



Trabalho de Conclusão de Curso

Proposta de um sistema de supervisão e controle de tanques acoplados

André Luiz de Oliveira Cezário
aloc@ic.ufal.br

Orientador:
Prof. Dr. João Raphael Souza Martins

Maceió, Maio de 2023

André Luiz de Oliveira Cezário

Proposta de um sistema de supervisão e controle de tanques acoplados

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas.

Orientador:

Prof. Dr. João Raphael Souza Martins

Maceió, Maio de 2023

Catlogação na fonte
Universidade Federal de Alagoas
Biblioteca Central
Divisão de Tratamento Técnico
Bibliotecária: Taciana Sousa dos Santos – CRB-4 – 2062

C425p Cezário, André Luiz de Oliveira.
Proposta de um sistema de supervisão e controle de tanques acoplados /
André Luiz de Oliveira Cezário. – 2023.
52 f. : il. color.

Orientador: João Raphael Souza Martins.
Monografia (Trabalho de Conclusão de Curso em Engenharia da
Computação) – Universidade Federal de Alagoas. Instituto de Computação.
Maceió, 2023.

Bibliografia: f. 50-52.

1. Controladores Lógicos Programáveis. 2. Sistemas de controle
(Computação). 3. Sistema de tanques acoplados. I. Título.

CDU: 004

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Engenharia de Computação do Instituto de Computação da Universidade Federal de Alagoas, aprovada pela comissão examinadora que abaixo assina.

Prof. Dr. João Raphael Souza Martins - Orientador
Instituto de Computação
Universidade Federal de Alagoas

Prof. Dr. Ícaro Bezerra Queiroz de Araújo - Examinador
Instituto de Computação
Universidade Federal de Alagoas

Prof. Me. Andressa Martins Oliveira - Examinador
Instituto de Computação
Universidade Federal de Alagoas

Agradecimentos

Agradeço, primeiramente, aos meus pais Gidelma e Luiz, por me ensinarem os princípios e valores que levo sempre comigo.

Aos meus irmãos que sempre estiveram comigo e me ajudaram direto e indiretamente durante toda minha jornada.

Aos meus amigos e colegas de curso, que estiveram comigo e compartilharam sonhos, desejos e frustrações, mostrando que eu não estava sozinho.

Agradeço aos meus professores, coordenadores, diretores, especialistas, mestres e doutores que agregaram conhecimento e experiências durante minha trajetória acadêmica.

Por fim, agradeço a todos que tornaram a realização desse sonho possível.

Resumo

O uso de recursos tecnológicos para o controle de processos e redução dos custos de produção tem se tornado crescentes na automação industrial, sendo cada vez mais indispensável o uso de controladores lógicos programáveis (CLP) e sistemas de controle e aquisição de dados supervisórios (SCADA). Essas tecnologias facilitam o desenvolvimento de sistemas de controle de modo mais rápido e eficiente, seja na automação de processos na indústria ou em plantas de bancadas didáticas no ensino de disciplinas de Engenharia. Contudo, quando se tratam de diferentes fabricantes a comunicação direta é inviabilizada devido ao uso de protocolos de comunicação distintos. Tal barreira é resolvida pelo protocolo de comunicação OPC, que permite a interoperabilidade entre os diferentes sistemas de fornecedores. Neste trabalho foi proposta a integração de uma plataforma de simulação do MATLAB com CLP para o controle de nível de líquidos em um sistema de tanques acoplados, utilizando controle PID em um CLP S7-1200 da Siemens, de uma planta modelada no MATLAB Simulink, no qual a comunicação entre o CLP e o MATLAB foi realizada por um servidor OPC KEPServerEX.V4. O intuito foi ampliar recursos para projetos de controle de plantas didáticas, auxiliando no processo de ensino-aprendizagem de estudantes do curso de Engenharia da Computação. Os resultados mostraram que o sistema controlado via CLP tem uma boa resposta, mantendo o controle mesmo diante as variações de setpoint. Além disso, a IHM desenvolvida fornece uma ferramenta para supervisão em tempo real de um sistema de controle de tanques acoplados, integrando os diferentes softwares e tecnologias em um único processo, podendo ainda ser escalada para outros tipos de plantas.

Palavras-chave: Tanques acoplados; CLP; OPC; MATLAB.

Abstract

The use of technological resources for process control and production cost reduction has become increasingly important in industrial automation, and the use of programmable logic controllers (PLC) and supervisory control and data acquisition (SCADA) systems is becoming more and more indispensable. These technologies facilitate the development of control systems in a faster and more efficient way, whether in the automation of industrial processes or in bench top plants in the teaching of engineering disciplines. However, when different manufacturers are involved, direct communication is not possible due to the use of different communication protocols. This barrier is solved by the OPC communication protocol, which allows interoperability between the different supplier systems. In this work we proposed the integration of a MATLAB simulation platform with a PLC for the control of liquid levels in a coupled tank system, using PID control in a Siemens PLC S7-1200, from a plant modeled in MATLAB Simulink, in which the communication between the PLC and MATLAB was done through an OPC server KEPServerEX.V4. The purpose was to extend resources for didactic plant control projects, helping in the teaching-learning process of Computer Engineering students. The results showed that the system controlled via PLC has a good response, maintaining control even when facing setpoint variations. Furthermore, the developed HMI provides a tool for real-time supervision of a coupled tank control system, integrating different software and technologies in a single process, and can also be scaled to other types of plants.

Keywords: Coupled tanks; PLC; OPC; MATLAB.

Lista de Figuras

| | | |
|----|---|----|
| 1 | Sistema em malha aberta | 16 |
| 2 | Sistema em malha fechada | 16 |
| 3 | Diagrama de blocos de um controlador PID | 18 |
| 4 | Sistema de tanques Quanser | 19 |
| 5 | Sistema de nível de líquido de um tanque | 21 |
| 6 | Sistema de dois tanques acoplados com interação | 22 |
| 7 | Sistema de dois tanques acoplados sem interação | 24 |
| 8 | Caso de uso típico de clientes e servidores OPC | 27 |
| 9 | Objetos criados por um cliente OPC DA | 28 |
| 10 | Arquitetura OPC UA | 30 |
| 11 | (a) Configuração 1 e (b) Configuração 2 | 33 |
| 12 | Modelagem do sistema de tanques acoplados proposta por Glauber R. Leite | 33 |
| 13 | Diagrama de blocos de configuração de controle PID no MATLAB | 34 |
| 14 | Diagrama da rede | 34 |
| 15 | Modelagem do sistema de tanques acoplados com controle via CLP | 35 |
| 16 | Diagrama de blocos de configuração de controle PID no CLP com OPC | 35 |
| 17 | Projeto de servidor OPC no KEPServerEX | 36 |
| 18 | Configuração do NetToPLCsim | 37 |
| 19 | Esquema de controle no PID_Compact | 37 |
| 20 | Controle PID do sistema no CLP | 43 |
| 21 | Especificações do sistema de controle no CLP | 43 |
| 22 | Controle PID no CLP (setpoint fixo) | 44 |
| 23 | Controle PID no MATLAB/Simulink | 45 |
| 24 | IHM para supervisão do sistema de tanques | 46 |
| 25 | Sistema controlado por CLP e MATLAB | 47 |

Lista de Símbolos

| | |
|---------------|-------------------------|
| $G(s)$ | Processo ou planta |
| $u(s)$ | Sinal de controle |
| $y(s)$ | Sinal de saída |
| $r(s)$ | Valor de referência |
| $C(s)$ | Controlador |
| $H(s)$ | Realimentação |
| $e(s)$ | Sinal de erro |
| K | Ganho proporcional |
| T_i | Tempo integrativo |
| T_d | Tempo derivativo |
| R | Resistência |
| C | Capacitância |
| \mathcal{L} | Transformada de Laplace |

Conteúdo

| | |
|---|------------|
| Lista de Figuras | iii |
| Lista de Símbolos | v |
| 1 Introdução | 10 |
| 1.1 Justificativa | 12 |
| 1.2 Objetivos | 13 |
| 1.2.1 Objetivo Geral | 13 |
| 1.2.2 Objetivos Específicos | 13 |
| 1.3 Resultados esperados | 13 |
| 1.4 Estrutura do texto | 14 |
| 2 Fundamentação Teórica | 15 |
| 2.1 Sistemas de controle | 15 |
| 2.1.1 Sistemas em Malha Aberta | 15 |
| 2.1.2 Sistemas em Malha Fechada | 16 |
| 2.2 Controladores | 17 |
| 2.2.1 Controlador PID | 17 |
| 2.3 Modelagem Matemática | 19 |
| 2.3.1 Sistemas de Nível de Líquidos | 20 |
| 2.3.2 Sistemas de um Tanque | 20 |
| 2.3.3 Sistemas de Tanques Acoplados com Interação | 22 |
| 2.3.4 Sistemas de Tanques Acoplados sem Interação | 24 |
| 2.4 OPC | 26 |
| 2.4.1 OPC Classic | 26 |
| 2.4.2 OPC Data Access | 27 |
| 2.4.3 OPC Alarms and Events | 28 |
| 2.4.4 OPC Historical Data Access | 28 |
| 2.4.5 OPC UA | 29 |
| 2.4.6 Arquitetura OPC UA | 30 |
| 3 Materiais e Métodos | 32 |
| 3.1 Sistema | 32 |
| 3.2 Modelagem | 33 |
| 3.3 Comunicação | 36 |
| 3.4 Controle e Supervisão | 37 |

| | | |
|----------|---|-----------|
| 4 | Resultados e Discussões | 42 |
| 4.1 | Controle PID no CLP | 42 |
| 4.2 | Controle PID no MATLAB/Simulink | 44 |
| 4.3 | IHM e Supervisão no CLP | 45 |
| 4.4 | Comparativo entre os Sistemas de Controle | 47 |
| 5 | Conclusão | 49 |
| | Referências bibliográficas | 50 |

1

Introdução

O domínio e conhecimento de novas tecnologias são pré-requisitos importantes para a formação de qualquer profissional, independente da sua área de atuação. E quando se trata de áreas ligadas à tecnologia, essa necessidade se torna cada vez mais frequente, dado que tais recursos tecnológicos podem facilitar a execução de processos e melhorar o desempenho. Na automação industrial, um dos ramos da Engenharia da Computação, é recorrente o uso de tecnologias para o controle de processos que reduzem os custos e melhoram as etapas de produção.

A automação industrial é descrita como todo processo que realiza tarefas e atividades de forma autônoma ou que auxilia o homem em suas tarefas diárias por meio do desenvolvimento de equipamentos de controle de variáveis que possibilita a flexibilização e a comunicação homem/máquina Santos u. a. (2015). Tal processo é denominado de sistema de controle, que consiste em subsistemas construídos com o objetivo de se obter uma saída desejada com um desempenho esperado, dada uma entrada especificada Nise (2010).

Em um sistema de controle à medida que os valores de entrada são alterados, espera-se que os valores de saída também se alterem, alcançando os resultados esperados no menor tempo possível e corrigindo alterações causadas por perturbações externas. Normalmente um controlador é adicionado ao sistema de controle que não atende tais requisitos, sendo os controladores que trabalham com o método proporcional-integral-derivativo (PID) os mais usados.

Um controlador PID utiliza o valor de erro, que é a diferença entre o valor de saída desejado e o valor de saída medido como entrada. Ele trabalha para atingir o valor de ajuste desejado minimizando o erro, ajustando a entrada do sistema. Com intuito de se obter um controle sem erros, o sinal controlado é realimentado e comparado ao sinal de referência. Um sinal de ação proporcional à diferença entre os sinais de entrada e saída é calculado pelo controlador e aplicado ao sistema para corrigir o erro Atiyah u. a. (2019).

O controle de sistemas de automação industrial é amplamente utilizado em controladores lógico programáveis (CLPs), ao invés do controle convencional de relés, pois oferece diversas vantagens, como: fácil instalação, menor custo de produção, maior segurança e qualidade. Os

CLPs suportam protocolos avançados de comunicação industrial e podem trocar informações a longas distâncias. Isso possibilita o controle e monitoramento centralizado de sistemas de automação industrial em longas distâncias e em ambientes de difícil acesso [Åström und Hägglund \(1995\)](#); [Friedland \(1985\)](#).

Um controlador lógico programável é definido pelo IEC (*International Electrotechnical Commission*) como um sistema eletrônico operando digitalmente, projetado para uso em ambiente industrial, que usa uma memória programável para a armazenagem interna de instruções orientadas para o usuário na implementação de funções específicas, tais como lógica, sequencial, temporização, contagem e aritmética, para controlar, através de entradas e saídas digitais ou analógicas, vários tipos de máquinas ou processos.

Diversos trabalhos e pesquisas têm sido realizados com o intuito de automatizar processos e permitir a intercomunicação entre dispositivos CLP e softwares como MATLAB, com base em protocolos de comunicação, como o *Open Platform Communications* (OPC). Tais estudos têm tanto finalidade em aplicações industriais, como na automatização de plantas de sistemas de controle quanto acadêmica, como na criação de bancadas didáticas e sistemas de supervisão e aquisição de dados (SCADA) para atividades práticas em disciplinas de automação industrial.

No trabalho de [Bagal u. a. \(2018\)](#) é apresentado um controle de processo de um motor DC, através de um CLP utilizando SCADA e MATLAB OPC Toolbox, no qual o CLP Mitsubishi FX2N 32MR se comunica com o servidor OPC KEPServerEX.V4 e este com os clientes SCADA e MATLAB, possibilitando que eles tenham acesso aos dados do processo em tempo real. É discutido o uso do servidor OPC DA para estabelecer o controle de tempo entre os sistemas SCADA, MATLAB e CLP, bem como a troca segura e confiável de dados entre dispositivos de diferentes fabricantes por meio do mesmo. A complexidade de controle do sistema não é o foco do estudo, dado que o processo é um motor DC, mas sim fornecer uma metodologia para estabelecer a comunicação entre os sistemas, logo não são explorados aspectos como IHM ou uso mais robusto de controle PID no CLP. O resultado do experimento demonstra a troca de dados eficiente e confiável entre CLP, SCADA e MATLAB.

[Santos u. a. \(2015\)](#) desenvolveram uma bancada didática baseada no CLP CompactLogix da Rockwell para controle de variáveis, com o intuito de melhorar e aperfeiçoar o aprendizado de alunos de Engenharias. Para tal, foi construída uma bancada que permite abordar tópicos de instrumentação e controle industrial através da operação de instrumentos utilizados em aplicações industriais e de malhas de controle de temperatura, nível, pressão e vazão. Não foi descrito o uso de softwares como MATLAB ou servidor OPC, somente a integração entre o CLP e os diferentes processos de controle na construção da bancada, sendo o principal foco fomentar recursos para ensino-aprendizagem dos acadêmicos.

[Silva u. a. \(2021\)](#) criaram um sistema de supervisão e controle automático de nível, temperatura, pressão e vazão de um reservatório, cuja arquitetura do sistema é composta por dois reservatórios, um CLP S7-1200, sensores, atuadores e os softwares Eclipse SCADA e MATLAB, no qual os dados são acessados do sistema supervisório pela rede através do protocolo OPC, con-

trolado por um controlador proporcional integral (PI) e apresentados em uma interface-homem-máquina (IHM). Os resultados de simulação e experimentais, considerando as mudanças na referência de nível de água do reservatório, evidenciaram a efetividade do controle e a sintonia adequada do controlador PI utilizado. O trabalho reforça a importância do uso das plantas didáticas como ferramenta educacional para o fortalecimento do processo de ensino-aprendizagem em disciplinas de controle e automação, assim como uso de protocolos de comunicação, que estão sendo cada vez mais difundidos na indústria.

Neste trabalho é proposto o uso do OPC para integração de uma plataforma de simulação do MATLAB com um CLP S7-1200 da Siemens para o controle de nível de líquidos em um sistema de tanques acoplados. Diferenciando-se dos demais por utilizar o recurso de controle PID automático do TIA, software integrado ao CLP, dispensando a modelagem do controlador no MATLAB e conseqüentemente facilitando o desenvolvimento de plantas no programa. O intuito é utilizar os diferentes equipamentos de laboratório na automação de sistemas de controle, proporcionando uma atividade prática em pequena escala de um processo industrial.

1.1 Justificativa

O uso de bancadas didáticas estimulam o processo de aprendizagem e melhoram as práticas do ensino técnico, visto que os alunos podem vivenciar a experiência de atuar sobre um processo real, que mesmo em menor escala, proporciona um contato mais próximo da indústria, do que uma aula teórica.

Conforme destacado por Santos *u. a.* (2015) plantas didáticas são plataformas tecnológicas e constituem alternativas práticas no ensino de controle de processos. A junção de tais sistemas didáticos com um CLP busca integrar o ensino na prática de disciplinas distintas dos cursos de Engenharia e áreas afins, favorecendo a interdisciplinaridade e possibilitando o contato com situações mais próximas da realidade.

Por outro lado, sistemas de automação industrial e controle de processos de diferentes fabricantes utilizam protocolos de comunicação distintos, na maioria das vezes proprietários, tornando uma desvantagem a sua utilização e dificultando a interoperabilidade entre os sistemas de supervisão, controle e automação de diferentes marcas. Tal fato ocorre na bancada de tanques acoplados da Quanser, disponível no laboratório de Sistema de Controle do IC UFAL, que não está integrada diretamente ao CLP da Siemens.

Nesse sentido, a implementação do padrão OPC para a comunicação entre uma planta no MATLAB e o controle PID em um CLP, fornece metodologias úteis e ágeis para parametrizar e monitorar processos entre diferentes sistemas, demonstrando a viabilidade e vantagens da tecnologia para maximizar a flexibilidade de processos industriais, além de promover seu conhecimento para estudantes e futuros profissionais.

1.2 Objetivos

1.2.1 Objetivo Geral

Integração de uma plataforma de simulação do MATLAB com CLP via OPC, possibilitando a interoperabilidade dos sistemas em um processo de automação industrial, ampliando recursos para projetos de controle de plantas didáticas e auxiliando no processo de aprendizagem e formação de estudantes do curso de Engenharia da Computação, através de uma experiência real do conteúdo teórico visto em sala com o prático em laboratório.

1.2.2 Objetivos Específicos

- Determinar as especificações funcionais e características dos sistemas a serem utilizados: OPC, MATLAB e e CLP.
- Modelar matematicamente um sistema de nível de dois tanques acoplados e simular no MATLAB/Simulink.
- Desenvolver um projeto de automação para um sistema de nível, de modo a proporcionar a supervisão, incluindo a aquisição de dados via MATLAB - OPC e controle PID do processo pelo CLP da Siemens.
- Fazer testes de validação do sistema para os dados provenientes da planta, a fim de verificar a eficiência do controle do processo, na automação do nível de líquidos dos tanques.

1.3 Resultados esperados

Espera-se com este trabalho proporcionar o contato de alunos do curso de Engenharia da Computação com tecnologias e equipamentos encontrados na indústria de forma integrada, comunicando modelos de plantas desenvolvidas no MATLAB com um CLP industrial, via OPC. Tal contato facilita a aquisição de conhecimento de uma nova tecnologia que não é tão abordada durante o curso, uma vez que o OPC tem diversas aplicações e são muito úteis para permitir a interoperabilidade entre sistemas e flexibilidade de processos industriais.

Além disso, almeja-se que com os resultados deste trabalho seja possível modelar e integrar outros sistemas, além do controle de nível em tanques acoplados, possibilitando a aquisição de conhecimento necessária para ampliar o uso de equipamentos do laboratório de forma mais integrada e interdisciplinar, promovendo uma melhor experiência de aprendizado durante as aulas experimentais, bem como dispendo de mais recursos didáticos a serem utilizados pelos docentes.

1.4 Estrutura do texto

A estrutura deste trabalho divide-se em: O Capítulo 2 trata da fundamentação teórica, abordando as principais teorias e modelagens para sustentar os métodos utilizados no decorrer do texto; O Capítulo 3 apresenta os materiais e métodos necessários para a reprodução do trabalho, assim como a metodologia adotada para alcançar os resultados; O Capítulo 4 demonstra os resultados obtidos ao produzir o modelo proposto e discute as suas principais implicações; Por fim, no Capítulo 5 são feitas as considerações finais e propostas de trabalhos futuros.

2

Fundamentação Teórica

2.1 Sistemas de controle

Um sistema de controle consiste em subsistemas e processos construídos com o objetivo de se obter uma saída desejada com um desempenho esperado, dada uma entrada especificada [Nise \(2010\)](#), sem a necessidade de intervenção humana.

Conforme destacado por [Bazanella \(2005\)](#), o sinal aplicado na entrada de um processo $G(s)$ é chamado de sinal de controle $u(s)$ ou variável manipulada, pois seu valor pode ser manipulado. O sinal de saída $y(s)$ do processo é chamado de variável controlada ou variável de processo, e é a variável cujo comportamento se deseja controlar. Em geral, um sistema controlado está suscetível a ação de eventos externos, chamados de perturbações.

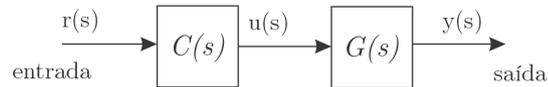
Logo, o objetivo de um sistema de controle é que à medida que os valores de entrada mudem, os valores de saída também mudem, atingindo os resultados esperados o mais rápido possível, e que as alterações causadas por perturbações ao sistema, sejam auto-corrigidas pelo mesmo. Geralmente um controlador $C(s)$ é adicionado na entrada do sistema que não atende tais requisitos, com o intuito de garantir estabilidade do sinal de saída. Esse sinal aplicado na entrada do controlador é chamado de valor de referência $r(s)$ ou setpoint.

2.1.1 Sistemas em Malha Aberta

Sistemas de controle de malha aberta são aqueles nos quais o sinal de saída não exerce nenhuma ação de controle no sistema, isto é, em um sistema de controle de malha aberta, o sinal de saída não é medido nem realimentado para comparação com a entrada [Ogata \(2010\)](#). Sistemas assim, em geral, são simples e baratos, visto que não envolvem equipamentos para a medição e/ou determinação do sinal de controle. Por outro lado, não podem ser utilizados na presença de perturbações, pois não estão preparados para lidar com esses eventos. Na Figura 1 abaixo é

ilustrado um sistema em malha aberta, como pode ser observado a saída não é medida ou realimentada para comparação com o valor da referência $r(s)$, desse modo cada entrada corresponde a uma condição fixa de operação.

Figura 1: Sistema em malha aberta



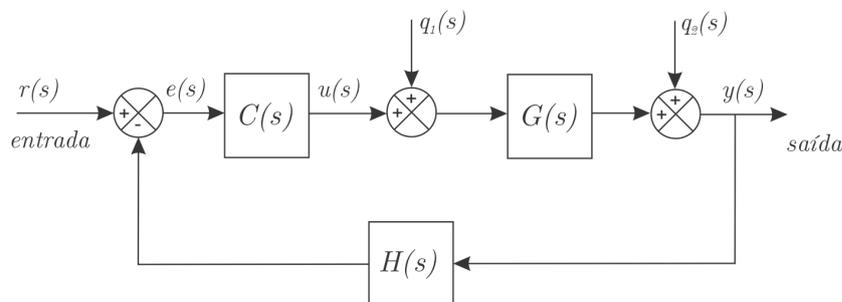
Fonte: Elaborado pelo próprio autor

2.1.2 Sistemas em Malha Fechada

Sistemas de controle de malha fechada são basicamente processos que possuem realimentação, em tais sistemas informações sobre a saída são utilizadas para determinar o sinal de controle que deve ser aplicado ao processo a cada instante de tempo. Isso é possível, a partir da medição do sinal de saída através de um sensor $H(s)$ que irá converter a grandeza física a ser controlada em um sinal.

Com o objetivo de tornar o sistema mais preciso e de fazer com que ele reaja à perturbações externas, o sinal de saída $y(s)$ é comparado com um sinal de referência $r(s)$ e a diferença entre esses dois sinais, chamada de sinal de erro $e(s)$, é utilizada para determinar o sinal de controle $u(s)$ que deve ser aplicado ao processo. Assim, o sinal de controle é determinado de forma a corrigir o desvio entre a saída e o sinal de referência. Esse processo que envolve a medida do sinal de saída e sua comparação com o sinal de referência recebe o nome de realimentação. O dispositivo que utiliza o sinal de erro para determinar ou calcular o sinal de controle a ser aplicado à planta é chamado de controlador $C(s)$ [Bazanella \(2005\)](#). Na Figura 2 é representado um sistema de controle em malha fechada:

Figura 2: Sistema em malha fechada



Fonte: Elaborado pelo próprio autor

onde $q_1(s)$ e $q_2(s)$ são as perturbações atuando respectivamente na entrada e na saída do processo.

2.2 Controladores

Um controlador automático compara o valor real de saída da planta com o valor desejado, determina o desvio e produz um sinal de controle que reduzirá o desvio a zero ou a um valor pequeno, [Ogata \(2010\)](#). O modo que o controlador produz o sinal de controle é chamado ação de controle. Dentre seus principais tipos, temos:

- **Ação de controle de duas posições ou on-off:** Nessa ação de controle, o elemento atuante tem apenas duas posições fixas, normalmente, on e off. É relativamente mais simples e barato, sendo muito utilizado em sistemas de controle domésticos e industriais por esse motivo. Pode ser descrito matematicamente, como:

$$u(t) = \begin{cases} \bar{u} & \text{se } e(t) > 0 \\ \underline{u} & \text{se } e(t) < 0 \end{cases} \quad (1)$$

no qual o sinal de controle $u(t)$ pode assumir apenas dois valores, conforme o erro $e(t)$ seja positivo ou negativo.

- **Ação proporcional:** Em controlador com ação proporcional, o sinal de controle aplicado a cada instante à planta é proporcional à amplitude do valor do sinal de erro:

$$u(t) = Ke(t) \quad (2)$$

- **Ação integral:** A ação de controle integral consiste em aplicar um sinal de controle $u(t)$ proporcional à integral do sinal $e(t)$:

$$u(t) = \frac{1}{T_i} \int_0^t e(\tau) d\tau \quad (3)$$

onde T_i é chamado de tempo integral.

- **Ação derivativa:** Em um controlador com ação derivativa, o valor da saída $u(t)$ do controlador é modificado a uma taxa de variação proporcional ao sinal de erro $e(t)$:

$$u(t) = T_d \frac{de(t)}{dt} \quad (4)$$

onde T_d é chamado de tempo derivativo.

2.2.1 Controlador PID

O controlador PID é resultado da combinação das ações proporcional, integral e derivativa para gerar um único sinal de controle, denominado controlador proporcional-integral-derivativo. Segundo [Bazanella \(2005\)](#), o controlador PID é capaz de eliminar erros de regime permanente,

através da ação integral, e antecipar o comportamento do processo, com a ação derivativa. A ação proporcional, por sua vez, faz com que o sistema reaja ao erro presente, conferindo-lhe uma reação imediata e, portanto, rápida às perturbações.

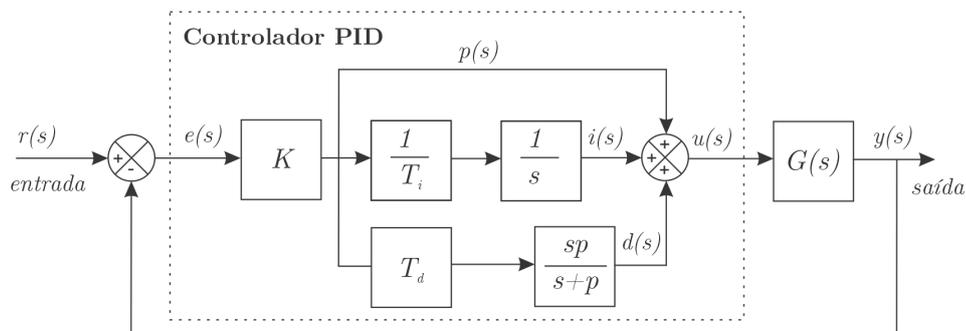
O sinal de controle gerado pelo controlador PID pode ser expresso como:

$$u(t) = K \left(e(t) + \frac{1}{T_i} \int_0^t e(\tau) d\tau + T_d \frac{de(t)}{dt} \right) \quad (5)$$

onde K é o ganho proporcional, T_i é o tempo integrativo e T_d é o tempo derivativo.

A Figura 3 mostra que o controlador PID pode ser visto como a soma de três sinais obtidos a partir do sinal de erro ($p(s), i(s), d(s)$):

Figura 3: Diagrama de blocos de um controlador PID



Fonte: Adaptado de Bazanella (2005)

A função de transferência de um controlador PID é dada por:

$$G_c(s) = K_P + \frac{K_I}{s} + K_D s \quad (6)$$

onde $K_P = K$, $K_I = K \frac{1}{T_i}$ e $K_D = K T_D$

A função de transferência do termo derivativo é representada por:

$$G_d(s) = \frac{K_D s}{\tau_d s + 1} \quad (7)$$

porém, usualmente, τ_d é muito menor que as constantes de tempo do próprio sistema, podendo assim ser desprezada Dorf und Bishop (2001).

Se fizermos $K_D = 0$, teremos um controlador proporcional e integral (PI):

$$G_c(s) = K_p + \frac{K_I}{s} \quad (8)$$

Por outro lado, se $K_I = 0$, obtemos um controlador proporcional derivativo (PD):

$$G_c(s) = K_p + K_D s \quad (9)$$

Muitos processos industriais são controlados usando controladores PID. Sua popularidade pode ser atribuída ao seu bom desempenho em uma ampla faixa de condições de operação, assim como sua simplicidade funcional que permite aos engenheiros operá-los de um modo simples e direto [Dorf und Bishop \(2001\)](#).

Controladores PID podem ser encontrados em ambiente industrial sob a forma de equipamentos single-loop digitais ou analógicos, que são equipamentos dedicados especificamente à execução de um algoritmo PID em uma malha de controle; como funções programadas em controladores lógicos programáveis (CLPs) ou em blocos funcionais que são executados em sistemas digitais de controle distribuídos (SDCDs); e em barramentos industriais [Bazanella \(2005\)](#).

Dentre os processos que podem ser controlados por um controlador PID estão os de sistemas de nível de líquidos, sendo o de tanques acoplados o mais comum. Na próxima seção será feito um estudo da modelagem matemática desses sistemas, a fim de compreender melhor o seu funcionamento e possibilitar a construção de modelos em softwares de simulação, como MATLAB.

2.3 Modelagem Matemática

O sistema de controle objeto de estudo neste trabalho é o de tanques acoplados da Quanser, no qual a planta é composta por uma bomba com um reservatório de água e dois tanques acoplados (Figura 4), o líquido é bombeado do reservatório para o primeiro tanque, flui para o segundo tanque e por fim para o reservatório novamente, completando o ciclo. Nas próximas subseções serão apresentados alguns tipos de sistemas de nível de líquidos, como monotanque e de tanques acoplados com e sem interação, sendo este último o sistema a ser controlado.

Figura 4: Sistema de tanques Quanser



Fonte: Quanser

2.3.1 Sistemas de Nível de Líquidos

Em geral, processos industriais de fluxo de líquidos estão envolvidos ao longo de tubos de conexão e reservatórios, sendo normalmente caracterizado por ser turbulento e não laminar. A análise desses sistemas, é dividida em fluxo laminar e fluxo turbulento, de acordo com o valor do número de *Reynolds*, que representa o regime de escoamento de um fluido sobre uma superfície. Se este número estiver entre 3.000 - 4.000, o sistema será turbulento, já se o valor for menor que 2.000, o fluxo será laminar.

No caso laminar, o fluxo ocorre em linhas de escoamento, sem turbulência. Sistemas que envolvem esse tipo de fluxo podem ser representados por equações diferenciais lineares. Por outro lado, sistemas que envolvem fluxo turbulento são frequentemente representados por equações diferenciais não lineares. No entanto, se a região de operação for limitada, essas equações diferenciais não lineares podem ser linearizadas [Ogata \(2010\)](#).

A modelagem matemática de sistemas de nível de líquido leva em consideração dois conceitos importantes, os de resistência e capacitância.

Consideremos o fluxo ao longo de uma tubulação curta, que conecta dois tanques:

- A resistência R ao fluxo de líquido é definida como a variação da diferença de nível entre os dois tanques necessária para causar variação unitária na vazão:

$$R = \frac{\text{variação na diferença de nível, m}}{\text{variação na vazão em volume, m}^3/\text{s}} \quad (10)$$

- A capacitância C do tanque é definida como a variação na quantidade de líquido armazenado necessária para causar uma mudança unitária no potencial, que nesse caso é a altura.

$$C = \frac{\text{variação na quantidade de líquido armazenado, m}^3}{\text{variação na altura, m}} \quad (11)$$

2.3.2 Sistemas de um Tanque

A Figura 5 ilustra um sistema de nível de líquido de um tanque, nesse sistema o líquido flui em uma válvula de restrição, na lateral do reservatório. Se esse fluxo for laminar, a relação entre a vazão e a altura do nível em regime permanente será dada por:

\bar{Q} é a vazão em volume em regime permanente (antes de ocorrer alguma variação) (m^3/s)

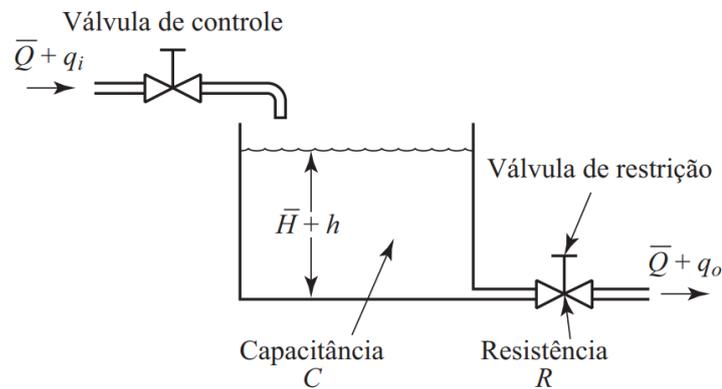
q_i é o desvio da vazão de entrada em relação a seu valor de regime permanente (m^3/s)

q_o é o desvio da vazão de saída em relação a seu valor de regime permanente (m^3/s)

\bar{H} é a altura do nível em regime permanente (antes que ocorra alguma variação) (m)

h é o pequeno desvio de nível a partir de seu valor de regime permanente (m)

Figura 5: Sistema de nível de líquido de um tanque



Fonte: Ogata (2010)

Considerando que o sistema seja linear ou linearizado e que o fluxo de entrada menos o fluxo de saída durante um pequeno intervalo de tempo dt é igual à quantidade adicional armazenada no reservatório, a equação diferencial desse sistema é dada por:

$$Cdh = (q_i - q_o) dt \quad (12)$$

Pela definição de resistência, a relação entre q_o e h é:

$$q_o = \frac{h}{R} \quad (13)$$

Para um valor constante de R , a equação diferencial desse sistema é:

$$RC \frac{dh}{dt} + h = Rq_i \quad (14)$$

Aplicando a Transformada de Laplace em ambos os membros da equação e considerando as condições iniciais nulas, temos:

$$(RCs + 1)H(s) = RQ_i(s) \quad (15)$$

onde $H(s) = \mathcal{L}[h]$ e $Q_i(s) = \mathcal{L}[q_i]$

Se q_i for considerada a entrada e h , a saída, a função de transferência do sistema é:

$$\frac{H(s)}{Q_i(s)} = \frac{R}{RCs + 1} \quad (16)$$

Contudo, se q_o for tida como a saída e a entrada permanecer a mesma, a função de transferência será:

$$\frac{Q_o(s)}{Q_i(s)} = \frac{1}{RCs + 1} \quad (17)$$

Que nos dá:

$$Q_0(s) = \frac{1}{R}H(s) \quad (18)$$

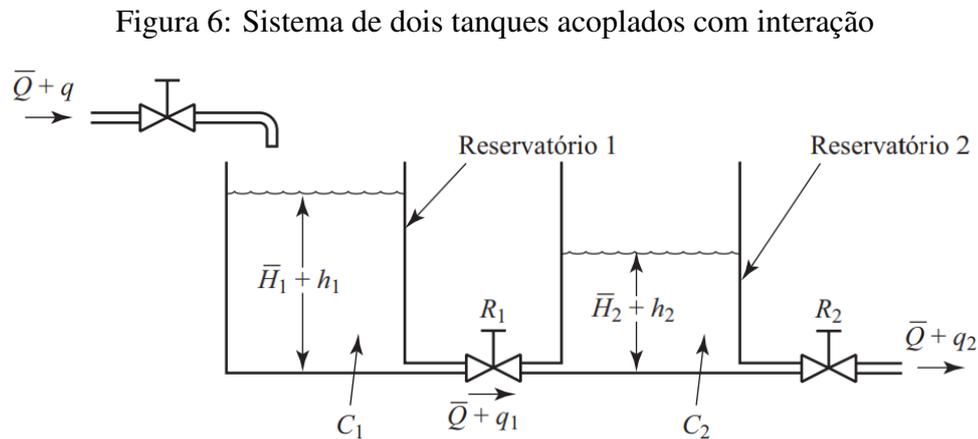
2.3.3 Sistemas de Tanques Acoplados com Interação

Considerando o sistema da Figura 6, os dois reservatórios interagem. Em tanques acoplados os estados do sistema são o nível no tanque 1 e o nível no tanque 2. Assim, a função de transferência do sistema não é o produto das funções de transferência de primeira ordem Ogata (2010).

\bar{Q} é a vazão em volume em regime permanente

\bar{H}_1 é o nível de líquido do reservatório 1 em regime permanente

\bar{H}_2 é o nível de líquido do reservatório 2 em regime permanente



Fonte: Ogata (2010)

Assumindo apenas pequenas variações das variáveis a partir dos valores de regime permanente, temos:

$$\frac{h_1 - h_2}{R_1} = q_1 \quad (19)$$

$$C_1 \frac{dh_1}{dt} = q - q_1 \quad (20)$$

$$\frac{h_2}{R_2} = q_2 \quad (21)$$

$$C_2 \frac{dh_2}{dt} = q_1 - q_2 \quad (22)$$

Logo, para o tanque 1

$$q - \frac{h_1 - h_2}{R_1} = C_1 \frac{dh_1}{dt} \quad (23)$$

$$R_1 q - h_1 + h_2 = C_1 R_1 \frac{dh_1}{dt} \quad (24)$$

A constante de tempo do tanque 1 é dada por:

$$\tau_1 = C_1 R_1 \quad (25)$$

Assim,

$$R_1 q = \tau_1 \frac{dh_1}{dt} + h_1 - h_2 \quad (26)$$

Aplicando a Transformada de Laplace em ambos os membros da equação, temos:

$$R_1 Q(s) = \tau_1 s H_1(s) + H_1(s) - H_2(s) \quad (27)$$

$$R_1 Q(s) = (\tau_1 s + 1) H_1(s) - H_2(s) \quad (28)$$

Para o tanque 2,

$$\frac{h_1 - h_2}{R_1} - \frac{h_2}{R_2} = C_2 \frac{dh_2}{dt} \quad (29)$$

$$C_2 R_2 \frac{dh_2}{dt} + h_2 + \frac{R_2}{R_1} h_2 = \frac{R_2}{R_1} h_1 \quad (30)$$

A constante de tempo do tanque 2 é dada por:

$$\tau_2 = C_2 R_2 \quad (31)$$

Portanto,

$$\tau_2 \frac{dh_2}{dt} + h_2 + \frac{R_2}{R_1} h_2 = \frac{R_2}{R_1} h_1 \quad (32)$$

Tomando a transformada de Laplace em ambos os lados:

$$\tau_2 s H_2(s) + H_2(s) + \frac{R_2}{R_1} H_2(s) = \frac{R_2}{R_1} H_1(s) \quad (33)$$

$$\left(\tau_2 s + 1 + \frac{R_2}{R_1} \right) H_2(s) = \frac{R_2}{R_1} H_1(s) \quad (34)$$

Isolando $H_1(s)$ na equação (28) e substituindo em (34), fazendo algumas operações algébricas, temos:

$$\left(\tau_1 \tau_2 s^2 + \tau_1 s + \frac{R_2}{R_1} \tau_1 s + \tau_2 s + 1 \right) H_2(s) = R_2 Q(s) \quad (35)$$

Da equação (25), vemos que (35) pode ser reescrita como:

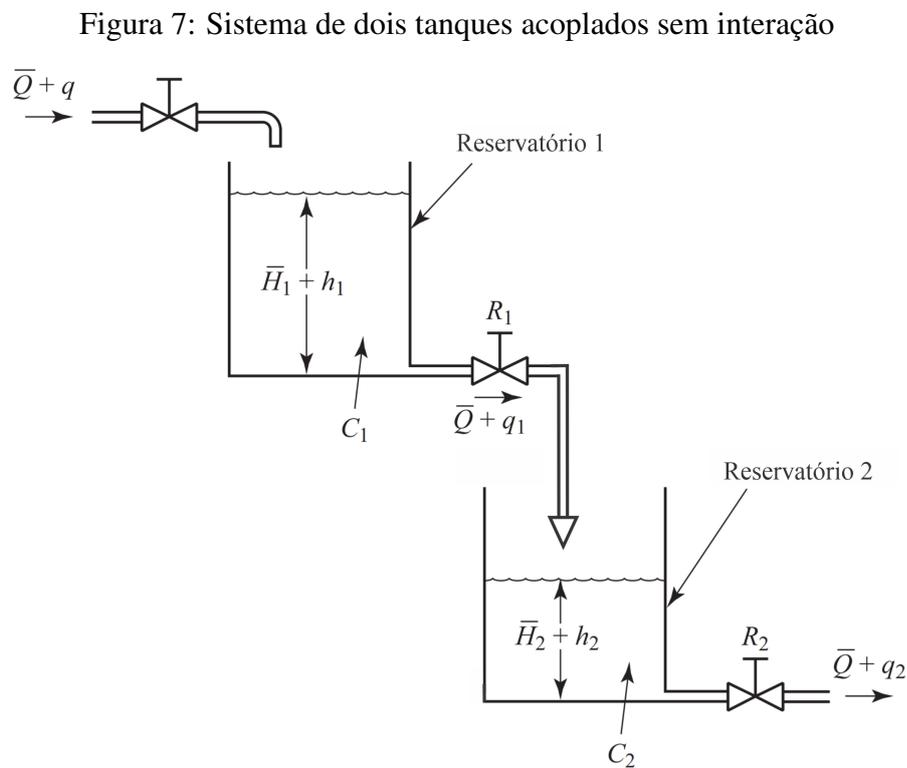
$$(\tau_1 \tau_2 s^2 + \tau_1 s + \tau_2 s + C_1 R_2 s + 1) H_2(s) = R_2 Q(s) \quad (36)$$

Considerando que $Q(s)$ é a entrada e $H_2(s)$, a saída, a função de transferência do sistema será:

$$\frac{H_2(s)}{Q(s)} = \frac{R_2}{\tau_1 \tau_2 s^2 + (\tau_1 + C_1 R_2 s + \tau_2) s + 1} \quad (37)$$

2.3.4 Sistemas de Tanques Acoplados sem Interação

A Figura 7 ilustra um sistema sem interação com fluxo de entrada $q(t)$ e fluxo de saída de produto $q_2(t)$, com objetivo de manter os níveis dos tanques 1 e 2 nas alturas h_1 e h_2 , respectivamente.



Fonte: Adaptado de Ogata (2010)

Aplicando a transferência de massa ao redor de cada tanque, em torno do primeiro tanque, temos:

$$C_1 \frac{dh_1}{dt} = q - q_1 \quad (38)$$

A resistência da válvula R_1 e a vazão de saída q_1 estão relacionadas por:

$$\frac{h_1}{R_1} = q_1 \quad (39)$$

Das equações (38) e (39):

$$q - \frac{h_1}{R_1} = C_1 \frac{dh_1}{dt} \quad (40)$$

Em estado estacionário,

$$q_s - \frac{h_{1s}}{R_1} = 0 \quad (41)$$

Logo, o desvio do estado estacionário é dado por [Sharma \(2018\)](#):

$$(q - q_s) - \frac{h_1 - h_{1s}}{R_1} = C_1 \frac{dh_1}{dt} \quad (42)$$

Reescrevendo a equação acima, como:

$$Q - \frac{H_1}{R_1} = C_1 \frac{dH_1}{dt} \quad (43)$$

onde $Q = q - q_s$ e $H_1 = h_1 - h_{1s}$,

Aplicando a Transformada de Laplace em (43),

$$Q(s) = C_1 s H_1(s) + \frac{H_1(s)}{R_1} \quad (44)$$

$$Q(s) = \frac{R_1 C_1 s + 1}{R_1} H_1 \quad (45)$$

Assim, a função de transferência do tanque 1 é dada por:

$$\frac{H_1(s)}{Q(s)} = \frac{R_1}{1 + \tau_1 s} \quad (46)$$

Analogamente, ao tanque 1 a função de transferência do tanque 2 é:

$$\frac{H_2(s)}{Q_1(s)} = \frac{R_2}{1 + \tau_2 s} \quad (47)$$

Aplicando a Transformada de Laplace em (39),

$$R_1 Q_1(s) = H_1(s) \quad (48)$$

Substituindo $H_1(s)$ acima, na equação (46), temos:

$$\frac{Q_1(s)}{Q(s)} = \frac{1}{1 + \tau_1 s} \quad (49)$$

Combinando as equações (47) e (49), obtemos a função de transferência do sistema:

$$\frac{H_2(s)}{Q(s)} = \frac{R_2}{(1 + s\tau_1)(1 + s\tau_2)} \quad (50)$$

2.4 OPC

Desenvolvida por volta de 1996, a primeira versão funcional do OPC (*Open Platform Communications*) é resultado de um consórcio de empresas fabricantes de sistemas de automação industrial, cujo o objetivo era promover a interoperabilidade dos dispositivos das redes industriais e de automação, facilitando a integração entre informações coletadas em chão de fábrica e setores administrativos [OPCFoundation \(2017\)](#).

Baseado em uma série de padronizações OLE/DCOM (*Object Linking and Embedding/Distributed Component Object Model*), no qual o OLE permite a conexão entre objetos de dados e o DCOM a troca de dados entre aplicativos e dispositivos, o OPC é o primeiro componente desenvolvido para atender necessidades específicas do domínio de automação, padronizando mecanismos de comunicação entre equipamentos de campo e banco de dados em centros de controle [Dang \(2007\)](#).

A finalidade do OPC é focar principalmente na troca de dados “brutos”, isto é, na leitura e gravação de dados entre um aplicativo e um dispositivo de controle de processo, de modo seguro e eficiente. Em outras palavras, o OPC pode ser considerado uma janela através da qual os dados da planta podem ser observados. E isso é possível, através da especificação de regras para troca de dados, implementados pelos fornecedores em interfaces de software.

2.4.1 OPC Classic

A OPC Foundation, organização sem fins lucrativos responsável pela elaboração e indicação de padrões para uso e desenvolvimento da tecnologia OPC, tem definido nos últimos anos uma série de interfaces de software para padronizar o fluxo de informações desde o nível de processamento até o nível de gerenciamento. Dentre os principais casos de uso estão as interfaces para aplicações de automação industrial, como IHMs (Interface Homem Máquina) e sistemas SCADA (Sistemas de Supervisão e Aquisição de Dados), atuando no consumo de dados de dispositivos e fornecendo dados, eventos e históricos para aplicativos de gerenciamento [Mahnke u. a. \(2009\)](#).

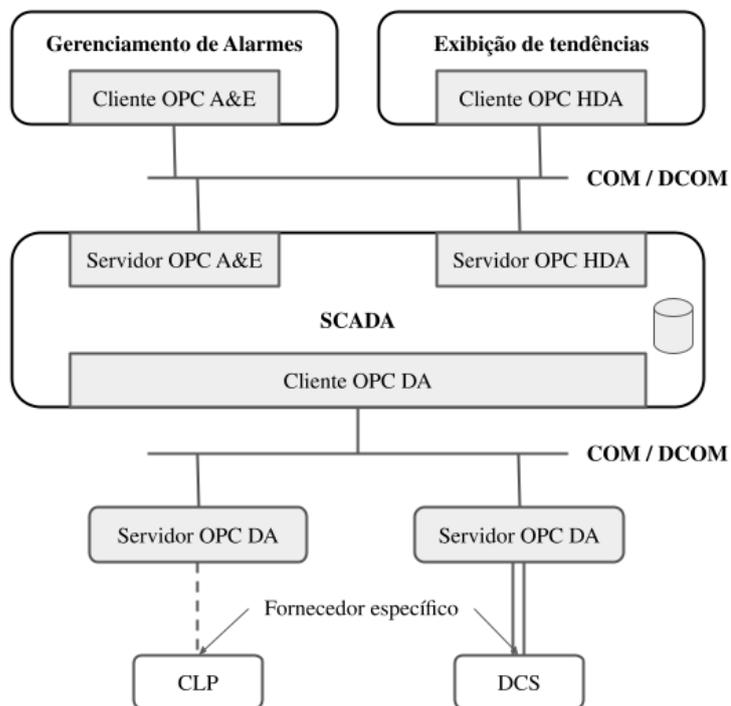
Com base nas necessidades das aplicações industriais foram desenvolvidas três especificações OPC: *Data Access* (DA), *Alarm & Events* (A&E) e *Historical Data Access* (HDA). DA fornece uma interface para dados de processo atualizados, AE define uma interface para monitoramento e reconhecimento de alarmes e eventos, e HDA descreve funções para acesso de dados arquivados.

O OPC clássico utiliza uma abordagem cliente-servidor para a troca de informações. Na qual, um servidor OPC encapsula a fonte de informações do processo como um dispositivo e fornece as informações por meio de sua interface.

O cliente OPC se conecta ao servidor OPC e pode acessar e consumir os dados disponíveis. Os aplicativos que consomem e disponibilizam dados podem ser cliente e servidor [Mahnke u. a.](#)

(2009). A Figura 8 ilustra um caso de uso típico de clientes e servidores OPC.

Figura 8: Caso de uso típico de clientes e servidores OPC



Fonte: Adaptado de [Mahnke u. a. \(2009\)](#)

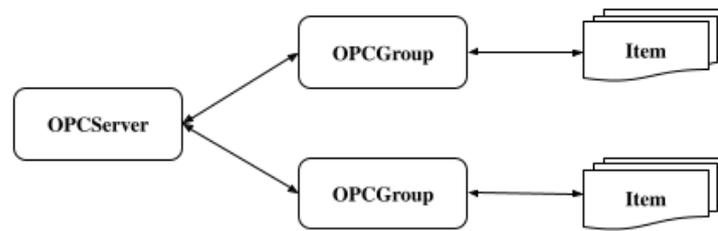
2.4.2 OPC Data Access

O *OPC Data Access* permite a leitura, gravação e monitoramento de variáveis contendo dados atuais de processos. Seu principal objetivo é mover dados em tempo real de CLPs (Controlador Lógico Programável), DCSs (Sistema de Controle Distribuído) e outros dispositivos de controle para IHMs.

Os servidores OPC DA permitem que clientes OPC DA acessem dados de variáveis internas, como o nível de água de um tanque, e escrevam valores de controle, como a abertura de uma válvula. Os clientes OPC DA selecionam as variáveis (itens OPC) que desejam ler, gravar ou monitorar no servidor. O cliente estabelece uma conexão com o servidor criando um objeto *OPCServer*. Este objeto oferece métodos para navegar pela hierarquia do espaço de endereço, localizando itens e suas propriedades, como tipo de dados e direitos de acesso [Santos \(2008\)](#); [Mahnke u. a. \(2009\)](#).

Para acessar os dados, o cliente organiza os itens OPC de mesma configuração, como tempo de atualização em um objeto *OPCGroup*. A Figura 9 mostra os diferentes objetos que o cliente OPC cria no servidor.

Figura 9: Objetos criados por um cliente OPC DA



Fonte: Elaborado pelo próprio autor

Uma vez adicionados a um grupo, os itens podem ser lidos ou gravados pelo cliente. A leitura dos dados do cliente é cíclica, monitorando as mudanças de valor no servidor. O cliente estabelece uma taxa de atualização no grupo dos itens de interesse, que é utilizada no servidor para verificar os valores que serão alterados. Ao final de cada ciclo, o servidor envia apenas os valores atualizados para o cliente.

O fornecimento de dados do OPC é em tempo real e podem não estar acessíveis a todo momento, por exemplo quando a comunicação com um dispositivo é interrompida temporariamente. O OPC clássico lida com este problema fornecendo um *timestamp* e uma etiqueta de qualidade para os dados entregues. A qualidade especifica se os dados são precisos (bons), indisponíveis (ruins) ou desconhecidos (incertos).

2.4.3 OPC Alarms and Events

A especificação *OPC Alarms and Events* define uma interface que permite a monitoração e reconhecimento de alarmes e eventos Santos (2008). Alarmes são notificações que alertam ao cliente sobre uma alteração inesperada na condição de um processo e que exige uma ação corretiva. Os eventos são notificações que comunicam ao cliente a ocorrência de uma mudança nas condições da planta, de um equipamento ou de uma variável. Nesse caso, o evento não necessita de uma medida corretiva, seu intuito é apenas indicar alguma mudança.

Desse modo, o OPC A&E fornece uma interface flexível para a transmissão de alarmes e eventos de processos. Se o cliente OPC A&E deseja receber notificações, ele deve se inscrever para tal, assim todas as notificações acionadas no servidor serão informadas. O cliente OPC se conecta criando um objeto *OPCEventServer* no servidor A&E e gerando um *OPCEventSubscription* usado para receber as mensagens do evento Mahnke u. a. (2009).

2.4.4 OPC Historical Data Access

Ao passo que o OPC DA dá acesso aos dados variáveis em tempo real, o *OPC Historical Data Access* permite o acesso a dados já armazenados. Com ele é possível acessar dados salvos na

memória de um equipamento, como o CLP ou então transferir dados para um software de supervisão ou até mesmo uma planilha em Excel. Um cliente OPC HDA pode criar diferentes objetos OPC no servidor HDA. O objeto *OPCHDAServer* é o de mais alto nível na estrutura hierárquica de objetos, sendo responsável por disponibilizar as funcionalidades de leitura, escrita e alteração de dados. Já a navegação no espaço de endereço do servidor é oferecida através do *OPCHDABrowser* Santos (2008).

2.4.5 OPC UA

As interfaces de OPC clássicas são baseadas nas tecnologias COM/DCOM da Microsoft. O uso de tais tecnologias apresentam vantagens e desvantagens a essa abordagem do OPC. A vantagem conforme destacado por Mahnke u. a. (2009) foi a redução do trabalho de especificação para definição de diferentes APIs, sem a necessidade de definir um protocolo de rede ou um mecanismo de comunicação interprocesso. A desvantagem é a dependência do sistema operacional Windows e os problemas do DCOM ao utilizar a comunicação remota com o OPC, visto que é difícil de configurar, tem tempos limite longos e não configuráveis e não pode ser usado para comunicação pela Internet.

Diante disso, a OPC Foundation desenvolveu uma versão totalmente revisada e expandida do padrão OPC que elimina os pontos fracos da tecnologia anterior. Esta versão independe do sistema operacional empregado, da linguagem de programação e de tecnologias proprietárias. Além disso, é independente do fabricante e suporta escalabilidade, alta disponibilidade e capacidade de Internet. Esta nova geração de tecnologia OPC foi lançada como OPC UA (*Unified Architecture*) Pogacean u. a. (2015).

Lançada em 2008, a OPC UA é uma arquitetura orientada a serviços (SOA) independente de plataforma que integra todas as funcionalidades das especificações do OPC Classic em uma estrutura escalável OPCFoundation (2017). É um padrão aberto e gratuito especificado na norma IEC62541, que faz o uso de um modelo de informação orientado a objetos, permitindo suporte a estruturas, objetos e máquinas de estado.

O OPC UA permite a troca de dados brutos e informações pré-processadas entre sistemas incorporados em sensores e dispositivos de campo, além de sistemas de ERP, MES e IHM, permitindo a disponibilidade de acesso aos dados para qualquer aplicação e a qualquer momento para pessoas autorizadas. Para isso, ele utiliza um protocolo binário baseado em TCP.

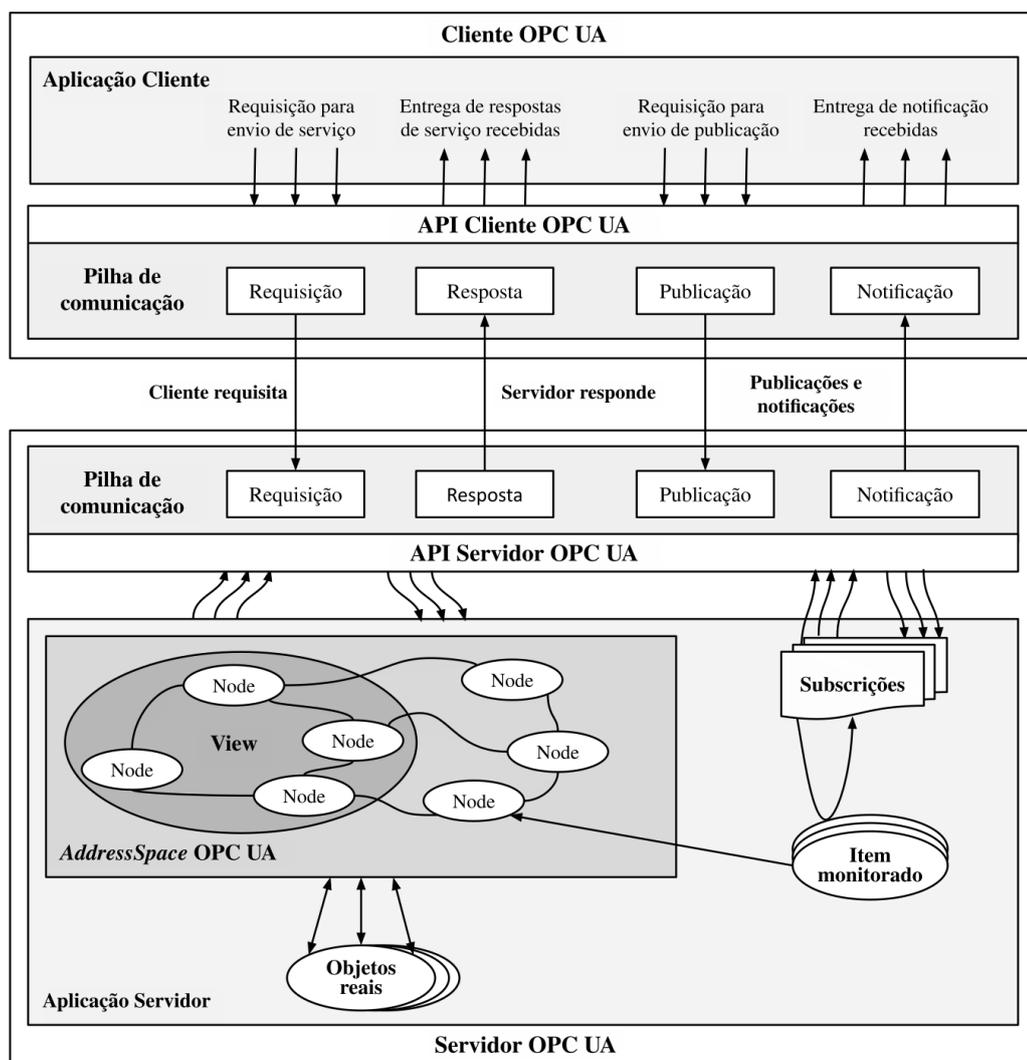
Conforme destacado por Pogacean u. a. (2015) o OPC UA além de suprir toda a funcionalidade do OPC Classic, fornece recursos para descrever procedimentos complexos em componentes padronizados orientados a objetos. Também permite que clientes que só suportam as regras básicas possam processar os dados mesmo sem conhecer a organização do servidor. Ainda segundo os autores, a aplicabilidade universal da tecnologia OPC UA permite a implementação de conceitos completamente novos de integração vertical, visto que ao conectar em cascata os componentes OPC UA, as informações podem ser enviadas de forma confiável e segura desde o

nível da produção para os sistemas ERP e MES. Como resultado, a tecnologia ocupa uma posição importante na visão de futuro da produção industrial como parte de um ambiente totalmente inteligente.

2.4.6 Arquitetura OPC UA

A arquitetura do protocolo OPC UA modela clientes e servidores para interagir como parceiros, conforme ilustrado na Figura 10. Nessa modelagem, cada sistema pode ter vários clientes e servidores, onde cada cliente pode interagir simultaneamente com um ou mais servidores e do mesmo modo um servidor com um ou mais clientes. Um aplicativo pode combinar componentes de servidor e cliente para permitir a interação com outros servidores e clientes.

Figura 10: Arquitetura OPC UA



Fonte: Adaptado de OPCFoundation (2017)

Conforme descrito pela OPCFoundation (2017) a arquitetura do OPC UA Client modela a

endpoint da integração cliente/servidor. Para tal, ele utiliza uma interface interna que isola o código do aplicativo do cliente de uma pilha de comunicação OPC UA. Essa interface é uma *API Client* utilizada para enviar e receber solicitações e respostas para o servidor. Já a arquitetura do OPC UA Server modela o *endpoint* do servidor das interações cliente/servidor. Nesse caso, o aplicativo do servidor é o código que executa a função do servidor, ele utiliza a *API Server* para enviar e receber mensagens dos clientes, que de modo similar à *API Client* isola o código do aplicativo, agora do servidor, da pilha de comunicação OPC UA.

O *AddressSpace* (Espaço de Endereçamento do Servidor OPC UA) é modelado como um conjunto de *Nodes* (Nós) acessíveis aos clientes por meio das interfaces e métodos OPC UA. Os *Nodes* são usados para representar objetos reais, suas definições e suas referências entre si. Objetos reais são objetos físicos ou de software que são acessíveis pelo aplicativo do servidor ou que ele mantém internamente.

Uma *View* (Visualização) é um subconjunto do *AddressSpace*. As *Views* são utilizadas para limitar os nós que o servidor torna visíveis para o cliente, restringindo assim o tamanho do endereçamento para as solicitações de serviço enviadas pelo cliente. De modo padrão a *View* é todo o *AddressSpace*, opcionalmente os servidores podem definir outras *Views*, ocultando alguns nós ou referências. Sua organização geralmente é hierárquica, facilitando a navegação pelos clientes [OPCFoundation \(2017\)](#).

De modo geral, OPC UA incorpora uma arquitetura robusta e de comunicação confiável, com mecanismos de monitoramento de tempo e detecção automática de falhas. As ferramentas de controle de falhas restabelecem de forma automática a comunicação entre o *OPC UA Client* e *OPC UA Server* sem perda de dados [Pogacean u. a. \(2015\)](#).

3

Materiais e Métodos

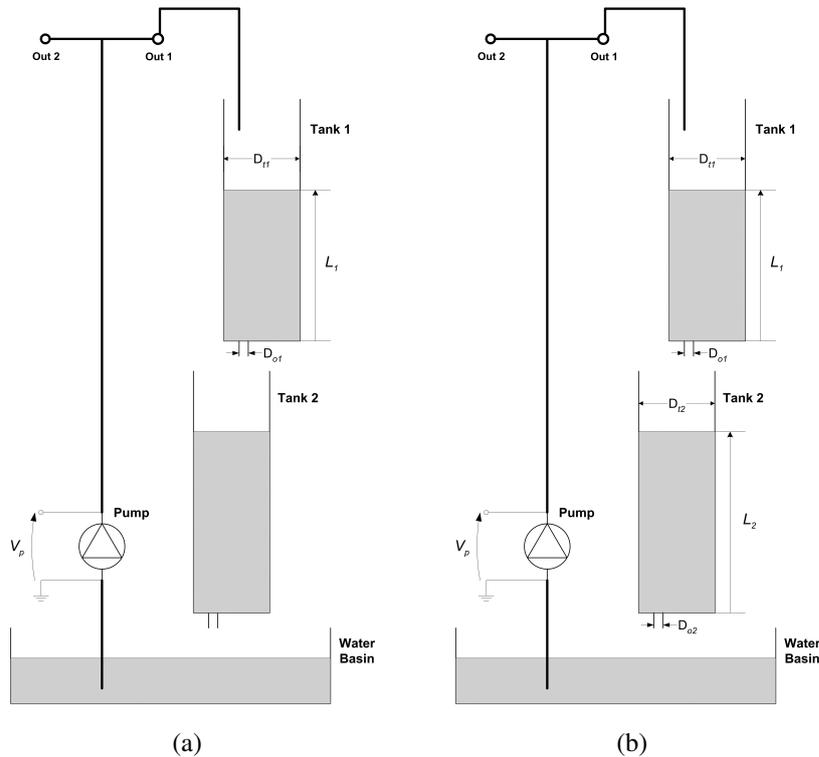
Neste capítulo serão apresentados os materiais e métodos necessários para atingir os resultados esperados. A metodologia é dividida em quatro seções, a Seção 3.1 apresenta o sistema de tanques a ser modelado; a Seção 3.2 traz a modelagem do sistema no MATLAB/Simulink com o controle PID desacoplado e os blocos OPC sendo utilizados para leitura/escrita das variáveis do processo; a Seção 3.3 aborda todo o mecanismo de comunicação necessário, com o servidor OPC KEPServerEX atuando no interfaceamento entre o MATLAB e o CLP; e por fim, a Seção 3.4 apresenta as configurações necessárias para realizar o controle e supervisão do sistema de tanques no CLP da Siemens.

3.1 Sistema

O sistema estudado neste trabalho foi o de tanques acoplados da Quanser. Trata-se de um protótipo de simulação de processos, que usa dois tanques interligados em cascata com fluxo regulado por uma bomba. Neste sistema, os tanques são montados, de modo que o fluxo do primeiro tanque (superior) possa fluir, através de um orifício de saída localizado na sua parte inferior para o segundo tanque (inferior), e assim para o reservatório de água principal. A bomba empurra a água verticalmente para dois orifícios de conexão rápida "Out1" e "Out2". As duas variáveis do sistema são medidas diretamente no equipamento de tanque acoplado, por meio de sensores de pressão e estão disponíveis para realimentação. Por fim, os níveis de água são marcados nos tanques 1 e 2 [Apkarian u. a. \(2013\)](#).

Para este sistema podem ser utilizadas duas configurações distintas, "*Configuração 1*" e a "*Configuração 2*" (Figura 11). Na Configuração 1, o objetivo é controlar o nível de água no Tanque 1, utilizando a vazão da bomba. Já na Configuração 2, o desafio é controlar o nível de água no Tanque 2, a partir do fluxo de água que sai do Tanque 1 [Apkarian u. a. \(2013\)](#). A configuração a ser utilizada nesta monografia será a primeira, implementando um sistema de controle para o nível do Tanque 1.

Figura 11: (a) Configuração 1 e (b) Configuração 2

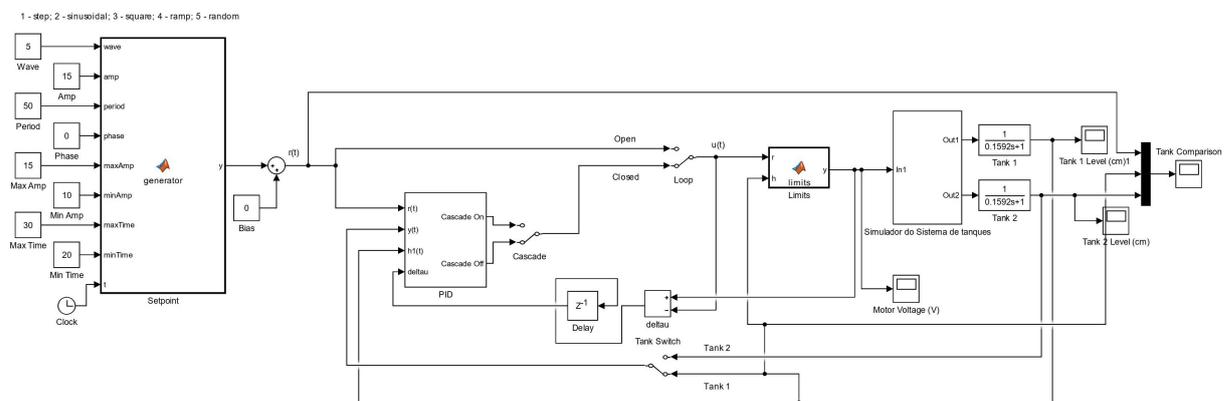


Fonte: Quanser

3.2 Modelagem

Para modelagem no MATLAB foi adaptada a planta desenvolvida pelo prof. Glauber Rodrigues Leite do IC UFAL (Figura 12), no qual o objetivo neste trabalho foi desacoplar o controle PID do sistema, visto que o intuito é realizar no CLP e acoplar os blocos de leitura e escrita do OPC, como ponto de partida para comunicação entre o MATLAB e o servidor OPC.

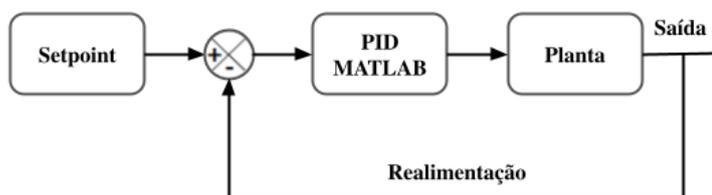
Figura 12: Modelagem do sistema de tanques acoplados proposta por Glauber R. Leite



Fonte: Glauber R. Leite

O diagrama de blocos da Figura 13 representa a configuração do controlador PID no MATLAB. A modelagem, controle e simulação são feitos diretamente no próprio software.

Figura 13: Diagrama de blocos de configuração de controle PID no MATLAB

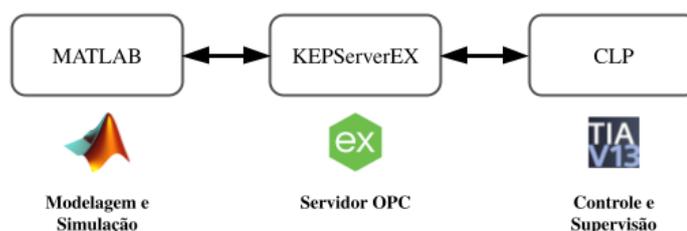


Fonte: Elaborado pelo próprio autor

O MATLAB é um software interativo que utiliza linguagem de alto nível, permitindo a execução de tarefas complexas, tais como, processamento de imagens, comunicações, aquisições de dados, sistemas de controle, robótica, entre outras funcionalidades [Iwasak \(2015\)](#).

Neste trabalho, o MATLAB foi utilizado para aquisição dos dados e modelagem do sistema. Como a comunicação entre ele e o CLP não pode ser feita de maneira direta, foi necessária a configuração de uma interface de comunicação entre os dois. Esta interface é possível através do uso dos blocos de comunicação de leitura e escrita do OPC Toolbox no MATLAB. A Figura 14 ilustra o diagrama de rede utilizado, com o OPC KEPServerEX sendo responsável por permitir o interfaceamento entre o MATLAB e o CLP.

Figura 14: Diagrama da rede



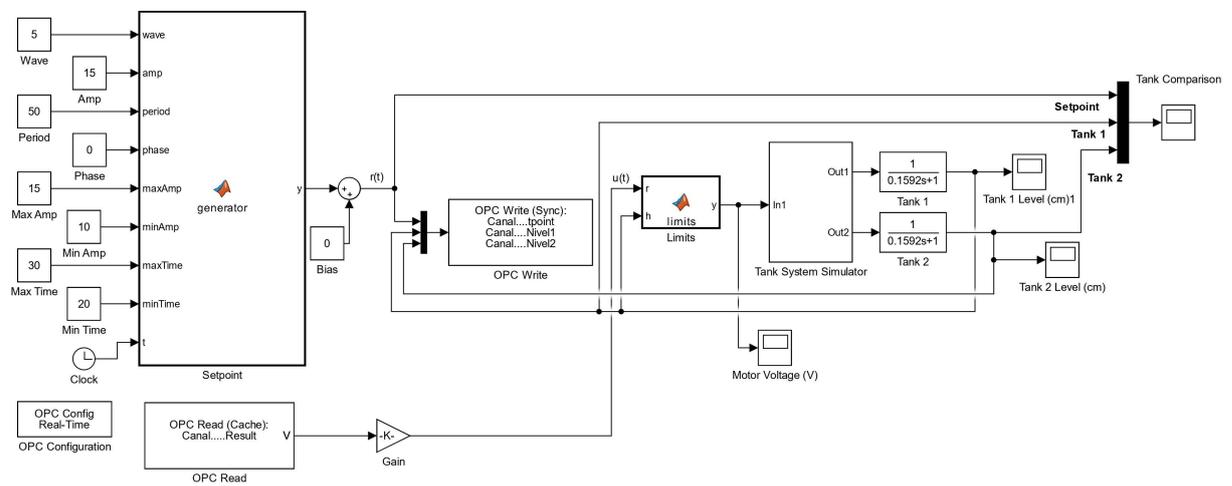
Fonte: Elaborado pelo próprio autor

O OPC Toolbox é uma ferramenta do MATLAB que permite a conexão com os servidores OPC DA (*Data Access*) e OPC HDA (*Historical Data Access*). É criado um objeto OPC DA para se comunicar com um servidor OPC, utilizando uma estrutura hierárquica para auxiliar no gerenciamento das conexões OPC e de tags do servidor. Com essa ferramenta é possível configurar e controlar todos os clientes, grupos e objetos do item, podendo modificar suas propriedades e valores em tempo real. O objeto OPC HDA é criado para se conectar a um servidor OPC HDA. Esse cliente permite que se navegue nos dados do servidor e recupere os IDs de cada item armazenado, esses IDs são utilizados para solicitar dados históricos a partir do servidor. Esses dados podem ser recuperados especificando, quais são os IDs, o período de tempo

para o qual deseja se recuperar os dados e parâmetros opcionais [Iwasak \(2015\)](#). Desse modo, os servidores OPC DA e OPC HDA dão acesso aos dados em tempo real e históricos diretamente do MATLAB e Simulink, possibilitando a leitura e o registro dos dados vindos do CLP, além da escrita de dados OPC no mesmo.

A Figura 15 exibe a modelagem no MATLAB Simulink do sistema de tanques da seção anterior, na parte inferior esquerda podem ser vistos os blocos de *OPC Config Real-Time* para configuração do servidor OPC e *OPC Read* para leitura da variável de sinal de controle utilizado do sistema. Na região central temos o bloco *OPC Write* para leitura do sinal de entrada e saída que serão utilizadas para supervisão e controle no CLP, setpoint, nível do tanque 1 e nível do tanque 2.

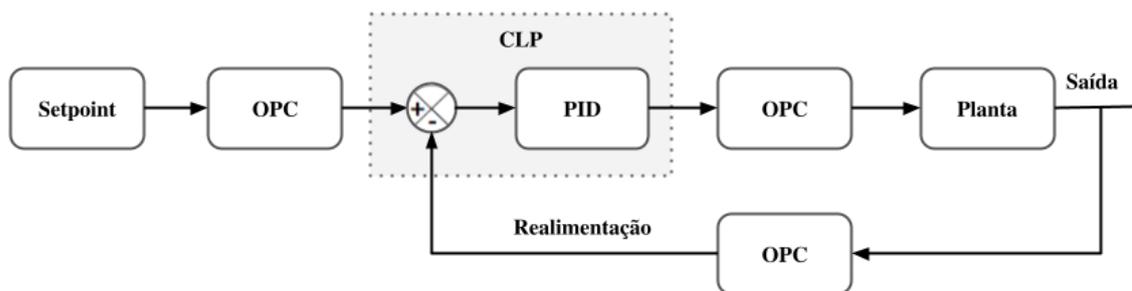
Figura 15: Modelagem do sistema de tanques acoplados com controle via CLP



Fonte: Elaborado pelo próprio autor

O diagrama de blocos da Figura 16 representa a configuração do controle PID no CLP. Este é o sistema a ser implementado neste trabalho, como OPC viabilizando a intercomunicação entre MATLAB (Setpoint e Planta) e CLP

Figura 16: Diagrama de blocos de configuração de controle PID no CLP com OPC



Fonte: Elaborado pelo próprio autor

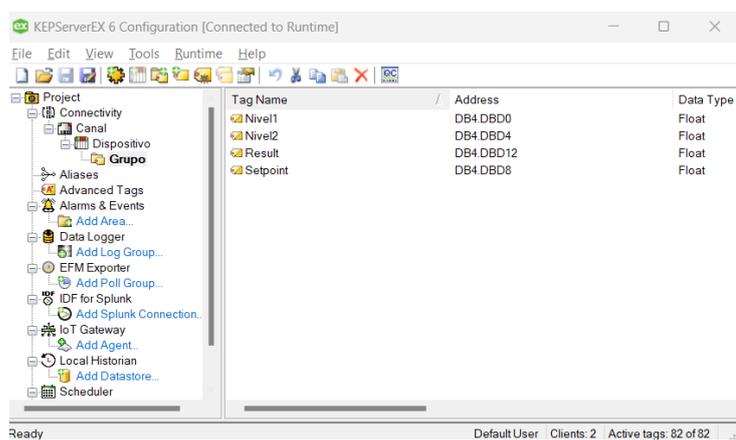
3.3 Comunicação

O sistema de controle de tanques acoplados proposto nesse trabalho será feito através do software MATLAB e do CLP S7-1200 da Siemens. Sendo o primeiro responsável pela modelagem e simulação e o segundo pelo controle do sistema. Devido a incompatibilidade entre o CLP e o MATLAB, o padrão de comunicação OPC é utilizado, permitindo a interoperabilidade entre essas duas tecnologias.

O servidor OPC utilizado foi o KEPServerEX, plataforma de software de comunicação criada pela Kepware e gerida pela Logitek, líder do setor que fornece uma única fonte de dados de automação industrial, possibilitando aos clientes se conectar, gerenciar, monitorar e controlar diversos dispositivos de automação e aplicativos de software por meio de uma interface de usuário intuitiva [Ossoftware](#).

A criação e configuração de um servidor OPC no KEPServerEX é bem simples, basta criar o **Projeto**, adicionar um **Canal** de comunicação do tipo Siemens TCP/IP Ethernet, visto que, esse será o protocolo de rede utilizado para transferir os dados entre o CLP e o OPC; Adicionar um **Dispositivo**, especificando o modelo do CLP e o ID, que é o IP de rede que está sendo utilizado no CLP. E por fim, adicionar um **Grupo** e suas **Tags**, definindo nome, endereço e tipo de dado. Na Figura 17 pode ser visto as tags utilizadas no projeto para o controle do sistema, sendo duas de saída: Nível 1, Nível 2, uma de entrada: Setpoint e uma de controle: Result.

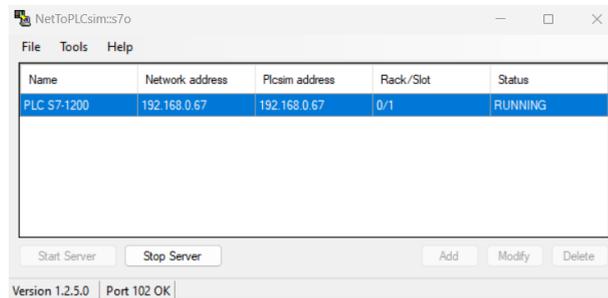
Figura 17: Projeto de servidor OPC no KEPServerEX



Fonte: Elaborado pelo próprio autor

Para possibilitar a comunicação entre o ambiente de simulação CLP (PLCSIM) e o OPC via TCP/IP, foi utilizada a extensão de rede pra PLCSIM, NetToPLCsim que permite que seja acessado o S7-PLCSIM da rede via comunicação TCP/IP (Iso-On-TCP), usando a interface de rede do computador no qual a simulação está sendo executada. Sendo extramente útil para testar uma aplicação cliente (SCADA, HMI) junto com o S7-PLCSIM, sem um CLP físico [NetToPLCsim](#). A configuração do NetToPLCsim, pode ser vista abaixo na Figura 18:

Figura 18: Configuração do NetToPLCsim



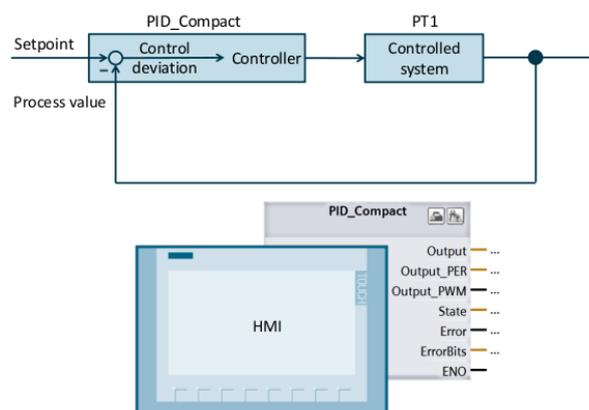
Fonte: Elaborado pelo próprio autor

3.4 Controle e Supervisão

O controle e supervisão do sistema foi feito em um CLP da Siemens, modelo S7-1200 CPU 1214C DC/DC/DC, o mesmo presente no laboratório de Sistemas de Controle da Universidade Federal de Alagoas. Para isso, será utilizado o ambiente de Automação Totalmente Integrada da Siemens, **TIA Portal**, software que otimiza todas as rotinas operacionais, de máquinas e processos, oferecendo um conceito de operação padronizada para controladores, interfaces homem-máquina (IHM) e acionamentos, e para armazenamento e consistência de dados compartilhados [Siemens](#).

O processo de controle no CLP ocorrerá a partir do uso do bloco **PID_Compact**, ferramenta do TIA Portal que fornece o controle de um sistema, com base na leitura do valor de processo medido e sua comparação com o setpoint, calculando a partir do desvio de controle resultante, um valor de saída para ajustar o desvio do ponto de ajuste ou a variável de perturbação, se necessário (Figura 19) [Siemens \(2019\)](#).

Figura 19: Esquema de controle no PID_Compact



Fonte: [Siemens \(2019\)](#)

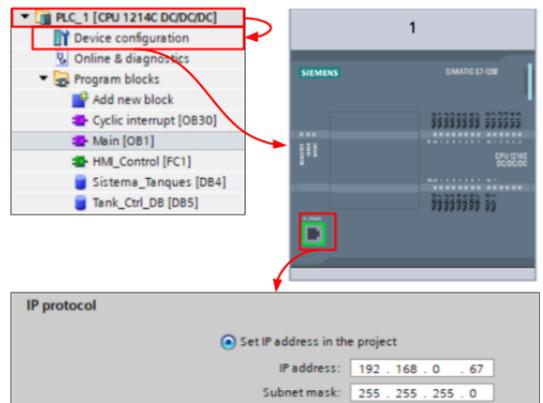
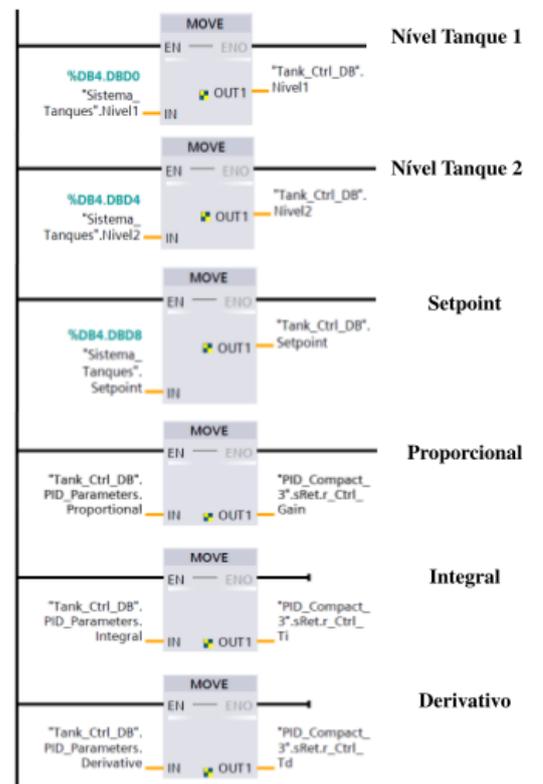
Conforme descrito por [Siemens \(2019\)](#), o PID_Compact possui a funcionalidade de comissionamento "*tuning*" com a qual os parâmetros P, I e D podem ser calculados automaticamente dependendo do sistema controlado. No entanto, também é possível especificar os parâmetros de controle manualmente. A sintonia automática é dividida em dois tipos:

1. **Pretuning:** O pré-ajuste determina a resposta do processo a um salto no valor de saída e procura o ponto de inflexão. Os parâmetros PID são calculados a partir do gradiente máximo e do tempo morto do sistema controlado.
2. **Fine tuning:** O ajuste fino gera uma oscilação constante e limitada do valor do processo. A amplitude e frequência desta oscilação são usadas para otimizar os parâmetros PID para o ponto de operação. Todos os parâmetros PID são recalculados a partir dos resultados. Os parâmetros PID do ajuste fino geralmente mostram melhor comportamento de comando e erro do que os parâmetros PID do pré-ajuste.

A supervisão do sistema será realizada em uma IHM do TIA Portal, no qual o usuário poderá acompanhar todo o processo de forma gráfica, além da entrada de parâmetros PID via interface, permitindo que o status de controle do sistema seja monitorado em tempo real.

A configuração do projeto no TIA para o controle do sistema de tanques é descrita na Tabela 1, a proposta é usar o ambiente de simulação do TIA PLCSIM para simular o CLP físico, podendo ser replicado pelos estudantes no equipamento disponível em laboratório.

Tabela 1: Configurações para controle do sistema no TIA Portal

| Nº | Ação | Observações |
|----|--|--|
| 1 | <p>Abra a configuração do dispositivo do controlador PLC_1 [CPU 1214C DC/DC/DC] e clique na porta RJ45, definindo o endereço de IP e a máscara da rede na qual o computador está conectado, o endereço de IP deve ser o mesmo utilizado no KeepServerEX e NetToPlc-sim.</p> |  |
| 2 | <p>Crie o bloco de programa principal Main [OB1], do tipo OB (Organization Block) e Program Cycle (bloco executado ciclicamente). No qual, serão colocadas as instruções que controlam o aplicativo, nível do tanque 1, tanque 2, setpoint, parâmetros PID.</p> <p>Os blocos MOVE são utilizados para mover os valores lidos via OPC do bloco de dados Sistema_Tanques [DB4] para o bloco de dados Tank_Ctrl_DB [DB5] nas variáveis de níveis do tanque 1, tanque 2 e setpoint. E nos de parâmetros PID para mover de Tank_Ctrl_DB [DB5] para o PID_Compact em Cyclic interrupt [OB30]. A operação é necessária para que seja possível ler e escrever dados no OPC e IHM.</p> |  |

Nº Ação Observações

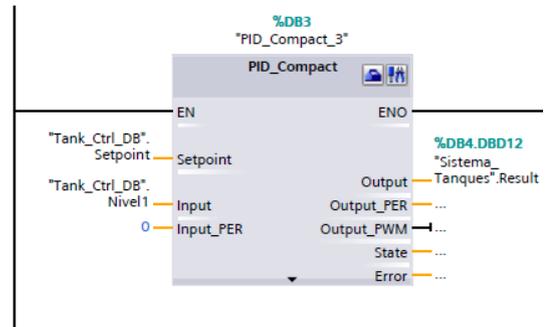
3 Defina um bloco de dados DB (Data Block) **Sistema_Tanques [DB_4]** para armazenar as variáveis que serão utilizadas para ler e/ou escrever dados no OPC. O offset deve ser o mesmo dos endereços atribuídos às tags OPC no KeepServerEX.

| Sistema_Tanques | | | |
|-----------------|----------|-----------|--------|
| | Name | Data type | Offset |
| 1 | Static | | |
| 2 | Nivel1 | Real | 0.0 |
| 3 | Nivel2 | Real | 4.0 |
| 4 | Setpoint | Real | 8.0 |
| 5 | Result | Real | 12.0 |

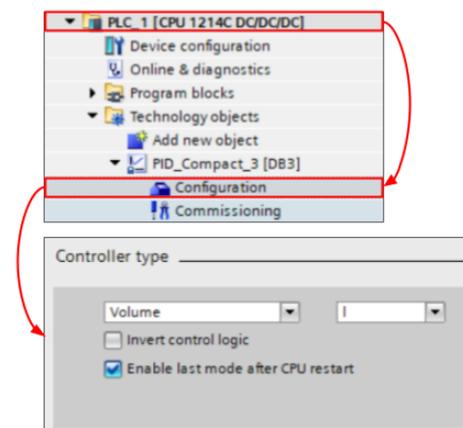
4 Defina um bloco de dados DB **Tank_Ctrl_DB [DB_5]** para armazenar todas as variáveis que serão utilizadas pelo PID_Compact e pela IHM.

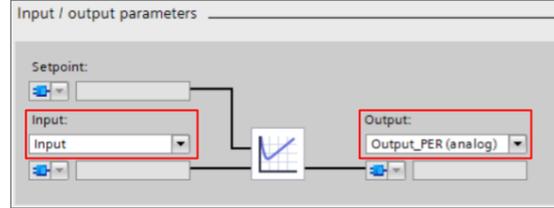
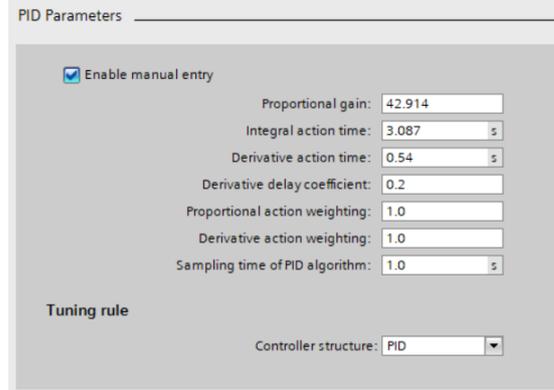
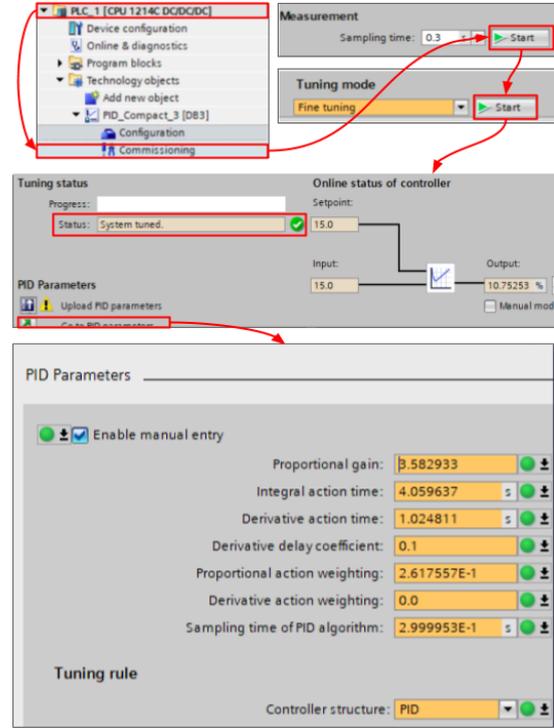
| Tank_Ctrl_DB | | | |
|--------------|----------------|-----------|-------------|
| | Name | Data type | Start value |
| 1 | Static | | |
| 2 | Nivel1 | Real | 0.0 |
| 3 | Nivel2 | Real | 0.0 |
| 4 | Setpoint | Real | 0.0 |
| 5 | PID_Parameters | Struct | |
| 6 | Proportional | Real | 0.0 |
| 7 | Integral | Real | 0.0 |
| 8 | Derivative | Real | 0.0 |
| 9 | PID_Mode | Struct | |
| 10 | Enable_Bit | Bool | false |
| 11 | Disable_Bit | Bool | false |
| 12 | Disable | Int | 0 |
| 13 | Enable | Int | 1 |

5 Crie o bloco responsável pelo controle do sistema, em **Cycle interrupt [OB30]**, do tipo OB e Cycle interrupt, que chama o controlador compacto ciclicamente a cada 100 milissegundos através dos blocos de simulação. Atribua o **Setpoint**, a variável de entrada **Nível1** (nível do tanque 1) e a saída **Result** (sinal controlado a ser utilizado para realimentação do sistema).



6 Abra o editor de configuração selecionando **PLC_1 [CPU 1214C DC/DC/DC]** → **Technology objects** → **PID_Compact_3** → **Configuration**. Em seguida o submenu **Controller type** nas configurações básicas, para definir que tipo de processo será controlado, nesse caso um sistema de tanques, cuja o tipo é **Volume** e a unidade de medida **l**.



| Nº | Ação | Observações |
|----|---|--|
| 7 | <p>Ajuste o submenu Input/Output parameters nas configurações básicas para definir os sensores/atuadores utilizados, para o sistema em questão a entrada é um ponto flutuante Input e saída um valor analógico Output_PER.</p> |  |
| 8 | <p>Abra os PID Parameters nas configurações avançadas, definindo a estrutura de controle para regra de ajuste como PID.</p> <p>Os parâmetros podem ser inseridos manualmente, mas isso não será feito, pois eles vão ser obtidos a partir do recurso de modo de ajuste, que ajustará automaticamente os três parâmetros de PID.</p> |  |
| 9 | <p>Abra o editor de configuração selecionando PLC_1 [CPU 1214C DC/DC/DC] → Technology objects → PID_Compact_3 → Commissioning. Inicie a medição, para que o sistema possa monitorar o setpoint e a variável de processo. Posteriormente, inicie o modo de ajuste em Fine tuning para obter os parâmetros PID, esse modo foi escolhido pois geralmente mostra melhor comportamento de comando e erro do que os parâmetros PID do Pretuning.</p> <p>Após a conclusão do Fine tuning, carregue os parâmetros PID determinados no projeto como valores iniciais no bloco de dados da instância do "PID_Compact", clicando em Upload PID parameters. Por fim, clique em Go to PID parameters para ver os parâmetros obtidos.</p> |  |



Resultados e Discussões

Neste capítulo serão apresentados os resultados e discussões acerca deste trabalho. O conteúdo esta dividido em 4 seções, a Seção 4.1 apresenta o controle PID do sistema utilizando o CLP; a Seção 4.2 apresenta os resultados do controle do sistema no MATLAB/Simulink; a Seção 4.3 traz os resultados do ambiente de supervisão desenvolvido em uma IHM, com uma visão geral de todo o processo, assim como, da planta, simulação, indicadores e controle PID; e na Seção 4.4 é feito um comparativo entre os resultados obtidos nas seções anteriores.

4.1 Controle PID no CLP

Esta seção traz os resultados do controle do sistema de tanques pelo CLP da Siemens. Os parâmetros PID utilizados foram os mesmos obtidos na Tabela 1, após o processo de *Fine tuning* no bloco de **PID_Compact** do TIA Portal (Tabela 2).

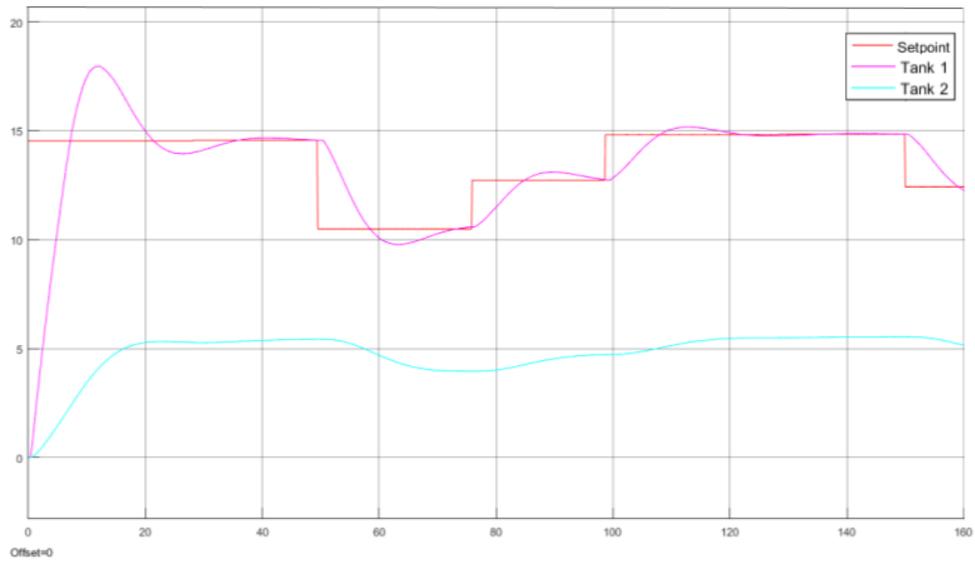
Tabela 2: Parâmetros PID para controle no CLP

| Indicador | Valor |
|----------------|---|
| Setpoint | 15 |
| Parâmetros PID | $K_p : 3,589$ $K_i : 4,060$ $K_d : 1,025$ |

Fonte: Elaborado pelo próprio autor

A Figura 20 abaixo exhibe o resultado do sistema de controle utilizando o bloco de controle PID do CLP, o setpoint é uma onda quadrada variável para simular melhor o controle automático do sistema mediante as variações e perturbações externas, que o mesmo pode sofrer ao longo do tempo. As ondas rosa e ciano representam o nível dos líquidos nos tanques 1 e 2, respectivamente, e a onda quadrada em vermelho, o setpoint.

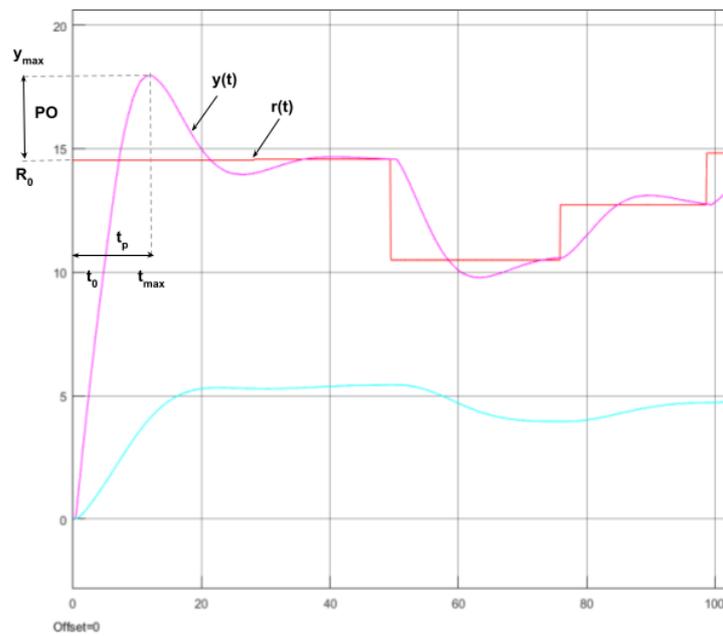
Figura 20: Controle PID do sistema no CLP



Fonte: Elaborado pelo próprio autor

Analisando as especificações de resposta para o sistema de controle apresentado na Figura 20, o valor máximo da resposta é denotado pela variável y_{max} e ocorre em um tempo t_{max} , como pode ser visto na Figura 21.

Figura 21: Especificações do sistema de controle no CLP



Fonte: Elaborado pelo próprio autor

Para essa resposta, o tempo de pico é encontrado usando:

$$t_p = t_{max} - t_0 \quad (51)$$

Sendo assim, o pico do sistema é dado em **12,2 segundos**.

Já o *overshoot*, isto é, o percentual de ultrapassagem do sinal é obtido por:

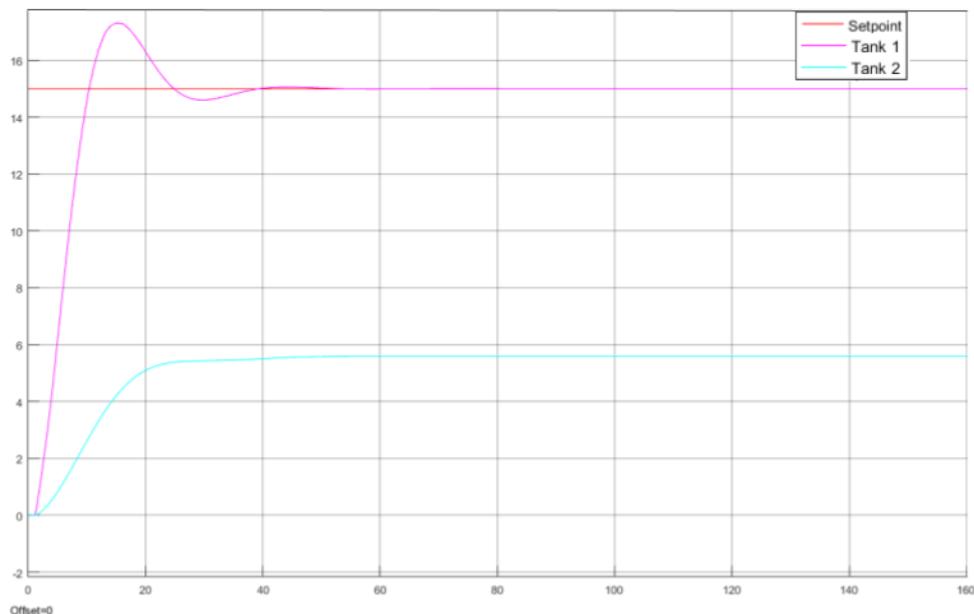
$$PO = \frac{100(y_{max} - R_0)}{R_0} \quad (52)$$

Utilizando as estatísticas do sinal no MATLAB, temos que $y_{max} = 17,96$ e $R_0 = 14,53$.

Portanto, o *overshoot* do sistema é de **23,6%**.

De modo geral, o sistema apresenta uma boa resposta mediante ao controle via CLP, obtendo uma estabilização por volta dos 40 segundos após o início da simulação. Como pode ser observado na Figura 22, que simula a saída do sistema fixado em um setpoint de amplitude de 15 cm.

Figura 22: Controle PID no CLP (setpoint fixo)



Fonte: Elaborado pelo próprio autor

4.2 Controle PID no MATLAB/Simulink

Nesta seção são apresentados os resultados obtidos pelo controle PID do sistema de tanques via MATLAB/Simulink. Os parâmetros proporcional, integral e derivativo utilizados foram os do modelo presente na Figura 12 e disponíveis na Tabela 3.

Na Figura 23 é apresentado o resultado do controle do sistema utilizando apenas o MA-

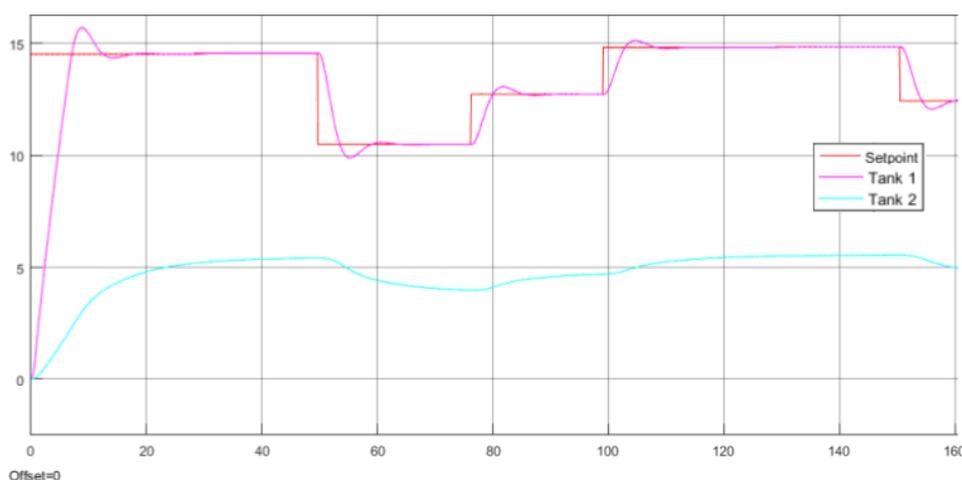
TLAB/Simulink, no qual as ondas azul e ciano representas os níveis no tanque 1 e 2, e a onda quadrada vermelha o setpoint.

Tabela 3: Parâmetros PID para controle no MATLAB/Simulink

| Indicador | Valor |
|----------------|-------------|
| Setpoint | 15 |
| Parâmetros PID | $K_p : 1$ |
| | $K_i : 0,8$ |
| | $K_d : 0,5$ |

Fonte: Elaborado pelo próprio autor

Figura 23: Controle PID no MATLAB/Simulink



Fonte: Elaborado pelo próprio autor

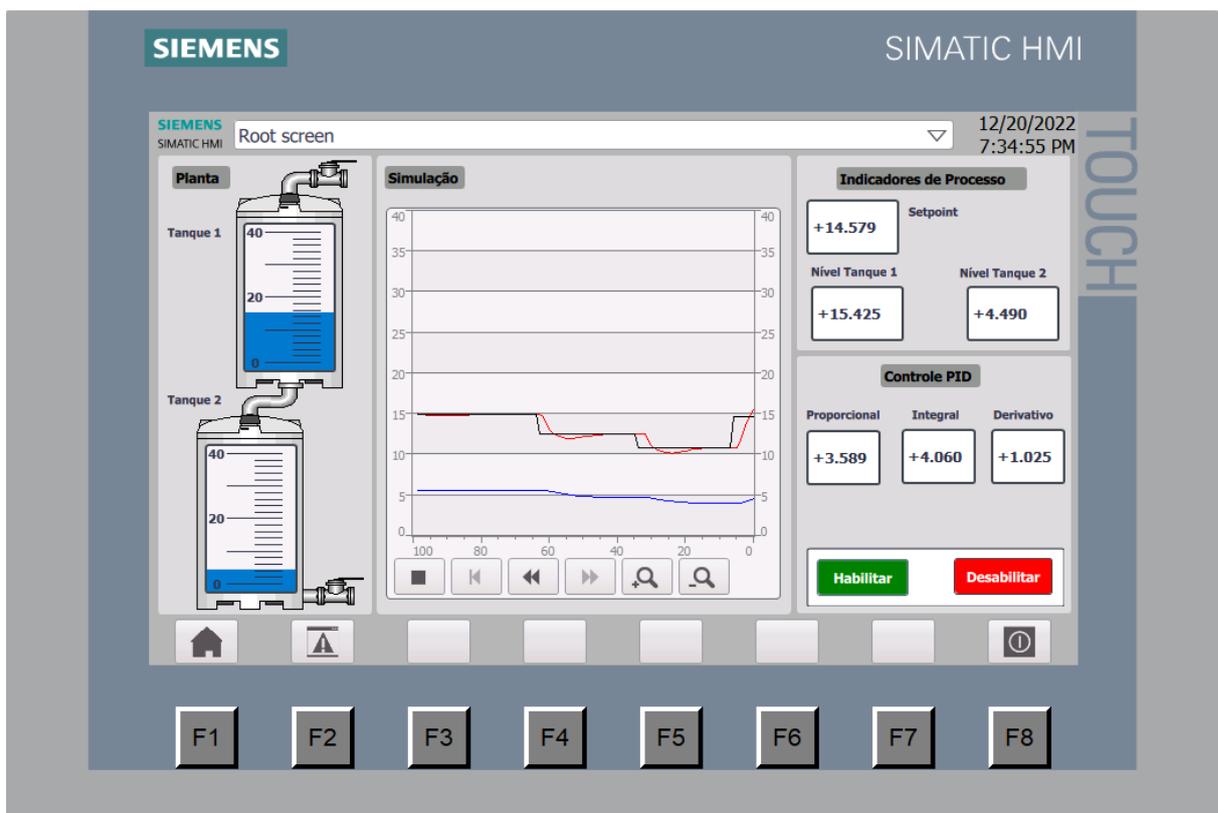
As especificações de resposta para o sistema apresentado na Figura 23, são de $y_{\max} = 15,70$ e $R_0 = 14,53$. Resultando em um tempo de pico de **8,8 segundos** e um *overshoot* de **8,05%**. Os resultados apresentam-se satisfatórios, com uma resposta mais rápida que o sistema controlado via CLP. Porém, nessa abordagem considera-se que o controle PID seja modelado no Simulink, conferindo uma complexidade adicional à modelagem do sistema em relação a abordagem anterior, que simplifica esse processo com a integração do CLP.

4.3 IHM e Supervisão no CLP

Utilizando as ferramentas disponíveis no TIA Portal foi construída uma IHM para supervisão do sistema (Figura 24), que exibe informações acerca da planta, com uma ilustração em tempo real do nível de líquido dos tanques 1 e 2; da simulação, com um gráfico exibindo os valores de

nível do tanque 1 (vermelho), tanque 2 (azul) e setpoint (preto) ao longo tempo; indicadores do processo; e os parâmetros para o controle PID, com as variáveis proporcional, integral e derivativa. Essas variáveis são inicializadas durante o início do processo, mas podem ser alteradas ao longo da execução, se necessário, refletindo diretamente nos resultados. Isso pode ser útil para avaliar, de modo simultâneo, o impacto da alteração dos parâmetros PID no controle do sistema.

Figura 24: IHM para supervisão do sistema de tanques



Fonte: Elaborado pelo próprio autor

A IHM desenvolvida neste trabalho fornece uma ferramenta para supervisão em tempo real de um sistema de controle de tanques acoplados, integrando os diferentes softwares e tecnologias presentes no laboratório em único processo. Essa integração e interoperabilidade entre os sistemas, só foi viável graças ao uso do protocolo OPC. Cabe destacar, que o intuito dessa proposta foi promover um uso mais amplo do aparato tecnológico disponível, de modo a subsidiar e ampliar o uso de propostas similares para metodologias de ensino práticas em disciplinas de Controle e Automação, utilizando MATLAB e CLP.

4.4 Comparativo entre os Sistemas de Controle

A latência em tempo real do modelo pode ser obtida na porta de saída do bloco **OPC Config** no Simulink. Essa variável é importante para o comparativo entre os sistemas discutidos nessa seção, dado que ela mensura o intervalo de tempo entre o recebimento de dados de entrada e a resposta a esses dados. Considerando o tempo da simulação de 200 segundos, a latência média através da rede foi de **27,16 milissegundos** para o sistema controlado pelo CLP.

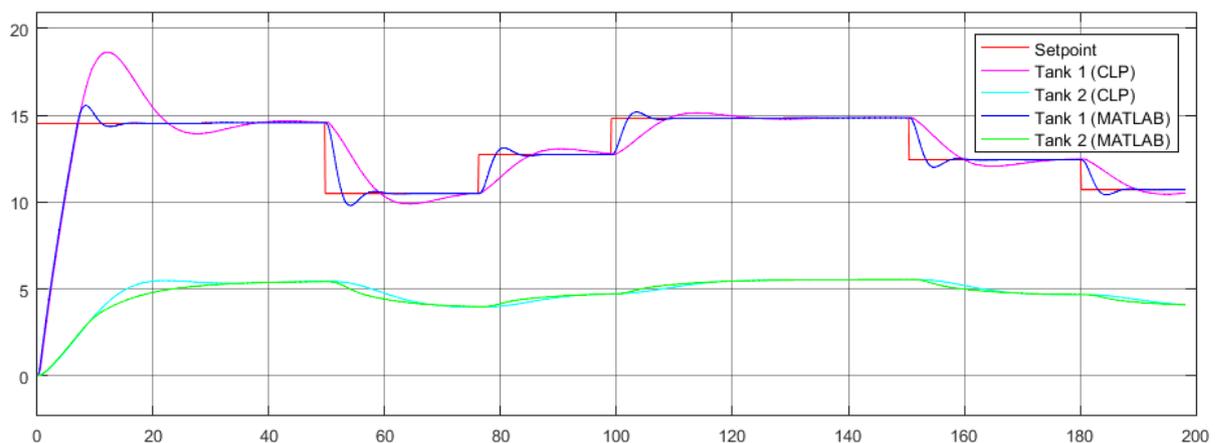
Analisando as saídas do sistema e comparando as duas metodologias de controle, vemos que no MATLAB/Simulink o *overshoot* é menor e o tempo de pico é atingido mais rápido, como pode ser visto na Tabela 4. Contudo, o sistema controlado via CLP ainda tem uma boa resposta, mantendo o controle mesmo diante as variações de setpoint e com uma diferença mais sutil na saída do tanque 2, quando comparado ao controle PID no MATLAB/Simulink.

Tabela 4: Especificações de resposta dos sistemas

| | Overshoot | Tempo de pico |
|-----------------|-----------|---------------|
| CLP | 23,6% | 12,2s |
| MATLAB/Simulink | 8,05% | 8,8s |

Fonte: Elaborado pelo próprio autor

Figura 25: Sistema controlado por CLP e MATLAB



Fonte: Elaborado pelo próprio autor

Tendo em vista que o controle no CLP depende da troca de dados na rede entre o MATLAB e o CLP, com intermédio do OPC, e que a latência média foi de cerca de 27 milissegundos, os resultados se mostram satisfatórios, dado que no Simulink esse processo ocorre de maneira direta. Isso demonstra que o controle PID em CLPs se apresenta como uma alternativa viável para

o desenvolvimento de sistemas de controle modelados no MATLAB, facilitando a construção e integração desses processos, visto que o controle é feito de modo automático no CLP. Desse modo, qualquer sistema que tenha sua planta desenvolvida no MATLAB pode ser controlado em um CLP, desde que sejam fornecidos o sinal de entrada e setpoint.

5

Conclusão

Neste trabalho foi desenvolvida uma plataforma de simulação e controle de um sistema de tanques acoplados, integrando MATLAB e CLP, com uso do protocolo OPC, em um único processo. A proposta apresentada fornece uma metodologia útil para construção de sistemas de controle modelados no MATLAB, simplificando a implementação de controladores PID, dado que esse controle passa a ser feito de modo automático no CLP.

Os resultados obtidos fornecem uma ferramenta para ampliar o desenvolvimento de projetos de controle em plantas didáticas integradas, visto que a interoperabilidade entre os sistemas é possível com o OPC. Consequentemente, a integração de processos industriais em pequena escala com MATLAB e CLP, tecnologias bastante utilizadas no curso de Engenharia de Computação do IC UFAL, porém sem uma integração prática para projetos de sistemas de controle, possibilita o uso de equipamentos do laboratório de forma mais unificada e interdisciplinar, promovendo uma melhor experiência de aprendizado durante as aulas experimentais, bem como dispondo de mais recursos didáticos a serem utilizados pelos docentes.

Nosso objeto de estudo foi um sistema de nível de líquidos, porém a integração entre MATLAB e CLP feitas, pode ser utilizada para qualquer outro tipo de sistema, dado que o MATLAB/Simulink fornece as variáveis de sinal de saída e setpoint, e o CLP retorna o sinal controlado. Logo, trata-se de um processo que independe de planta, desde que as variáveis de entrada sejam fornecidas, o sistema será controlado.

Portanto, essa proposta pode ser ampliada para modelar e integrar diferentes tipos de sistemas de controle, unificando processos e integrando equipamentos e tecnologias de modo funcional e prático.

Referências bibliográficas

- [Apkarian u. a. 2013] APKARIAN, Jacob ; LACHERAY, Hervé ; ABDOSSALAMI, Amin: Coupled Tanks Experiment for MATLAB /Simulink Users, 2013
- [Åström und Hägglund 1995] ÅSTRÖM, Karl Johan ; HÄGGLUND, Tore: *PID Controllers: Theory, Design, and Tuning*. ISA - The Instrumentation, Systems and Automation Society, 1995. – ISBN 1-55617-516-7
- [Atiyah u. a. 2019] ATIYAH, Bassam ; YADGAR, Saber ; GHAZY, Murad ; MOHSIN, Bassam ; HAMEED, Saber ; KHALAF, Murad: Comparison of MATLAB Simulink application with PLC application of real-time classical PID controllers in laboratory. In: *Periodicals of Engineering and Natural Sciences (PEN)* 7 (2019), 12, S. 2045–2056
- [Bagal u. a. 2018] BAGAL, K.N. ; KADU, C.B. ; PARVAT, B.J. ; VIKHE, P.S.: PLC Based Real Time Process Control Using SCADA and MATLAB. In: *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, 2018, S. 1–5
- [Bazanella 2005] BAZANELLA, Alexandre S.: *Sistemas de Controle: Princípios e Métodos de Projeto*. Editora UFRGS, 2005
- [Dang 2007] DANG, Tuan: Integration of Power Plant information system with Business information system in the open electricity market: challenges and solutions. In: *2007 5th IEEE International Conference on Industrial Informatics* Bd. 2, 2007, S. 1209–1213
- [Dorf und Bishop 2001] DORF, Richard C. ; BISHOP, Robert H.: *Modern control systems*. Rio de Janeiro, Brasil : Livros Técnicos e Científicos, 2001
- [Friedland 1985] FRIEDLAND, Bernard: *Control system design*. New York, NY : McGraw-Hill, November 1985 (McGraw-Hill series in electrical engineering)
- [Iwasak 2015] IWASAK, Ludmila Paola P.: *Desenvolvimento e implementação de um sistema de tanques acoplados*. 2015

- [Mahnke u. a. 2009] MAHNKE, W. ; LEITNER, S.H. ; DAMM, M.: *OPC Unified Architecture*. Springer Berlin Heidelberg, 2009 (SpringerLink: Springer e-Books). – URL <https://books.google.com.br/books?id=de9uLdXKj1IC>. – ISBN 9783540688990
- [NetToPLCsim] NETTOPLCSIM: *NetToPLCsim*.
<https://nettoplcsim.sourceforge.net/>. – Acessado em 12 de Novembro de 2022
- [Nise 2010] NISE, Norman S.: *Control Systems Engineering 6E Binder Ready Version*. Chichester, England : John Wiley & Sons, Dezember 2010
- [Ogata 2010] OGATA, Katsuhiko: *Engenharia de controle moderno*. São Paulo, Brasil : Pearson Education do Brasil, 2010
- [OPCFoundation 2017] OPCFOUNDATION: *Unified Architecture*. <https://opcfoundation.org/developer-tools/specifications-unified-architecture>. 2017. – [Acessado em 21 de Abril de 2022]
- [Osbssoftware] OSBSOFTWARE: *KEPServerEX*.
<https://osbssoftware.com.br/produto/kepserverex>. – Acessado em 10 de Outubro de 2022
- [Pogacean u. a. 2015] POGACEAN, Cristian ; BROSCHEI, Sebastian ; SÜSS, Georg: Implementando Comunicação OPC UA Determinística. In: *Westcon* (2015)
- [Santos u. a. 2015] SANTOS, G. A. ; SILVA, A. M. ; COSTA, H. W. S. ; FERREIRA, T. R. P. P. ; LIMA, M. F.: BANCADA DIDÁTICA BASEADA EM CLP COMPACTLOGIX DA ROCKWELL PARA USO NAS DISCIPLINAS DE AUTOMAÇÃO E INSTRUMENTAÇÃO INDUSTRIAL. In: *TECNOLOGIA INFORMAÇÃO 2* (2015), 10, Nr. 3
- [Santos 2008] SANTOS, Tiago Torres d.: *Utilização do protocolo OPC-DA na integração de um sistema OTS*, Pontifícia Universidade Católica do Rio Grande do Sul, Diplomarbeit, 2008
- [Sharma 2018] SHARMA, Aakansha: *Modeling, Simulation and Intelligent Control of Coupled Tank System*, 2018
- [Siemens] SIEMENS: *Siemens apresenta novidades do TIA Portal V15*.
<https://new.siemens.com/br/pt/empresa/release/siemens-apresenta-novidades-do-tia-portal-v15.html>. – Acessado em 13 de Novembro de 2022
- [Siemens 2019] SIEMENS: *PID control with PID_Compact*, 2019

- [Silva u. a. 2021] SILVA, Natasha ; CERVANTES, Ania ; GOMES, Flavio ; AGUEDO, Reinel: Supervision and control system of a didactic plant in the teaching of industrial automation and control. In: *2021 IEEE International Conference on Automation/XXIV Congress of the Chilean Association of Automatic Control (ICA-ACCA)*, 2021, S. 1–7