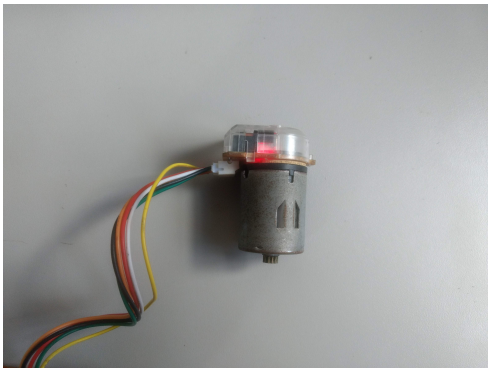

Figura 9 – Motor utilizado no trabalho

3.6.2 Circuito de acionamento de motor

.

Como mencionado no Cap 3 foi usado um conversor buck, que pode ser visto na Fig 10 , sendo este um conversor confiável quanto ao mantenimento da corrente em relação a tensão

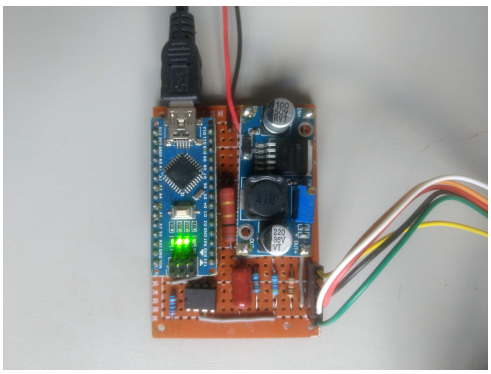

Figura 10 – Foto da placa montada com o arduino (a esquerda) e o conversor buck

3.7 Projeto do Controlador VRFT para Controle de Velocidade do Motor

A aplicação do método VRFT pode ser facilmente replicada para vários tipos de experimentos.Neste trabalho o método descrito no capitulo 3, foi empregado à um motor de corrente contínua a aplicação de um sinal de entrada, cálculo do erro, e construiu-se um modelo de referência. Como VRFT é um método de controle baseado em dados o primeiro passo é colher estes dados. Para isso, um sinal de entrada $u(t)$ na forma de tensão é aplicado ao motor, que por usa vez produz como saída um sinal de velocidade, tomado como snal de saída $y(t)$ da planta. Seguindo os procedimentos expostos no Cap 3, diferentes tipos de sinais com diferentes amplitudes, podem ser vistas na Figura 11 que mostram o sinal de entrada e a velocidade angular do motor lida em radianos por segundo.

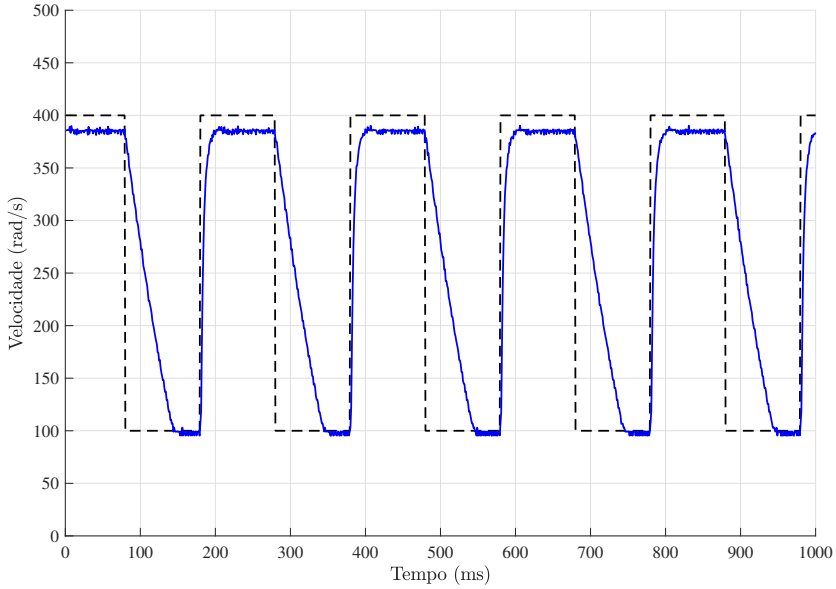

Figura 11 – Resposta de velocidade angular em malhar aberta (azul), considerando um sinal de entrada na forma de onda quadrada (tracejada em preto)

Tendo passado pelos processos descritos no Cap 3, os resultados se mostram similares independente da amplitude da tensão, ou mesmo do ruído, dado que esses estejam numa extensão razoável do funcionamento do motor. Na Figura 12 temos os dados utilizados para identificar os parâmetros do controlador, que se dá pelo cálculo do erro virtual a partir dos sinais de entrada.

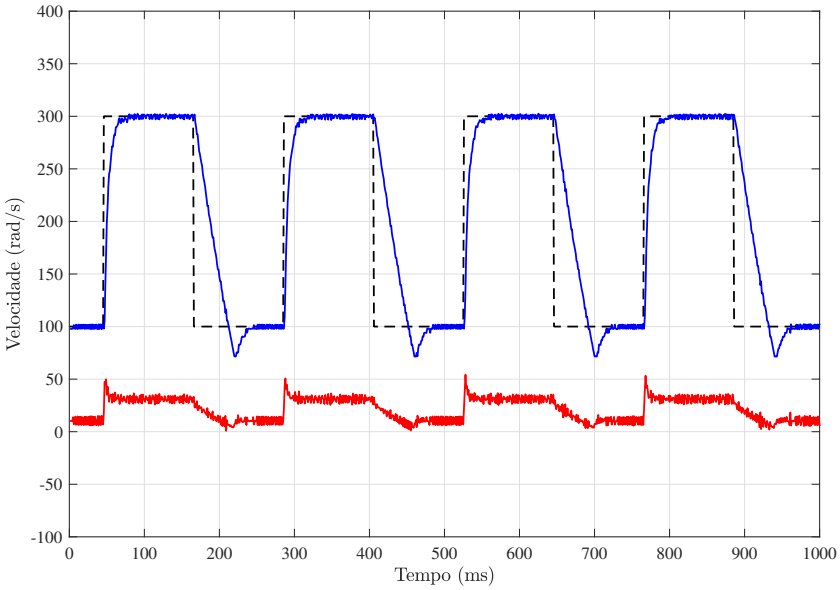

Figura 12 – Resposta de velocidade angular em malhar fechada (azul), considerando um sinal de entrada na forma de onda quadrada (tracejada em preto) e o erro virtual (em vermelho).

O sinal de entrada $u(t)$ aplicado procedente dos cálculos feitos no microcontrolador teve os valores mínimos e maximo definido a fim de que as leituras de velocidades fossem fidedignas. Uma vez que excitado acima do seu nível de saturação o motor não apresentaria diferença na velocidade.

3.8 Avaliação dos Controladores

Os resultados da utilização de dois controladores PI e PID são apresentados com o intuito de compará-los, sendo necessário pelo fato de que no método de referência virtual são feitas algumas suposições sobre o sistema a ser controlado, portanto ele pode ou não atender as exigências para o controle do sistema quando o controlador possuir um grau diferente da malha da planta. Temos a seguir a função de transferência do modelo de referência 3.36

$$T_z = \frac{0,09516}{z - 0,9048}. \tag{3.36}$$

A partir dos dados coletados em malha aberta, utilizando o modelo de referência (3.36) e adotando a estrutura de um controlador PI e um PID obteve-se, pelo uso da metodologia VRFT com a estimação por mínimos quadrados os seguintes parâmetros para o controlador PI temos -1.18 como ganhos proporcional, 1.55 o ganho integral.

Os valores obtidos para o controlador PID: 0.0017, 0.0002, 0.001. que representam os ganhos Kp, Ki, e Kd respectivamente. E que são as constantes proporcional, integral e derivativo são por consequência usados no controle de velocidade do motor. Temos portanto com esses valores a seguinte função de controle:

$$G_s = \frac{-0.0017s^2 + 0.0002s + 0.001}{s}. \tag{3.37}$$

4 RESULTADOS

O objetivo deste capítulo é apresentar os resultados obtidos quanto a aplicação dos controladores projetados.

A resposta temporal para o seguimento de uma referência na forma de onda quadrada é mostrada na Figura 13 usando um controlador PI e em seguida na Figura 14 com um controlador PID.

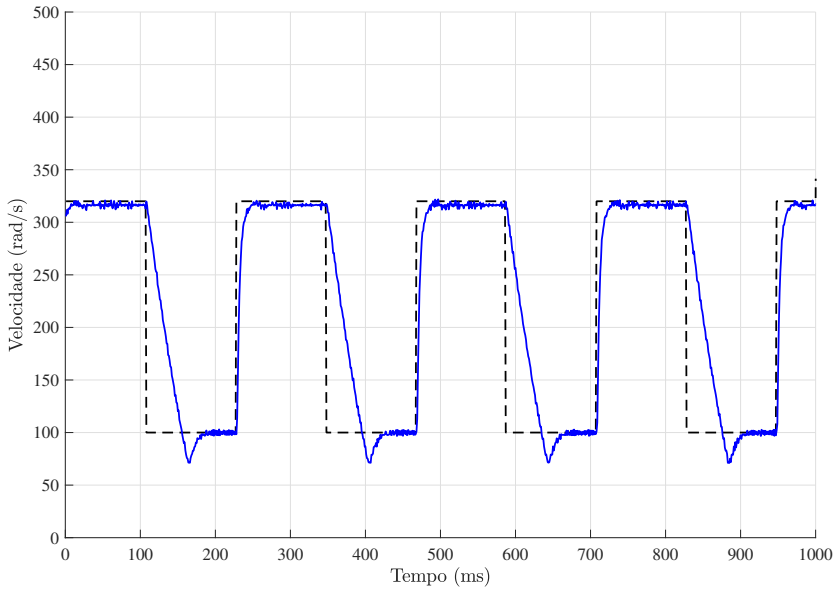

Figura 13 – Resposta temporal da velocidade angular em malha fechada (azul) seguindo uma referência de onda na forma quadrada (tracejada em preto).

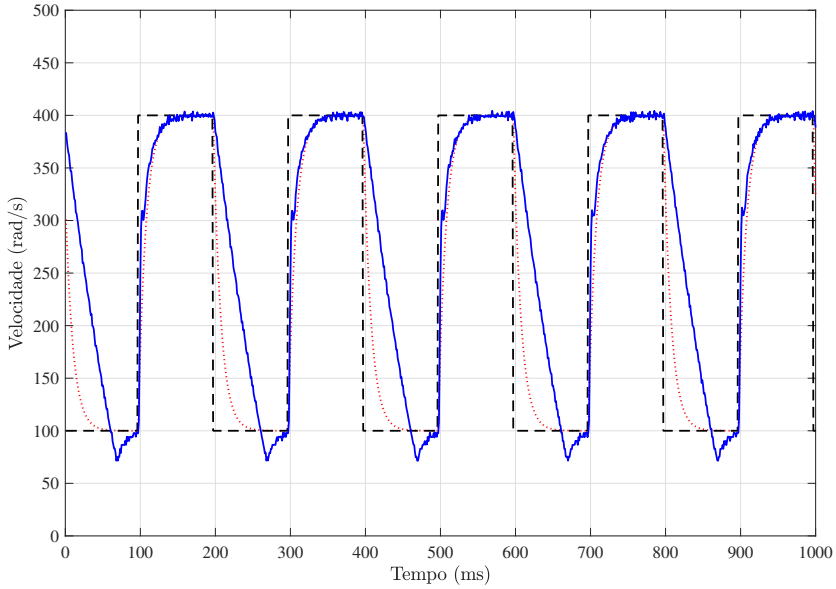

Figura 14 – Resposta temporal da velocidade angular em malha fechada (azul) seguindo uma referência de onda na forma quadrada (tracejada em preto), e o modelo de referência (pontilhado em vermelho).

Duas coisas são pertinentes de se observar nesse resultado quando uma carga é aplicada ao motor. Primeiro um erro considerável na fase desligada do ciclo, isso ocorre pois quando a chave semicondutora do circuito de acionamento está aberta, mesmo sobre o efeito da força contra eletromotriz gerada pelo motor, a corrente em seu induzido continua circulando no mesmo sentido por algum tempo devido ao circuito de roda livre criado pelo diodo em antiparalelo ao motor. Enquanto a corrente mantém seu sentido, um torque a favor do sentido de giro é produzido, de acordo com a lei de Lenz. Devido a perdas de energia em razão do efeito resistivo do enrolamento, correntes parasitas e histerese magnética, esta corrente cessa, o diodo passa a funcionar como uma chave aberta. Com isso, sem o torque, a desaceleração é mais rápida. Quando se acopla uma cara ao eixo do motor, o conjugado resistente aumenta, o que ajuda a vencer o conjugado (torque) provocado pela corrente remanescente no induzido, fazendo com que o rotor perca velocidade mais rapidamente como visto na 15 no momento em que o sinal passa da crista para o vale da onda. Segundo, o erro em regime permanente, esse se dá pelo fato de que o sinal de controle não pode passar de certo limite, estando no seu valor máximo de tensão o motor então não pode chegar a velocidade esperada.

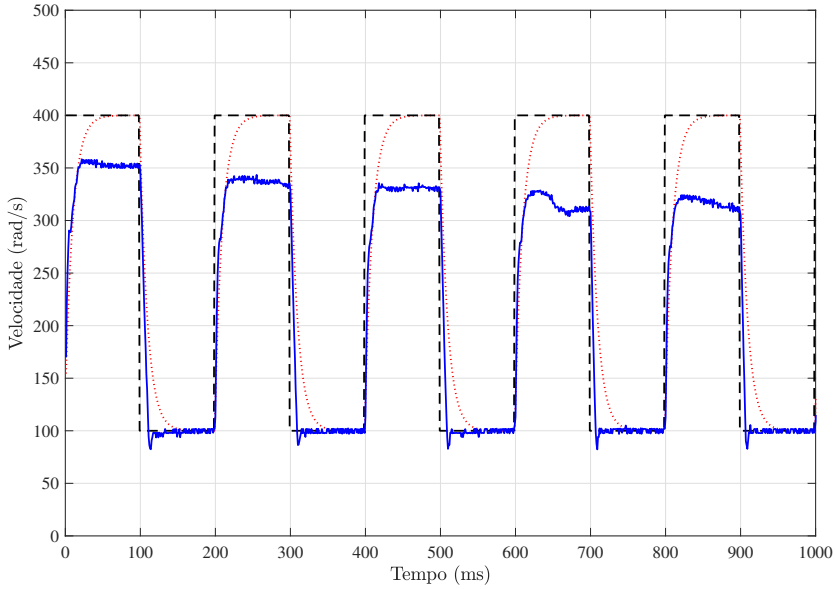

Figura 15 – Resposta temporal da velocidade angular em malha fechada (azul) agora com uma carga aplicada em seu eixo, e seguindo uma referência de onda na forma quadrada (tracejada em preto), e o modelo de referência (pontilhado em vermelho).

5 CONCLUSÕES E RECOMENDAÇÕES

O presente projeto teve o intento de projetar um controlador, e para tanto definir um modelo de referência, o qual, ao ser utilizado em métodos de controle embasados em modelo de referência resultaria em um desempenho satisfatório rejeitando perturbações e que se ajustasse à referência. A análise para obtenção do modelo foi fundamentada em trabalhos prévios, e usou-se mínimos quadrados como método matemático para estabelecer o mesmo.

Foi certificado que o controlador obtido cancela o polo do processo para a resposta à variação da referência quando se emprega um método usual na escolha do modelo de referência tendo em mente que a ordem do controlador seja igual ou maior que o da equação que rege do sinal controlado.

Com o auxílio da ferramenta de simulação Matlab/Simulink foi possível experimentar com pouco esforço diferentes formas de se obter o modelo de referência. O modelo proposto apresentou resultados satisfatórios, outorgando a proposta do trabalho. O método de referência virtual se mostrou promissor para aplicação adotada, não somente por sua facilidade de ser empregado como também pela sua enaltecida vantagem de não necessitar de um modelo do processo.

5.1 Recomendações para trabalhos futuros

Para a continuação do estudo de controladores pelo modelo de referencias sujere-se

-O desenvolvimento de um modelo de referência iterativo

-Adequar o modelo a processos de ordem maiores e estudar a resposta quanto a variação da referência e a rejeição a perturbação de maior ordem.

REFERÊNCIAS

BAZANELLA. *Data-driven controller design: the H2 approach*. [S.l.]: Springer, 2012. 300 p. ISBN 964-7445-88-1. Citado na página 18.

CAMPESTRINI, L. et al. Unbiased MIMO VRFT with application to process control. *Journal of Process Control*, 2016. Elsevier Ltd, v. 39, p. 35–49, 2016. ISSN 09591524. Disponível em: <http://dx.doi.org/10.1016/j.jprocont.2015.12.010>. Citado na página 9.

FUJISAKI, Y.; DUAN, Y.; IKEDA, M. System representation and optimal control in input-output data space. *IFAC Proceedings Volumes*, 2004. Elsevier, v. 37, n. 11, p. 185–190, 2004. Citado na página 16.

HJALMARSSON, H.; BIRKELAND, T. Iterative feedback tuning of linear time-invariant MIMO systems. *Proceedings of the IEEE Conference on Decision and Control*, 1998. v. 4, n. December, p. 3893–3898, 1998. ISSN 01912216. Citado na página 9.

HJALMARSSON, H.; GUNNARSSON, S.; GEVERS, M. A convergent iterative restricted complexity control design scheme. In: IEEE. *Proceedings of 1994 33rd IEEE Conference on Decision and Control*. [S.l.], 1994. v. 2, p. 1735–1740. Citado na página 15.

HOU, Z. The parameter identification, adaptive control and model free learning adaptive control for nonlinear systems. *Shenyang: North-eastern University*, 1994. 1994. Citado 2 vezes nas páginas 14 e 15.

HOU, Z. S.; WANG, Z. From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, 2013. Elsevier Inc., v. 235, p. 3–35, 2013. ISSN 00200255. Disponível em: <http://dx.doi.org/10.1016/j.ins.2012.07.014>. Citado 2 vezes nas páginas 9 e 13.

HUANG, B.; KADALI, R. *Dynamic modeling, predictive control and performance monitoring: a data-driven subspace approach*. [S.l.]: Springer, 2008. Citado na página 16.

KALMAN, R. E. A new approach to linear filtering and prediction problems. 1960. 1960. Citado na página 12.

KARIMI, A.; HEUSDEN, K. V.; BONVIN, D. Non-iterative data-driven controller tuning using the correlation approach. In: IEEE. *2007 European Control Conference (ECC)*. [S.l.], 2007. p. 5189–5195. Citado na página 16.

KARIMI, A.; MIŠKOVIĆ, L.; BONVIN, D. Convergence analysis of an iterative correlation-based controller tuning method. *IFAC Proceedings Volumes*, 2002. Elsevier, v. 35, n. 1, p. 413–418, 2002. Citado na página 15.

MARKOVSKY, I. et al. *Data driven simulation with applications to system identification*. IFAC, 2005. 970–975 p. ISSN 14746670. ISBN 008045108X. Disponível em: <http://dx.doi.org/10-.3182/20050703-6-CZ-1902.00163>. Citado na página 16.

NYQUIST, H. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 1928. IEEE, v. 47, n. 2, p. 617–644, 1928. Citado na página 11.

ROHRS, C. E. et al. Robustness of Adaptive Control Algorithms in the Presence of Unmodeled Dynamics. *Proceedings of the IEEE Conference on Decision and Control*, 1982. v. 1, n. September, p. 3–11, 1982. ISSN 01912216. Citado na página 13.

SAFONOV, M. G.; TSAO, T.-C. The unfalsified control concept: a direct path from experiment to controller. In: *Feedback Control, Nonlinear Systems, and Complexity*. [S.l.]: Springer, 1995. p. 196–214. Citado na página 15.

SOUZA, D. R.; SOUSA, D. J. Desbravando o pic18: Ensino didático. 2012. 2012. Citado 2 vezes nas páginas 27 e 28.

SPALL, J. C. Multivariate Stochastic Approximation Using a Simultaneous Perturbation Gradient Approximation. *IEEE Transactions on Automatic Control*, 1992. v. 37, n. 3, p. 332–341, 1992. ISSN 15582523. Citado na página 14.

THOMSEN, A. *O que é Arduino: conceito, benefícios e como utilizar - FilipeFlop*. 2020. Disponível em: <https://www.filipeflop.com/blog/o-que-e-arduino>. Citado na página 29.

WERBOS, P. J.; MILLER, W.; SUTTON, R. A menu of designs for reinforcement learning over time. *Neural networks for control*, 1990. MIT press Cambridge, MA, p. 67–95, 1990. Citado na página 16.

ZIEGLER, J. G.; NICHOLS, N. B. Optimum settings for automatic controllers. *InTech*, 1995. v. 42, n. 6, p. 94–100, 1995. ISSN 0192303X. Citado na página 9.