

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
CÂMPUS CORNÉLIO PROCÓPIO
DIRETORIA DE GRADUAÇÃO E EDUCAÇÃO PROFISSIONAL
DEPARTAMENTO ACADÊMICO DA ELÉTRICA
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

BRUNA FORTUNATO

**SIMULAÇÃO E CONTROLE VIA REDES DE SISTEMAS LINEARES
E INVARIANTES NO TEMPO**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO

2017

BRUNA FORTUNATO

**SIMULAÇÃO E CONTROLE VIA REDES DE SISTEMAS LINEARES
E INVARIANTES NO TEMPO**

Trabalho de Conclusão de Curso apresentado ao Departamento Acadêmico da Elétrica da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de Bacharel em Engenharia de Controle e Automação

Orientador: Cristiano Marcos Agulhari

CORNÉLIO PROCÓPIO

2017



Universidade Tecnológica Federal do Paraná
Campus Cornélio Procópio
Departamento Acadêmico de Elétrica
Curso de Engenharia de Controle e Automação



FOLHA DE APROVAÇÃO

Bruna Fortunato

Simulação e controle via redes de sistemas lineares e invariantes no tempo

Trabalho de conclusão de curso apresentado às 08:00hs do dia 09/06/2017 como requisito parcial para a obtenção do título de Engenheiro de Controle e Automação no programa de Graduação em Engenharia de Controle e Automação da Universidade Tecnológica Federal do Paraná. O candidato foi arguido pela Banca Avaliadora composta pelos professores abaixo assinados. Após deliberação, a Banca Avaliadora considerou o trabalho aprovado.

Prof(a). Dr(a). Cristiano Marcos Agulhari - Presidente (Orientador)

Prof(a). Dr(a). Luiz Francisco Sanches Buzachero - (Membro)

Prof(a). Dr(a). Kleber Romero Felizardo - (Membro)

Aos meus pais, pela união e fé.

AGRADECIMENTOS

Seria muito egoísmo resumir a minha trajetória acadêmica apenas em boas notas. Por isso, dedico este espaço para agradecer a todas as pessoas que estiveram em minha vida e me ensinaram a ser além de uma engenheira, uma pessoa melhor. Gostaria de agradecer aquele que me deu o dom da vida, do céu olhou para os meus sonhos e embora as circunstâncias fossem contrárias, suas promessas foram além do que eu sonhava. À Deus, minha eterna gratidão.

Agradeço aos meus pais, por todo apoio durante os momentos difíceis, por sempre e acima de tudo, acreditarem em meu potencial. Agradeço pelo esforço financeiro despendido para que eu pudesse realizar meu grande sonho.

Agradeço ao meu paciente orientador, Cristiano Marcos Agulhari, sempre reconhecido por mim como “Professor”, pelo exemplo em profissionalismo, caráter e ética. Agradeço a ele por toda a disponibilidade, por repetir várias vezes a mesma coisa, por me explicar fazendo gestos, desenhos, palavras e “quase virar de ponta cabeça” até eu dizer que havia entendido a matéria e ele ter certeza disso, as vezes até perguntando se falei “entendi” mentindo (risos). Agradeço a ele também por me ensinar a levar a vida menos a sério, pelas “piadas ruins” em momentos tensos, por toda compreensão, por me manter calma quando eu estava desesperada e por ter doado parte do conhecimento dele para mim, sempre acreditando em meu potencial como aluna.

Agradeço os meus amigos, Cleber, Keity, Manu e Lilian, por pacientemente ouvirem diversas vezes a frase “Eu não aguento mais!” e ainda assim, encontrarem diversos meios para me manter firme, cada um da sua maneira.

Agradeço ao grupo “Déboras” da primeira igreja presbiteriana de Cornélio Procópio, em especial à irmã Lúcia, por todo acolhimento e força.

Agradeço ao pessoal do Nuape da UTFPR, pelo apoio, em destaque as colaboradoras Tati, Ligia e a Milena.

Agradeço a família Bazílio por todo acolhimento e amparo aqui em Cornélio Procópio.

Agradeço aos meus colegas de classe e amigos da primeira turma de Engenharia de Controle e Automação da UTFPR, em especial, “a galera do “CIPECA”, ou “esquadrão suicida”, composto por: Thiago Honorato, Thí, Vini, Amanda, Marco e Pedro. Não menos especial

porém igualmente louco, Vagner, vulgo sequela, pelos cadernos emprestados e pelos grupos de estudo pré-prova em conjunto com a segunda turma de Engenharia de Controle e Automação.

Agradeço ao Professor Fabian Lara-Molina e a minha colega de classe Thamiris, pela parceria e oportunidade para realizar a iniciação em Robótica.

Agradeço ao Vinicius Sutério, o odioso mais adorável comigo, parceiro de estudos, de lanches, de passeios, embora ele sempre seja “críca” com a minha maneira de organizar as coisas (risos) e eu igualmente “críca” com ele nas demoras pelo “atendimento com o Cristiano”.

Agradeço ao pessoal do mestrado, Ítalo, Daniel, Layhon e Alex, pela boa companhia, por me auxiliarem sempre que possível, pelos cafés na copa do CIPECA regados a conhecimento inútil que geravam descontração em momentos de tensão e pela amizade que se estendeu para fora da universidade.

Agradeço a todos os professores da UTFPR que compartilharam parte do seu saber, ministrando suas disciplinas. Em especial, os professores que de alguma forma marcaram a minha trajetória acadêmica: Emerson, André Shiguemoto, Wagner Endo, Murilo Silva, Fábio Duran, Vallim, Luiz Francisco Buzachero e ao eterno candidato para a vaga de piadista no “A praça é nossa”, o professor migué, digo, Cristiano Marcos Agulhari (risos).

A todos os citados, o meu muito obrigada.

“Circunstâncias dizem que não dá, falam que você está tão perto de parar, coração valente, é assim que quero hoje te chamar...
Coração valente você não pode parar”

Anderson Freire

RESUMO

FORTUNATO, Bruna. Simulação e Controle via redes de sistemas lineares e invariantes no tempo. 60 f. Trabalho de Conclusão de Curso – Engenharia de Controle e Automação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2017.

O objetivo deste trabalho é implementar uma *interface* que simula um sistema de controle Linear e Invariante no Tempo (LIT) via rede. Para isso, será considerado um dos fatores degenerativos que ocorrem frequentemente em transmissões de pacotes de informações: os atrasos entre as trocas de informações. Assim, será proposto um ambiente interativo em que o usuário poderá inserir e simular um sistema de controle sujeito a um atraso invariante ao tempo, afim de verificar o comportamento do sistema de controle projetado quando inserido em um ambiente real com tais fatores degenerativos de comunicação. Esta *interface* será submetida a uma simulação de um sistema de controle de velocidade de um motor DC, proposto por Meshram e Kanojiya (2012). Em seguida, atrasos serão inseridos afim de verificar o desempenho do sistema de controle proposto, caso fosse inserido em um ambiente real. O objetivo principal desta *interface* é ser uma ferramenta facilitada e gratuitamente disponibilizada para testes dos alunos e pesquisadores da área de Engenharia de Controle e Automação.

Palavras-chave: Controle, Sistemas LIT, Redes, NCS

ABSTRACT

FORTUNATO, Bruna. Simulation and Control via networks of linear and time invariant systems. 60 f. Trabalho de Conclusão de Curso – Engenharia de Controle e Automação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2017.

The objective of this work is to implement an interface that simulates a Linear Time Invariant (LTI) system via a network. Thus, it will be considered one of the degenerative factors that happen frequently in information package transmissions: the delay between the information exchanges. Hence, it will be proposed an interactive environment wherein the user will be able to insert and simulate a control system subject to a time-invariant delay in order to verify the behavior of the developed control system when inserted in a real environment with such degenerative communication factors. This interface will be submitted to a simulation of a speed control system of a DC motor, proposed by Meshram and Kanojiya (2012). Then it will be inserted delays in order to verify the performance of the proposed control system as if it was inserted in a real environment. The main objective of this interface is to be an easy and freely available tool for experiments of students and researchers in the Automation and Control Engineering field.

Keywords: Control, LTI Systems, Networks, Network Control Systems, NCS.

LISTA DE FIGURAS

FIGURA 1	– Propriedade distributiva em um LIT	20
FIGURA 2	– Representação de um processo de digitalização	23
FIGURA 3	– Modelo implementado para aplicar o primeiro método de Ziegler e Nichols	28
FIGURA 4	– Curva de resposta para o primeiro método de Ziegler e Nichols	28
FIGURA 5	– Esquema de um Sistema de Controle após a sintonização por Ziegler e Nichols	29
FIGURA 6	– Fluxo de informações entre as camadas	30
FIGURA 7	– Fluxo de informações entre as camadas	31
FIGURA 8	– Modelo de um NCS	34
FIGURA 9	– Diagrama de blocos para o primeiro método de Ziegler e Nichols	39
FIGURA 10	– Curva de resposta para a sintonia do PID	40
FIGURA 11	– Representação da reta tangente e os parâmetros para sintonização do PID	40
FIGURA 12	– Resposta do PID por Ziegler e Nichols implementado em malha fechada no Simulink	41
FIGURA 13	– Resposta do PID implementado em malha fechada no Simulink	42
FIGURA 14	– Diagrama de Bode da planta	43
FIGURA 15	– Compensador digital inserido em um sistema de controle DC	44
FIGURA 16	– Resposta ao degrau em domínio discreto	45
FIGURA 17	– Modelo de um NCS encapsulado pela <i>interface</i>	46
FIGURA 18	– <i>Interface</i> projetada para simulação de um NCS	47
FIGURA 19	– Parâmetros ampliados da <i>Interface</i> projetada para simulação de um NCS	48
FIGURA 20	– Respostas ampliadas da <i>Interface</i> projetada para simulação de um NCS	49
FIGURA 21	– Instruções ampliadas da <i>Interface</i> projetada para simulação de um NCS	50
FIGURA 22	– Modelo de um NCS encapsulado pela <i>interface</i>	50
FIGURA 23	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 1ms	52
FIGURA 24	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 2ms	52
FIGURA 25	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 3ms	53
FIGURA 26	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 4ms	53
FIGURA 27	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 5ms	54
FIGURA 28	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 6ms	55
FIGURA 29	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 0.1s	55
FIGURA 30	– Modelo de um NCS encapsulado pela <i>interface</i> com atraso de 4s	56

LISTA DE SIGLAS

NCS	Networked Control System
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
OSI	Open Systems Interconnection
UTFPR	Universidade Tecnológica Federal
GUI	Graphical User Interface
LIT	Lineares e Invariantes no Tempo
BIBO	Bounded Input Bounded Output
A/D	Conversão Analógico/Digital
D/A	Digital Analógico
ZOH	zero-order-hold
A/D	contínuo/discreto
SISO	Single Input Single Output
DC	direct current
PID	Proporcional Integrativo e Derivativo

SUMÁRIO

1	INTRODUÇÃO	12
1.1	PROBLEMA ESTUDADO	15
1.2	JUSTIFICATIVA	15
1.3	OBJETIVOS	15
1.3.1	objetivos gerais	15
1.3.2	Objetivos Específicos	16
2	FUNDAMENTAÇÃO TEÓRICA	17
2.1	SINAIS E SISTEMAS DISCRETOS	17
2.2	O CONCEITO DE LINEARIDADE	18
2.2.1	Resposta de um sistema linear	19
2.3	SISTEMAS LINEARES E INVARIANTES NO TEMPO - LIT	19
2.4	SISTEMAS DIGITAIS	22
2.4.1	Amostragem em sistemas digitais	24
2.5	CONTROLE DE SISTEMAS EM MALHA ABERTA E MALHA FECHADA	26
2.5.1	Sintonização de Controladores PID por Ziegler e Nichols	27
2.6	PADRONIZAÇÃO DE REDE BASEADA NO MODELO OSI	29
2.6.1	A camada de enlace	31
2.6.2	A camada de transporte	32
2.7	PROTOCOLOS UDP	32
2.8	SISTEMAS DE CONTROLE VIA REDE - NCS	33
2.8.1	Atrasos em Sistemas de Controle via redes - NCS	35
3	MATERIAIS E MÉTODOS	37
3.1	REVISÃO BIBLIOGRÁFICA	37
3.2	FORMULAÇÃO E SIMULAÇÃO DO MODELO	37
4	RESULTADOS E DISCUSSÕES	39
4.1	PROJETO DO CONTROLADOR	39
4.2	PROJETO DA INTERFACE	45
4.3	VALIDAÇÃO DOS RESULTADOS	50
4.3.1	O sistema de controle de um motor DC com a inserção de atrasos invariantes no tempo	51
5	CONCLUSÃO	57
	REFERÊNCIAS	59

1 INTRODUÇÃO

Atualmente, fatores como a redução do custo, tempo e atendimento aos requisitos básicos de projeto têm sido um fator importante para sistemas de controle. Assim, em muitos casos, a realização de simulações via *software* têm sido uma importante alternativa, sendo estas voltas à obtenção de respostas aproximadas ao comportamento real de uma planta industrial, antecipando com um menor custo possíveis e tempo.

O processo para o desenvolvimento de um *software* deve seguir um padrão de desenvolvimento, ou seja, este processo deve possuir requisitos claros e bem definidos. Tais requisitos envolvem: o estudo do problema, entendendo claramente as reais necessidades do usuário; a busca por uma solução, que deve atender as necessidades específicas do utilizador com baixo custo e à implementação, que corresponde a implantação do projeto, de fato (BERNARDO; KON, 2008). Na implementação, busca-se a criação, alteração e refinamento dos modelos obtidos. Assim, opta-se pela realização de testes em ambiente virtual (FANTINATO et al., 2005). Neste cenário, é interessante estudar métodos que proporcionem ao utilizador um ambiente virtual facilitado e confiável para o estudo e a realização de testes.

Por meio da realização de testes em ambiente virtual, é possível obter uma série de vantagens, tais como: aumentar do ciclo de vida e qualidade do produto; reproduzir e conservar o sistema real, por meio da alteração de diversos parâmetros, em ambiente virtual; gerar diversas combinações de comandos e operações, para diferentes tipos de análises; e detectar falhas específicas, não identificadas na prática, para posterior estudo (BERNARDO; KON, 2008). Assim, o ambiente acadêmico e industrial são os principais utilizadores desta metodologia (FANTINATO et al., 2005).

No ambiente acadêmico, os testes são voltados à simulação, sendo possível o auxílio de uma plataforma didática, que aproxima o aluno a problemas reais voltados à engenharia, desde que, para isso, seja elaborada uma estratégia didático-pedagógica por parte do professor, favorecendo a transmissão e o desenvolvimento de conceitos relacionados à área de formação (CARVALHO; BARONE, 2009).

Na indústria, os testes são realizados de maneira automatizada, ou seja, voltados à verificação de um processo, sendo aplicados durante a fase de projeto e implementação de novos sistemas, *interfaces* ou formas de controle (FANTINATO et al., 2005). Nesse sentido, busca-se aproximar o comportamento real do objeto estudado, com base na análise e compreensão dos parâmetros que o compõem (FANTINATO et al., 2005).

O sistema de manufatura atual, em geral, é composto por sensores, atuadores e controladores, fisicamente distribuídos, que se comunicam via canais (ou barramentos) de rede, formando uma malha de controle remota (GODOY et al., 2008). A rede é responsável por receber e transmitir pacotes de mensagens amostradas pelos sensores, bem como enviar sinais dos controladores aos atuadores da planta (GODOY et al., 2008). Este tipo de implementação em sistemas industriais, fechados sob uma rede de comunicação, é denominado Sistema de Controle via Redes (NCS) (GODOY et al., 2008).

Como o NCS é um modelo para controle, a estabilidade e desempenho são fatores importantes a serem levados em consideração. Desta forma, durante o seu projeto, a transmissão das informações deve ser eficiente e otimizada e para isso, os impactos que atrasos de comunicação exercem sobre a estabilidade dos sistemas de controle devem ser considerados. Assim, para uma transmissão ser eficiente e otimizada, o projeto de um NCS deve levar em consideração o compartilhamento das mensagens entre os dispositivos conectados, que devem atender aos requisitos temporais da rede (GODOY et al., 2008). Logo, houve a necessidade em se criar padrões arquiteturais, contendo camadas de comunicação com funções bem definidas e um protocolo padrão para a comunicação entre as camadas que compõem um NCS. Dentre os protocolos ressaltam-se os denominados TCP e o UDP (GODOY et al., 2008).

A escolha entre os protocolos TCP ou UDP varia de acordo com a aplicação da rede. Basicamente, o protocolo TCP é viável quando torna-se necessário confirmar a chegada e a integridade de cada mensagem transmitida via rede (TORRES, 2015). Porém, este protocolo, devido à sua complexidade referente à implementação de técnicas que visam implementar a confirmação de cada mensagem transmitida, torna-se lento e inviável em aplicações que manipulam poucos dados (TORRES, 2015). Por isso, torna-se necessário a utilização de um protocolo mais simples, sendo frequente o uso do protocolo UDP (TORRES, 2015).

Sob os impactos que o NCS pode causar em sistemas de controle, destacam-se os atrasos de comunicação (*Delays*), bem como a discretização realizada pelos processos digitais. Os atrasos são gerados devido à quantidade de componentes ligados a rede que necessitam enviar/receber informações em tempo real, ocasionando uma saturação da rede (TANENBAUM, 2003). A discretização está relacionada à proximidade da informação recebida/transmitida pela

rede e gerada por um dispositivo que atua com um sinal analógico. Assim, ambos devem ser considerados, afim de garantir que seus efeitos não prejudiquem o desempenho do sistema (GODOY et al., 2008).

Neste Trabalho de Conclusão de Curso (TCC), o objetivo é implementar uma *interface* facilitada de rede de comunicação serial integrada, que leva em consideração, além da velocidade e eficiência na transmissão de pacotes de dados, os atrasos presentes na dinâmica de um sistema de controle via rede (NCS). O padrão arquitetural do NCS apresentado é baseado no modelo de sete camadas OSI, onde o foco são as camadas de transporte e enlace, sendo esta última responsável por captar ou enviar informações para componentes físicos em uma sistema. Como a comunicação entre os sensores e os atuadores é realizada via rede, torna-se necessário definir um protocolo padrão universal para comunicação entre os componentes do sistema. Neste trabalho, será utilizado o protocolo de transmissão UDP. Esta rede foi programada pelo aluno de Engenharia da Computação, Vinicius Henrique Moraes de Oliveira, estudante da UTFPR de Cornélio Procópio.

Esta *interface* foi criada para que o usuário possa inserir atrasos de rede durante a simulação, podendo assim, prever o comportamento do sistema o mais próximo possível de um ambiente real. Além disso, o usuário poderá definir qualquer valor referente ao período de amostragem da rede. Este é um fator relevante para o usuário, pois, pode-se definir a capacidade de refinamento dos dados da rede do NCS. Portanto, é possível avaliar qualitativamente a capacidade de refinamento da rede, na transmissão de pacotes de informações durante a comunicação entre os dispositivos conectados em um NCS e a sua influência sobre a precisão na resposta de controle desejada.

Todo o ambiente virtual gráfico será realizado via *software* Matlab/Simulink®. Para a interação entre o usuário e os dispositivos digitais, através de elementos gráficos, utilizou-se a *Toolbox* GUI, desenvolvida especificamente para o processamento de imagens. Esta *interface* irá permitir que o usuário defina os parâmetros da planta, controlador e rede, sem que para isso seja necessário qualquer conhecimento prévio de modelagem de sistemas de controles invariantes no tempo via *software* Matlab/Simulink®.

Por meio da *interface* desenvolvida, será possível a realização de testes automatizados em pesquisas, desenvolvimento e aperfeiçoamento de produtos relacionados à área de Engenharia de Controle e Automação, sem que para isso seja necessário a aquisição e custeamento de uma plataforma para simulações, ou ainda a implementação e obtenção de uma resposta indesejada de um sistema de controle já projetado, quando inserido em um ambiente real que apresente falhas de comunicação via rede, o que ocasionaria em um prejuízo para o projetista.

Estando ciente da relevância do objeto de estudo, buscou-se realizar um levantamento teórico, realizar testes preliminares para compreensão da dinâmica de rede e os seus fatores deteriorantes. A organização deste documento se dá na seguinte forma: primeiramente, serão apresentadas as situações que motivaram esta proposta de estudo, bem como os objetivos da pesquisa. O restante deste documento está organizado conforme descrito a seguir. Na seção 2, apresenta-se uma breve fundamentação teórica. Na seção 3, é apresentada a metodologia utilizada para o controle. Os resultados obtidos, referentes à influência da rede, são apresentados na seção 4. O cronograma de atividades do TCC é mostrada na seção 5.

1.1 PROBLEMA ESTUDADO

O presente trabalho apresenta uma *interface* alternativa para a simulação de sistemas LIT em um NCS. Assim, será possível obter respostas aproximadas a um ambiente real, com baixo custo, fácil manuseio e sem a necessidade em se conhecer a linguagem de programação para a construção de diagramas de blocos via *software* Matlab/Simulink[®].

1.2 JUSTIFICATIVA

A contribuição deste trabalho se insere no desenvolvimento de uma *interface* para controle via redes, com o auxílio do Matlab/Simulink[®] e da *toolbox* GUI, para que seja facilitada a interação do usuário na simulação de sistemas de controle invariantes no tempo via redes NCS. O desempenho do NCS modelado deve se assemelhar a uma rede de comunicação inserida em um ambiente real. Para isto, serão adotados os modelos clássicos de rede OSI, com comunicação via protocolos UDP e simulações de atrasos relacionados a informações provenientes de sensores e atuadores.

1.3 OBJETIVOS

1.3.1 OBJETIVOS GERAIS

O objetivo geral deste trabalho é desenvolver uma *interface* para simulação e controle via redes baseada em sistemas lineares e invariantes no tempo (LIT). Esta *interface* será aplicada em uma comunicação entre um controlador PID e um motor DC já desenvolvido, que propõe o controle da velocidade angular a partir do torque aplicado. Porém, a *interface* apresentada poderá ser utilizada futuramente para pesquisas envolvendo alunos do curso de Engenharia de Controle e Automação. Este produto propõe uma simulação facilitada que proporcione o

entendimento de sistemas LIT que trafegam informações via redes NCS.

1.3.2 OBJETIVOS ESPECÍFICOS

- Entender a comunicação de um NCS, aplicada via *Simulink/Matlab*®;
- Entender a forma de transmissão e recebimento de cada pacote de dados entre cada nó do NCS;
- Aplicar um controlador e uma planta, cujas respostas já são conhecidas, afim de verificar a eficiência da comunicação entre os dispositivos, conectados entre os nós do NCS;
- Variar e simular atrasos em um sistema de controle LIT;
- Realizar o levantamento dos requisitos necessários para um usuário simular um NCS via rede, em uma *interface*;
- Projetar a *interface* via *toolbox* GUI e realizar testes de simulação para verificar se a transmissão ocorre de acordo com o modelo adotado via *Simulink/Matlab*®;
- Analisar o desempenho de um sistema de controle LIT mediante simulações de atrasos e discretizações computacionais no *interface* criada, com o objetivo de estudar o desempenho desta quando implementado em um ambiente real.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 SINAIS E SISTEMAS DISCRETOS

Nos dias atuais, com a introdução de tecnologias e dispositivos para a comunicação entre o homem-máquina, nota-se um estudo crescente envolvendo a otimização e eficiência das informações trocadas. Tais informações são representadas por sinais.

Os sinais dependem do tipo de tecnologia empregado em um equipamento e podem ser classificados em duas naturezas: contínuos, com variações em sua amplitude pelo tempo ou discretos, onde são representados, basicamente, por valores obtidos em um intervalo tempo limitado ou discretos, em que um sinal é especificado apenas para valores discretos do tempo (LATHI et al., 2005). Tais sinais podem ser representados por uma variável n , que assume valores inteiros e $x[n]$ que representa o n -ésimo termo da sequência de x (LATHI et al., 2005).

Um sinal de tempo contínuo, $x(t)$, pode se tornar discreto. Para isso, o sinal analógico deve ser obtido a partir de um determinado número de amostragens e pode ser expresso por $x(nT)$, onde T representa o período de amostragem do sinal contínuo (sendo o seu inverso a frequência de amostragem) (OPPENHEIM, 1999). Não se considera que $x(nT)$ não possui valores para n não inteiros, mas sim, que o valor de $x(nT)$ é indefinido para n não inteiro (OPPENHEIM, 1999).

Uma das operações realizadas em sinais discretos é a do deslocamento. Matematicamente, esta operação consiste em um sinal, representado por $x[n] = x(nT)$, que é deslocado por meio de amostras representadas por M . Esta operação irá resultar em um novo sinal $x_s[n]$, conforme a Eq.(1).

$$x_s[n] = x[n - M] \quad (1)$$

Assim, se o valor de deslocamento se $M \geq 0$, o sinal $x[n]$ é deslocado para a direita, caracterizando um atraso; caso $M \leq 0$, o deslocamento é para a esquerda, ocasionando um avanço (LATHI et al., 2005). Assim, $x_s[n]$ pode resultar em uma sequência de sinais em avanços

ou atrasos.

Um sistema em tempo discreto é uma sequência que processa uma entrada de números $x[n]$ resultando em outra sequência de números como saída, $y[n]$ (LATHI et al., 2005). Porém, poderá haver mais de uma entrada e saída em um sistema discreto. Assim, o número de entradas ou saídas produzidas em um sinal afeta diretamente a sua estabilidade. A estabilidade em um sistema discreto é um fator importante a ser considerado.

2.2 O CONCEITO DE LINEARIDADE

Dentre as categorias para classificações existentes em sistemas, existe a categoria dos sistemas Lineares e Invariantes no Tempo, ou LIT. Um sistema é linear se sua resposta for proporcional à perturbação aplicada (LATHI et al., 2005). Assim, se em um sistema LIT forem aplicadas as entradas x_1 e x_2 , respectivamente, suas saídas y_1 e y_2 poderão ser estudadas separadamente, conforme a Eq.(2) (LATHI et al., 2005). Esta Propriedade é denominada aditividade.

$$x_1 + x_2 \rightarrow y_1 + y_2 \quad (2)$$

Um sistema LIT também deve satisfazer a condição de homogeneidade ou propriedade escalar, onde estados com valores k , reais ou imaginários, deve possuir a mesma amplitude caso a entrada seja incrementada em um novo valor k . Assim, a saída do sistema também representará o valor k incrementado, conforme a Eq.(3) (LATHI et al., 2005).

$$kx \rightarrow ky \quad (3)$$

Ambas as propriedades, aditividade e homogeneidade, podem ser combinadas em uma única propriedade: O princípio da superposição, expressado na Eq.(4). A aditividade implica na homogeneidade, porém, a homogeneidade nem sempre implica na aditividade (LATHI et al., 2005).

$$k_1x_1 + k_2x_2 \rightarrow k_1y_1 + k_2y_2 \quad (4)$$

Sistemas LIT possuem parâmetros invariantes no tempo (ou parâmetros constantes). Para um dado sistema LIT, se uma entrada é atrasada em T segundos, sua saída também será atrasada pelo mesmo tempo, mantendo seu comportamento (LATHI et al., 2005).

Redes compostas por elementos resistivos, indutivos e capacitivos (RLC) são sistemas

lineares e invariantes no tempo (LATHI et al., 2005). Estes sistemas podem ser escritos em equações diferenciais lineares com coeficientes constantes. Se esses coeficientes forem variantes no tempo, o sistema é dito variante no tempo.

2.2.1 RESPOSTA DE UM SISTEMA LINEAR

A estabilidade de um sistema está relacionada à resposta total dele, ou seja, a soma da resposta natural (ligada ao transitório do sistema) e da resposta forçada, conforme a Eq.(5) (NISE; SILVA, 2002).

$$\textit{Resposta total} = \textit{Resposta natural} + \textit{Resposta forçada} \quad (5)$$

A resposta natural depende apenas do sistema e descreve a forma como o sistema dissipa ou obtém a energia, quando excitado (NISE; SILVA, 2002). Já a resposta forçada, por outro lado, é dependente do sistema (NISE; SILVA, 2002).

Para o controle, é interessante que a resposta natural do sistema tenda a zero ou apenas oscile (NISE; SILVA, 2002). Quando a resposta natural do sistema tende a ser maior que a resposta forçada, o sistema tende a atingir a condição de instabilidade e pode se tornar autodestrutivo, caso os limitadores não façam parte do projeto (NISE; SILVA, 2002). Caso o sistema seja estável, suas respostas transitórias e erro em regime permanente podem ser projetadas.

2.3 SISTEMAS LINEARES E INVARIANTES NO TEMPO - LIT

A resposta de um sistema linear, $y[n]$, será o resultado da superposição de impulsos deslocados (OPPENHEIM, 2010). Além disso, a propriedade de invariância no tempo garante que as respostas de um sistema submetidos aos impulsos deslocados no tempo são apenas o resultado de versões deslocadas no tempo dessas respostas (OPPENHEIM, 2010). Assim, a representação por soma de convolução em um sistema LIT é resultado na união dos conceitos de respostas de sistemas invariantes e lineares, sendo estas a base para a definição de diversas propriedades específicas para um sistema LIT (OPPENHEIM, 1999).

Considerando a convolução de sistemas discretos como sendo a soma das respostas a uma sequência de impulsos, $h[n - k]$, onde $x[k]$ são os pesos de uma combinação linear, tem-se que a saída $y[n]$ pode ser calculada como mostrado na Eq.(6) (OPPENHEIM, 2010).

$$y[n] = \sum_{k=-\infty}^{+\infty} x[k]h[n-k] = x[n] * h[n] \quad (6)$$

A propriedade distributiva está ligada a interconexão de sistemas. Considere dois sistemas, conforme as Eq.(7-8).

$$y_1[n] = x[n] * h_1[k] \quad (7)$$

$$y_2[n] = x[n] * h_2[n] \quad (8)$$

Sendo h_1 e h_2 suas respostas impulsivas respectivamente e $x[n]$ entradas idênticas, suas saídas, $y_1[n]$ e $y_2[n]$, podem ser adicionadas em série ou em paralelo, pois produzirão respostas idênticas, conforme as Eq.(9-10) (OPPENHEIM, 2010).

$$y[n] = x[n] * h_1[n] + x[n] * h_2[n] \Rightarrow \quad (9)$$

$$\Rightarrow y[n] = x[n] * (h_1[n] + h_2[n]) \quad (10)$$

A Fig.1 ilustra em diagrama de blocos os efeitos dessa propriedade (OPPENHEIM, 1999).

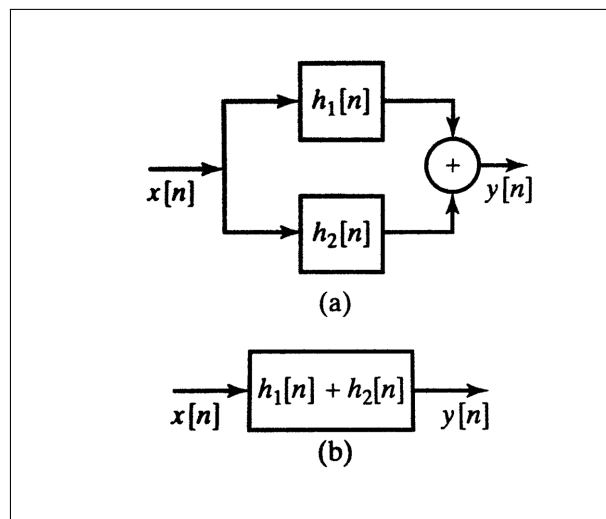


Figura 1: Propriedade distributiva em um LIT
Fonte: Oppenheim (1999)

A propriedade distributiva é resultado das propriedades de combinação linear e comutatividade da convolução (OPPENHEIM, 1999). Por sua vez, propriedade associativa é ilustrada

conforme a Eq.(12) (OPPENHEIM, 2010).

$$x[n] * (h_1[n] * h_2[n]) = (x[n] * h_1[n]) * h_2[n] \quad (12)$$

Assim, de acordo com a Eq.(12), duas respostas ao impulso, $h_1[n]$ e $h_2[n]$ podem ser convoluídas juntas ou separadamente a uma mesma entrada $x[n]$, que produzirão a mesma saída (OPPENHEIM, 2010).

Os sistemas LIT também podem ser com memória ou sem memória. Quando o sistema LIT é sem memória, sua saída em qualquer instante depende apenas do valor de entrada naquele instante atual (OPPENHEIM, 2010). Caso o sistema LIT possua resposta ao impulso, $h[n] \neq 0$, o sistema possuirá memória, dependendo de valores passados ou futuros (OPPENHEIM, 2010).

Com relação à estabilidade de um LIT, este é denominado BIBO-estável se, quando submetido a entradas limitadas, sempre que produzir à saídas limitadas (OPPENHEIM, 2010). Assim, um sistema estável LIT deve possuir resposta ao impulso somável, sendo que cada resposta deve possuir um valor absoluto, B_h , conforme a Eq.(13) (OPPENHEIM, 1999).

$$B_h = \sum_{k=-\infty}^{+\infty} |h[k]| < \infty \quad (13)$$

Em geral, um sistema com resposta ao impulso com duração finita é estável os valores em sua magnitude forem finitos (OPPENHEIM, 1999). Caso a duração seja infinita, o sistema será instável como ocorre no sistema acumulador. Dada uma entrada ao degrau, $u[n]$, a soma dos seus valores em amplitude tendem ao infinito, devido à duração infinita de sua resposta ao impulso (OPPENHEIM, 1999). Este é um exemplo de respostas com amplitude finitas porém instáveis.

Para analisar a BIBO estabilidade, ou seja, a produção de entradas limitadas por saídas limitadas em um sistema linear e invariante no tempo, com função de transferência $G(z)$, deve-se observar o módulo do seus polos possuem módulo menor que 1, ou seja, se estão estritamente dentro do círculo unitário (CASTRUCCI, 2006).

O conceito de convolução é útil em sistemas LIT com atrasos ideais. Considerando um sistema como sendo composto por um atraso ideal, $y[n] = x[n - n_d]$, sendo n_d o valor de deslocamento do atraso, sua resposta impulsiva será $h[n] = \delta[n - n_d]$ e sua relação de acordo com a convolução é dada pela Eq.(14) (OPPENHEIM, 1999).

$$x[n] * \delta[n - n_d] = \delta[n - n_d] * x[n] = x[n - n_d] \quad (14)$$

Assim, de acordo com a Eq.(14), a convolução de um impulso deslocado com qualquer sinal $x[n]$ resulta no com o deslocamento de $x[n]$ no valor onde o impulso ocorre.

Como o atraso é uma implementação recorrente em sistemas lineares, a Eq.(14) é uma ferramenta para análise muito útil, pois simplifica a análise de associações em sistemas LIT (OPPENHEIM, 1999). Considerando um sistema com diferenças progressivas em série com atrasos de uma amostra, de acordo com a propriedade comutativa da convolução, a ordem de associação entre os sistemas não importa, desde que eles sejam LIT (OPPENHEIM, 1999) conforme as Eq.(15-16).

$$h[n] = (\delta[n+1] - \delta[n]) * (\delta[n-1]) \quad (15)$$

$$h[n] = \delta[n-1] * (\delta[n+1] - \delta[n]) \quad (16)$$

Assim, é possível obter o mesmo resultado tanto atrasando o sistema e calculando suas diferenças progressivas quanto calculando suas diferenças progressivas e o atrasando conforme o exemplo da Eq.(15-17) (OPPENHEIM, 1999).

$$h[n] = \delta[n] - \delta[n-1] \quad (17)$$

Além disso, o resultado entre a convolução de uma amostra em atraso com um sinal que possua uma diferença progressiva, produz como resposta um sistema de diferenças regressivas, conforme demonstrado na Eq.(17) (OPPENHEIM, 1999).

2.4 SISTEMAS DIGITAIS

Os controladores contínuos podem ser construídos utilizando dispositivos analógicos, tais como resistores, capacitores e amplificadores operacionais. Contudo, a grande maioria dos sistemas utilizados atualmente fazem uso de um computador digital,, microprocessadores ou microcontroladores. Desta forma, tem se tornado cada vez mais comum a utilização de sistemas híbridos, capazes de realizar operações com sinais contínuos ou discretos (CASTRUCCI, 2006). Os computadores digitais trabalham com *hardware* específicos para entradas e saídas que implementam controladores. A operação de digitalização é ilustrada na Fig.2.

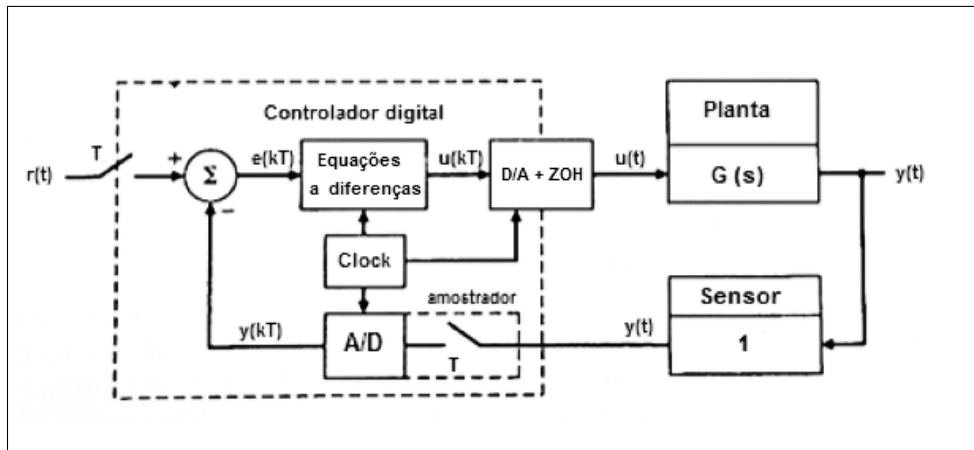


Figura 2: Representação de um processo de digitalização
Fonte: Franklin et al. (1990)

No controlador ocorre um cálculo computacional, onde se relacionam um sinal de erro $e(kT)$ e uma compensação dinâmica (FRANKLIN et al., 1990). Esta compensação dinâmica é realizada por um controlador digital, que faz uso de um atuador para potencializar a resposta (CASTRUCCI, 2006). A diferença entre uma representação contínua e uma representação digital é que a última opera com amostras de sinais de saída da planta, ao invés de trabalhar com os sinais de saída puramente contínuos. Além disso, a dinâmica de um controlador, é realizada via equações a diferenças (FRANKLIN et al., 1990).

A primeira ação para converter um sinal analógico em digital é chamada conversão Analógico Digital, ou A/D. Quando um sinal entra por esta primeira ação, é realizada uma conversão de um sinal analógico $y(t)$ em um valor digital (FRANKLIN et al., 1990). A conversão de um sinal analógico $y(t)$ ocorre repetitivamente a cada T intervalos de segundos. Desta forma, T representa o período de amostragem e sua inversa é a frequência de amostragem, dada em ciclos por segundo ou Hz. O sinal amostrado depende do período de amostragem e portanto é representado por $y(kT)$ (FRANKLIN et al., 1990).

Também existe um amostrador e conversor A/D para comandos de entrada $r(t)$ ou sinal de referência, produzindo seu equivalente discreto $r(kT)$, que é comparado com $y(kT)$ para ser subtraído, gerando um sinal discreto de erro $e(kT)$ (FRANKLIN et al., 1990).

O resultado da equação de diferenças é um sinal discreto $u(kT)$ amostrado a cada instante de tempo. Este sinal é convertido a um sinal contínuo, por um conversor D/A (CASTRUCCI, 2006). Esta conversão ocorre por meio de um segurador de ordem zero, que aproxima a função $u(t)$ em um cada intervalo por um valor constante (CASTRUCCI, 2006).

O conversor D/A calcula um número binário para ser um sinal contínuo, como uma tensão analógica (CASTRUCCI, 2006). Normalmente também se utiliza um segurador de or-

dem zero ZOH para manter um valor obtido até que ocorra o próximo período de amostragem. O sinal resultante, $u(t)$ é então aplicado para o atuador como se fosse um sinal puramente contínuo (FRANKLIN et al., 1990).

Para um computador realizar o cálculo de equações diferenciais em tempo real utiliza-se o método de Euler, representado matematicamente pela Eq.(18), desde que $\delta t \neq 0$.

$$\dot{x} = \lim_{\delta t \rightarrow 0} \frac{\delta x}{\delta t} \quad (18)$$

Para o caso de sistemas discretos, utilizados em sistemas digitais a aproximação da derivada \dot{x} dada pela Eq.(18) é associada a aproximações retangulares (FRANKLIN et al., 1990). Assim, considerando uma aproximação retangular progressiva, o valor aproximado da dinâmica é relacionada como a Eq.(19).

$$\dot{x}(tk) \cong \frac{x(k+1) - x(k)}{T} \quad (19)$$

Onde:

- T corresponde ao intervalo em segundos dado por $T = tk + 1 - tk$;
- O instante de tempo atual é $t_k = kT$;
- $x(k)$ é o valor de x em t_k ;
- $x(k+1)$ é o valor de x em t_{k+1} .

2.4.1 AMOSTRAGEM EM SISTEMAS DIGITAIS

Uma sequência de amostras $x[n]$ é obtida a partir de um sinal de tempo contínuo, $x_c(t)$ de acordo com a relação na Eq.(20). Esta relação é válida para $-\infty < n < +\infty$ (OPPENHEIM, 1999).

$$x[n] = x_c(nT) \quad (20)$$

A partir da Eq.(20), T é o período de amostragem e sua inversa, $f_s = 1/T$ é denominada frequência de amostragem em amostras por segundo, normalmente expressa em radianos por segundo, sendo portanto dada por $\omega_s = 2\pi/T$, sendo a sua unidade expressa em radianos por segundo (OPPENHEIM, 1999).

O processo de amostragem consiste em aplicar um modulador de trem de impulsos ou $s(t)$ ponderados em uma sequência de tempo discreto $x_c(t)$. Assim, uma sequência amostrada $x_s(t)$ pode ser expressa de acordo com a Eq.(21) (OPPENHEIM, 2010).

$$x_s(t) = x_c(t) \cdot s(t) = \sum_{n=-\infty}^{\infty} x_c(nT) \delta(t - nT) \quad (21)$$

Onde o modulador de trem de impulsos é, $s(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$.

Para obter a relação no domínio da frequência de um sinal amostrado em um conversor A/D ideal, considera-se um sinal $x_s(t)$ com aplicação da transformada de Fourier na Eq.(21). Portanto, o resultado final corresponde à convolução entre x_c e a transformada de um trem de impulso $s(t)$ periódicos no domínio da frequência de acordo com a Eq.(23) (OPPENHEIM, 1999).

$$X_s(j\omega) = \frac{1}{2\pi} X_c(j\omega) * S(j\omega) \quad (23)$$

Desta forma, para sinais com variáveis discretas e considerando $S(j\omega)$ um trem de impulsos, $X_s(j\omega)$ é dado pela Eq.(24).

$$X_s(j\omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c((j\omega) - k(j\omega_s)) \quad (24)$$

Assim, de acordo com a Eq.(24), nota-se que $x_s(t)$ consiste em réplicas repetidas periodicamente da transformada de Fourier deslocadas no tempo, isto é, $X_c(j\omega)$ (OPPENHEIM, 1999). Em outras palavras, tais réplicas são deslocadas por um número inteiro e sobrepostas para representar uma transformada periódica relacionada a um trem de impulsos (OPPENHEIM, 1999). Conclui-se, portanto, que a periodicidade é um fator a ser considerado para a amostragem e conversão de um sinal contínuo em discreto.

Um efeito gerado pela alteração na taxa de amostragem é chamado de *aliasing*. O *aliasing* ocorre se forem usadas baixas taxas de amostragem em relação à frequência do sinal contínuo (OPPENHEIM, 1999). Assim, para evitar esse efeito de distorção na reconstrução de sinais, a taxa de amostragem do modulador de trem de impulsos, ω_s , deve ser maior ou igual ao dobro do valor da maior componente de frequência do sinal contínuo, ω_N , conforme a relação na Eq.(25). Caso contrário, o sinal será distorcido por *aliasing*.

$$\omega_s \geq 2\omega_N \quad (25)$$

Assim, caso a frequência de amostragem do sinal obedeça a Eq.(25), um filtro de passa baixas ideal poderá reconstruir o sinal discreto para seu equivalente contínuo (OPPENHEIM, 1999).

2.5 CONTROLE DE SISTEMAS EM MALHA ABERTA E MALHA FECHADA

Existem basicamente dois tipos de sistemas para controle: sistemas de controle de malha aberta e sistemas de controle de malha fechada.

No sistema de controle de malha aberta, o sinal de saída não é medido nem realimentado e portanto, caso não seja aplicado um projeto de controle adequado, este tipo de sistema pode apresentar respostas indesejáveis. Assim, cada entrada de referência corresponde uma condição fixa de operação (OGATA, 2010). Caso ocorram perturbações, estas são acrescentadas após a atuação do controlador (NISE; SILVA, 2002). Assim, sistemas em malha aberta são caracterizados por serem sensíveis a perturbações e não eficazes para a correção após seus efeitos (NISE; SILVA, 2002).

No sistema de controle de malha fechada, o sinal do erro atuante, que é a diferença entre o sinal de referência e sua realimentação (que pode ser o próprio sinal de saída ou uma função do sinal de saída e suas derivadas/integrais), realimenta um controlador que tem por objetivo minimizar o erro e acertar a saída para que esta convirja a um valor desejado (OGATA, 2010).

A grande vantagem do sistema de controle de malha fechada é o fato de que o controle é menos sensível a perturbações externas ou variações internas nos parâmetros do sistema, devido à realimentação, que basicamente mede a diferença entre o sinal de referência e o comportamento do sistema (OGATA, 2010). Caso haja uma diferença entre os sinais medidos, o sistema aciona a planta para realizar uma ação de controle (NISE; SILVA, 2002). Caso contrário, a planta não será acionada, pois a diferença entre os sinais medidos não existirá (NISE; SILVA, 2002).

O uso de componentes relativamente imprecisos e baratos para obtenção de um sistema de controle, pode ser considerado em sistemas de malha fechada (OGATA, 2010). Porém, a estabilidade deste tipo de sistema é um fator a ser considerado, ao passo de que em sistemas de malha aberta, ela já é garantida, devido à sua precisão (OGATA, 2010).

Na teoria de controle, as funções de transferência são comumente usadas para definir a relação de entrada e de saída de componentes ou um sistema que pode ser modelado através de equações diferenciais lineares e invariantes no tempo (OGATA, 2010).

Uma função de transferência de um sistema representado por uma equação diferencial linear invariante no tempo, conforme a Eq.(26) é definida como a relação entre a transformada de Laplace da saída, $Y(s)$ (ou resposta) e a transformada de Laplace da entrada, $X(s)$ (ou sinal de excitação), admitindo-se toda as condições iniciais nulas (NISE; SILVA, 2002).

$$\frac{Y(s)}{X(s)} = \frac{b_0s^m + b_1s^{m-1} + b_{m-1}s}{a_0s^n + a_1s^{n-1} + a_{n-1}s + a_n} \quad (26)$$

A ordem do sistema corresponde ao maior expoente do denominador da Eq.(26). Utilizando uma função de transferência torna-se possível representar a dinâmica de um sistema por meio de uma equação algébrica em s , embora esta não forneça nenhuma informação relativa à estrutura física do sistema (OGATA, 2010).

2.5.1 SINTONIZAÇÃO DE CONTROLADORES PID POR ZIEGLER E NICHOLS

O controlador Proporcional, Integral e Derivativo, PID, é amplamente utilizado em automação industrial (OGATA, 2010). A utilidade de controladores PID se dá pelo fato de que em geral, em projetos envolvendo sistemas de controle, não se conhece o modelo matemático da planta e portanto, métodos analíticos não são aplicáveis, tornando necessário o ajuste em campo (OGATA, 2010). Segundo Ogata (2010), a representação matemática de um PID é definida conforme a Eq.(27).

$$G_c(s) = K_P \left(1 + \frac{1}{T_I s} + T_D s \right), \quad (27)$$

Sendo,

- K_P o ganho proporcional ao erro;
- $1/T_i$ a taxa de restabelecimento do sistema, que influencia no regime estacionário;
- T_d , que é o tempo derivativo relacionado à minimização do erro em regime transitório.

Ziegler e Nichols desenvolveram dois métodos de sintonização que permitem o ajuste de um PID (OGATA, 2010). O primeiro método é baseado na resposta ao degrau e o segundo, baseado em um ganho e período crítico da planta (OGATA, 2010). Estas técnicas são eficazes em diversos tipos de ambientes industriais e nela são abrangidos parâmetros parasitas e peculiaridades que os modelos propostos anteriormente raramente incorporavam (CASTRUCCI, 2006).

O primeiro método consiste na estimação de parâmetros do PID da Eq.(27), baseados na resposta em malha aberta após excitação ao degrau, conforme a Fig.3.

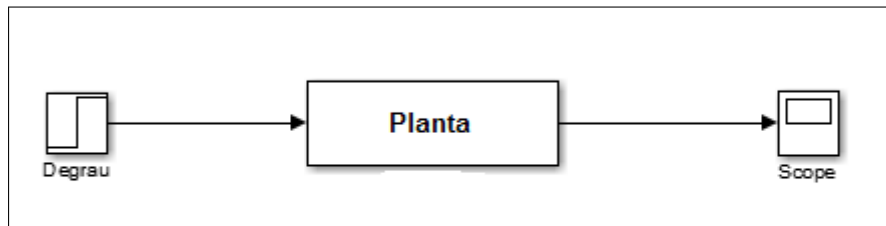


Figura 3: Modelo implementado para aplicar o primeiro método de Ziegler e Nichols

Fonte: autoria própria

Se a planta possuir integradores ou polos complexos conjugados dominantes, sua resposta será em forma de S, conforme a Fig.4, sendo aplicável portanto o primeiro método para sintonização de controladores PID (OGATA, 2010).

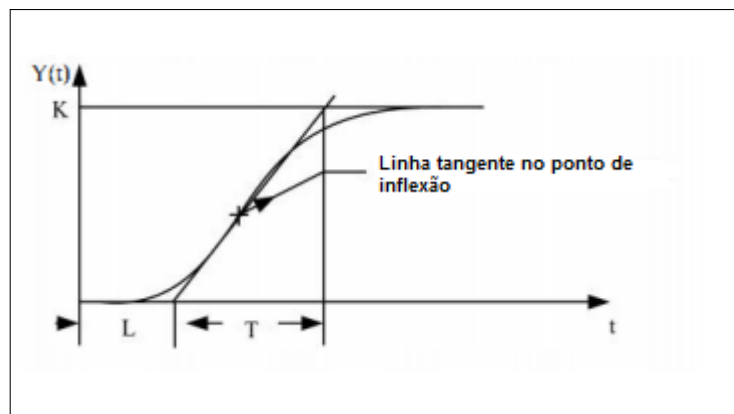


Figura 4: Curva de resposta para o primeiro método de Ziegler e Nichols

Fonte: Ogata (2010)

A curva S pode ser representada pelas constantes de atraso, L , tempo T e um valor de amplitude K , correspondente à intersecção da curva e o valor da resposta em regime estacionário. Tais parâmetros são definidos pela resposta da planta, conforme a Fig.4 (OGATA, 2010). Assim, uma linha tangente traçada no ponto de inflexão da curva de resposta da planta e determinando-se a intersecção da linha tangente com o eixo dos tempos e a linha $c(t) = K$ (OGATA, 2010).

De acordo com a Fig.4, uma função de transferência pode ser estabelecida conforme a relação descrita na Eq.(28) (OGATA, 2010).

$$\frac{Y(s)}{X(s)} = \frac{Ke^{-Ls}}{Ts + 1} \quad (28)$$

Os parâmetros $\frac{1}{T_I s}$, $T_D s$ e K_c da Eq.(27), para a sintonização do PID, podem ser ajustados conforme a Tab.1.

Tabela 1: Regra de sintonização baseada no degrau, proposta por Ziegler e Nichols

Tipo de controlador	K_C	T_D	$1/T_I$
P	$\frac{T}{L}$	∞	0
PI	$0.9\frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2\frac{T}{L}$	$2L$	$0.5L$

Após a determinação dos parâmetros do PID, o sistema de controle pode então ser realimentado com um compensador $G_p(s)$ em paralelo com uma planta $G_c(s)$, de acordo com a Fig.5.

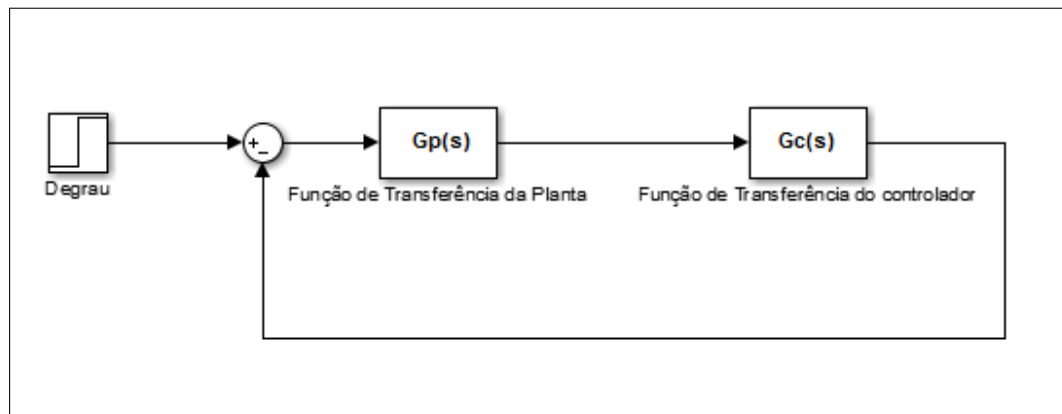


Figura 5: Esquema de um Sistema de Controle após a sintonização por Ziegler e Nichols
Fonte: Autoria Própria

Os métodos de Ziegler e Nichols servem como base para sintonia de ganhos em um PID. Além deste ajuste pode-se realizar uma sintonia fina, que corresponde à mudança de parâmetros em torno da região dos polos e zeros determinados, afim de que o sistema apresente uma resposta melhorada em malha fechada.

2.6 PADRONIZAÇÃO DE REDE BASEADA NO MODELO OSI

O modelo OSI (*Open Systems Interconnection*), ilustrado na Fig.6, é um modelo padrão de rede, composto por sete camadas, criado para facilitar a interconexão entre diferentes tipos de sistemas de computadores (TORRES, 2015).

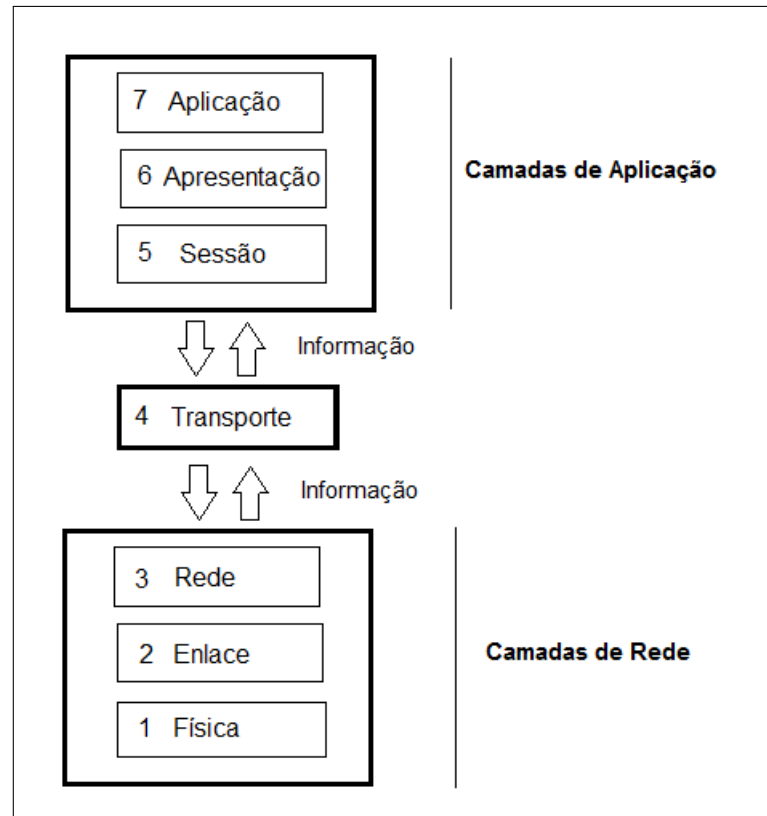


Figura 6: Fluxo de informações entre as camadas.
Fonte: Torres (2015)

Basicamente, o modelo OSI define sistematicamente e organizadamente a função de cada uma das setes camadas que o compõem (TANENBAUM, 2003). Assim, ele também é muito utilizado como referência para a criação de protocolos de comunicação apropriados (TORRES, 2015).

Para preservar a origem de cada dado transmitido por cada uma das setes camadas do modelo OSI, a informação flui por meio de um mecanismo denominado encapsulamento. Ou seja, durante a transmissão de um dado, uma camada recebe a informação, proveniente de uma camada superior, acrescenta informações de acordo com sua funcionalidade, disponibilizando novamente a informação para uma camada inferior a ela, conforme ilustrado na Fig.7 (TORRES, 2015).

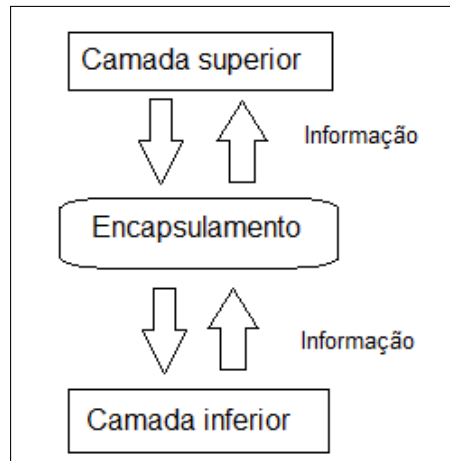


Figura 7: Fluxo de informações entre as camadas.

Fonte: Autoria Própria

O mecanismo de encapsulamento é útil para a verificação e detecção de erros de transmissão entre pacotes (TANENBAUM, 2003). A troca de mensagens pode ocorrer em via dupla porque cada divisão entre as funcionalidades das camadas pode receber ou enviar dados simultaneamente. Esta divisão entre as funcionalidades das camadas no modelo OSI são organizadas em três grupos: aplicação, transporte e rede, conforme ilustrado na Fig.6 (TORRES, 2015).

As camadas de rede são responsáveis pela transmissão de dados para dispositivos físicos envolvidos em um processo e são por isso, são consideradas, camadas de baixo nível (TORRES, 2015). Essa transmissão é dinâmica e calculada com base em dados estatísticos de rotas, entre a origem e o destino, afim de evitar componentes defeituosos, por exemplo (TANENBAUM, 2003).

A camada de transporte, é responsável por filtrar os dados, provenientes das camadas inferiores a ela e enviá-los a camada de rede (TORRES, 2015).

As camadas de aplicação, por sua vez, são as de mais alto nível e têm a função de traduzir em linguagem de máquina, dados que serão utilizados em aplicativos computacionais (TORRES, 2015).

Para este trabalho de conclusão de curso, serão estudadas as camadas que envolvem o transporte e o enlace.

2.6.1 A CAMADA DE ENLACE

Os dados fluem entre as camadas de rede em pequenas unidades de informações, geralmente desorganizadas, devido às prioridades entre as informações a serem transmitidas. Assim,

torna-se necessário criar uma camada que organiza os dados para que a informação possa obedecer a ordem necessária. A camada responsável por organizar pequenas unidades de dados é denominada camada de enlace (TANENBAUM, 2003).

A camada está ligada diretamente à camada de rede. Ela é responsável portanto por captar ou enviar pacotes de dados ligados à terceira camada (rede) e organizá-los, por meio do encapsulamento em quadros (TANENBAUM, 2003). Neste encapsulamento é possível encontrar a origem e destino dos dados (TANENBAUM, 2003).

O tipo de enquadramento pode ser por meio da contagem de caracteres (TANENBAUM, 2003). Neste método, realiza-se a criação de um cabeçalho no fluxo de dados, que identifica o caractere de *bytes* no quadro e informa quantos *bytes* deverão ser entregues, sendo esta informação utilizada para confirmação de chegada da mensagem (TANENBAUM, 2003).

Cada quadro, enviado pela conexão recebe um caractere de identificação e desta forma, a camada de enlace garante que cada quadro enviado será recebido e em uma ordem correta (TANENBAUM, 2003).

2.6.2 A CAMADA DE TRANSPORTE

A camada de transporte, ou quarta camada do modelo OSI, é responsável por retirar ou enviar os dados das camadas de aplicação e transportá-los para as camadas de rede, em forma de pacotes (TORRES, 2015).

Os pacotes, em geral, são enviados ou recebidos pela rede em uma sequência intercalada, devido a alta taxa de solicitação de serviços que a rede possui (TORRES, 2015). Para definir a ordem de intercalação para envio de pacotes, prioridades podem ser acrescentadas em um protocolo de comunicação (TANENBAUM, 2003).

Os protocolos de comunicação podem ser classificados em dois tipos: O protocolo TCP (*Transmission Control Protocol*) e o protocolo UDP (*User Datagram Protocol*) (TORRES, 2015).

2.7 PROTOCOLOS UDP

A função de cada camada deve ser escolhida com base em protocolos padronizados internacionalmente (TANENBAUM, 2003). Um protocolo é um conjunto de regras que controla o formato e o significado dos pacotes ou mensagens que são recebidas ou emitidas entre entidades pares, dentro de uma camada para implementar definições de serviços, ou *interfaces*,

a serem realizados entre camadas (TANENBAUM, 2003).

O protocolo UDP é um protocolo para transporte de dados, que pode operar em redes *offline* (TORRES, 2015). Este protocolo não verifica se a mensagem foi entregue ao destino e se limita apenas ao transporte de dados (TORRES, 2015). Assim, seu uso é útil em aplicações que manipulam poucas informações e que não estão sujeitas ao congestionamento de dados (TORRES, 2015).

O congestionamento de dados ocorre quando a rede recebe uma quantidade de dados além do que ela pode transmitir (TORRES, 2015). Assim, a rede faz com que os dados aguardem para ser transmitidos e em muitos casos, podem descartar dados de mensagens (perda de pacote de dados) (TORRES, 2015).

A comunicação por meio do protocolo TCP é mais complexa em relação a comunicação UDP (TORRES, 2015). Nela, o próprio protocolo organiza os dados recebidos - a distorção de dados é um problema recorrente em grandes redes - e verifica se todos os pacotes de dados chegaram corretamente ao seu destino (TORRES, 2015).

2.8 SISTEMAS DE CONTROLE VIA REDE - NCS

A tecnologia de redes industriais foi desenvolvida afim de aperfeiçoar a comunicação entre sensores e atuadores que estão distribuídos fisicamente em locais diferentes (FANTINATO et al., 2005).

O uso de redes de comunicação serial, integradas com sistemas de controle via rede (NCS), para substituir a tradicional ligação ponto a ponto, realizada por cabos de rede, têm sido frequente, uma vez que a NCS permite a descentralização do controle e favorece o desenvolvimento de sistemas de controle com modularidade, facilidade de diagnóstico e baixo custo (WALSH; YE, 2001). Automóveis inteligentes, aeronaves, plantas de manufatura e outros contextos, que necessitam o uso de uma comunicação controlada e automática, são os principais utilizadores de um NCS.

Aplicações recentes de sistemas de controle distribuído demonstram uma nova abordagem envolvendo redes de comunicação serial. A definição característica de um NCS é que a informação de referência (entrada de referência, saída da planta, controle de saída, etc) é alterada via rede entre componentes do sistema de controle (sensores, controladores, atuadores, etc) fisicamente distribuído em lugares distintos (WALSH; YE, 2001). A Fig.8 ilustra uma típica configuração e o fluxo de informação de um NCS (ZHANG et al., 2001).

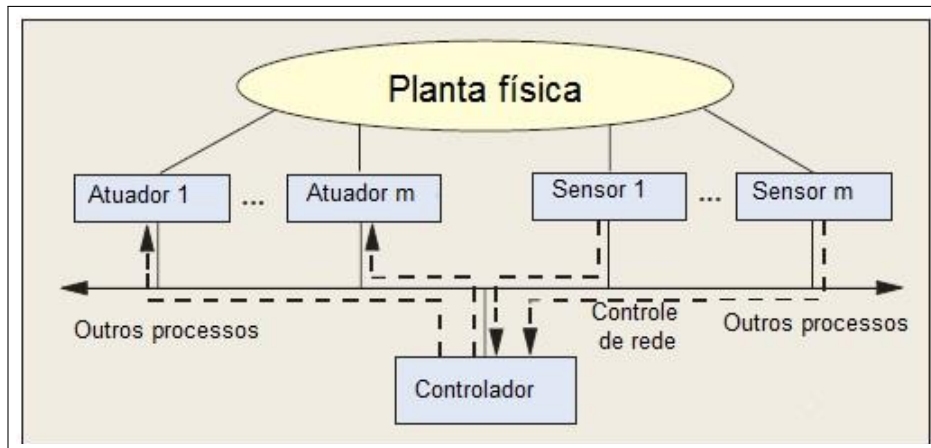


Figura 8: Modelo de um NCS
Fonte:ZHANG et al.

De acordo com a Fig.(8), o sinal de controle é enviado para o atuador através de uma mensagem encaminhada via rede, enquanto o sensor amostra o sinal de saída da planta e retorna a mesma informação através da rede de comunicação em que está inserido (ZHANG et al., 2001). É importante ressaltar que tanto o sensor quanto o atuador estão inseridos em uma malhada de controle fechada sob uma rede de comunicação em tempo real (GODOY et al., 2008).

Devido à inserção de uma rede serial para comunicação, o sistema cabeado é reduzido, facilitando a manutenção e prevenção de falhas no sistema, aumentando sua robustez (WALSH; YE, 2001). Desta forma, as redes de comunicação serial são usadas para transmitir informações e sinais de controle entre componentes de um sistema fisicamente organizado e outros subsistemas, tais como computadores supervisórios, controladores, entradas e saídas inteligentes (sensores inteligentes e atuadores) (ZHANG et al., 2001). Nesse sentido, o principal objetivo para controle, é a coordenação e a comunicação desses processos, de forma que atendam a um bom funcionamento, com desempenho global garantido (ZHANG et al., 2001).

O desempenho e a estabilidade de um NCS em geral são afetados pela presença da rede de comunicação e fatores degenerativos, como amostragem de sinais e atrasos nos sinais de comunicação entre sensores, atuadores e controladores do sistema (WALSH; YE, 2001). Com a inclusão de uma rede para a comunicação no sistema de controle, alguns fatores como os atrasos de comunicação gerados pela rede e a discretização do sinal enviado do sensor acabam exercendo influência sobre o desempenho da rede, podendo levar o comportamento do sistema até a instabilidade (GODOY et al., 2008).

2.8.1 ATRASOS EM SISTEMAS DE CONTROLE VIA REDES - NCS

Em geral, os estudos realizados envolvendo redes e controle NCS são realizados de forma independente e suas abordagens, em geral, desconsideram o problema clássico envolvendo escalonamento de tarefas em um sistema de tempo real e os atrasos envolvendo a comunicação entre o sensor, controlador e atuador (ZHANG et al., 2001).

O atraso de comunicação em um NCS representa um dos principais fatores que influenciam em seu desempenho e estabilidade. Por isso, a influência dos atrasos em mensagens tem sido uma área bastante pesquisada e diversas metodologias tem sido desenvolvidas para compensar esses efeitos (GODOY et al., 2008). Técnicas robustas que consideram as transmissões de atrasos variantes no tempo e intervalos de transmissão ou perda de pacotes no tempo como sendo uma faixa de valores degradantes no desempenho do sistema são objetivos de estudo frequentes (TANG et al., 2015).

O compartilhamento de informação da rede com outros subsistemas ou tempo de processamento que os dispositivos de *hardware* consomem para retornar uma informação de volta para a rede são exemplos de problemas recorrentes em um NCS que influenciam no desempenho de um NCS, podendo levar o NCS a instabilidade (WALSH; YE, 2001). Este compartilhamento depende do tipo de protocolo escolhido para comunicação, da velocidade de transmissão (período de amostragem), tamanho das mensagens de dados, períodos de amostragem dos dispositivos e porcentagem de mensagens não recuperadas (GODOY et al., 2008).

Assim, existe a teoria de escalonamento, baseada na ordem de execução de tarefas, para que seja cumprida a ordem de execução e requisitos temporais de mensagens (WALSH; YE, 2001). Basicamente, existem dois tipos de algoritmos para escalonamento: O algoritmo estático e o dinâmico (WALSH; YE, 2001).

Outro critério muito visado em projetos de um NCS leva em consideração o cumprimento de requisito temporal de cada mensagem (*deadline*), que diz respeito à transmissão de mensagens dentro de um tempo limitado e menor em relação ao tempo de amostragem, afim de evitar a sobreposição e perdas de mensagens que podem deteriorar o desempenho do sistema (GODOY et al., 2008).

Devido à limitação do tamanho dos pacotes de informações transmitidas na camada de dados e ao congestionamento da rede, um controlador ou atuador não recebe pacotes imediatamente enviados de um sensor ou controlador (MO; XIE, 2013). Assim, neste estudo não será considerado o controle sobre os atrasos mas sim, os efeitos que os atrasos podem causar em um sistema de controle via rede.

Considerando-se um sistema genérico SISO representado em espaço de estados, conforme a Eq.(29), onde A , B e C representam respectivamente as matrizes do sistema, dos estados de entrada e a equação de saída, sendo uma combinação linear entre as variáveis de estado e as de entrada, utilizada para quaisquer outras variáveis do sistema. Além disso, x e u são respectivamente o estado e os valores de entrada em \mathfrak{R}^n e \mathfrak{R}^m . A , B e C são matrizes que possuem dimensões compatíveis.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \quad (29)$$

Considerando um atraso de tempo, τ_k^{sc} , nos estados do sinal de referência do sensor e τ_k^{ca} , como sendo o atraso de tempo inserido no controlador, a representação em espaço de estados é definida conforme a Eq.(30).

$$\begin{aligned} \dot{x} &= Ax + Bu(t - \tau_k^{ca}) \\ y &= Cx(t - \tau_k^{sc}) \end{aligned} \quad (30)$$

Aplicando a transformada de Laplace, admitindo que $X(s)$ que todas as condições iniciais sejam nulas, obtém-se a função de transferência do sistemas, $H(s)$ Eq.(31)

$$H(s) = C(sI - A)^{-1} B.e^{-\tau_k^{sc}s}.e^{-\tau_k^{ca}s} \quad (31)$$

Então, o tempo de atraso total, τ_k é igual a soma entre $\tau_k^{sc} + \tau_k^{ca}$ (MO; XIE, 2013). A influência de tais atrasos, em um controle realimentado NCS pode ser definida, então, a partir do seu comportamento inicial $X(s)$ e final $Y(s)$, conforme a Eq.(32).

$$\frac{Y(s)}{U(s)} = e^{-(\tau_k^{sc} + \tau_k^{ca})s} G(s) \quad (32)$$

A partir da Eq.(32), considera-se que τ_k obedece a uma distribuição uniforme em um intervalo fechado de tempo (MO; XIE, 2013).

3 MATERIAIS E MÉTODOS

Este capítulo apresenta os passos que foram seguidos para o desenvolvimento deste trabalho. Portanto, as seguintes etapas são propostas na metodologia: revisão bibliográfica, formulação da *interface* proposta e simulação do modelo matemático que descreve o motor DC, para posterior constatação dos efeitos que atrasos invariantes no tempo exercem sobre sistemas LIT, que serão os resultados para atender a proposta deste trabalho.

3.1 REVISÃO BIBLIOGRÁFICA

Primeiramente realizou-se uma revisão bibliográfica sobre alguns trabalho já realizados na área. Neste trabalho, os autores utilizados para o NCS foram os Zhang et al. (2001) e Walsh e Ye (2001) e para a rede e seus protocolos de comunicação TANENBAUM (2003) e Torres (2015). Para a simulação e validação dos resultados obtidos, foi utilizado o estudo sobre o controle da velocidade angular a partir de um torque aplicado. Este estudo foi proposto por Meshram e Kanojiya (2012).

3.2 FORMULAÇÃO E SIMULAÇÃO DO MODELO

A verificação da modelagem do NCS se deu por meio da comparação entre os efeitos dos atrasos gerados entre a transmissão dos sinais de entrada e a resposta de controle utilizando uma rede de transmissão. Para isso, foram implementados uma planta e um controlador, conectados por uma rede. O processo para transmissão se deu da seguinte forma: a planta solicitou via rede a comunicação com dispositivos externos, enviado um sinal por meio de uma resposta em frequência.

A rede de transmissão recebeu o sinal em frequência e o discretizou, enviando o resultado a um novo nó de rede, onde estava inserido o controlador. O controlador, por sua vez, estava em série à planta e realizou uma determinada ação de controle, devolvendo estas informações novamente à rede de transmissão.

Considerando atrasos invariantes no tempo inseridos no sistema de controle, implementou-se um modelo de planta de um motor DC com compensação do tipo PID, visando o controle da velocidade angular por meio do torque aplicado, proposto por Meshram e Kanojiya (2012).

A sintonização do PID foi a mesma realizada por Meshram e Kanojiya (2012), ou seja, aplicando o primeiro método proposto por Ziegler e Nichols. Após a sintonização do compensador, realizou-se o seu equivalente discreto, inserindo-o na rede juntamente com a função de transferência em domínio contínuo do motor DC. Desta forma, atrasos foram inseridos no sistema de controle, via *interface*, com o objetivo de se constatar seus efeitos, caso o sistema de controle projetado estivesse submetido aos seus efeitos em um ambiente real.

A rede utilizada foi modelada atendendo a um padrão arquitetural baseado no modelo OSI com protocolos de comunicação UDP. As simulações abrangeram a camada de enlace, responsável por se comunicar com dispositivos externos à rede. Esta *interface* foi realizada por meio da *toolbox* GUI do Matlab/Simulink[®] e têm por objetivo apresentar, por meio do fornecimento de parâmetros de simulação e coeficientes da função de transferência de uma planta e um controlador, respostas de sistemas de controle. Assim, o ambiente se torna facilitado, uma vez que não exige do usuário qualquer conhecimento prévio, referente ao manuseio e construção de diagramas de blocos de controle via Matlab/Simulink[®].

Além dos fatores citados, este *software* representa uma alternativa de um baixo custo, estando disponível para uso nos casos em que estudantes e pesquisadores do curso de Engenharia de Controle e Automação da UTFPR do câmpus de Cornélio Procópio necessitem realizar simulações de sistemas de controle via rede considerando atrasos de comunicação.

4 RESULTADOS E DISCUSSÕES

4.1 PROJETO DO CONTROLADOR

Esta sessão apresenta os passos seguidos para o projeto e sintonização do controlador PID, pelo método de Ziegler e Nichols. O projeto tem por objetivo apresentar o menor *overshoot* e tempo de estabilização possíveis para um modelo de planta proposto por Meshram e Kanojiya (2012). O projeto do controlador foi idealizado por Meshram e Kanojiya (2012), para que fossem atendidas algumas restrições de projeto, tais como, o menor tempo de estabilização e *overshoot* possíveis na resposta controlada. Foi pressuposto que o sistema é LIT.

O modelo de planta, conforme Meshram e Kanojiya (2012) é dado na Eq.(33).

$$\frac{Y(s)}{U(s)} = \frac{0.015}{0.01s^2 + 0.14s + 0.40015} \quad (33)$$

Por meio da função de transferência descrita na Eq.(33), aplicou-se o primeiro método de Ziegler e Nichols. Primeiramente, inseriu-se uma perturbação na planta em malha aberta, representado por um degrau de amplitude unitária que iniciou-se em 1s conforme a Fig.9.

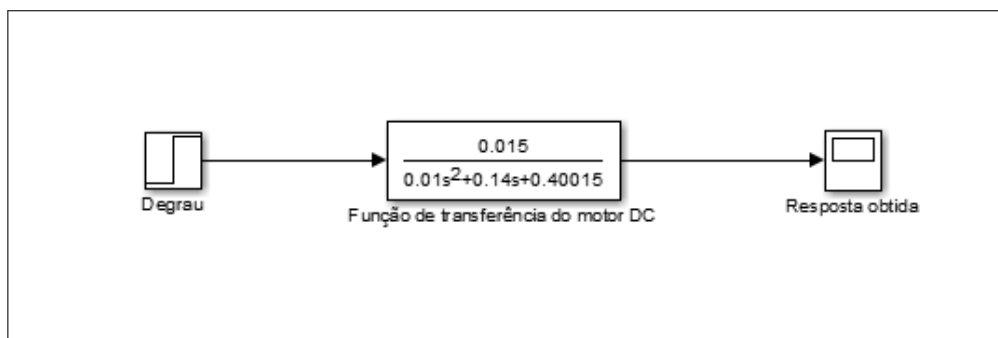


Figura 9: Diagrama de blocos para o primeiro método de Ziegler e Nichols
Fonte: Autoria própria

A partir da aplicação e simulação ao sinal degrau, obteve-se a resposta da planta conforme a Fig.10.

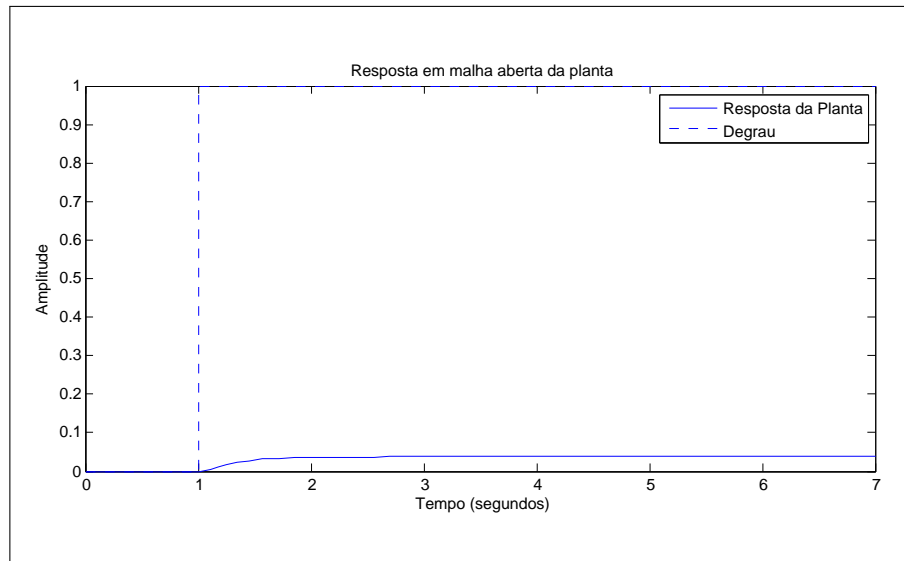


Figura 10: Curva de resposta para a sintonia do PID

Fonte: Autoria própria

Da Fig. 10, traçou-se uma linha tangente ao ponto de inflexão e retiraram-se os parâmetros referentes as constantes de atraso L , tempo T e amplitude K , conforme representado na Fig.11.

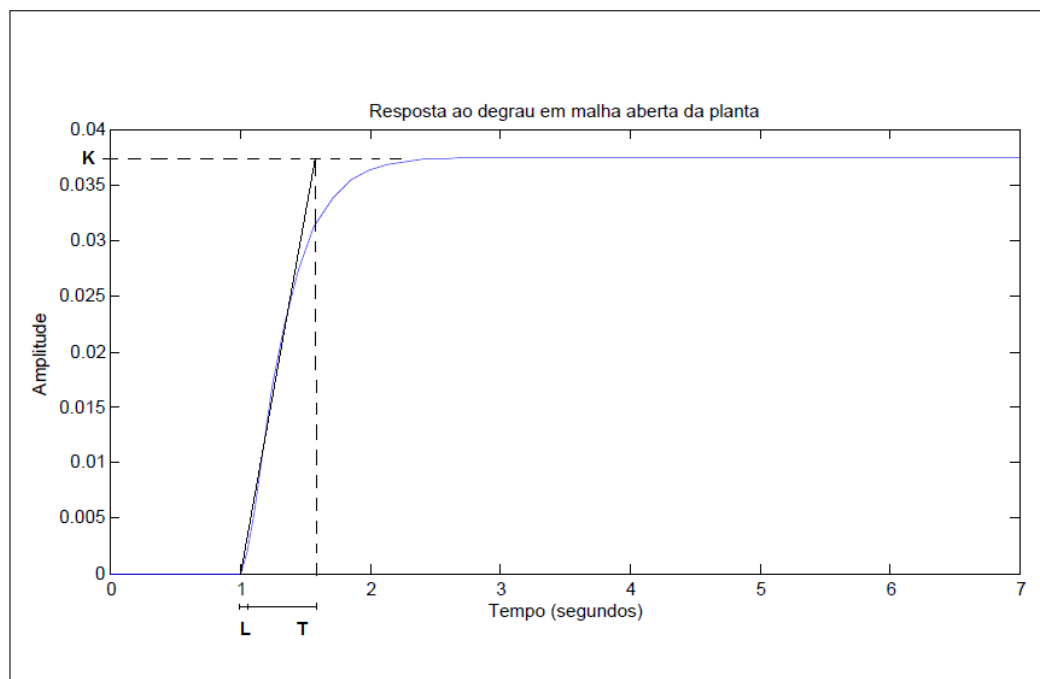


Figura 11: Representação da reta tangente e os parâmetros para sintonização do PID

Fonte: Autoria própria

Assim, os valores retirados correspondem a $L = 0.05$, $T = 0.5$ e $K = 0.037$. Tais valores foram inseridos na Tab.(1), de onde retirou-se os valores $T_D = 0.02$ e $T_I = 0.005$ que deram origem aos ganhos derivativo $K_d = 8.1081$ e integrativo $K_i = 3243.24$, sendo o ganho proporcional linearizado por K cujo valor é $K_p = 324.324$.

Como o PID possuía um ganho derivativo não causal, um filtro foi considerado em sua função de transferência. Desta forma, um PID com ganhos não ideais foi representado pela Eq.(34), sendo o parâmetro N a constante associada ao filtro passa-baixas.

$$G_c(s) = K_p + K_i + \frac{1}{s} + K_d \frac{N}{1 + \frac{1}{s}} \quad (34)$$

Expandiu-se o denominador da Eq.(35) e aplicou-se os valores sintonizados com $N = 10$, obtendo a função de transferência do controlador $G_c(s)$.

$$G_c(s) = \frac{s^2[405.4] + s[6486] + 3.2410^{-4}}{s(s + 10)} \quad (35)$$

Inseriu-se a Eq.(35) para ser simulada juntamente com a planta descrita na Eq.(33) em malha fechada, sem o uso da rede, conforme a Fig.12.

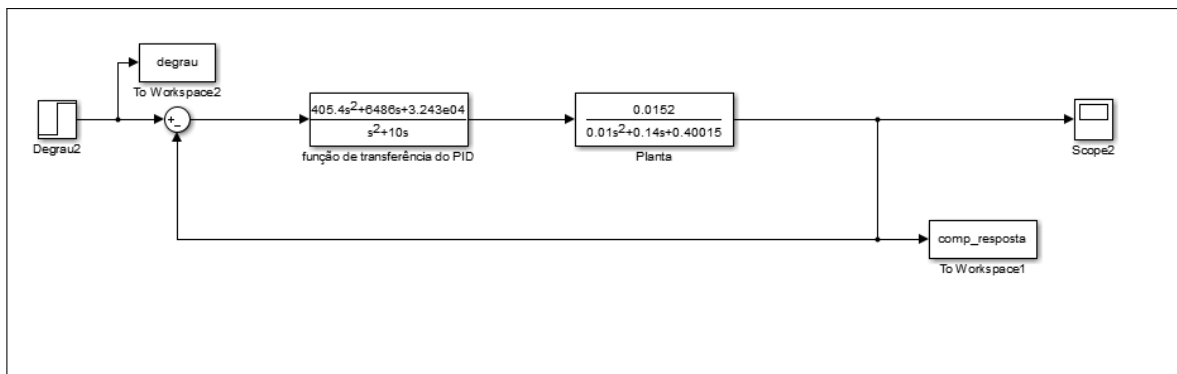


Figura 12: Resposta do PID por Ziegler e Nichols implementado em malha fechada no Simulink
Fonte: Autoria própria

Obteve-se o comportamento de resposta obtido dado de acordo com a Fig.13.

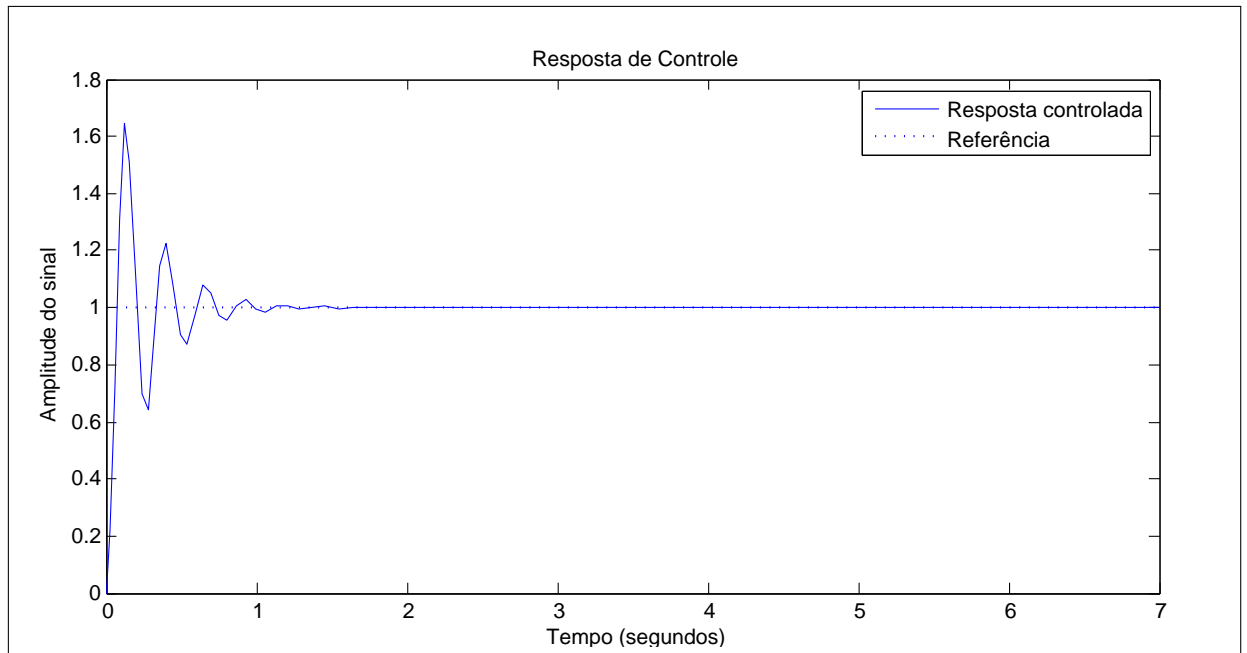


Figura 13: Resposta do PID implementado em malha fechada no Simulink
Fonte: Autoria própria

De acordo com a Fig.13 observou-se que o *overshoot* era de 64% , com um tempo de estabilização do sistema inferior a 2 segundos ou $2s$.

Analisou-se então, a faixa de frequência para que o controlador pudesse atuar sem que ocorresse o fenômeno *aliasing*. Após realizada a discretização, obteve-se o diagrama de Bode da função de transferência da planta referente ao motor DC, de acordo com a Fig.14.

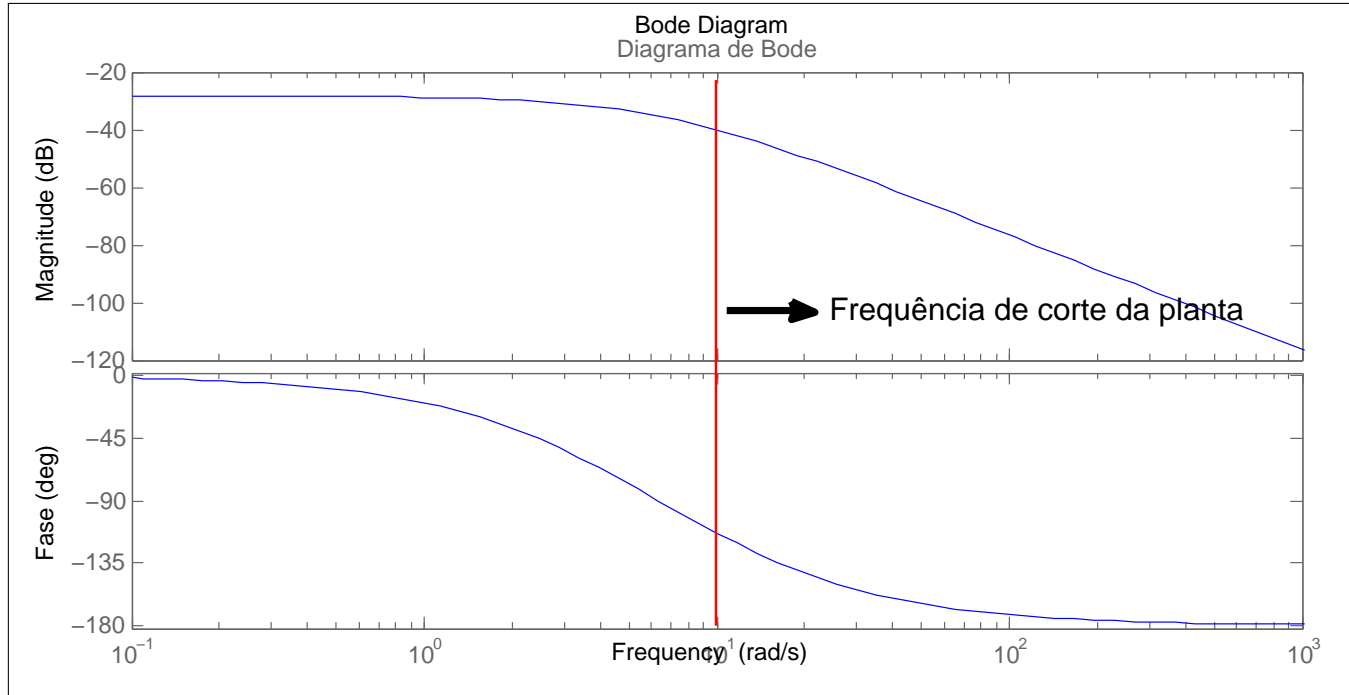


Figura 14: Diagrama de Bode da planta
Fonte: Autoria própria

Pelo diagrama de Bode da Fig.14, observou-se que a frequência de corte, ω_N , é aproximadamente 10 rad/s. Para que não ocorresse *aliasing*, a frequência de amostragem do controlador, ω_s , deveria ser maior que o dobro da frequência de corte da planta, conforme a Eq.(25).

Assim, considerou-se a frequência de amostragem do controlador como sendo duas vezes maior que a frequência de corte da função de transferência do motor DC de acordo com a Eq.(36).

$$\omega_s \geq 2\omega_N = 20 \quad (36)$$

O período de amostragem do controlador, T_s , é o inverso da frequência de amostragem obtida pela Eq.(36). A presente *interface* leva em consideração períodos de amostragem com valores em unidades de segundos. Assim, buscou-se relacionar a frequência encontrada para uma frequência equivalente em unidades de segundo, sendo esta relação dada pelo seu período T_s conforme a Eq.(37).

$$\omega_s = \frac{2\pi}{T_s} \geq 2\omega_N \quad (37)$$

Pela Eq.(37), o período em segundos deve obedecer a a Eq.(38).

$$T_s \leq \frac{\pi}{\omega_s} \leq \frac{2\pi}{10} \quad (38)$$

Portanto, considerou-se para o período de amostragem em segundos a relação da Eq.(39).

$$T_s = \frac{2\pi}{20} \quad (39)$$

Pela Eq.(39), constatou-se que o período de amostragem em unidades de segundo T_s apresentou um valor aproximado de 0.3142 segundos. Portanto, valores inferiores a este, não ocasionariam o fenômeno *aliasing*. Assim, para atender a esta exigência, considerou-se o período de amostragem em segundos do controlador com o valor $T_s = 0.001s$.

Como a rede trabalha em domínio discreto, realizou-se a discretização equivalente pelo método de aproximação por Tustin, realizado por meio do comando “c2d” do Matlab/Simulink[®]. A transformação para o equivalente discreto está de acordo com a Eq.(40).

$$G_c(z) = \frac{[405.4]z^2 - z[804.38] + 399}{z^2 - [1.99]z + 0.99} \quad (40)$$

O método para aproximação foi o ZOH, com o período de amostragem em $T_s = 0.001s$. Nota-se que este período de amostragem é o menor possível que o Matlab/Simulink[®] trabalha. Afim de verificar a equivalência entre o comportamento do sistema no domínio contínuo e discreto, realizou-se a montagem do sistema dado pelas Eq.(33-40) via Matlab/Simulink[®], onde o objetivo era observar o comportamento de resposta referente à saída da planta em malha fechada, conforme a Fig.15.

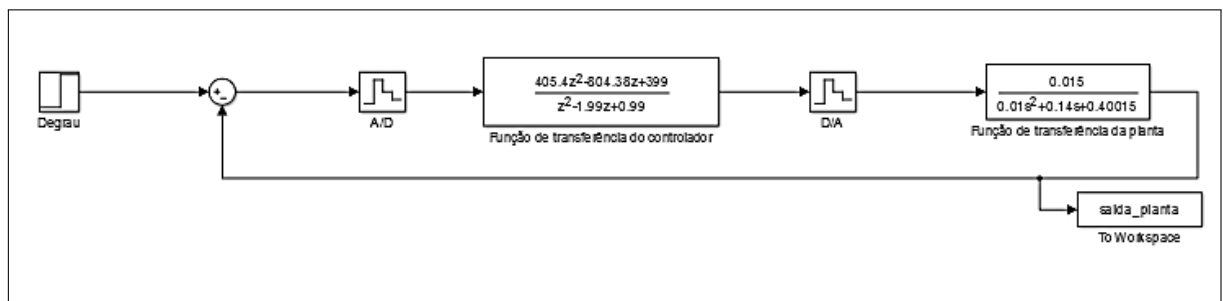


Figura 15: Compensador digital inserido em um sistema de controle D
Fonte: Autoria própria

Então, realizou-se a simulação no tempo de 10s. O resultado obtido esteve conforme ilustrado na Fig.16.

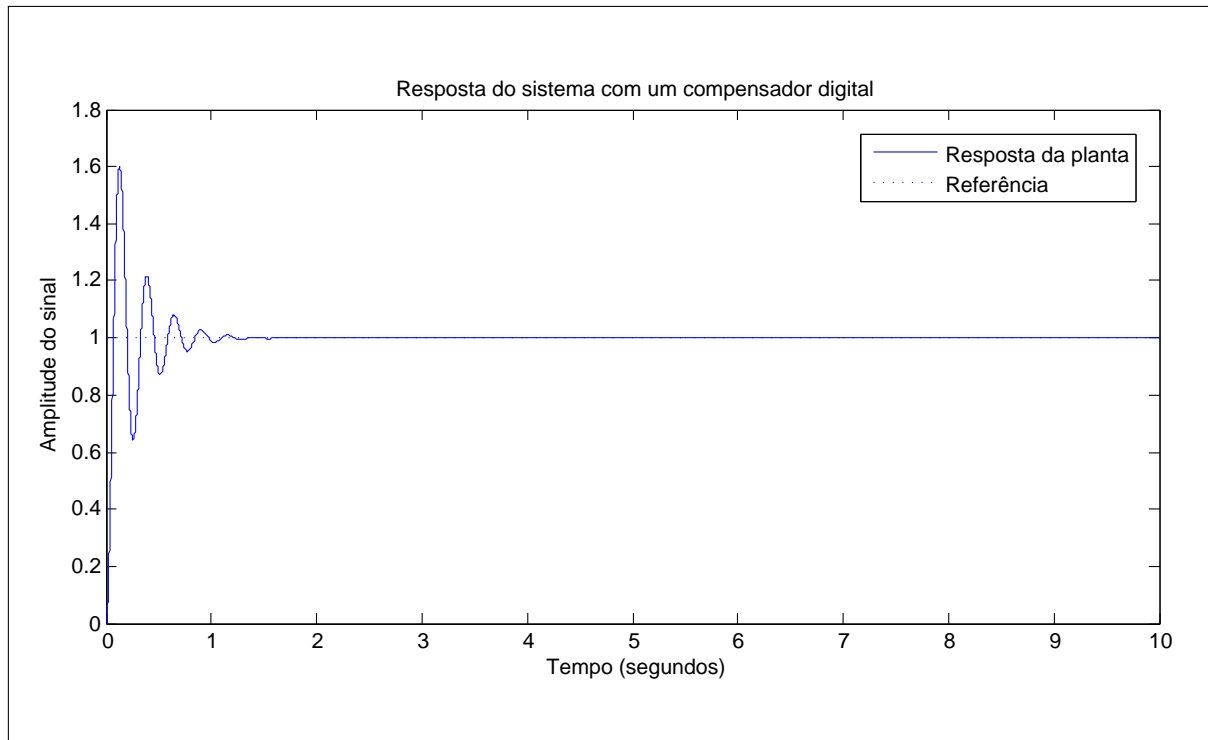


Figura 16: Resposta ao degrau em domínio discreto
Fonte: Autoria própria

De acordo com a Fig.16, verificou-se que a resposta foi equivalente ao domínio contínuo ilustrado na Fig.13.

4.2 PROJETO DA INTERFACE

Esta sessão apresenta os passos seguidos para entendimento, aprimoramento da comunicação, inserção de atrasos e criação de uma *interface* para um sistema de controle via rede - NCS. Primeiramente foi implementada a comunicação via redes, via protocolo UDP, entre um modelo de planta e controlador genéricos. Esta implementação se deu por meio de um diagrama de blocos, implementado no Matlab/Simulink[®]. Em seguida, foram inseridas funções que implementariam atrasos invariantes no tempo durante a transmissão de pacotes de informações entre o controlador e a planta, via rede.

O modelo da *interface* foi projetado pela *toolbox* GUI, via Matlab/Simulink[®] para possibilitar a inserção de parâmetros referentes à rede, atrasos invariantes no tempo, função de transferência da planta e controlador. Esta *interface* encapsulará toda a comunicação de um NCS, simulada no Simulink[®].

Com base nas teorias aplicadas a sistemas de controle via rede, modelou-se um sistema via *software* Matlab/Simulink[®] que possibilitou a simulação de um sistema de controle con-

tendo as principais características de um NCS. O sistema para simulação, conforme a Fig.17, foi então encapsulado por uma *interface*.

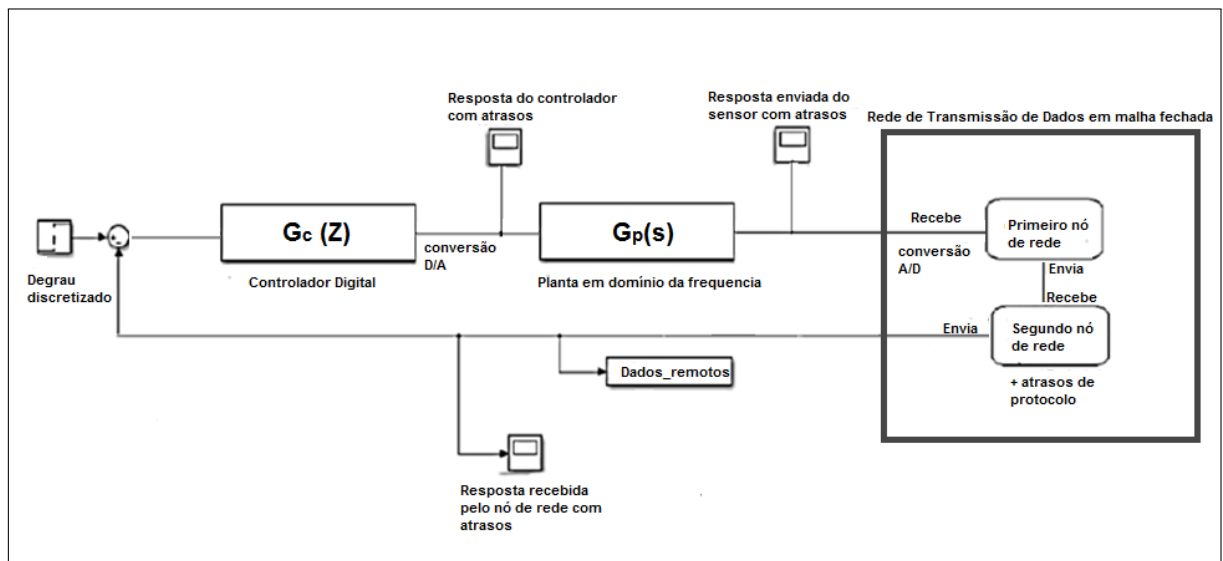


Figura 17: Modelo de um NCS encapsulado pela interface
Fonte: Autoria própria

Assim, através dos parâmetros inseridos na *interface*, ocorreu a transmissão ao controle digital via rede, implementado via *software* Matlab/Simulink[®]. O processo ocorreu da seguinte maneira: primeiramente a planta, $G_p(s)$, recebeu uma perturbação ao degrau e enviou a resposta em frequência com atrasos. Os atrasos foram provenientes de um suposto sensor fisicamente localizado na planta, o qual era responsável por enviar informações para a rede.

Na rede, as informações com atrasos foram convertidas em sinais discretos. Os pacotes de informações digitais foram enviados para um segundo nó de rede, que devolveu para a realimentação do sistema de controle no somador. Porém, durante o envio de informações, os pacotes situados no segundo nó de rede sofreram perdas relacionadas aos protocolos de informações UDP de um NCS, tais como baixa prioridade sobre a entrega de mensagens, perda de pacotes, etc. Tais perdas deram origem a um segundo atraso, que se somou ao atraso da informação proveniente do sensor.

Desta forma, a informação total para realimentação da malha de controle foi disponibilizada com atrasos provenientes de um sensor e de perda de pacotes de rede. Houve-se então, no somador uma comparação entre o sinal de referência aplicado (no caso um degrau discretizado) e a informação digitalizada entregue com atrasos.

O resultado da comparação realizada no somador foi disponibilizado ao controlador digital, $G_c(z)$, onde com base nas informações recebidas foi determinada uma ação de controle. Após a compensação, o controlador disponibilizou a informação compensada em forma

contínua, sendo esta conversão realizada por um ZOH, ligado à saída da ação de controle com um período de conversão. Finalmente, o atuador, acoplado fisicamente à planta, recebeu a mensagem de controle e atuou sobre o sistema conforme a ação determinada de controle.

É importante ressaltar que todos os períodos de amostragem e discretização necessários à rede foram configurados para serem iguais ao período de amostragem do controlador, afim de evitar possíveis sobreposições de sinais, ocasionados durante os processos para digitalização e comunicação remotas.

A partir do processo encapsulado foram obtidas respostas três respostas gráficas, sendo elas referentes à resposta em frequência da planta enviada para a rede por meio de um sensor contendo atrasos; O sinal de rede recebido com novos atrasos causados pelo protocolo UDP correspondente a ação do controlador digital; e a resposta entregue ao atuador na planta, fisicamente acoplado à ela.

As respostas de controle foram equivalentes a visualização dos *scopes* inseridos no *software* Matlab/Simulink[®] da Fig17. Para serem ilustradas na tela da *interface*, foi necessário a realização de algumas configurações necessárias ao funcionamento de um NCS.

Os seguintes passos apresentam as configurações necessárias para funcionamento correto da *interface*.

- Inicializar o *software* Matlab/Simulink[®];
- Indicar o diretório onde esta armazenada a *interface*;
- Após indicar diretório, digitar no *prompt* de comando do *software* do Matlab/Simulink[®] o comando: *inicializarNCS*. Após esta requisição, a *interface* será visualizada, conforme a Fig.18.

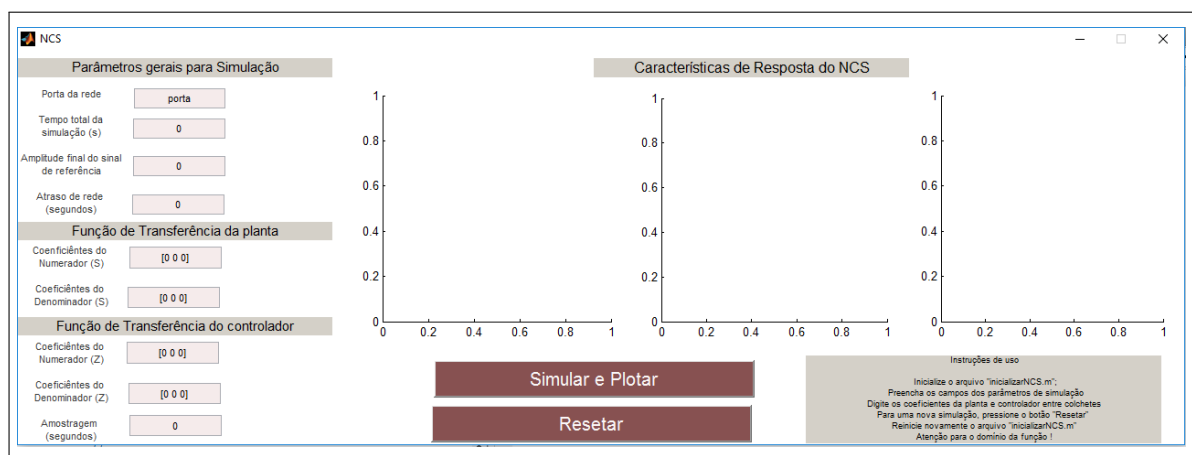


Figura 18: Interface Projetada para simulação de um NCS

Fonte: Autoria Própria

- Digitar os *Parâmetros gerais da simulação*, *Função de transferência da planta* e *Função de transferência do controlador*, situados na parte esquerda da tela, exibidos em detalhes na Fig.19.

The image shows a software interface for NCS simulation. It is divided into three main sections, each with a title bar and several input fields:

- Parâmetros gerais para Simulação:**
 - Porta da rede: porta
 - Tempo total da simulação (segundos): 0
 - Amplitude final do sinal de referência: 0
 - Atraso de rede (segundos): 0
- Função de Transferência da planta:**
 - Coefficientes do Numerador (s): [0 0 0]
 - Coefficientes do Denominador (s): [0 0 0]
- Função de Transferência do controlador:**
 - Coefficientes do Numerador (Z): [0 0 0]
 - Coefficientes do Denominador (Z): [0 0 0]
 - Amostragem (segundos): 0

Figura 19: Parâmetros ampliados da Interface projetada para simulação de um NCS
Fonte: Autoria Própria

- Cada parâmetro geral citado exige a inserção de informações específicas. Primeiramente, é necessário informar a *Porta de rede*, para comunicação dos dados entre a rede e seus componentes de controle; em seguida, é requerido o *tempo total de simulação* em unidades de segundos, bem como a *amplitude do sinal de referência* sendo este um degrau unitário. Por último, informa-se o *Atraso de rede*, que diz respeito aos valores em segundos de espera que a rede considerará para realizar a transmissão de mensagens de controle. Tais atrasos representam o tempo ocioso que a rede permanece enquanto espera um sinal proveniente do sensor e/ou planta.
- Digitar nos campos correspondentes a *Função de Transferência da Planta*, compreendidos entre os coeficientes do numerador e denominador da planta em domínio contínuo.
- No campo *Função de Transferência do Controlador*, deve-se inserir os coeficientes do numerador e denominador, provenientes da função de transferência em domínio discreto.

É importante frisar que no campo *amostragem* do compensador digital, deve-se inserir o período de amostragem, sendo os valores deste maiores que 1μ . Este é o valor mínimo que o *software* Matlab/Simulink[®] considera para a simulação do processo.

- Pressione o botão *Simular e Plotar* e aguarde o retorno da mensagem “A simulação foi concluída!” no *prompt* de comando do Matlab/Simulink[®], indicando a finalização da simulação. As respostas de controle, juntamente com o sinal de referência serão então visualizadas na *interface*, conforme o exemplo da Fig.20.

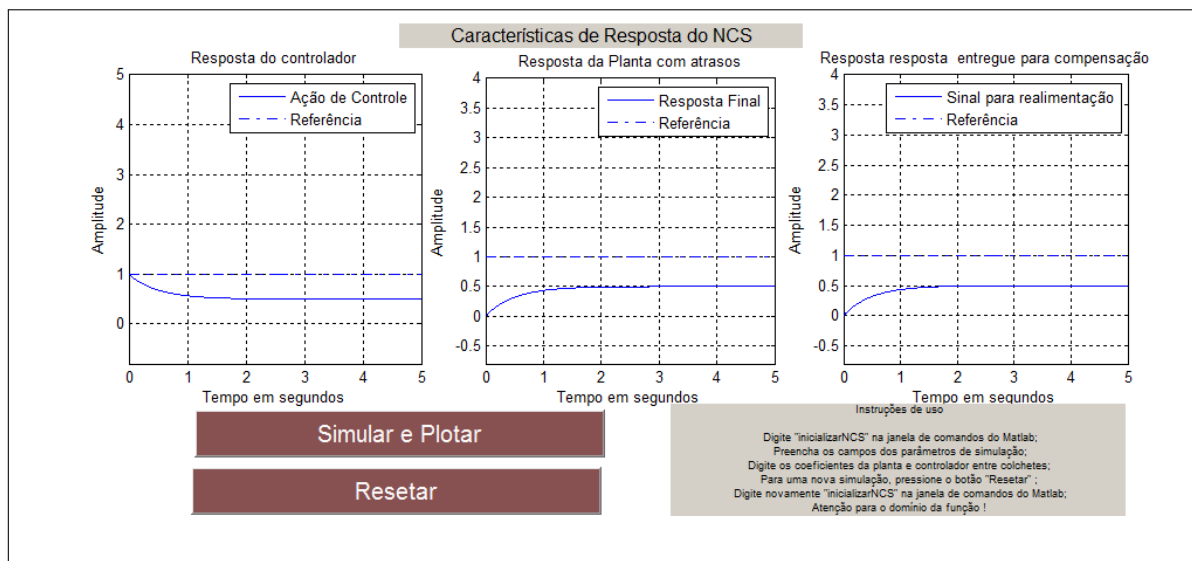


Figura 20: Respostas ampliadas da *Interface* projetada para simulação de um NCS

Fonte: Autoria Própria

- Os gráficos da Fig.20 correspondem as respostas com atrasos para um sinal tipo degrau, sendo sua amplitude definida pelo usuário. O primeiro gráfico, cujo nome é “Resposta do controlador com atrasos”, corresponde à ação de controle mediante aos atrasos de informações; o segundo gráfico com o nome de “Resposta da planta com atrasos”, corresponde a resposta final da planta, mediante a ação de controle e com atraso do sensor e por fim, o gráfico “Resposta entregue para compensação” corresponde a mensagem entregue à realimentação do sistema em malha fechada para realização de uma atuação de controle.
- Para uma nova simulação eficiente, as portas da rede devem ser novamente abertas e limpas de dados provenientes de simulações anteriores. Por isso, deve-se pressionar o comando *Reinicializar* e retomar a chamada da *interface*, digitando *InicializarNCS* na janela de comando do *software* Matlab/Simulink[®].
- Também é possível visualizar um guia, contendo as instruções de uso, na própria *interface*, conforme a Fig.21.

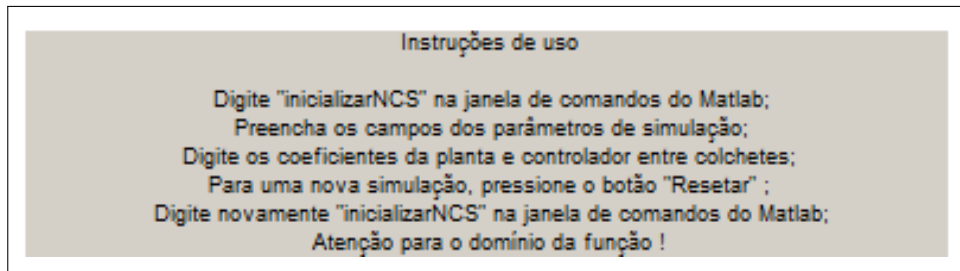


Figura 21: Instruções ampliadas da *Interface* projetada para simulação de um NCS
Fonte: Autoria Própria

4.3 VALIDAÇÃO DOS RESULTADOS

A verificação dos efeitos gerados por atrasos de informações dos pacotes de informações trocadas entre a função de transferência do controlador e da planta foi executada por meio de simulações, realizadas na *interface* proposta e com o auxílio do *software* Matlab/Simulink®. Assim, os passos para realização da validação sobre os efeitos gerados foram compreendidos por meio da inserção de atrasos nos protocolos de comunicação da rede.

O comportamento entre o controlador e a planta foi avaliado mediante o efeito de tais atrasos, manipulados via *interface*, onde foram obtidas diversas simulações sobre a curva de resposta do sistema de controle em questão. A Fig.22 ilustra a forma de comunicação do NCS projetado e encapsulado por meio *interface* proposta. Os gráficos gerados na *interface* foram os mesmos que a ferramenta *scope* geraria em um Simulink®.

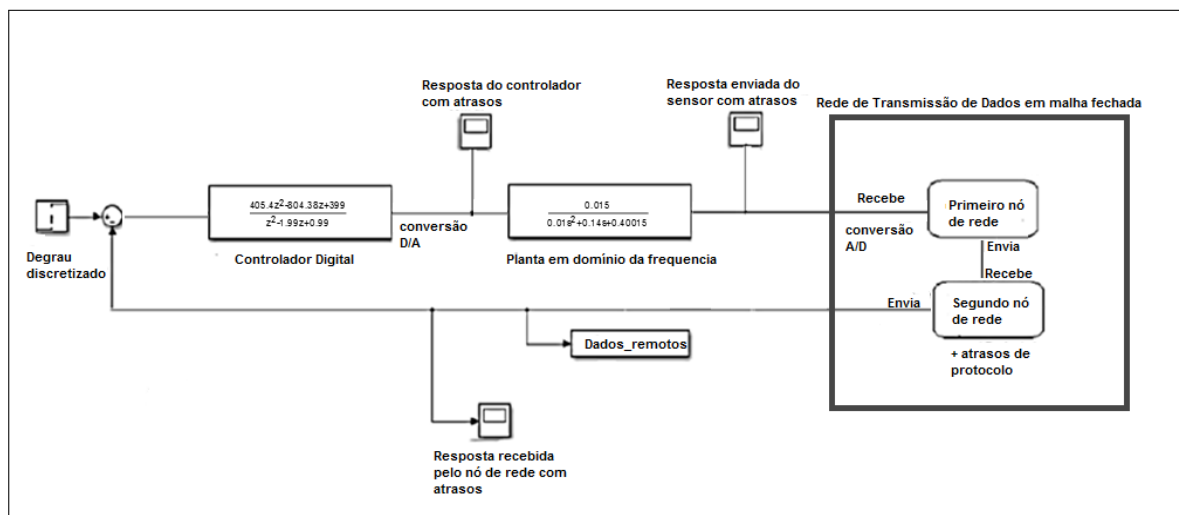


Figura 22: Modelo de um NCS encapsulado pela *interface*
Fonte: Autoria própria

Assim, mediante a ilustrado na Fig.22, a inserção dos atrasos de sinais enviados pela rede foi entre os processos de envio envolvendo a emissão e leitura de mensagens realizadas

remotamente se deram por meio da aplicação do protocolo de comunicação UDP na arquitetura de rede. Sendo assim, o protocolo UDP, utilizado na rede, para a comunicação entre a planta e o controlador, recebeu um sinal com um atraso pré definido e um período de amostragem da rede, que fluiu entre um processo de controle. Foi suposto que o controlador e a planta estavam fisicamente distantes entre si e assim como ocorre em um NCS a malha de controle esteve fechada sob uma rede que realizou a comunicação e interação entre os elementos de controle.

Com a inserção de atrasos invariantes no tempo entre os pacotes de informações trocados pelo controlador e a planta, a função de transferência do sistema simulada esteve conforme descrito na Eq.(41).

$$\frac{Y(s)}{U(s)} = e^{-(\tau_k^{sc} + \tau_k^{ca})s} G(s) \quad (41)$$

Sendo $G(s)$ a função de transferência entre do controlador e da planta e $e^{-(\tau_k^{sc} + \tau_k^{ca})s}$ os erros presentes na espera das informações trocadas entre a planta, o controlador e o sensor, via rede descritos nas Eq.(33-35).

O atuador e o sensor foram embutidos na modelagem da função de transferência da planta, uma vez que o controle se deu por meio do controle da velocidade angular do motor DC a partir do torque gerado. Com isso, a constatação dos efeitos que os atrasos invariantes no tempo geraram em um NCS se deu por meio da inserção e simulação de uma função de transferência de um motor DC com o controle da sua velocidade angular por meio do torque aplicado. Tal compensação foi realizada com um controlador Proporcional, Integrativo e Derivativo ou PID, sintonizado pelo primeiro método de Ziegler e Nichols.

Assim, com a função de transferência do controlador PID e do motor DC, foi possível analisar, via *interface* projetada, o comportamento do sistema quando submetido a atrasos constantes e invariantes ao tempo. Este tipo de controle e planta adotado com base no estudo apresentado por Meshram e Kanojiya (2012), supondo o sistema LIT.

4.3.1 O SISTEMA DE CONTROLE DE UM MOTOR DC COM A INSERÇÃO DE ATRASOS INVARIANTES NO TEMPO

Foram simuladas várias situações com diferentes atrasos durante a emissão das respostas do sensor físico da planta e na rede para transmissão de dados, afim de analisar a comunicação do sistema de controle projetador via rede - NCS.

Os atrasos inseridos compreendiam nos valores progressivos de $1ms$ a $6ms$, além de $4s$. Cada gráfico desta *interface* proposta ilustrou o comportamento de resposta da planta, bem

como a ação de controle desprendida pelo PID e o sinal entregue ao atuador físico da planta. A descrição de tais simulações e os resultados gráficos obtidos estão apresentados em seguida.

O primeiro passo desta simulação foi compreendida na análise do comportamento da planta, controlador e a o sinal entregue para compensação do erro, considerando o valor de $1ms$. O resultado foi retornado via *interface* de acordo com a Fig.23.

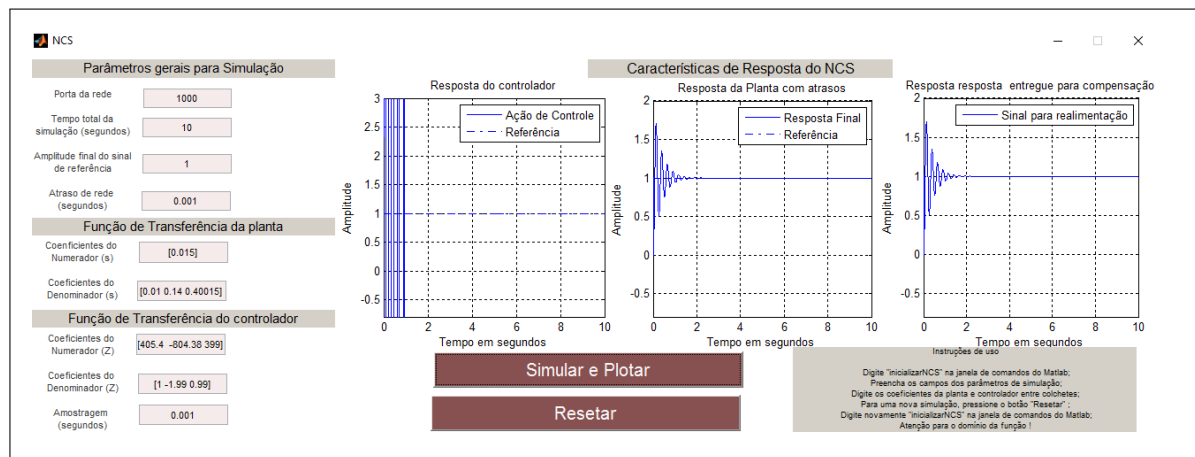


Figura 23: Modelo de um NCS encapsulado pela *interface* com atraso de $1\mu s$
Fonte: Autoria própria

Constatou-se que o desempenho do sistema de controle projetado via rede - NCS, atendeu aos requisitos, apresentando uma resposta bem semelhante ao projeto de controle com resposta referente à Fig.16. Observou-se uma grande ação de controle por parte do PID, compreendido entre os dois primeiros instantes de tempo, para minimização do erro de resposta em relação a trajetória a ser seguida, sendo neste caso um degrau de amplitude unitária.

Inseriu-se então, um tempo de atraso para envio e transmissão, compreendido em $2ms$, conforme a Fig.24.

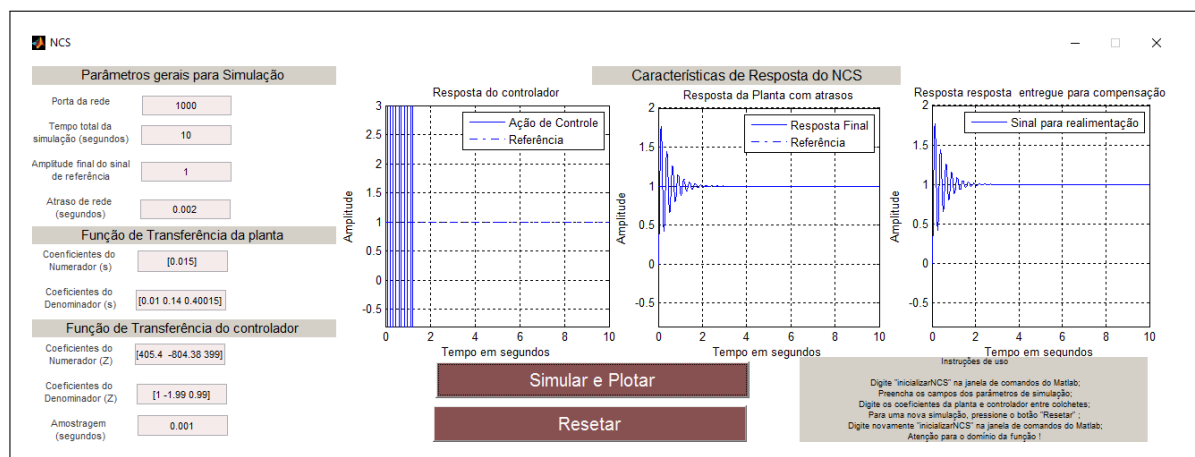


Figura 24: Modelo de um NCS encapsulado pela *interface* com atraso de $2ms$
Fonte: Autoria própria

Por meio da Fig.24, observou-se que a amplitude das ondas durante o regime transitório do sistema apresentou um ligeiro aumento nos três gráficos, sendo que o tempo de estabilização foi aproximado em torno de 2s.

Novamente, inseriu-se um novo atraso de comunicação correspondente a 3ms e realizou-se a simulação. Após o tempo de execução total de 10s do sistema de controle, obteve-se o sistema de controle simulado com o comportamento descrito na a Fig.25.

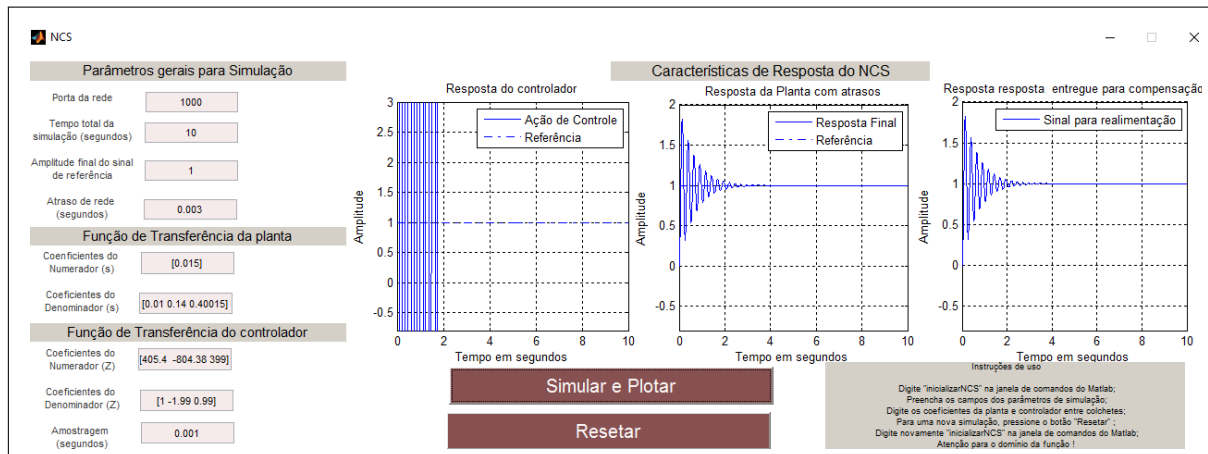


Figura 25: Modelo de um NCS encapsulado pela interface com atraso de 3ms

Fonte: Autoria própria

Observou-se pela Fig.25 que o sistema de controle obteve resposta similar à um sistema sem atraso, porém seu regime transitório excedeu a 2s. Sobre o comportamento da resposta do controlador com atrasos, observou-se um ligeiro aumento da ação de controle em relação ao desempenho apresentado na Fig.24.

Realizou-se uma nova simulação do NCS, inserindo um atraso de 4ms, conforme a Fig.26.

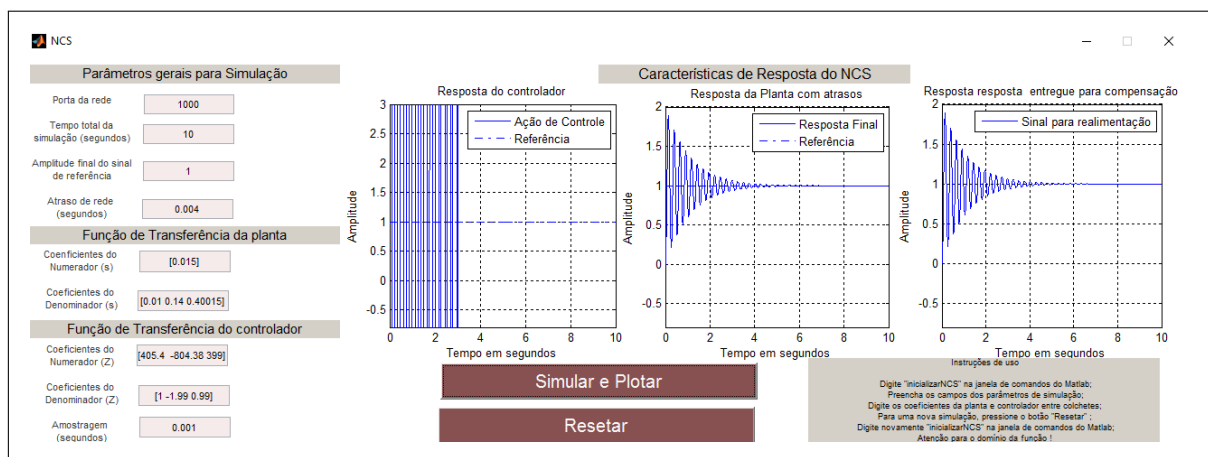


Figura 26: Modelo de um NCS encapsulado pela interface com atraso de 4ms

Fonte: Autoria própria

De acordo com a Fig.26, observou-se que o sistema apresentou um maior esforço do PID, no que se refere ao aumento sobre as frequências de ações de controle. Este foi consequência da inserção do atrasos no sensor da planta, que sofreu alterações significativas; devido aos atrasos e ações de controle, observou-se que o regime transitório da planta foi estendido a 4s porém ainda foi possível seguir o degrau unitário.

Inseriu-se, então, um atraso de 5ms, sendo seus resultados dados de acordo com a Fig.27.

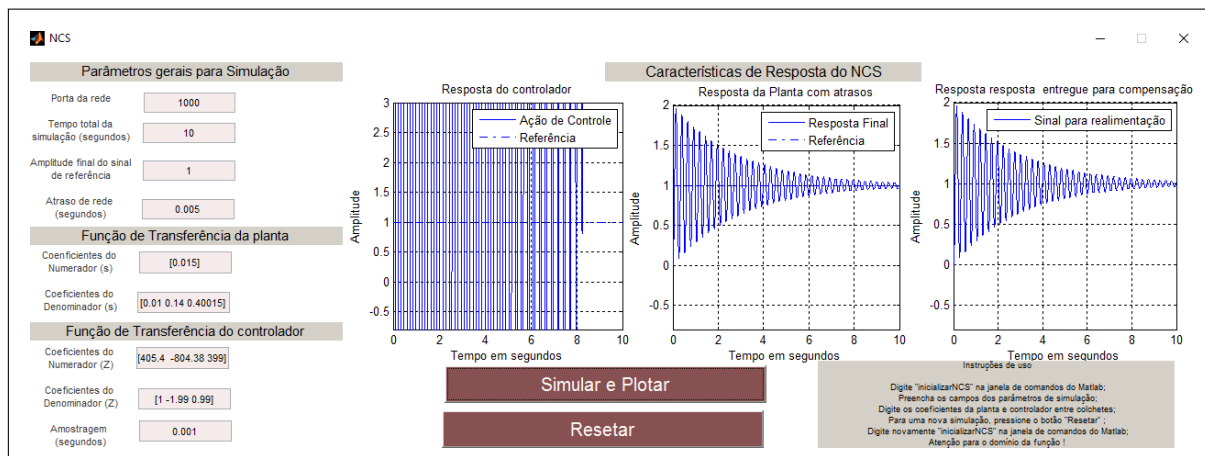


Figura 27: Modelo de um NCS encapsulado pela interface com atraso de 5ms
Fonte: Autoria própria

Observou-se pela Fig.27 que houve um aumento considerável no tempo de estabilização do sistema, sendo este superior a 10s. Além disso a resposta da planta controlada com atrasos pareceu se comportar com amplitudes de ondas cada vez mais atenuadas devido à alta energia desprendida por parte do PID, que fez com que o sinal entregue ao atuador forçasse a estabilização na amplitude unitária, referente ao degrau aplicado.

Buscou-se analisar então o limite para a variação do tempo de estabilização no sistema. Para isso, definiu-se um atraso de 6ms, sendo seus resultados conforme a Fig.28.

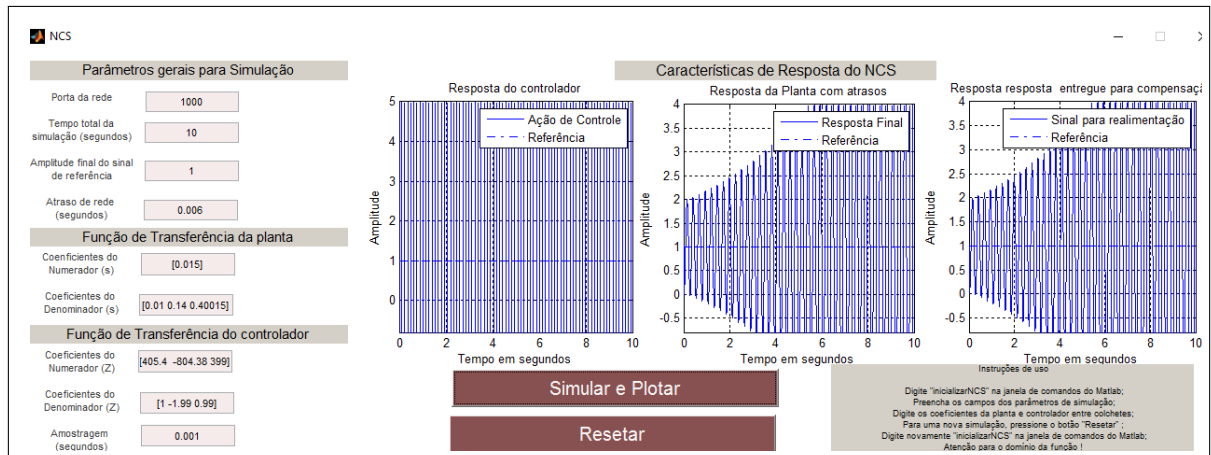


Figura 28: Modelo de um NCS encapsulado pela interface com atraso de 6ms
Fonte: Autoria própria

Verificou-se que o sistema não se estabilizou em 10 segundos, onde seu comportamento aparentou possuir amplitudes cada vez maiores, mesmo com o esforço tendendo ao infinito por parte do PID.

Buscou-se analisar então verificar o comportamento do sistema com a inserção de atrasos maiores. Para isso, definiu-se um atraso de 0.1s, ou seja, 10 vezes maior ao seu atraso inicial, sendo seus resultados conforme a Fig.29.

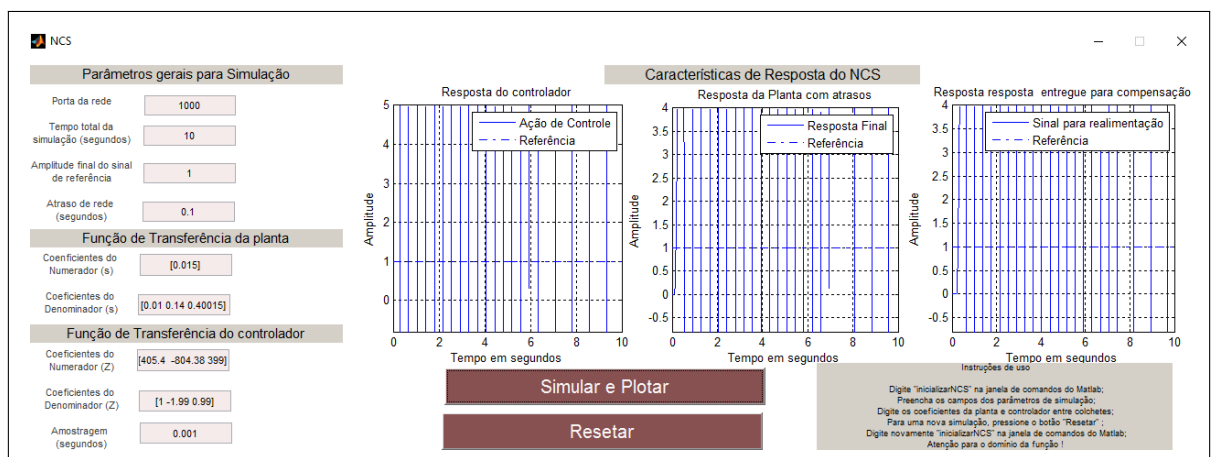


Figura 29: Modelo de um NCS encapsulado pela interface com atraso de 0.1s
Fonte: Autoria própria

Pela Fig.29, observou-se que o sistema apresentou um comportamento instável, pois a amplitude das ondas em seu transitório tendeu ao infinito. Desta forma, o PID, mesmo com ganhos tendendo ao infinito não conseguiu sequer afetar significativamente o sistema de controle implementado.

Estabeleceu-se um atraso de 4s, sendo seu comportamento dado pela Fig.30.

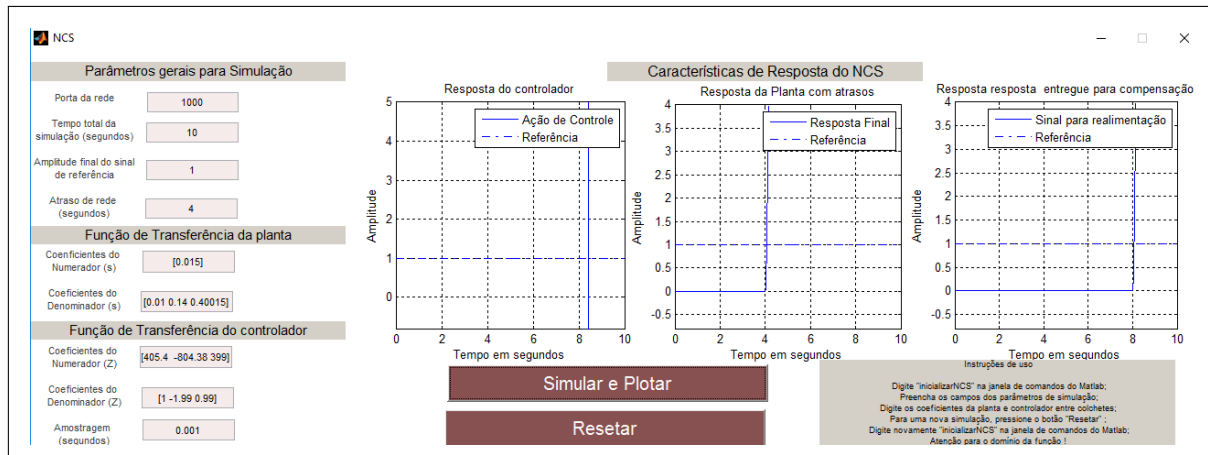


Figura 30: Modelo de um NCS encapsulado pela interface com atraso de 4s
Fonte: Autoria própria

Com a inserção de um atraso na entrega de pacotes de 4s, observou-se que os sinais de respostas dos três gráficos assumiram amplitudes tendendo ao infinito. Desta forma, mesmo com uma energia tendendo ao infinito por parte do PID, não foi possível estabilizar o sistema.

De acordo com o início das respostas ilustradas nos três gráficos da Fig.30, constatou-se que os atrasos foram inseridos na emissão do sinal proveniente do sensor no motor DC e do controlador para o atuador. Notou-se que as ondas correspondentes às respostas da planta com atrasos e a mensagem captada pelo sensor assumiram a respostas controladas iniciaram o seu comportamento com um atraso em aproximadamente 4s e 8s respectivamente, devido ao tempo de espera da rede em 4s para receber a mensagem do sensor na planta somados aos 4s de tempo que a rede demorou para emitir a mensagem do controlador.

A resposta do controlador com atrasos apresentou uma pequena demora para demonstrar uma ação perceptível, devido à sensibilidade característica do controlador PID em perceber alterações e compensá-las. Porém, tal sensibilidade não afetou o sistema durante os testes realizados, uma vez que o desempenho da planta atendeu aos requisitos nas primeiras simulações, apresentando mudanças gráficas significativas somente quando submetida a atrasos constantes e maiores.

5 CONCLUSÃO

Com relação à comunicação remota entre a planta e o controlador, realizada via rede, conclui-se que de fato ela houve, devido a interação entre os elementos encapsulados e as respostas gráficas do sistema de controle. Este encapsulamento se deu por meio da inserção de parâmetros via *interface*, que proporcionou de forma didática, facilitada a visualização dos sinais detectados em cada ponto de nó de rede pré definido. Além disso, a vantagem em se usar a *interface* via o *software* Matlab/Simulink[®] está em seu baixo custo e na grande acessibilidade, uma vez que a maior parte do corpo acadêmico utiliza esta ferramenta para a realização de pesquisas envolvendo controle e automação de sistemas.

O sistema em estudo foi modelado por função de transferência em domínio da frequência e inserido em um sistema de controle digital em malha fechada através da comunicação via rede. As respostas coletadas nos pontos de nó de rede deram origem a três gráficos significativos para projetos em sistemas de controle: o primeiro, intitulado “resposta do controlador com atrasos”, que possui uma resposta levando em conta atrasos de comunicação com o controlador, originado por uma entrega de informação contendo atrasos de entrega de pacotes somados a atrasos de mensagens do sensor pela rede; o segundo chamado “Resposta da planta com atrasos” que possui atrasos de comunicação entre o sensor e a rede e; o terceiro intitulado “Resposta para realimentação com atrasos”, que ilustra a informação recebida via rede com atrasos somados por possíveis perdas, mensagens corrompidas ou prioridades inferiores sobre demais mensagens que também necessitam a transmissão instantânea para posterior atuação sobre o controle da velocidade pelo torque em um motor DC.

Sobre os atrasos aplicados na rede, observou-se que a inserção dos mesmos durante a emissão das mensagens via rede proporcionou a princípio uma resposta estabilizada inferior a 2s durante a inserção do atraso de 1ms. Porém, com a inserção progressiva de atrasos de 2ms e 3ms notou-se que o tempo de estabilização foi sendo prolongado, além de exigir um maior esforço do controlador PID.

Quando foi inserido o atraso de 4ms notou-se um maior esforço do PID para manter a planta seguindo o degrau unitário. Porém, a estabilização do sistema foi ligeiramente maior a

2s.

Com a inclusão do atraso de $5ms$ notou-se que o PID precisou realizar uma ação de controle muito maior em relação às simulações anteriores, sendo o sistema não estabilizável durante o tempo total de simulação, ou seja, $10s$. Esta resposta se manteve em amplitudes ainda maiores quando o sistema de controle foi submetido a $6ms$ mesmo quando o PID realizou uma ação de controle tendendo ao infinito. O mesmo comportamento ocorreu, sendo instável do início da perturbação e tendendo ao infinito as amplitudes de resposta da planta quando o NCS foi submetido a um atraso de transmissão de $0.1s$.

Sobre os deslocamentos dos sinais no tempo, constatou-se que de fato ela ocorreu, sendo nítido o deslocamento do sinal para o lado direito durante as simulações presentes no atraso de $4s$, onde também se constatou que os atrasos foram inseridos somente durante a emissão das mensagens em todos os nós de rede, uma vez que os três comportamentos se deram no mesmo instante de tempo, com um leve retardo de resposta do PID devido a sensibilidade de controle implementado.

É notório que no sistema de controle a sintonização proposta por Ziegler e Nichols não considera atrasos entre as informações trocadas na rede. Desta forma, caso seja necessário o uso de uma rede para trocar informações entre uma planta e um atuador, deve-se considerar os fatores degenerativos de rede tais como atrasos.

Com a inserção dos atrasos via rede, dependendo do valor de espera, o sistema tornou-se instável, não sendo possível assim prever o seu comportamento. Desta forma, conclui-se que os efeitos causados por atrasos na rede tornam o sistema instável. Sendo assim, o motor DC, com sintonização por Ziegler e Nichols, caso fosse inserido em uma rede onde seja necessário durante seu processo compensatório a comunicação remota com um controlador possivelmente poderia ter um desempenho não desejado. Tal afirmação se justifica devido à exposição do sistema de controle frente aos fatores degenerativos presentes em um NCS, tais como prioridade sobre mensagens, perda de pacotes, baixo tempo para aquisição de uma mensagem de um sensor localizado na planta, dentre outros. Tais fatores, desde que superiores a $3ms$ gerariam uma resposta indesejada que posteriormente poderia até se tornar instável, caso a troca de mensagens via rede excedesse à $6ms$.

Vale lembrar que tais efeitos puderam ser visualizados após ser constatado que a sintonização proposta por Ziegler e Nichols não leva em conta atrasos inerentes à comunicação via rede. Assim sendo, técnicas robustas para controle de sistemas LIT podem ser implementadas, afim de minimizar tais efeitos de comunicação e considerando os atrasos como sendo incertezas sobre variações dos parâmetros envolvidos no processo.

REFERÊNCIAS

- BERNARDO, P. C.; KON, F. A importância dos testes automatizados. **Engenharia de Software Magazine**, v. 1, n. 3, p. 54–57, 2008.
- CARVALHO, A. S.; BARONE, D. A. C. Uma plataforma computacional para o ensino de engenharia e controle. **COBENGE, Recife (PE)**, 2009.
- CASTRUCCI, P. **Controle Automático–Teoria e Projeto**. 5^a.ed. Rio de Janeiro: Elsevier, 2006.
- FANTINATO, M. et al. Autotest–um framework reutilizável para a automação de teste funcional de software. **Cad. CPqD Tecnologia**, v. 1, n. 1, p. 119–131, 2005.
- FRANKLIN, G. F.; POWELL, J. D.; WORKMAN, M. L. **Digital Control of Dynamic Systems**. Second. Reading Massachusetts: Addison-Wesley, 1990. (Electrical and Computer Engineering; Control Engineering). A SRL reference.
- GODOY, E. P.; PORTO, A. J. V.; INAMASU, R. Y. Proposta de utilização de hardware-in-the loop no desenvolvimento de sistemas de controle via redes can. **sistema**, v. 5, p. 6, 2008.
- LATHI, B. P. et al. **Linear systems and signals**. [S.l.]: Oxford University Press New York:, 2005.
- MESHARAM, P. M.; KANOJIYA, R. G. Tuning of pid controller using ziegler-nichols method for speed control of dc motor. In: **IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)**. [S.l.: s.n.], 2012. p. 117–122.
- MO, H.; XIE, M. Modelling and analysis of transmission delays and packet dropouts on the reliability of digital networked control systems. In: **2013 Chinese Automation Congress**. [S.l.: s.n.], 2013. p. 187–192.
- NISE, N. S.; SILVA, F. R. da. **Engenharia de sistemas de controle**. [S.l.]: LTC, 2002.
- OGATA, K. **Engenharia de controle moderno**. 5^a.ed. Rio de Janeiro: Prentice Hall, 2010.
- OPPENHEIM, A. V. **Discrete-time signal processing**. [S.l.]: Pearson Education India, 1999.
- OPPENHEIM, A. V. **Sinais e sistemas**. [S.l.]: Prentice-Hall, 2010.
- TANENBAUM, A. S. **Redes de Computadores. São Paulo: Ed.** [S.l.]: Campus, 2003.
- TANG, B.; WANG, J.; ZHANG, Y. A delay-distribution approach to stabilization of networked control systems. **IEEE Transactions on Control of Network Systems**, v. 2, n. 4, p. 382–392, Dec 2015. ISSN 2325-5870.
- TORRES, G. **Redes de computadores**. [S.l.]: Novaterra Editora e Distribuidora LTDA, 2015.
- WALSH, G. C.; YE, H. Scheduling of networked control systems. **IEEE Control Systems**, IEEE, v. 21, n. 1, p. 57–65, 2001.

ZHANG, W.; BRANICKY, M. S.; PHILLIPS, S. M. Stability of networked control systems. **IEEE Control Systems**, IEEE, v. 21, n. 1, p. 84–99, 2001.