

UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

## **Trabalho de Conclusão de Curso**

**Controle Preditivo por Subespaços como um serviço em  
nuvem**

Victor Rafael Bezerra Maciel

Campina Grande - PB

Dezembro de 2022

Victor Rafael Bezerra Maciel

## **Controle Preditivo por Subespaços como um serviço em nuvem**

Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Engenheiro Eletricista.

Área de Concentração: Controle de Sistemas Dinâmicos

Universidade Federal de Campina Grande - UFCG

Centro de Engenharia Elétrica e Informática - CEEI

Departamento de Engenharia Elétrica - DEE

Coordenação de Graduação em Engenharia Elétrica - CGEE

Rafael Bezerra Correia Lima, D.Sc.

(Orientador)

Campina Grande - PB

Dezembro de 2022

Victor Rafael Bezerra Maciel

## **Controle Preditivo por Subespaços como um serviço em nuvem**

*Trabalho de Conclusão de Curso submetido à Coordenação de Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Engenheiro Eletricista.*

Trabalho aprovado. Campina Grande - PB, 20 de Dezembro de 2022:

---

**George Acioli Júnior**

Universidade Federal de Campina Grande  
Avaliador

---

**Rafael Bezerra Correia Lima**

Universidade Federal de Campina Grande  
Orientador

Campina Grande - PB

Dezembro de 2022

*Este trabalho é dedicado aos meus pais, Moizes e Viviane, que batalharam todos os dias para que eu tivesse a oportunidade de cursar engenharia, ao meu irmão Vinícius, que me ensinou a amar e a ver a vida com leveza e a todos os meus amigos pelo companheirismo e apoio ao longo dessa jornada.*

# Agradecimentos

Muitas pessoas foram fundamentais durante a minha trajetória. Gostaria de agradecer e dedicar este trabalho, em especial, as seguintes pessoas:

Aos meus pais, Moizes e Viviane, por todo amor, apoio, compreensão e conselhos que me deram ao longo da vida.

Ao meu irmão Vinicius (*In Memoriam*), que sempre me apoiou e tanto me ensinou ao longo de sua breve vida.

Aos amigos que fiz no ensino médio Thiago, Rhamon, Nicolas, Natália e Raíssa, que sempre me divertem e escutam meus problemas.

Aos amigos que fiz na universidade, em especial a minha amiga Meirelly e a todos os membros do grupo "Não Cola", que tornaram esses anos da graduação mais fáceis e felizes.

A toda a minha família, em especial aos meus avós.

Ao professor Rafael, meu orientador de PIBIC e do TCC, por ter me dado uma oportunidade no LIEC, por toda paciência e apoio ao longo desses anos.

Por fim, agradeço a Deus por me dar força e sabedoria para passar por todos os momentos de dificuldade.

*“Ó, vida futura! Nós te criaremos.”*  
*Carlos Drummond de Andrade*

# Resumo

É proposto nesse trabalho um algoritmo para controladores preditivos por subespaços para sistemas em rede, que pode ser estendido para sistemas de controle em nuvem. O controle preditivo por subespaços é uma metodologia de controle baseada em dados, que une as vantagens do controle preditivo baseado em modelo, como robustez, capacidade de lidar com restrições e simplicidade no tratamento de sistemas multivariáveis, mas eliminando a necessidade de um modelo do processo, que é uma etapa onerosa. O controle preditivo por subespaços faz uso de dados experimentais do processo para construir a sua lei de controle. Para tanto, o algoritmo de controle utiliza projeções do método de identificação por subespaços, que necessitam somente de decomposições QR de matrizes de Hankel dos dados de entrada e saída. O esquema de controle em rede proposto consiste em um controlador preditivo para gerar uma sequência de sinais de controle futuros e um compensador de atrasos de rede no lado do atuador. A arquitetura é toda baseada em dados, o que minimiza os custos de identificação e potencializa o uso prático do controlador em rede. Um novo método de sintonia para controladores preditivos por subespaços é proposto, empregando otimização multiobjetivo para obter parâmetros de sintonia que atendam requisitos expressos como constantes de tempo desejadas. São apresentados tópicos ligados à identificação por subespaços em malha aberta e ao uso desses dados para construção das matrizes de predição. Aspectos relacionados à sintonia e performance de controladores preditivos são discutidos, com a aplicação do método de sintonia desenvolvido a modelos univariáveis genéricos e a um modelo simplificado de um fracionador de petróleo pesado. Por fim, é apresentado a concepção de um controlador preditivo por subespaços em rede que é aplicado em um modelo univariável e multivariável de primeira ordem por meio de simulações no ambiente do Simulink®.

**Palavras-chave:** controle preditivo, identificação por subespaços, controle voltado a dados, computação em nuvem, controle em rede, IoT, sintonia de controlador.

# Abstract

This work proposes an algorithm for subspace predictive controllers for networked systems, which can be extended to cloud control systems. Subspace predictive control is a data-based control methodology, which combines the advantages of model-based predictive control, such as robustness, ability to deal with constraints, and simplicity in the treatment of multivariable systems, but eliminating the need for a process model, which is a time-consuming step. Subspace predictive control makes use of experimental process data to construct its control law. For that, the control algorithm uses projections of subspace identification methods, which require only QR decompositions of Hankel matrices of input and output data. The proposed network control scheme consists of a predictive controller to generate a sequence of future control signals and a network delay compensator on the actuator side. The architecture is entirely data-based, which minimizes identification costs and potentializes the practical use of the networked controller. A new tuning method for subspace predictive controllers is proposed, employing multiobjective optimization to obtain tuning parameters that meet requirements expressed as desired time constants. Topics related to the identification of open-loop subspaces and the use of these data for the construction of prediction matrices are presented. Aspects related to the tuning and performance of predictive controllers are discussed, with the application of the developed tuning method to generic univariate models and to a simplified model of a heavy oil fractionator. Finally, the design of a predictive controller for networked subspaces is presented, which is applied in univariable and multivariable first-order models through simulations in Simulink<sup>®</sup>.

**Keywords:** predictive control, subspace identification, data-driven control, cloud computing, networked control, IoT, controller tuning.



# Lista de ilustrações

Figura 1 – Representação gráfica do CCS . . . . .	2
Figura 2 – Propriedade geométrica entre $\mathbf{Y}_f$ , $\mathbf{X}_f$ e $\mathbf{U}_f$ . . . . .	7
Figura 3 – Evolução das variáveis de um sistema com controle preditivo . . . . .	11
Figura 4 – Algoritmo: controle preditivo por subespaços baseado em dados . . . . .	15
Figura 5 – Representação do espaço das variáveis de decisão e o espaço objetivo correspondente . . . . .	18
Figura 6 – Representação do conjunto ótimo de Pareto e da Fronteira de Pareto . . . . .	19
Figura 7 – Representação do ponto de Utopia e da solução de compromisso . . . . .	20
Figura 8 – Algoritmo: sintonia do SPC baseada na solução de compromisso . . . . .	23
Figura 9 – Arquitetura típica de um NCS . . . . .	25
Figura 10 – Algoritmo: compensador para lidar com problemas de comunicação no canal direto . . . . .	27
Figura 11 – Representação do fluxo de dados no bloco controlador e preditor SPC . . . . .	29
Figura 12 – Algoritmo: controlador e preditor para lidar com problemas de comunicação no canal de realimentação . . . . .	30
Figura 13 – Experimento em malha aberta - FOPDT . . . . .	32
Figura 14 – Avaliação em malha fechada: $\tau = 2$ - FOPDT . . . . .	33
Figura 15 – Avaliação em malha fechada: $\tau = 4$ - FOPDT . . . . .	33
Figura 16 – Avaliação em malha fechada: $\tau = 8$ - FOPDT . . . . .	34
Figura 17 – Experimento em malha aberta - SOPDT . . . . .	35
Figura 18 – Avaliação em malha fechada: $\tau = 2$ - SOPDT . . . . .	36
Figura 19 – Avaliação em malha fechada: $\tau = 3.5$ - SOPDT . . . . .	36
Figura 20 – Avaliação em malha fechada: $\tau = 5.7$ - SOPDT . . . . .	37
Figura 21 – Experimento em malha aberta - Fracionador de petróleo pesado 2x2 . . . . .	38
Figura 22 – Resposta em malha fechada: $\tau_1 = 40$ , $\tau_2 = 50$ - Fracionador de petróleo pesado 2x2 . . . . .	39
Figura 23 – Sinais de controle: $\tau_1 = 40$ , $\tau_2 = 50$ - Fracionador de petróleo pesado 2x2 . . . . .	39
Figura 24 – Resposta em malha fechada: $\tau_1 = 25$ , $\tau_2 = 32$ - Fracionador de petróleo pesado 2x2 . . . . .	40
Figura 25 – Sinais de controle: $\tau_1 = 25$ , $\tau_2 = 32$ - Fracionador de petróleo pesado 2x2 . . . . .	40
Figura 26 – Distribuição dos atrasos ao longo do tempo: NCS - FOPDT . . . . .	41
Figura 27 – Resposta em malha fechada: NCS - FOPDT . . . . .	42
Figura 28 – Experimento em malha aberta - 2x2 FOPDT . . . . .	43
Figura 29 – Distribuição dos atrasos ao longo do tempo: NCS - 2x2 FOPDT . . . . .	44
Figura 30 – Resposta em malha fechada: NCS - 2x2 FOPDT . . . . .	45
Figura 31 – Sinais de controle em malha fechada: NCS - 2x2 FOPDT . . . . .	45

# Lista de tabelas

Tabela 1 – Parâmetros de sintonia e métricas - FOPDT . . . . .	33
Tabela 2 – Parâmetros de sintonia e métricas - SOPDT . . . . .	35
Tabela 3 – Parâmetros de sintonia e métricas - Fracionador de petróleo pesado 2x2	38
Tabela 4 – Métricas - SPC/NCS: FOPDT . . . . .	42
Tabela 5 – Métricas - SPC/NCS: 2x2 FOPDT . . . . .	46

# Lista de abreviaturas e siglas

NCS	<i>Networked Control System</i>
CCS	<i>Cloud Control Systems</i>
MPC	<i>Model Predictive Control</i>
SPC	<i>Subspace Predictive Control</i>
LQR	<i>Linear Quadratic Regulator</i>
LQG	<i>Linear Quadratic Gaussian</i>
MOO	<i>Multiobjective Optimization</i>
PRBS	<i>Pseudorandom Binary Sequence</i>
PSO	<i>Particle Swarm Optimization</i>
IoT	<i>Internet of Things</i>
RFID	<i>Radio-Frequency Identification</i>
CAC	<i>Controller to Actuator Channel</i>
SCC	<i>Sensor to Controller Channel</i>
FOPDT	<i>First Order Plus Dead Time</i>
SOPDT	<i>Second Order Plus Dead Time</i>
IAE	<i>Integrated Absolute Error</i>
TV	<i>Total Variation</i>
RMSE	<i>Root Mean Squared Error</i>

# Lista de símbolos

$U_p$	Matriz de Hankel em bloco de sinais passados
$U_f$	Matriz de Hankel em bloco de sinais futuros
$\Gamma_i$	Matriz de observabilidade estendida
$\Delta_i$	Matriz de controlabilidade estendida invertida
$H_i$	Matriz triangular inferior de Toeplitz
$\Sigma ()$	Matriz de covariância
$A/B$	Projeção ortogonal do espaço linha da matriz A no espaço linha da matriz B
$\Delta$	Operador diferencial: $1 - q^{-1}$
$A \otimes B$	Produto de Kronecker de A por B
$A^\dagger$	Pseudo-inversa da matriz A
$\text{diag}(a, \dots, z)$	Matriz diagonal
$\mathbf{x}$	Vetor de variáveis de decisão
$\mathbf{F}(\mathbf{x})$	Vetor de funções objetivo
$\mathbf{F}^\circ$	Ponto de Utopia

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1</b>
<b>1.1</b>	<b>Objetivos</b>	<b>3</b>
1.1.1	Objetivos Gerais	3
1.1.2	Objetivos Específicos	3
<b>1.2</b>	<b>Organização do trabalho</b>	<b>4</b>
<b>2</b>	<b>CONTROLE PREDITIVO POR SUBESPAÇOS</b>	<b>5</b>
<b>2.1</b>	<b>Identificação por subespaços</b>	<b>5</b>
2.1.1	Definições	5
2.1.2	Propriedades geométricas de sistemas determinísticos	7
2.1.3	Cálculos das matrizes do sistema	9
<b>2.2</b>	<b>Controle Preditivo por Subespaços</b>	<b>9</b>
2.2.1	Contextualização com o Controle Preditivo baseado em Modelo	9
2.2.2	Introdução ao Controle Preditivo por Subespaços	11
2.2.3	Lei de controle	12
<b>3</b>	<b>SINTONIA DE CONTROLADOR PREDITIVO POR SUBESPAÇOS</b>	<b>16</b>
<b>3.1</b>	<b>Contextualização do problema</b>	<b>16</b>
<b>3.2</b>	<b>Otimização multiobjetivo</b>	<b>17</b>
3.2.1	Definições	17
3.2.2	Solução de compromisso	18
<b>3.3</b>	<b>Sintonia baseada na solução de compromisso</b>	<b>20</b>
3.3.1	Determinação dos horizontes $N_p$ e $N_c$	20
3.3.2	Determinação das matrizes $Q_k$ e $R_k$	21
<b>4</b>	<b>SISTEMAS DE CONTROLE EM REDE</b>	<b>24</b>
<b>4.1</b>	<b>Introdução</b>	<b>24</b>
<b>4.2</b>	<b>Compensação do atraso e a perda de dados no canal direto</b>	<b>26</b>
<b>4.3</b>	<b>Compensação do atraso e a perda de dados no canal de realimentação</b>	<b>28</b>
<b>5</b>	<b>RESULTADOS</b>	<b>31</b>
<b>5.1</b>	<b>Sintonia SPC</b>	<b>32</b>
5.1.1	Caso univariável: modelo FOPDT	32
5.1.2	Caso univariável: modelo SOPDT	34
5.1.3	Caso multivariável: modelo 2x2 fracionador de petróleo pesado	35
<b>5.2</b>	<b>SPC em rede</b>	<b>41</b>

5.2.1	Caso univariável: modelo FOPDT . . . . .	41
5.2.2	Caso multivariável: 2x2 FOPDT . . . . .	42
<b>6</b>	<b>CONCLUSÕES . . . . .</b>	<b>47</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS . . . . .</b>	<b>49</b>

# 1 Introdução

Computação em nuvem é um paradigma emergente que oferece *hardware* e *software* como serviço. O surgimento em larga escala de plataformas de computação em nuvem traz grandes oportunidades sem precedentes para plantas industriais complexas, viabilizando o aumento da eficiência dos sistemas de controle e redução de custos operacionais. Além disso, essa nova arquitetura permite a integração com uma série de serviços de nuvem que estão em plena evolução e tem grande potencial, tais como, *cloud analytics*, *machine learning*, armazenamento de dados em larga escala, supervisão, entre outros (CHENARU et al., 2015).

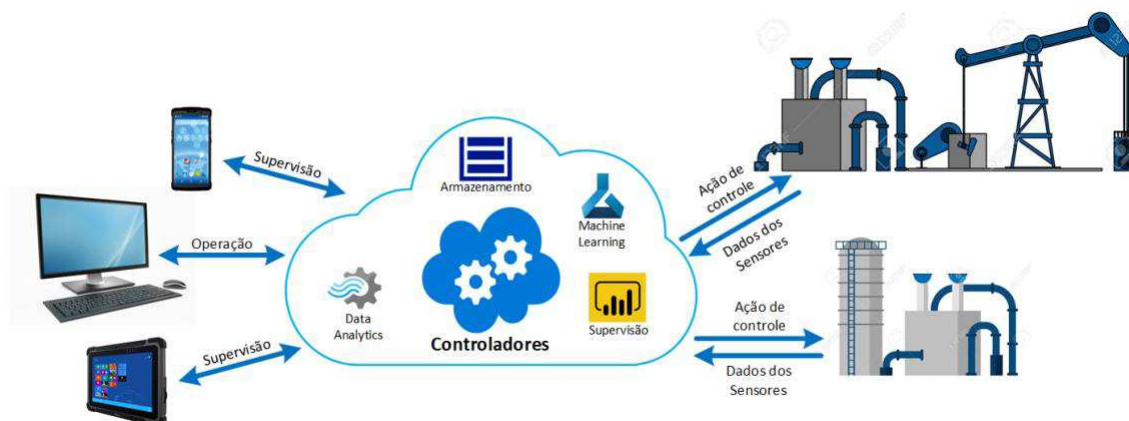
A adoção de computação em nuvem permite a integração vertical dos diferentes níveis de produção industrial, desde o nível de supervisão até o nível de planejamento corporativo, possibilitando melhoria de desempenho, redução de custos, maior flexibilidade, tolerância a falhas, escalabilidade e agilidade (CHENARU et al., 2015), (XIA, 2015). Além disso, a oferta de serviços de automação industrial em nuvem reduz o tempo e o custo na implantação de novos sistemas, uma vez que os atuais sistemas de automação industrial são complexos e requerem grande esforço humano para serem construídos (HEGAZY; HEFEEDA, 2014).

Tecnologias de comunicação em rede têm se desenvolvido bastante e, recentemente, seu uso em sistemas de controle está cada vez mais comum. Este tipo de sistema de controle no qual controlador, atuadores, sensores e sistema supervisor são conectados por meio de redes de comunicação é chamado na literatura de Sistema de Controle em Rede (NCS). Esta arquitetura é desenvolvida para lidar com os problemas inerentes à comunicação em rede, tais como atrasos intermitentes, perdas de pacotes, problemas de quantização, etc, que têm grande impacto em aplicações em tempo real tais como controle de processos dinâmicos (XIA, 2015).

Arquiteturas de controle em nuvem também apresentam os problemas com os quais o NCS tem que enfrentar. Deste modo, uma nova arquitetura de sistemas de controle é concebida, na qual estende a metodologia NCS para uma arquitetura em nuvem, chamado de Sistemas de Controle em Nuvem (CCS). Sendo assim, o CCS é capaz de modelar o algoritmo de controle como um serviço de nuvem, desacoplando através da internet, os operadores, controladores, sensores e atuadores, além de oferecer as vantagens citadas anteriormente para aplicações em nuvem. É ilustrado na Fig. 1 o diagrama típico de um sistema de controle em nuvem.

Uma metodologia de controle bastante empregada em NCSs é o controle preditivo. O Controle Preditivo baseado em Modelo (MPC) teve origem no final da década de 1970,

Figura 1 – Representação gráfica do CCS



Fonte: Autoria própria

e vem se desenvolvendo bastante desde então. O MPC compreende um vasto conjunto de metodologias que empregam modelos do processo de modo a buscar o sinal de controle ótimo que minimize uma certa função de custo (CAMACHO; ALBA, 2013). As metodologias MPC apresentam uma série de vantagens que as tornam bastante atrativas, tais como: apresentar uma formulação que é aplicável a sistemas multivariáveis de forma simples; ter a capacidade de lidar com diferentes tipos de restrições em um sistema de controle; ser capaz de executar otimização de processos de forma *online* (WANG, 2009). A maioria dos métodos MPC baseiam-se nas seguintes ideias: uso explícito de modelos para predição dos sinais de saída do sistema em instantes futuros; lei de controle dependente das predições; cálculo da sequência de controle por meio da minimização de uma função de custo; a utilização de um horizonte de predição móvel que se desloca a cada instante de tempo.

Por utilizar explicitamente o modelo do processo para realizar predições, o primeiro passo para o projeto de um controlador MPC é a obtenção de um modelo. Os processos de identificação, entretanto, apresentam algumas restrições. A identificação de modelos paramétricos, envolvem necessariamente uma ordem reduzida em uma estrutura pré-especificada, que pode omitir parte da complexidade e dinâmica do processo (HUANG; KADALI, 2008). Além disto, esta etapa de modelagem é o passo mais oneroso e demorado do projeto de controle. Uma maneira de evitar a necessidade de um modelo explícito é utilizando abordagens voltadas a dados. Neste tipo de metodologia, as matrizes de predição são obtidas diretamente dos dados do processo, eliminando a necessidade de escolha de um modelo para a planta. Neste contexto, surge uma alternativa que contempla as vantagens dos controladores MPC, mas sem necessitar de um modelo do processo: o método de Controle Preditivo por Subespaços (SPC). Métodos SPC utilizam como preditores um certo conjunto de matrizes intermediárias do processo de identificação por subespaços.

O controle preditivo é particularmente interessante no contexto de sistemas de



controle em rede, pois com esta formulação é possível obter sinais de controle ótimos em uma janela futura. Tais sinais podem ser utilizados para lidar com eventuais problemas de comunicação comuns em sistemas de controle em rede, tais como atrasos intermitentes e perdas de pacotes. Como o uso do modelo no controle preditivo limita-se a construir os preditores, abordagens voltadas a dados como o SPC têm bastante potencial de uso em sistemas de controle em rede, visto que somente com dados do processo é possível obter os preditores necessários para a lei de controle, além de que podem aproveitar serviços oferecidos em nuvem como *machine learning* e *data analytics*.

Apesar de promissora, esta abordagem carece de métodos de sintonia para o ajuste dos diversos parâmetros do controlador do tipo SPC consolidados na literatura. A maioria dos métodos de sintonia destinam-se a métodos MPC e exigem que o projetista possua um modelo do processo, algo que no contexto do SPC é inadequado. Neste trabalho é proposto adicionalmente um método de sintonia de controladores preditivos por subespaços, no qual os parâmetros de sintonia deste tipo de controlador são obtidos por meio de um processo de otimização multiobjetivo.

## 1.1 Objetivos

### 1.1.1 Objetivos Gerais

Desenvolvimento de uma metodologia de controle preditivo baseado em dados, através de subespaços, como um serviço de nuvem. O desafio encontra-se em desenvolver uma variação de algoritmo SPC que acomode os atrasos de comunicação inerentes em sistemas distribuídos através da internet e garanta performance e estabilidade mesmo no cenário de perda de pacotes.

### 1.1.2 Objetivos Específicos

No contexto dos objetivos gerais, os objetivos específicos são definidos da seguinte forma:

- Desenvolver um método de sintonia para obtenção de parâmetros ótimos de sintonia do SPC a partir de requisitos temporais;
- Desenvolver uma variação de algoritmo SPC que inclua atrasos de comunicação variáveis ao longo do tempo;
- Validar o método de sintonia proposto para diferentes requisitos desejados;

- Validar analiticamente o algoritmo de controle em rede proposto para garantir estabilidade e performance no cenário de atrasos e perdas de pacotes, inerentes a aplicações com conexão em rede;
- Validar os algoritmos desenvolvidos em simulações computacionais e aplicações industriais.

## 1.2 Organização do trabalho

O trabalho está estruturado em 5 capítulos, incluindo este introdutório, conforme a seguir.

O **Capítulo 2** apresenta o referencial teórico para o Controle Preditivo por Subespaços. A teoria de identificação por subespaços é detalhada, mostrando que, a partir de propriedades geométricas de matrizes de dados é possível obter um modelo do sistema. A partir disto, é apresentado que matrizes intermediárias do processo de identificação podem ser utilizadas como matrizes de predição, sendo então empregadas na definição da lei de controle preditiva.

O **Capítulo 3** propõe um novo método de sintonia, adaptando conceitos e ideias encontrados na literatura de sintonia MPC. O algoritmo de sintonia permite que o usuário defina constantes de tempo como requisitos a serem alcançados. Os parâmetros de sintonia são obtidos a partir da resolução de um problema de otimização multiobjetivo. Aspectos ligados a escolha de alguns parâmetros são discutidas, bem como no impacto que eles têm na dinâmica em malha fechada.

O **Capítulo 4** aborda a solução proposta no âmbito SPC para lidar com atrasos intermitentes de comunicação e perdas de pacotes presentes em sistemas de controle em rede. São mostradas as adaptações necessárias tanto no controlador, que agora se comunica com os atuadores e sensores por meio de uma rede de comunicação, como no lado do atuador, onde um *buffer* é implementado para receber os pacotes de dados com os sinais de controle.

O **Capítulo 5** apresenta os resultados alcançados para o método de sintonia, que é aplicada para alguns tipos de processo e para diferentes requisitos. Métricas de performance são apresentadas e discutidas de modo a avaliar as dinâmicas alcançadas em malha fechada. Por fim, são mostrados alguns resultados para a solução proposta para o SPC em rede, comparando-o com o caso sem adaptações.

Por fim, no **Capítulo 6** aborda as conclusões do trabalho e apresenta propostas futuras de modo a complementar as atividades desenvolvidas.

## 2 Controle Preditivo por Subespaços

### 2.1 Identificação por subespaços

O modelo é uma representação matemática que busca descrever adequadamente um sistema físico, biológico ou de informação. Em geral, para a engenharia a obtenção do modelo é apenas uma das etapas de um desenvolvimento mais amplo, como em simulações, sistemas de predição, diagnóstico de falhas, processamento de sinais, monitoramento de qualidade e segurança, estimação de estados e, por fim, controle de sistemas baseado em modelos. Em um sentido prático, a qualidade do modelo é determinada pelo objetivo ao qual o modelo pretende alcançar, isto é, incertezas do modelo são permitidas desde de que o sistema desenvolvido seja robusto. Sendo assim, devido a complexidade e o conhecimento prévio limitado do sistema, a modelagem física é bastante onerosa e, para muitas aplicações, a forma mais apropriada e eficiente de se modelar um sistema dinâmico é por meio de dados experimentais, no que é chamado identificação de sistemas (KEESMAN, 2011), (OVERSCHEE; MOOR, 1996). Métodos de identificação por subespaços são um tipo de método paramétrico de identificação, nos quais o modelo do sistema a ser identificado é obtido no formato em espaço de estados (OVERSCHEE; MOOR, 1996).

#### 2.1.1 Definições

Inicialmente, considera-se o modelo determinístico, isto é, sem influência de ruído nas medições, no qual estão disponíveis  $s$  medições dos  $m$  sinais de entrada  $u(k)$  e dos  $l$  sinais de saída  $y(k)$  em malha aberta:

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) \\y(k) &= Cx(k) + Du(k)\end{aligned}\tag{2.1}$$

onde os parâmetros a serem encontrados são as matrizes  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times m}$ ,  $C \in \mathbb{R}^{l \times n}$  e  $D \in \mathbb{R}^{l \times m}$ . Além disso, a ordem  $n$  do sistema é desconhecida.

Na literatura sobre os métodos de identificação por subespaços, são utilizadas matrizes de Hankel para compactar os dados de entrada e saída do sistema. Este tipo de matriz tem a propriedade que cada diagonal ascendente da esquerda para a direita possui os mesmos elementos. A matriz de Hankel de entrada  $\mathbf{U}$  é dada por:

$$\mathbf{U} = \begin{pmatrix} \mathbf{U}_p \\ \mathbf{U}_f \end{pmatrix} = \begin{pmatrix} \mathbf{U}_{0|i-1} \\ \mathbf{U}_{i|2i-1} \end{pmatrix} = \begin{pmatrix} u(0) & u(1) & \dots & u(j-1) \\ u(1) & u(2) & \dots & u(j) \\ \vdots & \vdots & & \vdots \\ u(i-1) & u(i) & \dots & u(i+j-2) \\ u(i) & u(i+1) & \dots & u(i+j-1) \\ u(i+1) & u(i+2) & \dots & u(i+j) \\ \vdots & \vdots & & \vdots \\ u(2i-1) & u(2i) & \dots & u(2i+j-2) \end{pmatrix}, \quad (2.2)$$

onde o número de linhas  $i$  é um índice definido pelo usuário que deve ser pelo menos maior que a ordem máxima do sistema que se deseja identificar ( $i \geq n$ ). O número de colunas  $j$  é simplesmente dado por  $j = s - 2i + 1$ , isto é, são utilizados todos os dados disponíveis do processo. De acordo com a convenção adotada, o subscrito “p” denota que são dados passados e o subscrito “f” por sua vez denota que são dados futuros (OVERSCHEE; MOOR, 1996). A matriz de Hankel de saída  $\mathbf{Y} = (\mathbf{Y}_p \mid \mathbf{Y}_f)^T$  é definida de modo similar a equação (2.2).

A partir das matrizes de Hankel de entrada e saída que contêm dados passados, obtém-se a seguinte matriz na notação de Willems:

$$\mathbf{W}_p = \begin{pmatrix} \mathbf{U}_p \\ \mathbf{Y}_p \end{pmatrix} \quad (2.3)$$

Os algoritmos de identificação por subespaços utilizam-se das matrizes de controlabilidade e observabilidade para calcular as matrizes que descrevem o modelo do sistema. Assumindo que o sistema é observável, a matriz de observabilidade estendida  $\Gamma_i$  tem posto  $n$  e é dada por:

$$\Gamma_i = \begin{pmatrix} C & CA & CA^2 & \dots & CA^{i-1} \end{pmatrix}^T \quad (2.4)$$

Assumindo também que o sistema é controlável, a matriz de controlabilidade estendida invertida  $\Delta_i$  é definida como:

$$\Delta_i = \begin{pmatrix} A^{i-1}B & A^{i-2}B & \dots & AB & B \end{pmatrix} \quad (2.5)$$

Por fim, define-se a matriz triangular inferior de Toeplitz  $H_i$  como sendo:

$$H_i = \begin{pmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \dots & D \end{pmatrix} \quad (2.6)$$

### 2.1.2 Propriedades geométricas de sistemas determinísticos

Conforme mostrado por (OVERSCHEE; MOOR, 1996), é possível obter as seguintes relações entre as matrizes de Hankel de entrada e saída:

$$\mathbf{Y}_p = \Gamma_i \mathbf{X}_p + H_i \mathbf{U}_p \quad (2.7)$$

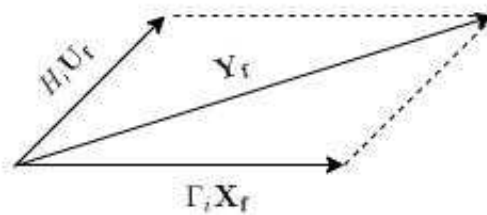
$$\mathbf{Y}_f = \Gamma_i \mathbf{X}_f + H_i \mathbf{U}_f \quad (2.8)$$

$$\mathbf{X}_f = A_i \mathbf{X}_p + \Delta_i \mathbf{U}_p \quad (2.9)$$

A prova das relações (2.7)-(2.9) podem ser obtidas diretamente do modelo em espaço de estados (2.1). As matrizes  $\mathbf{X}_p$  e  $\mathbf{X}_f$  denotam as matrizes de Hankel dos estados  $x(k)$  passados e futuros, respectivamente.

A interpretação geométrica de (2.8) é mostrada na Fig. 2. Nela, é possível notar que os vetores no espaço linha da matriz de saídas futuras  $\mathbf{Y}_f$  são uma soma da combinação linear dos vetores do espaço linha dos estados futuros  $\mathbf{X}_f$  com a combinação linear dos vetores do espaço linha das entradas futuras  $\mathbf{U}_f$ .

Figura 2 – Propriedade geométrica entre  $\mathbf{Y}_f$ ,  $\mathbf{X}_f$  e  $\mathbf{U}_f$



Fonte: Autoria própria.

De acordo com as equações (2.7)-(2.9), é possível notar que a sequência de estados  $\mathbf{X}_f$  e a matriz de observabilidade estendida  $\Gamma_i$  podem ser determinados diretamente dos dados de entrada e saída, sem a necessidade de conhecer as matrizes  $A$ ,  $B$ ,  $C$  e  $D$ . Antes, entretanto, é necessário assumir as seguintes premissas (OVERSCHEE; MOOR, 1996):

1. O sinal de entrada  $u(k)$  é persistentemente excitante com ordem  $2i$ , isto é, a matriz de covariância dos sinais de entrada  $\Sigma(\mathbf{U}, \mathbf{U})$  possui posto igual a  $2mi$ . A grosso modo, um sinal é considerado persistentemente excitante se o seu espectro contém uma quantidade suficientemente grande de harmônicas (LJUNG, 1971);
2. A intersecção entre o espaço linha das matrizes de entradas futuras  $\mathbf{U}_f$  e de estados passados  $\mathbf{X}_p$  é vazia, ou seja, o sistema está em malha aberta.

Segundo as relações dadas por (2.7)-(2.9), é possível notar que a matriz de saídas futuras  $\mathbf{Y}_f$  pode ser predita por meio da matriz de entradas futuras  $\mathbf{U}_f$  e da matriz de

Hankel de entradas e saídas passadas  $\mathbf{W}_p$ . Assim, busca-se o preditor linear que seja solução do seguinte problema de mínimos quadrados (FAVOREEL; MOOR; GEVERS, 1999):

$$\hat{\mathbf{Y}}_f = L_w \mathbf{W}_p + L_u \mathbf{U}_f \quad (2.10)$$

$$\min_{L_w, L_u} = \left\| \mathbf{Y}_f - \begin{pmatrix} L_w & L_u \end{pmatrix} \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix} \right\|_F^2,$$

que pode ser reescrito em termos da projeção ortogonal do espaço linha de  $\mathbf{Y}_f$  no espaço linha de  $(\mathbf{W}_p \ \mathbf{U}_f)^T$ :

$$\hat{\mathbf{Y}}_f = \mathbf{Y}_f / \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix} = \mathbf{Y}_f \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix}^\dagger \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix} \quad (2.11)$$

Esta projeção pode ser calculada de maneira bastante numericamente robusta fazendo uso da seguinte decomposição QR:

$$\begin{pmatrix} \mathbf{W}_p \\ \mathbf{Y}_f \\ \mathbf{U}_f \end{pmatrix} = \begin{pmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \\ R_{31} & R_{32} & R_{33} \end{pmatrix} \begin{pmatrix} Q_1^T \\ Q_2^T \\ Q_3^T \end{pmatrix},$$

$$\hat{\mathbf{Y}}_f = L \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix} = \begin{pmatrix} L_w & L_u \end{pmatrix} \begin{pmatrix} \mathbf{W}_p \\ \mathbf{U}_f \end{pmatrix}, \quad (2.12)$$

onde:

$$L = \begin{pmatrix} L_w & L_u \end{pmatrix} = \begin{pmatrix} R_{31} & R_{32} \end{pmatrix} \begin{pmatrix} R_{11} & 0 \\ R_{21} & R_{22} \end{pmatrix}^\dagger, \quad (2.13)$$

com  $L_w \in \mathbb{R}^{il \times i(m+l)}$  sendo as primeiras  $i(m+l)$  colunas de  $L$  e  $L_u \in \mathbb{R}^{il \times im}$  correspondendo as demais colunas. É possível provar que, por meio de  $L_w$  é possível estimar  $\Gamma_i$  e  $\hat{\mathbf{X}}_f$ . Entretanto, antes disso, é preciso reduzir a dimensão de  $L_w$ , de modo a evitar que o ruído acabe sendo modelado como parte da dinâmica do sistema. Assim, por inspeção dos valores singulares dominantes de  $L_w$ , a ordem do modelo a ser identificado é obtida e  $L_w$  é aproximado por:

$$\begin{aligned} L_w &= \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix} \\ &\approx U_1 S_1 V_1^T \end{aligned} \quad (2.14)$$

É possível demonstrar que assumindo que o número de colunas das matrizes de Hankel de dados  $\mathbf{Y}_f$ ,  $\mathbf{U}_f$  e  $\mathbf{W}_p$  é suficientemente grande (idealmente  $j = \infty$ ), a matriz de observabilidade  $\Gamma_i$  e a estimativa dos estados  $\hat{\mathbf{X}}_f$  são dadas por:

$$\Gamma_i = U_1 S_1^{1/2} \quad (2.15)$$

$$\hat{\mathbf{X}}_f = S_1^{1/2} V_1^T \mathbf{W}_p \quad (2.16)$$

### 2.1.3 Cálculos das matrizes do sistema

Os algoritmos de identificação por subespaços diferem entre si sobre o modo como as matrizes  $A$ ,  $B$ ,  $C$  e  $D$  do sistema são calculadas. Alguns utilizam a matriz de observabilidade  $\Gamma_i$ , outros empregam a matriz estimada dos estados  $\hat{\mathbf{X}}_f$  ou mesmo usam uma combinação de ambos. Aqui, será empregado somente  $\hat{\mathbf{X}}_f$  obtido em (2.16) para o processo de identificação. Demais métodos podem ser encontrados em (OVERSCHEE; MOOR, 1996).

Definindo as seguintes matrizes:

$$\begin{aligned}\bar{X}_i &= [\hat{x}(i) \quad \hat{x}(i+1) \quad \dots \quad \hat{x}(i+j-2)] \\ \bar{X}_{i+1} &= [\hat{x}(i+1) \quad \dots \quad \hat{x}(i+j-2) \quad \hat{x}(i+j-1)] \\ \bar{U}_{i|i} &= [u(i) \quad \dots \quad u(i+j-2)] \\ \bar{Y}_{i|i} &= [y(i) \quad \dots \quad y(i+j-2)]\end{aligned}$$

Pode-se facilmente mostrar que, por meio destas matrizes, é possível obter a seguinte relação (OVERSCHEE; MOOR, 1996), (KATAYAMA, 2006):

$$\begin{bmatrix} \bar{X}_{i+1} \\ \bar{Y}_{i|i} \end{bmatrix} = \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} \bar{X}_i \\ \bar{U}_{i|i} \end{bmatrix} \quad (2.17)$$

que é um sistema linear de equações, que pode ser resolvida pelo método dos mínimos quadrados para determinar as matrizes  $A$ ,  $B$ ,  $C$  e  $D$ .

O método de identificação por subespaços traz algumas inovações em relação as abordagens mais clássicas. Em primeiro lugar, é possível observar que o único parâmetro definido pelo usuário é a ordem do sistema, que pode ser facilmente escolhido a partir da inspeção de valores singulares já mencionada na Seção 2.1.2, enquanto que os métodos clássicos contêm diversos parâmetros a serem definidos. A escolha da ordem do sistema, aliás, possibilita a redução da ordem do modelo de uma maneira bastante direta. Em segundo lugar, o método por subespaços são em geral mais rápidos do que boa parte dos métodos tradicionais, visto que empregam algoritmos rápidos como decomposição QR e decomposição em valores singulares, além de, por não ser um método iterativo, não apresenta problemas de convergência (OVERSCHEE; MOOR, 1996).

## 2.2 Controle Preditivo por Subespaços

### 2.2.1 Contextualização com o Controle Preditivo baseado em Modelo

Os conceitos que levaram ao desenvolvimento do controle preditivo moderno foram introduzidos inicialmente na década de 1960 por meio dos estudos do Regulador Quadrático

Linear (LQR) por Rudolf E. Kalman. Nele, o algoritmo do controlador consegue estabilizar planta lineares sem restrições por meio da minimização de uma função de custo quadrática que possui um horizonte infinito. A combinação do LQR com a poderosa teoria do Filtro de Kalman para estimação de estados a partir de dados de entrada e saída de processos resultou no Regulador Linear Gaussiano (LQG). Apesar de bastante promissora e ser capaz de controlar qualquer planta linear sem restrições, o impacto da contribuição em aplicação industriais na época foi pequeno, motivado pela não consideração de restrições às variáveis do processo, incertezas do modelo que impactam na robustez e, sobretudo, pelo desconhecimento ou descrença sobre o LQG por parte da comunidade de controle de processos da época. Neste sentido, surgiu na indústria um de uma metodologia de controle baseada em modelo mais geral, na qual a minimização da função de custo é resolvida *on-line* a cada execução de controle, considerando as restrições as variáveis do processo no problema de otimização (QIN; BADGWELL, 2003). Ainda na década de 1960, conceitos fundamentais como o horizonte de controle móvel foram propostos, mas também sem grande impacto na indústria (MORARI; LEE, 1999).

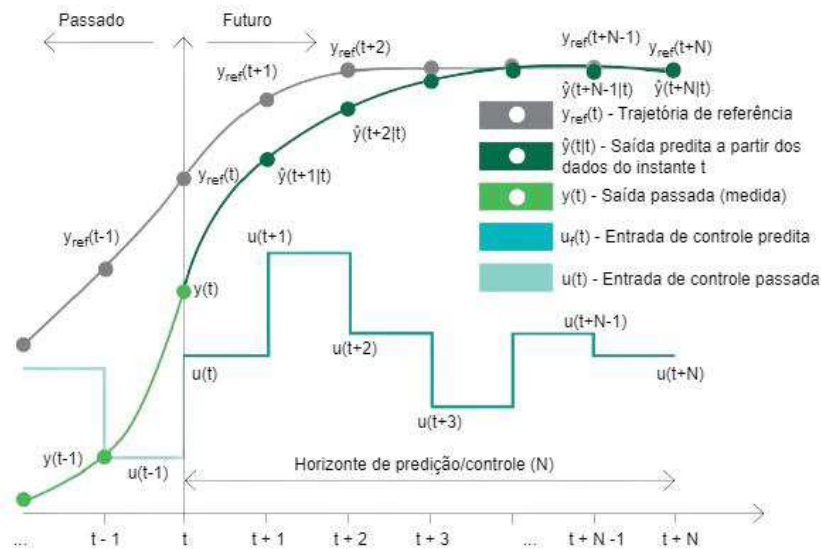
A primeira publicação de aplicações de controladores preditivos foi realizada por Richalet *et al.* em 1978, que foi aplicada a uma coluna de destilação em uma refinaria de petróleo, a um gerador a vapor de uma usina termelétrica e a uma planta completa de fabricação de PVC, no qual a maioria dos processos são controlados através da estratégia proposta (RICHALET *et al.*, 1978). Desde então, várias formulações e variações foram desenvolvidas e elas, em geral, utilizam um modelo explícito para prever os estados futuros do processo. A partir dos sinais preditos ao longo da janela de predição, sinais futuros de controle são calculados de modo que minimize uma certa função de custo e atenda certos critérios de restrições às variáveis. Uma revisão dos métodos de controle preditivo baseado em modelo (MPC) pode ser encontrada em (CAMACHO; ALBA, 2013) e a Fig. 3 mostra como se dá a evolução típica na dinâmica das variáveis de um controlador preditivo.

De maneira geral, os controladores preditivos baseado em modelo apresentam as seguintes características (CAMACHO; ALBA, 2013), (WANG, 2009):

1. As saídas futuras  $\hat{y}(t+k|t)$  do processo são preditas ao longo do horizonte  $N$ , chamado de horizonte de predição a cada instante de amostragem, utilizando o modelo do processo e dependem dos sinais de entrada e saída conhecidos (ou passados) do processo, além do sinal de controle futuro  $u(t+k|t)$ . O horizonte de predição determina quantos instantes de amostragem no futuro o serão previstos.
2. A partir das saídas preditas, o controlador calcula um conjunto de sinais de controle futuro  $u(t+k|t)$  ao longo do horizonte  $N$  que busque manter a saída predita  $\hat{y}(t+k|t)$  o mais próximo possível da trajetória de referência  $y_{ref}(t+k|t)$ . Em geral, o critério a ser minimizado é função do erro quadrático entre estes dois sinais e, na maioria



Figura 3 – Evolução das variáveis de um sistema com controle preditivo



Fonte: [ABB \(2021\)](#), Adaptado.

dos casos, também considera a variação do sinal de controle. Para o caso em que o modelo é linear e não há restrições às variáveis do processo, uma solução explícita pode ser obtida. Caso contrário, é necessário realizar um processo de otimização iterativo para obter  $u(t+k|t)$  ótimo.

3. O sinal de controle  $u(t|t)$  é aplicado ao processo e, o processo é repetido no próximo instante de amostragem, atualizando-se os vetores  $y(t)$  e  $u(t)$  passados com os novos valores obtidos. Este é o chamado princípio do horizonte de predição/controlador móvel.

As metodologias MPC apresentam diversas vantagens em relação a outras estratégias de controle tradicionais, tais como: apresentar uma formulação que é aplicável a sistemas multivariáveis de forma simples; ter a capacidade de lidar com diferentes tipos de restrições em um sistema de controle; ser capaz de executar otimização de processos de forma *online* ([WANG, 2009](#)).

### 2.2.2 Introdução ao Controle Preditivo por Subespaços

Por utilizar o modelo do processo para realizar previsões, a primeira etapa do projeto do controlador MPC é a obtenção de um modelo, que posteriormente é usado para a construção dos preditores utilizados na lei de controle. Conforme já citado na Seção 2.1, os modelos podem ser obtidos a partir de relações matemáticas ou a partir de algum processo de identificação e ambas são etapas onerosas para o projeto do sistema de controle ([MACIEL; LIMA, 2022](#)). Apesar de, em geral, serem mais simples do que a modelagem matemática, os processos de identificação apresentam algumas restrições. A identificação

de modelos paramétricos tais como função de transferência ou modelo em espaço de estados geralmente envolvem uma redução de ordem aplicada a uma estrutura pré-especificada, que pode omitir parte da complexidade da dinâmica do processo. Por outro lado, a modelagem não paramétrica como modelagem pela resposta ao impulso, ao degrau ou baseada na resposta em frequência, são limitados por restrições à dinâmica imposta pelo tipo de sinal de entrada aplicado e por restrições físicas impostas pelos atuadores (HUANG; KADALI, 2008). Revisões sobre os diversos métodos de identificação podem ser encontradas em (UNBEHAUEN; RAO, 1998), (KEESMAN, 2011) e (ISERMANN; MÜNCHHOF, 2011).

Para o projeto de controle do tipo MPC, o modelo é utilizado somente para a construção dos preditores utilizados pelo algoritmo de controle. Sendo assim, um modo de evitar o processo de obtenção de um modelo paramétrico é por meio de abordagens de controle preditivo voltadas a dados. De maneira geral, o paradigma de controle orientado a dados implica no desenvolvimento de algoritmos e estratégias de controle diretamente dos dados, sem nenhuma etapa intermediária de identificação de sistemas. Este tipo de método de controle pode ser especialmente útil em situações nas quais a identificação de um modelo de processo é difícil e demorada, como quando os dados são afetados por ruído ou apresentam dinâmica não linear (PERSIS; TESI, 2019). Em todas as formulações de controle orientado a dados, a principal questão diz respeito a como garantir a substituição efetiva de modelos do processo por dados. De maneira geral, é possível provar que dados persistentemente excitantes permitem a construir matrizes que podem substituir modelos de sistemas (WILLEMS et al., 2005), (PERSIS; TESI, 2019).

O controle preditivo por subespaços (SPC) é uma estratégia de controle voltada a dados e surge como uma alternativa aos tradicionais controladores preditivos baseados em modelo. Introduzido pela primeira vez em 1999, o SPC une o método de identificação por subespaços com o MPC (FAVOREEL; MOOR; GEVERS, 1999). É uma abordagem voltada a dados, que não necessita de um modelo, realizando as predições por meio das matrizes de preditores obtidas durante a fase intermediária do processo de identificação por subespaços (HUANG; KADALI, 2008). Neste sentido, o SPC consegue fornecer as mesmas vantagens do MPC, mas sem necessitar de um modelo do processo, tornando o projeto do controlador mais simplificado. Mais detalhes sobre a abordagem por subespaços em controladores preditivos pode ser encontrada em (HUANG; KADALI, 2008), (FAVOREEL; MOOR; GEVERS, 1999), (VAJPAYEE; MUKHOPADHYAY; TIWARI, 2017).

### 2.2.3 Lei de controle

O problema de otimização que caracteriza a lei de controle SPC sem restrições às variáveis do processos busca o sinal ótimo de controle  $u(k)$  que minimize a seguinte função

de custo:

$$\min_{u(k)} J = \sum_{i=1}^{N_p} (\hat{y}(k+i) - r(k+i))^T Q_k (\hat{y}(k+i) - r(k+i)) + \sum_{i=1}^{N_c} \Delta u(k+i)^T R_k \Delta u(k+i) \quad (2.18)$$

onde  $\Delta = (1 - q^{-1})$ , com  $q^{-1}$  denotando o operador de atraso unitário. O vetor  $\hat{y}(k)$  é a saída predita,  $Q_k \in \mathbb{R}^{l \times l}$ ,  $R_k \in \mathbb{R}^{m \times m}$  são as matrizes de ponderação do erro de rastreamento da referência e do esforço de controle, respectivamente,  $N_p$  é o horizonte de predição e  $N_c$  é o horizonte de controle ( $N_p, N_c \leq M$ ). A função de custo (2.18) pode ainda ser escrita sob a forma matricial. Definindo as sequências do esforço de controle futuro  $\Delta u_f$ , da saída predita  $\hat{y}_f$ , da referência  $r_f$  e os sinais de entrada e saída passados incrementais  $\Delta w_p$ :

$$\Delta u_f = \begin{pmatrix} \Delta u(k+1) \\ \Delta u(k+2) \\ \vdots \\ \Delta u(k+N_c) \end{pmatrix}, \quad \hat{y}_f = \begin{pmatrix} \hat{y}(k+1) \\ \hat{y}(k+2) \\ \vdots \\ \hat{y}(k+N_p) \end{pmatrix}, \quad r_f = \begin{pmatrix} r(k+1) \\ r(k+2) \\ \vdots \\ r(k+N_p) \end{pmatrix}$$

$$\Delta u_p = \begin{pmatrix} \Delta u(k-M+1) \\ \Delta u(k-M+2) \\ \vdots \\ \Delta u(k) \end{pmatrix}, \quad \Delta y_p = \begin{pmatrix} \Delta y(k-M+1) \\ \Delta y(k-M+2) \\ \vdots \\ \Delta y(k) \end{pmatrix}, \quad \Delta w_p = \begin{pmatrix} \Delta y_p \\ \Delta u_p \end{pmatrix}$$

E a nova equação da função de custo:

$$J = (\hat{y}_f - r_f)^T Q (\hat{y}_f - r_f) + \Delta u_f^T R \Delta u_f \quad (2.19)$$

onde as matrizes  $Q \in \mathbb{R}^{N_p l \times N_p l}$ ,  $R \in \mathbb{R}^{N_c m \times N_c m}$  são definidas por meio do produto de Kronecker ( $\otimes$ ):  $Q = I_{N_p} \otimes Q_k$  e  $R = I_{N_c} \otimes R_k$ , sendo  $Q$  positiva semi-definida e  $R$  é positiva definida (VAJPAYEE; MUKHOPADHYAY; TIWARI, 2017).

De acordo com a formulação matemática da identificação por subespaços obtida na Seção 2.1.3, é possível observar que o termo  $\hat{y}_f$  pode ser reescrito em termos das matrizes de predição  $L_w$  e  $L_u$ , sendo a primeira coluna de  $\hat{\mathbf{Y}}_f$  dado por (2.12). Neste caso, o horizonte do modelo  $M$  é igual ao número de linhas  $i$  da matriz de blocos de Hankel da literatura de identificação por subespaços. Os horizontes  $N_p$  e  $N_c$  são usados para truncar os preditores, resultando em

$$\hat{y}_f = \tilde{L}_w w_p + \tilde{L}_u u_f \quad (2.20)$$

e os preditores truncados são definidos, usando a notação do MATLAB<sup>®</sup>, como:

$$\tilde{L}_w = L_w(1 : N_p l, :) \quad (2.21)$$

$$\tilde{L}_u = L_u(1 : N_p l, 1 : N_c m) \quad (2.22)$$

O vetor  $\hat{y}_f$  encontrado em (2.20), entretanto, não pode ser utilizado na função de custo (2.19), visto que nesta pretende-se encontrada o sinal de controle incremental  $\Delta u$  ótimo. Entretanto, é possível escrever  $\hat{y}_f$  em termos de dados incrementais da forma:

$$\begin{aligned}\Delta \hat{y}_f &= \tilde{L}_w \Delta w_p + \tilde{L}_u \Delta u_f \\ \hat{y}_f &= \frac{1}{\Delta} \tilde{L}_w \Delta w_p + \frac{1}{\Delta} \tilde{L}_u \Delta u_f \\ &= F_1 y(k) + F_2 \tilde{L}_w \Delta w_p + F_2 \tilde{L}_u \Delta u_f \\ \hat{y}_f &= F_1 y(k) + \tilde{L}_{wI} \Delta w_p + \tilde{L}_{uI} \Delta u_f\end{aligned}\quad (2.23)$$

onde

$$\begin{aligned}F_1 &= \begin{pmatrix} I_l \\ I_l \\ \vdots \\ I_l \end{pmatrix} \in \mathbb{R}^{N_p l}, \quad F_2 = \begin{pmatrix} I_l & 0 & \dots & 0 \\ I_l & I_l & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ I_l & I_l & \dots & I_l \end{pmatrix} \in \mathbb{R}^{N_p l \times N_p l} \\ \tilde{L}_{wI} &= F_2 \tilde{L}_w \\ \tilde{L}_{uI} &= F_2 \tilde{L}_u\end{aligned}\quad (2.24)$$

$$\tilde{L}_{uI} = F_2 \tilde{L}_u \quad (2.25)$$

Substituindo (2.23) em (2.19) e tomando  $tr\left(\frac{\partial J}{\partial \Delta u_f}\right) = 0$ , encontra-se a seguinte lei de controle:

$$\Delta u_f = -K_e (y(k) - r(k+1)) - K_w \Delta w_p \quad (2.26)$$

onde as matrizes de ganho  $K_e \in \mathbb{R}^{N_c m \times 1}$  e  $K_w \in \mathbb{R}^{N_c m \times M(l+m)}$  são definidas como:

$$K_e = (R + \tilde{L}_{uI}^T Q \tilde{L}_{uI})^{-1} \tilde{L}_{uI}^T Q F_1 \quad (2.27)$$

$$K_w = (R + \tilde{L}_{uI}^T Q \tilde{L}_{uI})^{-1} \tilde{L}_{uI}^T Q \tilde{L}_{wI} \quad (2.28)$$

De acordo com o princípio do horizonte de predição móvel, aplica-se ao processo somente o primeiro elemento do vetor  $\Delta u_f$ , sendo recalculado a cada instante de amostragem. Neste caso, somente os  $m$  primeiros elementos de (2.26) são implementados, resultando na lei de controle:

$$\Delta u(k+1) = -k_e (y(k) - r(k+1)) - k_w \Delta w_p \quad (2.29)$$

$$u(k+1) = \Delta u(k+1) + u(k) \quad (2.30)$$

onde  $k_e$  e  $k_w$  são as primeiras  $m$  linhas de  $K_e$  e  $K_w$ , respectivamente. Mais detalhes da implementação de controladores preditivos por subespaços podem ser encontrados em (FAVOREEL; MOOR; GEVERS, 1999), (HUANG; KADALI, 2008) e (VAJPAYEE; MUKHOPADHYAY; TIWARI, 2017). A Fig. 4 mostra o pseudocódigo do cálculo dos ganhos do controlador preditivo por subespaços.

Figura 4 – Algoritmo: controle preditivo por subespaços baseado em dados

```

// Algoritmo para cálculo dos ganhos do controlador e lei de
// controle SPC para o controlador sem restrições às variáveis do
// processo
Dados:  $u, y$ 
// Obter os dados de entrada  $u$  e saída  $y$  de um experimento em malha
// aberta.
Entrada:  $N_p, N_c, R_k, Q_k$ 
// Escolher os parâmetros de sintonia: horizontes  $N_c$  e de predição
//  $N_p$ , matrizes de ponderação  $R_k, Q_k$ .
1 Montar as matrizes de Hankel  $\mathbf{U}_p, \mathbf{U}_f, \mathbf{Y}_p$  e  $\mathbf{Y}_f$ , de acordo com (2.2) e concatenar
// as matrizes de dados passados  $\mathbf{W}_p$  conforme (2.3).
2 Calcular os preditores  $L_w$  e  $L_u$  a partir da decomposição QR das matrizes de
// Hankel  $\mathbf{W}_p, \mathbf{U}_f$  e  $\mathbf{Y}_f$ , conforme (2.13).
3 Obter os preditores diferenciais truncados  $\tilde{L}_{wI}$  e  $\tilde{L}_{uI}$  a partir dos preditores  $L_w$  e
//  $L_u$ , conforme (2.21), (2.22), (2.24) e (2.25).
4 Calcular as matrizes de ganho  $K_e$  e  $K_w$  de acordo com (2.27) e (2.28),
// respectivamente, e truncar as  $m$  saídas
//  $m$  é a quantidade de sinais de controle
5 para cada  $k$  faça
6   |   Calcular o sinal de de controle de acordo com (2.29) e (2.30) e aplicá-lo ao
//   |   processo.
//  $k$  é o instante de amostragem

```

Fonte: Autoria própria

## 3 Sintonia de Controlador Preditivo por Subespaços

### 3.1 Contextualização do problema

Desde a sua introdução em meados de 1970, as formulações do MPC evoluíram consideravelmente, tornando-se mais flexível ao passo que sua complexidade e número de parâmetros de sintonia aumentaram. Nos algoritmos mais modernos, os parâmetros de sintonia incluem os horizontes de predição e controle; as matrizes que ponderam as magnitudes das variáveis controladas, as magnitudes das variáveis manipuladas e a agressividade do esforço de controle; parâmetros de trajetória de referência, pesos para as restrições suaves e o horizonte do modelo. Devido ao potencial de utilização do MPC no controle de processos complexos e com restrições às variáveis do processo, aliado ao grande número de parâmetros de sintonia, surgiram diversos métodos para sintonizar controladores MPC (ALHAJERI; SOROUGH, 2020). Alguns métodos apresentam-se na forma de equações que descrevem os parâmetros de sintonia em termos dos parâmetros da dinâmica do processo, tais como tempo morto, ganhos, constantes de tempo e o período de amostragem. Outros consistem em problemas de otimização cujas soluções são os valores dos parâmetros de sintonia. Existem ainda alguns métodos de sintonia heurísticos. Uma revisão detalhada de métodos de sintonia consolidados pode ser encontrada em (GARRIGA; SOROUGH, 2010), (ALHAJERI; SOROUGH, 2020).

De acordo com o que já foi comentando no Capítulo 2, as características apresentadas pelo SPC o tornam um método de controle bastante promissor. Entretanto, ao contrário das metodologias MPC, a abordagem por subespaços carece de métodos de sintonia consolidados na literatura que sejam capazes de realizar o ajuste dos diversos parâmetros deste tipo de controlador. Em geral, os métodos de sintonia dedicados ao MPC exigem que o projetista possua um modelo do processo, algo que no contexto do SPC é inadequado (MACIEL; LIMA, 2022). O trabalho mais consolidado na literatura que trata da sintonia do SPC, propõe a sintonia dos horizontes de predição e controle levando em conta critérios de estabilidade, deixando a cargo do projetista a escolha das matrizes de ponderação (SEDGHIZADEH; BEHESHTI, 2018).

Conforme é possível observar em (2.18), os controladores preditivos por subespaços apresentam quatro parâmetros que influenciam na dinâmica em malha fechada: as matrizes de ponderação  $Q_k$  e  $R_k$ , são as matrizes de ponderação do erro de rastreamento da referência e do esforço de controle, respectivamente; e os horizontes de predição  $N_p$  e de controle  $N_c$ . Entretanto, nos casos multivariáveis, a escolha das matrizes de ponderação adequadas

torna-se extremamente complexa, visto que as dimensões das matrizes dependem diretamente da quantidade de entradas e saídas do processo. Neste contexto, o trabalho propõe o desenvolvimento de um método de sintonia de controladores preditivos por subespaços, no qual os parâmetros de sintonia que definem a agressividade e robustez da resposta em malha fechada são obtidos por meio de um processo de otimização multiobjetivo. Adaptado do método de sintonia MPC de Yamashita *et al.*, a solução obtida busca uma relação de compromisso entre os objetivos concorrentes, de modo a aproximar as saídas do processo por um conjunto de trajetórias de referências definidas pelo projetista (YAMASHITA *et al.*, 2016). Mais detalhes da implementação do método de sintonia e discussão e avaliação dos resultados podem ser encontrados em (MACIEL; BARROS; LIMA, 2021) e (MACIEL; LIMA, 2022).

## 3.2 Otimização multiobjetivo

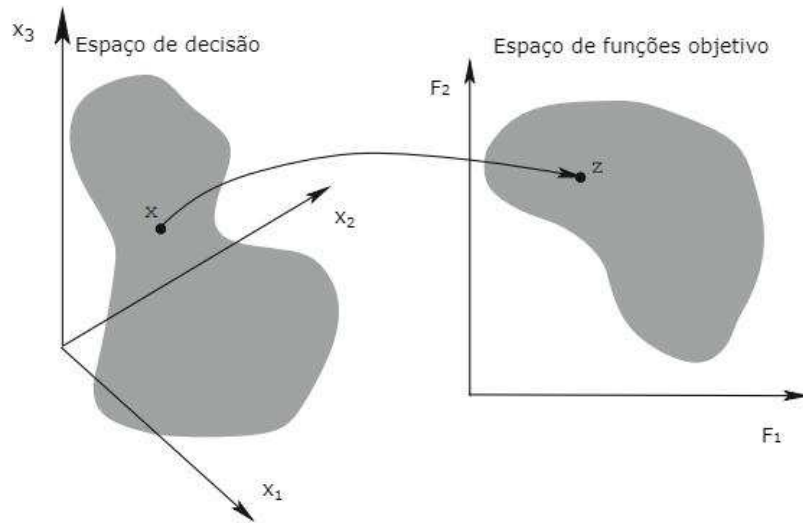
### 3.2.1 Definições

Otimização multiobjetivo (MOO) é o processo de otimizar de forma sistemática e simultânea um conjunto de funções objetivo que busca um conjunto de soluções ótimas ou uma solução única para problemas que apresentem objetivos conflitantes (MARLER; ARORA, 2004). A otimização multiobjetivo tem uma grande importância em diversas áreas, visto que muitos problemas de otimização do mundo real são modelados usando múltiplos objetivos concorrentes. Um problema geral de otimização multiobjetivo pode ser definido como:

$$\begin{aligned} \min_{\mathbf{x}} \mathbf{F}(\mathbf{x}) &= (F_1(\mathbf{x}) \ F_2(\mathbf{x}) \ \dots \ F_w(\mathbf{x}))^T \\ \text{sujeito a:} \quad &g_j(\mathbf{x}) \leq 0, j = 1, \dots, z \\ &h_l(\mathbf{x}) = 0, l = 1, \dots, e \end{aligned} \quad (3.1)$$

onde  $\mathbf{F}(\mathbf{x})$  é o vetor que compreende as  $w$  funções objetivo  $F_i(\mathbf{x})$ , que são funções das  $n_d$  variáveis de decisão  $\mathbf{x}$ . As funções  $g_j(\mathbf{x})$  e  $h_l(\mathbf{x})$  definem o espaço de projeto realizável, ou espaço das variáveis de decisão,  $\mathbf{X} = \{\mathbf{x} \in \mathbb{R}^{n_d} \mid g_j(\mathbf{x}) \leq 0, h_l(\mathbf{x}) = 0 \ \forall j = 1, \dots, z, l = 1, \dots, e\}$  e estão relacionados com as restrições de desigualdade e de igualdade, respectivamente. O conjunto de pontos determinados por  $\mathbf{F}(\mathbf{x})$  formam o espaço de funções objetivo, ou espaço alcançável,  $\mathbf{Z} = \{\mathbf{z} \in \mathbb{R}^w \mid \mathbf{z} = \mathbf{F}(\mathbf{x}), \mathbf{x} \in \mathbf{X}\}$ . Esta é uma das grandes diferenças entre a otimização de objetivo único e otimização multiobjetivo: além do espaço formado pelas variáveis de decisão, que é comum a todos os problemas de otimização, na otimização multiobjetivo as funções objetivo constituem um espaço multidimensional. A Fig. 5 ilustra esses dois espaços e um mapeamento entre eles (DEB, 2014). Neste sentido, ser realizável implica que nenhuma restrição é violada, enquanto que ser alcançável requer que um ponto no espaço de objetivo é mapeado para um ponto no espaço de variáveis de decisão. Cada ponto no espaço das variáveis de decisão é mapeado para um ponto no espaço de funções objetivo, mas o inverso pode não ser verdadeiro; todo ponto no espaço

Figura 5 – Representação do espaço das variáveis de decisão e o espaço objetivo correspondente



Fonte: (DEB, 2014), Adaptado.

objetivo  $\mathbf{z} \in \mathbf{Z}$  não corresponde necessariamente a um único ponto  $\mathbf{x} \in \mathbf{X}$ . Deste modo, mesmo para o caso onde não há restrições as variáveis, apenas alguns pontos no espaço de critério são alcançáveis (MARLER; ARORA, 2004).

Na literatura de otimização multiobjetivo, o conjunto ótimo de Pareto é definida como:

$$\mathbf{P}^* = \{\mathbf{x}^* \in \mathbf{X} \mid \nexists \mathbf{x} \in \mathbf{X} : \mathbf{F}(\mathbf{x}) \leq \mathbf{F}(\mathbf{x}^*)\} \quad (3.2)$$

A fronteira de Pareto é a imagem do conjunto ótimo de Pareto, ou seja:

$$\mathbf{FP}^* = \{\mathbf{F}(\mathbf{x}) = (F_1(\mathbf{x}) \ F_2(\mathbf{x}) \ \dots \ F_w(\mathbf{x}))^T \mid \mathbf{x} \in \mathbf{P}^*\} \quad (3.3)$$

A Fig 6 mostra uma representação gráfica desses dois conjuntos.

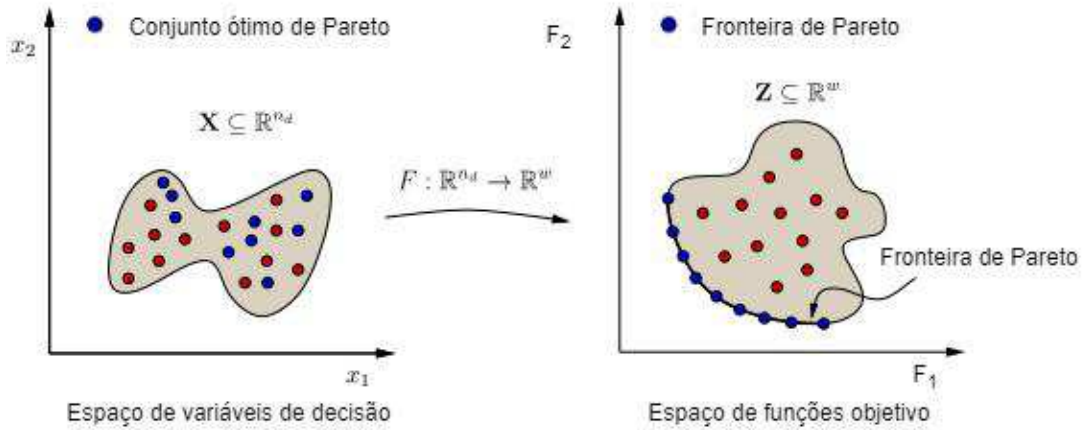
### 3.2.2 Solução de compromisso

A solução para um problema de otimização multiobjetivo pode ser obtida por meio de diversas técnicas de programação matemática, que podem ser classificadas de acordo com o momento na qual usuário é solicitado a fornecer informações de preferência entre as funções objetivo:

- Antes da busca pela solução: método de programação de metas, método lexicográfico, etc.
- Após a busca pela solução: método da combinação linear de pesos, método de métricas ponderadas, método da  $\varepsilon$ -restrição, etc.



Figura 6 – Representação do conjunto ótimo de Pareto e da Fronteira de Pareto



Fonte: (JAIMES et al., 2009), Adaptado.

- Durante a busca pela solução: método de Geoffrion-Dyer-Feinberg, método de Tchebycheff, método de pontos de referência, etc.

Uma revisão completa das técnicas de programação matemática para obter soluções para problemas MOO podem ser encontradas em (MARLER; ARORA, 2004), (JAIMES et al., 2009). Dado o problema de otimização (3.1), a solução mais direta busca no espaço  $\mathbf{X}$  do ponto cuja imagem seja o ponto Utopia. Também conhecido como ponto ideal, o ponto de Utopia é definido como sendo o ponto do espaço  $\mathbf{Z}$  que contempla a minimização de cada uma das funções objetivo:

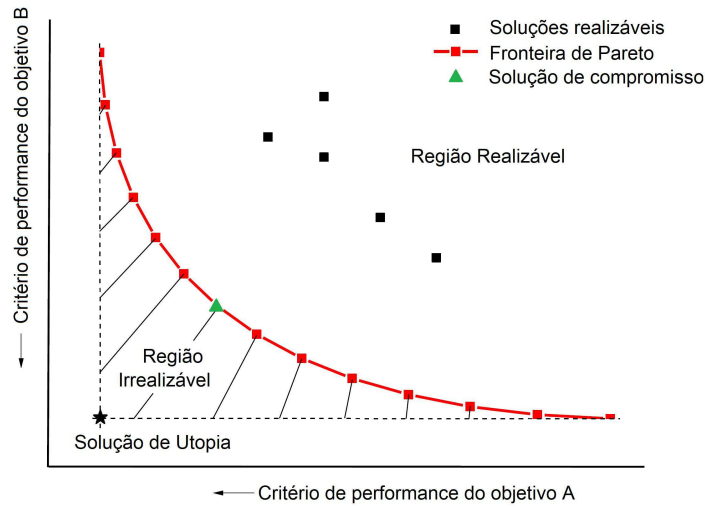
$$\mathbf{F}^\circ = \{(F_1^\circ \ F_2^\circ \ \dots \ F_i^\circ)^T \mid F_i^\circ = \min_{\mathbf{x}} \{F_i(\mathbf{x}) \mid \mathbf{x} \in \mathbf{X}\} \ \forall i = 1, 2, \dots, w\} \quad (3.4)$$

Em geral, o ponto de Utopia não é alcançável. Entretanto, ele pode ser utilizado durante o processo de otimização pelo método da solução de compromisso. Esta técnica é atrativa para diversas aplicações práticas devido a grande complexidade de representar com precisão as preferências do tomador de decisão em funções de utilidade (DEB, 2014). Na solução de compromisso, o algoritmo buscará o ponto alcançável mais próximo do ponto de Utopia  $\mathbf{F}^\circ$ . Esta medida de proximidade pode ser a distância euclidiana, mas não é necessário restringir-se a ela (MARLER; ARORA, 2004). Assim, a solução de compromisso é definida como:

$$\min_{\mathbf{x}} \|\mathbf{F}(\mathbf{x}) - \mathbf{F}^\circ\| = \sqrt{\sum_{i=1}^k (F_i(x) - F_i^\circ)^2} \quad (3.5)$$

A Fig. 7 mostra graficamente a solução de compromisso e o ponto de Utopia para um problema MOO com duas funções objetivo.

Figura 7 – Representação do ponto de Utopia e da solução de compromisso



Fonte: (SCHWEIDTMANN et al., 2018), Adaptado.

### 3.3 Sintonia baseada na solução de compromisso

Conforme já citado anteriormente, existem quatro parâmetros de sintonia para o SPC: as matrizes de ponderação  $Q_k$  e  $R_k$  e os horizontes de previsão  $N_p$  e controle  $N_c$ . Entretanto, para manter o processo de otimização mais simples, preferiu-se focar somente em obter valores ótimos para  $Q_k$  e  $R_k$ , enquanto que os horizontes são escolhidos a partir de recomendações encontradas na literatura de sintonia MPC.

#### 3.3.1 Determinação dos horizontes $N_p$ e $N_c$

O horizonte de controle determina o número de variáveis de controle que são calculadas a cada instante de amostragem pelo controlador. Valores pequenos de  $N_c$  resultam em sistemas mais lentos e estáveis, enquanto valores maiores levam a respostas mais agressivas, porém, com menos robustez e maior esforço computacional (GARRIGA; SOROUGH, 2010), (ALHAJERI; SOROUGH, 2020). Geralmente são utilizados valores pequenos para o horizonte de controle ( $N_c \leq 5$ ).

O horizonte de previsão determina o tamanho da janela de previsão das variáveis controladas que são calculadas pelo controlador a cada instante de amostragem. Recomenda-se que o  $N_p$  seja grande o suficiente para englobar a dinâmica do processo, pois, a robustez e estabilidade do controlador aumentam com valores maiores de  $N_p$ . Contudo, nota-se que acima de certos valores não há melhoria significativa de performance em malha fechada, ao passo que há um aumento significativo do custo computacional (ALHAJERI; SOROUGH, 2020). Os métodos analíticos de sintonia MPC encontrados na literatura em geral são descritos em termos de métricas de performance em malha aberta (GARRIGA; SOROUGH,

2010), (ALHAJERI; SOROUSH, 2020), tais como atrasos, constantes de tempo e tempos de acomodação. Para o método proposto aplicável ao SPC, o  $N_p$  é escolhido em função do tempo de acomodação. Neste caso, o horizonte deve corresponder ao valor em instantes de amostragem que equivale ao tempo necessário para que a resposta ao degrau em malha aberta fique entre 80% a 90% do valor de regime permanente,  $0.8T_s \leq N_p \leq 0.9T_s$  (YAMASHITA et al., 2016).

### 3.3.2 Determinação das matrizes $Q_k$ e $R_k$

As matrizes  $R_k$  e  $Q_k$ , por outro lado, são obtidas por meio de um processo de otimização multiobjetivo. O problema MOO é estabelecido como a busca da solução ótima que minimize o erro quadrático entre a resposta em malha fechada  $y_i(k)$  e uma trajetória de referência definida pelo usuário  $y_{i,ref}(k)$ , para cada uma das  $i$  variáveis controladas do processo. Assim, a função objetivo  $F_i$  é:

$$F_i(x) = \sum_{k=1}^{t_{tun}} (y_{i,ref}(k) - y_i(k))^2, \quad i = 1, \dots, l \quad (3.6)$$

onde  $t_{tun}$  é o horizonte de sintonia. O valor de  $t_{tun}$  deve ser escolhido grande o suficiente para que as funções objetivo  $F_i$  consigam capturar a dinâmica do processo na simulação feita ao longo do processo de sintonia. De modo a não utilizar modelos ao longo do processo de sintonia, as saídas  $y_i(k)$  são calculadas de acordo com lei de controle preditivo por subespaços já definida (2.23). Para o método de sintonia proposto, as matrizes de ponderação são diagonais, isto é,  $Q_k = \text{diag}(q_1, \dots, q_l)$  e  $R_k = \text{diag}(r_1 \dots r_m)$ , então o vetor de variáveis de decisão é definido como  $\mathbf{x} = (q_1 \dots q_l \ r_1 \dots r_m)$  e corresponde aos  $l + m$  parâmetros de sintonia do controlador a serem obtidos pelo algoritmo de otimização.

Com o propósito de fornecer ao usuário uma método que seja simples de aplicar e que tenha potencial de aplicações práticas, escolheu-se que as trajetórias de referência são definidas por meio da resposta de um sistema  $G_{ref_N}(s)$  de ordem  $N$  para cada uma das  $i$  variáveis controladas. A ordem do sistema de referência deve ser obtida por meio da inspeção de valores singulares dominantes da matriz  $L_w$  em (2.14), já citada na Seção 2.1.2. No processo de sintonia, deve-se aplicar um sinal pseudo-randômico (PRBS) a entrada de  $G_{ref_N}(s)$ , de modo a garantir que ao longo do processo de otimização o algoritmo esteja simulando o processo em diferentes pontos de operação. Nas trajetórias de referência são geradas por funções de transferência da forma:

$$G_{i,ref_N}(s) = \frac{1}{(\tau_{ref_{i_1}} + 1)(\tau_{ref_{i_2}} + 1) \dots (\tau_{ref_{i_N}} + 1)} e^{-\theta_i s} \quad (3.7)$$

As constantes de tempo  $\tau_{ref_{i,N}}$  são os requisitos temporais definidos pelo projetista, escolhidas de acordo com a agressividade e desempenho que ele quer que o controlador preditivo obtenha para cada uma das  $i$  variáveis controladas, enquanto que  $\theta_i$  é o atraso estimado para cada uma das  $i$  variáveis controladas.

O ponto de Utopia  $\mathbf{F}^\circ(x)$  é obtido como solução do seguinte problema de otimização:

$$F_i^\circ(\mathbf{x}) = \min_{\mathbf{x}} F_i(x), \quad i = 1, \dots, w \quad (3.8)$$

$$\text{sujeito a:} \quad LB \leq \mathbf{x} \leq UB \quad (3.9)$$

onde  $w = l + m$ . Finalmente, a sintonia é obtida por meio da solução de compromisso, minimizando a distância euclidiana em relação à solução de Utopia:

$$\arg \min_{\mathbf{x}} \|\mathbf{F}^\circ - \mathbf{x}(x)\|^2 \quad (3.10)$$

$$\text{sujeito a:} \quad LB \leq \mathbf{x} \leq UB \quad (3.11)$$

onde  $\mathbf{F}^\circ(x)$  é o vetor com as componentes  $F_i^\circ$ .  $LB$  e  $UB$  significam os limites inferiores e superiores dos parâmetros de sintonia, respectivamente. Os problemas de otimização dos pontos de Utopia (3.8) e (3.9) e o problema de otimização para a solução de compromisso (3.10) e (3.11) são resolvidos com o método de otimização por enxame de partículas (PSO). Mais detalhes sobre o PSO e sua aplicação na sintonia SPC podem ser encontrados em (MACIEL; BARROS; LIMA, 2021).

O problema de otimização proposto é em geral não-convexo e apresenta múltiplas soluções locais. Contudo, estas soluções locais são suficientes para que o sistema consiga atender aos requisitos temporais desejados. Ademais, havendo multiplicidade destas soluções, é razoável estabelecer que um dos elementos das matrizes diagonais  $R_k$  ou  $Q_k$  é constante ao longo do processo de otimização. Deste modo, há uma diminuição da complexidade e conseqüente redução do custo computacional empregado, sem haver alguma piora considerável dos resultados obtidos.

Figura 8 – Algoritmo: sintonia do SPC baseada na solução de compromisso

```

// Algoritmo para obtenção dos parâmetros de sintonia  $N_c$ ,  $N_p$ ,  $R_k$  e
//  $Q_k$  para a lei de controle SPC sem restrições às variáveis do
// processo
Dados:  $\tilde{L}_{wI}$  e  $\tilde{L}_{uI}$ 
// Obter os preditores diferenciais truncados  $\tilde{L}_{wI}$  e  $\tilde{L}_{uI}$  de acordo
// com o algoritmo encontrado na Fig. 4.
Entrada:  $N$ ,  $t_{tun}$ 
// Escolher a ordem da função de transferência de referência e o
// horizonte de sintonia.
Entrada:  $\tau_{refi,1}, \dots, \tau_{refi,N}, \theta_i$ 
// Escolher as constantes de tempo para cada uma das  $i$  variáveis
// controladas e fornecer os atrasos estimados
Entrada:  $N_c, N_p$ 
// Escolher os horizontes de controle e predição:  $N_c \leq 5$  e
//  $0.8T_s \leq N_p \leq 0.9T_s$ .
Entrada:  $LB, UB$ 
// Escolher os limites inferior e superior, respectivamente para as
// variáveis de decisão  $\mathbf{x} = (q_1 \dots q_l r_1 \dots r_m)$ 
1 Montar as funções de transferência  $G_{refiN}(s)$  a partir das constantes de tempo  $\tau_{iN}$ 
// e atrasos  $\theta_i$  informados pelo usuário.
2 para cada  $y_i$  faça
3   | Obter  $y_{i ref}$  ao aplicar o sinal do tipo PRBS em  $G_{refiN}(s)$  ao longo do
//   | horizonte de sintonia  $t_{tun}$ .
4   | Definir as funções objetivo  $F_i$  a partir de (3.6)
5   | Obter os pontos de Utopia a partir do problema de otimização (3.8) e (3.9)
//   |
//   | Caso com saída única
6 se  $l = 1$  então
7   | Usar diretamente as soluções de Utopia obtidas na etapa 5
//   |
//   | Caso com múltiplas saídas
8 senão
9   | Usar o ponto de Utopia  $\mathbf{F}^\circ$  obtido nas etapas 5 para encontrar a solução de
//   | compromisso do problema de otimização (3.10) e (3.11)
10 Obter as matrizes de ganho do controlador SPC dadas por (2.27) e (2.28)

```

Fonte: Autoria própria

## 4 Sistemas de Controle em Rede

### 4.1 Introdução

Sistemas de controle com componentes espacialmente distribuídas são comuns na indústria e já são utilizados a décadas, tais como em refinarias, processos químicos em geral e usinas elétricas. Neste tipo de sistema de controle, atuadores, sensores, controladores e o próprio processo a ser controlado estão alocados fisicamente em locais distintos. Geralmente, as partes do sistema estão conectadas por meio de cabos a um controlador central, de onde as informações coletadas pelos sensores são monitoradas e, de acordo com a estratégia de controle, norteiam as políticas de controles que são enviadas aos atuadores. Contudo, esta configuração de sistema de controle apresenta algumas desvantagens significantes, tais como alto custo associado ao cabeamento, dificuldades de introduzir novos componentes ou alterar a distribuição espacial destes. ([ANTSACLIS; BAILLIEUL, 2007](#)).

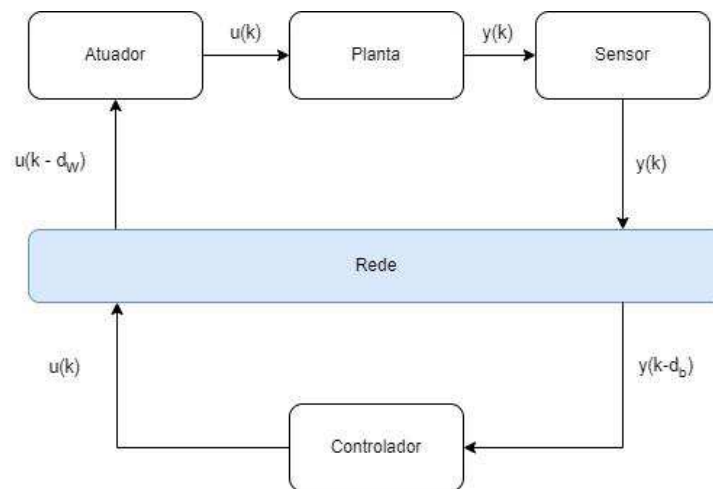
Nos últimos anos, as tecnologias associadas a comunicação em rede evoluíram significativamente, como os microprocessadores de baixo custo e crescimento expressivo da Internet das Coisas (IoT). O termo Internet das Coisas foi utilizado inicialmente por Kevin Ashton em 1999 para descrever um sistema no qual dispositivos físicos podem ser conectados à internet por meio de sensores. Inicialmente, a expressão descrevia o uso de etiquetas de Identificação por Rádio-Frequência (RFID) para contar e rastrear objetos sem a necessidade de intervenção humana, promovendo a conexão destas com a internet ([ROSE; ELDRIDGE; CHAPIN, 2015](#)). Entretanto, IoT pode ser definido de maneira mais ampla como uma infraestrutura dinâmica com capacidade de se autoconfigurar, baseando-se em padrões e protocolos de comunicação. Além disto, os dispositivos IoT, sejam eles físicos ou virtuais, possuem identificações e atributos próprios e são capazes de empregar interfaces inteligentes e formar uma rede de informações ([LI; XU; ZHAO, 2015](#)).

Apesar do termo ser relativamente recente, conceito de empregar redes de comunicação para monitorar e controlar dispositivos é mais antiga. Em 1977, Paraskevagos patenteou um sistema capaz de medir automaticamente o consumo de redes elétricas por meio da rede telefônica, além de permitir o gerenciamento de cargas remotamente ([ROSE; ELDRIDGE; CHAPIN, 2015](#)). Os diversos avanços tecnológicos nas áreas de chips e redes de comunicação, tais como miniaturização, conectividade com menor custo e maior velocidade, aliados as tendências de mercado como *Big Data* e computação na nuvem fizeram com que o mercado de dispositivos IoT crescesse consideravelmente nos últimos anos. Em 2019, o mercado de soluções IoT chegara a US\$ 212 bilhões e a previsão é de que chegue a US\$ 1.6 trilhão em meados de 2025 ([STATISTA, 2019](#)). Este crescimento acelerado motivou o desenvolvimento de diversas linhas de pesquisa nos mais variados

campos de aplicação, incluindo a de controle de processos dinâmicos utilizando dispositivo IoT.

Deste modo, aplicações de soluções em rede empregadas em malhas de controle tornaram-se cada vez mais comuns. Este tipo de sistema de controle é chamado Sistemas de Controle em Rede (NCS), no qual planta, sensor, atuador, controlador e comando de referência estão conectados na rede (XIA et al., 2013). Uma arquitetura típica de NCS é mostrado na Fig. 9

Figura 9 – Arquitetura típica de um NCS



Fonte: Autoria própria.

Contudo, as mudanças ocasionadas pela implementação desta nova arquitetura provocam mudanças significativas na análise e *design* do sistema de controle. Neste tipo de topologia, é necessário considerar explicitamente as interconexões entre os componentes, isto é, a rede de comunicação passa a afetar de forma significativa no comportamento dinâmico do sistema. Sobretudo em aplicações de sistemas de controle distribuídos, questões associadas a protocolos de comunicação para roteamento e escalonamento, algoritmos de sistemas em tempo real para lidar com restrições do tipo *hard* e tipo *soft* (ANTSAKLIS; BAILLIEUL, 2007). Além disto, transmitir dados por meio de redes envolve perdas intermitentes de pacotes de dados e atrasos aleatórios de comunicação, bem como problemas de sincronia, erros de quantização, restrições de largura de banda e outros efeitos não previsíveis. Estas limitações deterioram a performance em malha fechada, podendo ainda levar o sistema à instabilidade, (XIA, 2015). Neste contexto, o controle preditivo apresenta-se como uma técnica de controle que possibilita lidar bem com os problemas de comunicação, visto que o controlador pode trabalhar diretamente com previsões da variável controlada e consegue calcular variáveis de controle futuras. Neste trabalho, será feita a análise da solução proposta por Xia, Fu, Liu para lidar com os atrasos de comunicação e perdas de pacotes que ocorrem nos canais direto e de realimentação, isto é, entre o canal controlador-

atuador e canal sensor-controlador, respectivamente (XIA; FU; LIU, 2011).

## 4.2 Compensação do atraso e a perda de dados no canal direto

Os atrasos de comunicação ou perda de pacotes no Canal entre o Controlador e o Atuador (CAC) significam que o processo pode deixar de receber o sinal de controle no momento adequado, algo que pode levar a perda de performance e, eventualmente, pode levar a instabilidade. Para lidar com a perda de pacotes, em geral, existem dois tipos de soluções comuns na literatura: esquema de entrada zero, isto é, o sinal do atuador do processo é configurado como zero enquanto um novo pacote não chegar; ou esquema de entrada mantida, no qual o sinal de controle é mantido igual ao último pacote que chegou ao atuador (XIA; FU; LIU, 2011).

A solução proposta consiste em, ao invés de enviar para o atuador somente o sinal de controle para o próximo instante dado por (2.29) e (2.30), transmitir a sequência futura de sinais de controle para um compensador. Este compensador atua como um *buffer* de sinais de controle. Caso haja um atraso de comunicação ou perda de pacote, o compensador ainda é capaz de fornecer um sinal de controle adequado, escolhendo o sinal de controle mais recente disponível no lado da planta. Conforme mostrado em (2.26), o controlador é capaz de gerar uma sequência de variação de controle futuro  $\Delta u_f$ , que compreende  $\Delta u(k+1)$  até  $\Delta u(k+N_c)$ . Assim, o horizonte de controle determina o tamanho do *buffer* que pode ser gerado. Deste modo, ao contrário do caso em que o controlador é executado localmente, utilizar um  $N_c \leq 5$  iria limitar consideravelmente a capacidade de compensar os problemas de comunicação no canal direto. O aumento no esforço computacional causado pelo horizonte de controle maior é justificável, visto que o controlador está sendo executado remotamente na rede. Assim, no contexto de NCS deste trabalho, utilizar-se-á o horizonte de predição igual ao horizonte de controle, isto é,  $N_c = N_p = M$ .

Nesta configuração, controlador envia o pacote de dados para a planta calculado no instante  $k$ :

$$\Delta u_f(k+1|k) = \begin{pmatrix} \Delta u(k+1|k) \\ \Delta u(k+2|k) \\ \vdots \\ \Delta u(k+N_c|k) \end{pmatrix} \quad (4.1)$$

No lado da planta, o compensador armazenará o vetor  $\Delta u_f$  mais recente que foi transmitido com sucesso. Em um instante de amostragem  $k_o$  qualquer, o compensador buscará em seu *buffer* qual é a variação de controle mais recente disponível, de modo que:

$$\begin{aligned} \Delta u(k_o) &= \Delta u(k_o|k_o - i) \\ i &= \arg \min_i u(k_o|k_o - i) \end{aligned} \quad (4.2)$$



É mostrado em (XIA et al., 2013) e (XIA, 2015) que o desempenho em malha fechada na presença de atrasos de comunicação empregando o esquema NCS proposto assemelha-se bastante ao desempenho do sistema em malha fechada sem os atrasos, desde que as seguintes condições sejam aceitas:

1. O atraso aleatório seja limitado e tenha um valor menor ou igual ao horizonte de predição  $M$ ;
2. O número de perdas de pacote consecutivos seja limitado e tenha um valor menor ou igual ao horizonte de predição  $M$ .

A Fig. 10 mostra o algoritmo do compensador que é executado a cada instante de amostragem, localmente. É possível observar que a solução apresentada no presente trabalho difere do compensador proposto por Xia, visto que em seu algoritmo SPC a ação integral não é incluída no controlador, de modo que ele trabalha com  $u_f$  e não com  $\Delta u_f$  (XIA et al., 2013), (XIA, 2015).

Figura 10 – Algoritmo: compensador para lidar com problemas de comunicação no canal direto

```
// Algoritmo do compensador para obtenção do sinal de controle
Entrada:  $\Delta u_{f,rec}$ 
// Sequência de variação do sinal de controle, de tamanho  $M$ 
1 se  $timestamp_{rec} > timestamp_{arm}$  então
2    $\Delta u_{f,arm} = \Delta u_{f,rec}$ 
   // Verifica se a sequência  $\Delta u_{f,rec}$  (recebido) é mais recente do que
   // a última sequência disponível  $\Delta u_{f,arm}$ . Caso sim, atualiza o
   // buffer.
3 Obter variação do sinal de controle:  $\Delta u(k) = \Delta u_{f,arm}(:, 1)$ 
   // Extrai a variação do sinal de controle para o instante  $k$  de
   // interesse como sendo o primeiro elemento do buffer
4 Calcula o sinal de controle para o instante  $k$  como sendo:  $u(k) = \Delta u(k) - u(k - 1)$ 
   e aplica-o ao processo
   // Salva dados que são necessários para os cálculos no próximo
   // instante de amostragem
5 Salva o sinal de controle  $u(k)$  atualizado para ser utilizado no cálculo no próximo
   instante de amostragem
6 Atualiza o buffer, retirando o primeiro elemento da fila e armazenando para o
   próximo instante de amostragem
```

Fonte: Autoria própria

### 4.3 Compensação do atraso e a perda de dados no canal de realimentação

Para lidar com os atrasos de transmissão e perdas de dados no Canal entre o Sensor e o Controlador (SCC), (XIA; FU; LIU, 2011) apresentam um método de estimação ótima por meio de um filtro de Kalman modificado construído a partir do modelo do processo. Esta solução é baseada no método proposto por Shi, Xie, Murray, na qual o filtro de Kalman é utilizado em uma rede com atraso de pacotes, cuja análise é feita por uma abordagem probabilística. Neste contexto, o estimador mantém um *buffer* de comprimento  $D$  para que a cada tempo  $k$  o estimador seja capaz de recuperar todos os pacotes de dados disponíveis até o tempo  $k - D + 1$  (SHI; XIE; MURRAY, 2009).

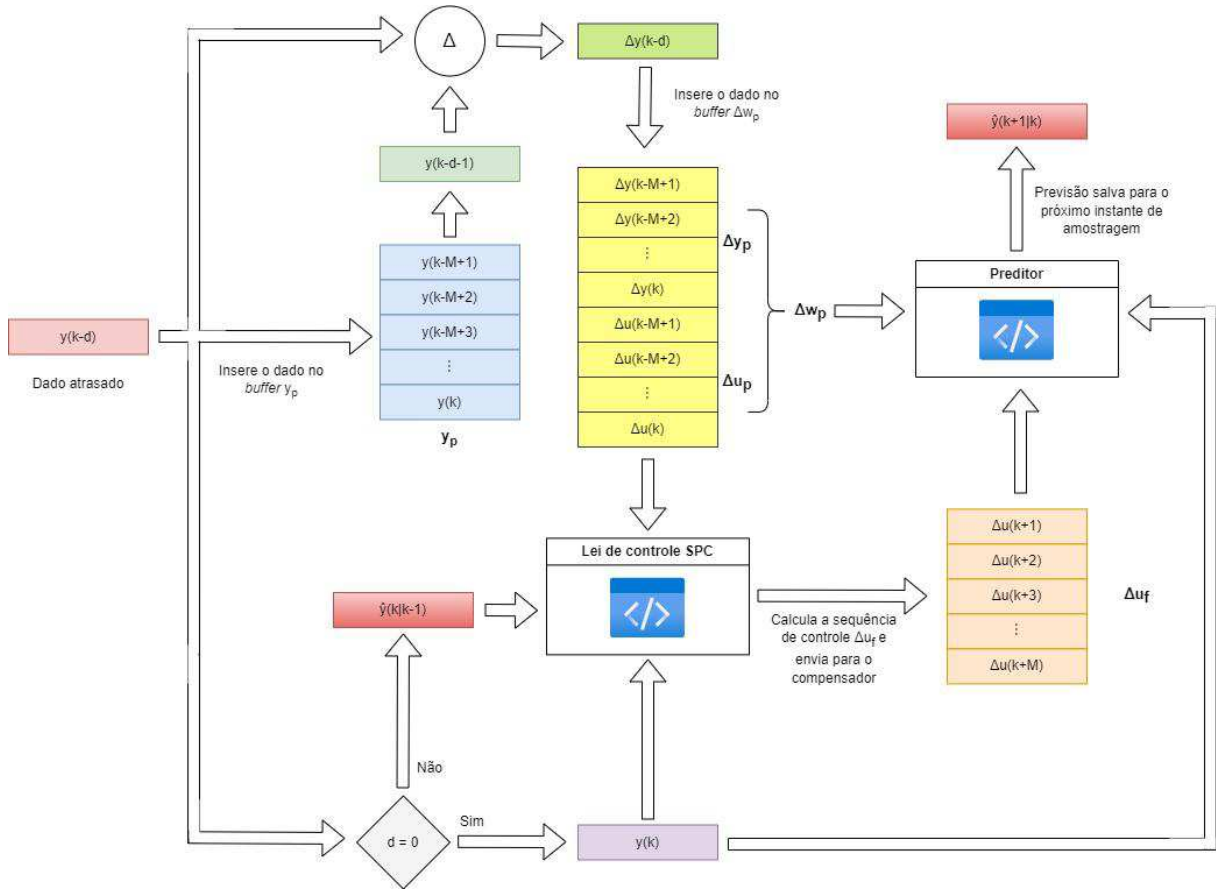
Sendo assim, o método proposto por Xia aplicável ao MPC utiliza um filtro de Kalman para estimar os estados futuros  $\hat{x}(k + 1)$  do processo e utilizá-los para calcular o sinal de controle. Assim, um *buffer* é criado no controlador para armazenar os novos dados recebidos do sensor e recalculá-los para obter o novo  $\hat{x}(k|k)$ . A partir deste novo  $\hat{x}(k|k)$ , o controlador pode obter um novo sinal de controle que é enviado para o atuador pela rede. Caso o controlador receba um novo dado com atraso  $y(k - d)$ , este novo dado será armazenado na célula correspondente do *buffer*, de modo a melhorar as predições obtidas pelo filtro de Kalman (XIA; FU; LIU, 2011).

Entretanto, a abordagem acima utiliza explicitamente um modelo do processo para a construção do filtro de Kalman, algo que é indesejado no SPC. Contudo, conforme mostrado por (OVERSCHEE; MOOR, 1996) e (COCK; MOOR; LEUVEN, 2003), é possível obter um filtro de Kalman de estados não estacionários diretamente dos dados sem utilizar um modelo do processo. Ademais, a formulação matemática do SPC apresenta  $\hat{y}_f$ , que é a sequência das saídas futuras do processo. Deste modo, é possível obter uma predição da saída  $\hat{y}(k + 1)$  como sendo o primeiro bloco de tamanho  $l$  de  $\hat{y}_f$ . Em notação do MATLAB®, tem-se que:

$$\hat{y}_f = \begin{pmatrix} \hat{y}(k + 1) \\ \hat{y}(k + 2) \\ \vdots \\ \hat{y}(k + N_p) \end{pmatrix} \rightarrow \hat{y}(k + 1) = \hat{y}_f(l, :) \quad (4.3)$$

A sequência  $\hat{y}_f$  pode ser obtida por meio de (2.23). Contudo, caso fossem utilizados somente os valores preditos  $\hat{y}(k)$  na lei de controle, o sistema estaria em malha aberta, ficando extremamente dependente da qualidade das predições e não tendo qualquer resposta a perturbações na saída do processo. Uma maneira de diminuir o erro de predição é atualizar o vetor que armazena os dados passados  $\Delta w_p$  com os dados medidos atrasados  $y(k - d)$  que chegam ao controlador, utilizando a mesma ideia de inserir os dados passados para melhorar o preditor do método proposto por Xia para o MPC (XIA; FU; LIU, 2011).

Figura 11 – Representação do fluxo de dados no bloco controlador e predictor SPC



Fonte: Autoria própria.

Entretanto, para o caso do SPC proposto neste trabalho, a inserção dos dados atrasado não é trivial, pois, a amostra a ser inserida é, na verdade, o valor da variação da saída neste instante, isto é, o valor  $\Delta y(k-d)$ . Para lidar com esta situação, foi proposta a utilização de um *buffer* que armazene os dados de saída passados  $\hat{y}_p$  que irá armazenar os sinais preditos  $\hat{y}(k)$  passados. A medida em que amostras atrasadas  $y(k-d)$  medidas do processo cheguem ao controlador, estas devem ser guardadas no *buffer*  $y_p$ . Assim, o controlador irá buscar no *buffer* o valor armazenado de  $\hat{y}(k-d-1)$ , seja ele um dado predito ou medido, de modo a atualizar o valor de  $\Delta y(k-d) = y(k-d) - \hat{y}(k-d-1)$  no vetor de dados  $\Delta w_p$  empregado no predictor e na lei de controle. A Fig. 11 mostra o fluxo dos dados no controlador associado ao predictor que está sendo executado na rede ao receber um pacote  $y(k-d)$  atrasado.

As condições para garantir que a estratégia apresentada seja efetiva são as mesmas já citadas para o caso do canal direto: o atraso de comunicação e as perdas consecutivas de pacotes devem ser limitados ao comprimento  $M$  dos *buffers* (XIA; FU; LIU, 2011). A Fig. 12 mostra o pseudocódigo associado ao bloco controlador e predictor no contexto

do SPC proposto para lidar com problemas de comunicação no canal entre o sensor e o controlador.

Figura 12 – Algoritmo: controlador e preditor para lidar com problemas de comunicação no canal de realimentação

```
// Algoritmo do controlador associado ao preditor para obtenção do
// sinal de controle

Dados:  $y_{rec}$ 
// Amostra recebida da saída do processo, atrasada em  $d$  instantes de
// amostragem
Entrada:  $K_e, K_w$ 
// Ganhos do controlador preditivo sem restrições obtidos a partir
// de (2.27) e (2.28).
Entrada:  $\tilde{L}_{wI}, \tilde{L}_{uI}$ 
// Preditores diferenciais obtidos a partir de (2.24) e (2.25).

1 Calcula o atraso  $d$  associado a amostra recebida, isto é,  $y_{rec} = y(k - d)$ .
2 se  $d = 0$  então
3    $y_{atual} = y_{rec}$ 
   // Para o caso sem atraso, a saída recebida é utilizada diretamente
   // pela lei de controle e ela é utilizada para atualizar os buffers
   // do preditor.
4 senão
5    $y_{atual} = y_{pred}$ 
   // Para o caso com atraso, a saída recebida e com atraso é utilizada
   // para atualizar os buffers do preditor e um valor predito na
   // última iteração é utilizado na lei de controle.
6 Buscar no buffer  $y_p$  a saída anterior, isto é  $y_{ant} = y(k - d - 1)$ .
7 Atualizar os buffers  $y_p$  e  $\Delta y_p$  com o  $y_{rec}$  e  $\Delta y_{rec} = y_{rec} - y_{ant}$ , respectivamente.
8 Aplicar  $y_{atual}$  diretamente à lei de controle (2.26) e enviar para o processo a
   sequência  $\Delta u_f$ .
9 Utilizar (2.23) para calcular a predição  $\hat{y}_{ref}$ .
10 Atualizar  $y_{pred} = \hat{y}(k + 1|k)$  como o primeiro bloco  $l$  de  $\hat{y}_{ref}$  e salvá-lo para a
    próxima iteração.
```

Fonte: Autoria própria

## 5 Resultados

De modo a verificar a aplicabilidade do SPC, bem como verificar a efetividade do método de sintonia proposto e da solução para lidar com os problemas de comunicação do SPC/NCS, foram realizados testes em modelos computacionais no ambiente Simulink®. Foram escolhidos modelos de Primeira Ordem com Atraso (FOPDT) e Segunda Ordem com Atraso (SOPDT), tanto para os casos univariáveis como multivariáveis, visto que tais modelos descrevem uma ampla gama de processos.

Para avaliar o desempenho do controlador em malha fechada, foi utilizado o índice de Erro Absoluto Integrado (IAE). Neste caso, o IAE é calculado a partir da diferença entre a saída do processo em malha fechada  $y(k)$  e o sinal de referência  $r(k)$ . Este indicador fornece uma medida do quão rápida é a resposta em malha fechada, isto é, quanto maior, mais rápida e agressiva é a resposta. Ele é dado por:

$$IAE = \sum_{k=0}^n |r(k) - y(k)| \quad (5.1)$$

Para avaliar a entrada de controle  $u(k)$ , é utilizado o índice de Variação Total (TV), que é dado pela soma de todas as variações do sinal de controle. Assim, quanto menor o TV, mais suave é o sinal de controle:

$$TV = \sum_{k=0}^n |u(k) - u(k-1)| \quad (5.2)$$

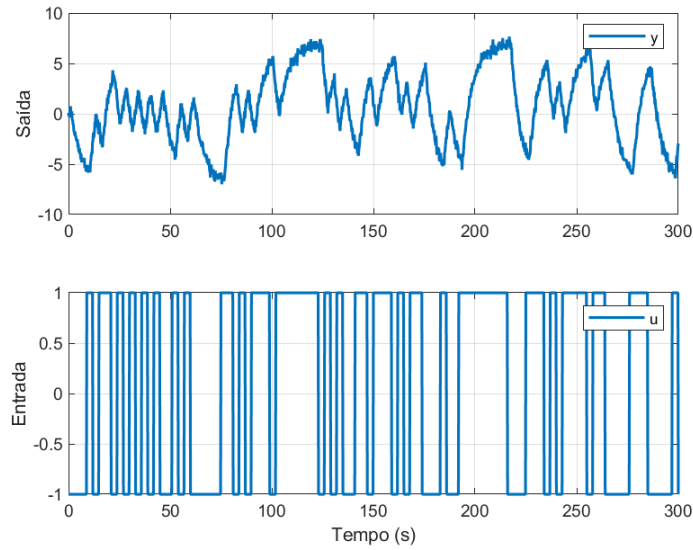
Por fim, para medir o grau de proximidade entre a resposta em malha fechada  $y(k)$  e a trajetória de referência desejada  $y_{ref}(k)$ , escolheu-se avaliar a Raiz do Erro Médio Quadrático (RMSE). Esta métrica permite avaliar se o processo de sintonia forneceu uma resposta em malha fechada que se aproxima da trajetória planejada. Ele é dado por:

$$RMSE = \sqrt{\sum_{k=0}^n \frac{(y_{ref}(k) - y(k))^2}{n}} \quad (5.3)$$

Em todos os casos, de modo a simular os efeitos de ruídos de medição no processo de coleta de dados em malha aberta e no desempenho do sistema em malha fechada com o controlador, foram adicionados às saídas do processo um sinal do tipo ruído branco  $w(k)$ , de modo que  $y_{med}(k) = y(k) + w(k)$ . Em todos os casos, o ruído é definido em termos da sua densidade espectral de potência.

Os problemas de otimização para a sintonia do controlador foram todos resolvidos com uso de PSO, que faz parte da *Global Optimization Toolbox*. Mais detalhes sobre a implementação da função e do algoritmo podem ser encontrados em ([MATHWORKS, 2022](#)).

Figura 13 – Experimento em malha aberta - FOPDT



Fonte: Autoria própria.

## 5.1 Sintonia SPC

### 5.1.1 Caso univariável: modelo FOPDT

Um modelo genérico do tipo FOPDT é dado por:

$$G(s) = \frac{K}{\tau s + 1} e^{-\theta s} \quad (5.4)$$

O modelo escolhido a ser controlado tem  $K = 7$ ,  $\tau = 5$  e  $\theta = 1.6$ :

$$G(s) = \frac{7}{5s + 1} e^{-1.6s} \quad (5.5)$$

Para a obtenção dos dados utilizados para o cálculo das matrizes de predição e dos ganhos do controlador, foi simulado um experimento em malha aberta com a aplicação de um PRBS à entrada do modelo (5.5), conforme mostra na Fig. 13. O período de amostragem escolhido foi de  $T_s = 0.2s$ . A potência do sinal de ruído do tipo ruído branco foi de 0.005.

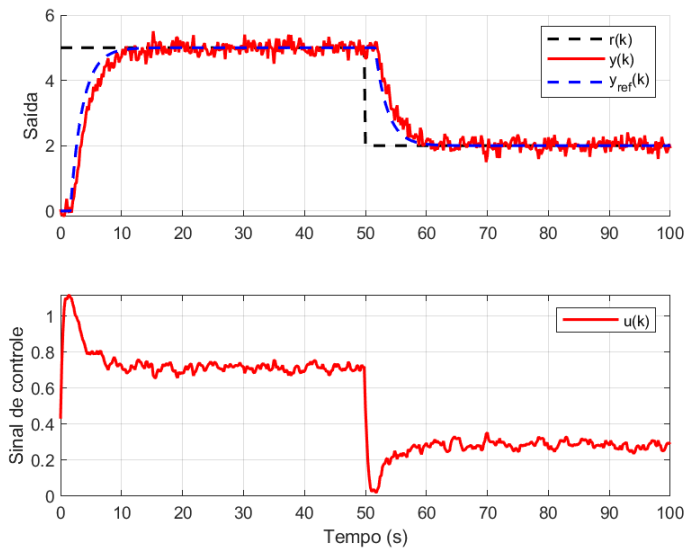
Para avaliar os resultados do método de sintonia, foram escolhidas funções de referência do tipo FOPDT, com parâmetros  $K = 1$ ,  $\theta = 1.6$  e valores de  $\tau$  iguais a 2, 4 e 8. Conforme já discutido, o parâmetro  $Q_k$  é mantido constante para simplificar o problema de otimização. A variável de decisão neste caso é  $\mathbf{x} = (R_k)$ , com limites superior e inferior iguais a 10000 e 1, respectivamente. Os horizontes escolhidos foram  $N_p = 50$  e  $N_c = 3$ . Os parâmetros de sintonia obtidos por meio do processo de otimização, bem como as métricas de desempenho em malha fechada podem ser observados na Tab. 1 e as dinâmicas em malha fechada podem ser observadas nas Fig. 14, Fig. 15 e Fig. 16, respectivamente.

Tabela 1 – Parâmetros de sintonia e métricas - FOPDT

$\tau$	$N_p$	$N_c$	$Q_k$	$R_k$	RMSE	IAE	TV
2.0	50	3	1	179.05	0.2686	239.1351	7.8679
4.0	50	3	1	1912.14	0.1989	266.0203	4.2258
8.0	50	3	1	9515.23	0.4010	328.0676	2.0791

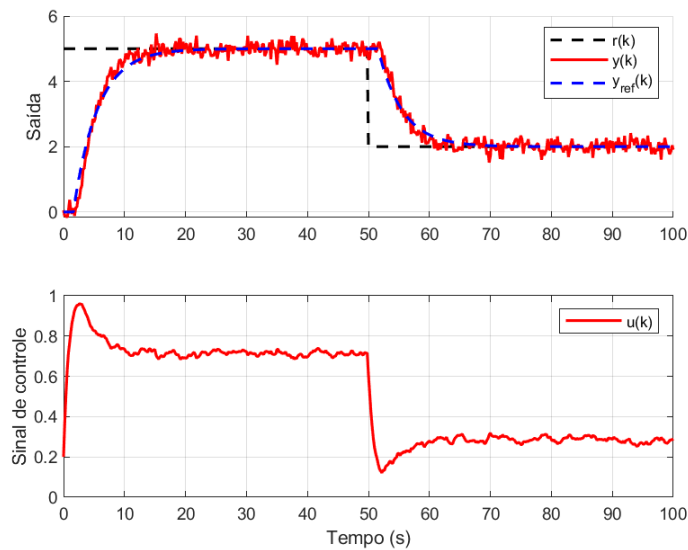
Fonte: Autoria própria.

Figura 14 – Avaliação em malha fechada:  $\tau = 2$  - FOPDT

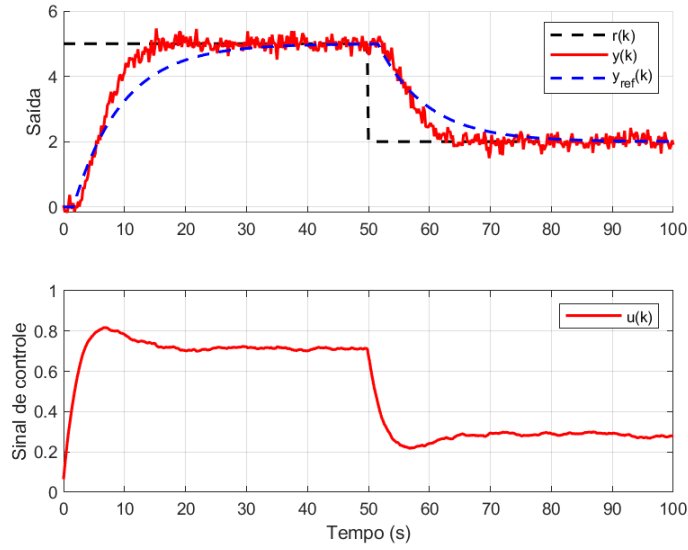


Fonte: Autoria própria.

Figura 15 – Avaliação em malha fechada:  $\tau = 4$  - FOPDT



Fonte: Autoria própria.

Figura 16 – Avaliação em malha fechada:  $\tau = 8$  - FOPDT

Fonte: Autoria própria.

A partir dos dados obtidos, é possível observar que, trajetórias de referência com dinâmicas mais rápidas levam a respostas em malha fechada mais agressivas. Conforme esperado, quanto maior o valor de  $R_k$ , mais lenta será a resposta, resultando em um menor TV e maior IAE. O caso em que a trajetória de referência melhor se aproxima da resposta em malha fechada é quando  $\tau = 5$ , que apresenta o menor RMSE entre todos os casos.

### 5.1.2 Caso univariável: modelo SOPDT

Um modelo genérico do tipo SOPDT é dado por:

$$G(s) = \frac{K}{(\tau_1 s + 1)(\tau_2 s + 1)} e^{-\theta s} \quad (5.6)$$

O modelo escolhido a ser controlado tem  $K = 12$ ,  $\tau_1 = 6$ ,  $\tau_2 = 3$  e  $\theta = 2$ :

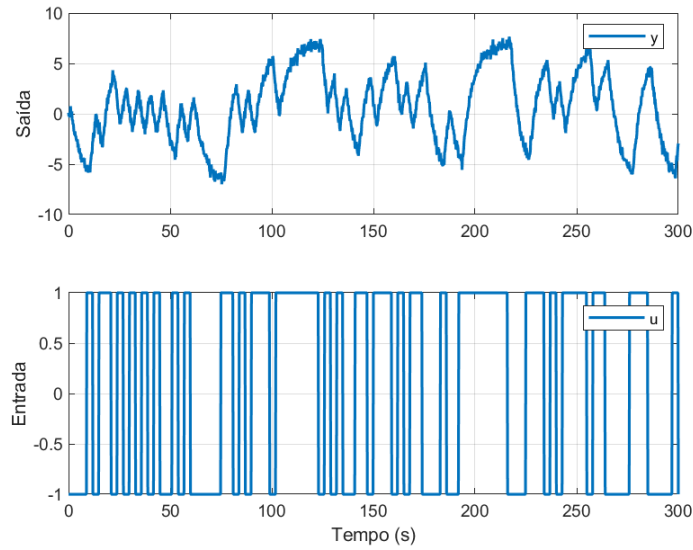
$$G(s) = \frac{12}{(6s + 1)(3s + 1)} e^{-2s} \quad (5.7)$$

Os dados do experimento em malha aberta são mostrados na Fig. 17. Novamente, o período de amostragem escolhido foi de  $T_s = 0.2s$  e a potência do sinal de ruído foi de 0.005.

Para o método de sintonia proposta, foram selecionadas funções de referência do tipo SOPDT (5.6), com parâmetros  $K = 1$ ,  $\theta = 2$  e valores de  $\tau_1 = \tau_2 = \tau$  iguais a 2, 3.5 e 5.7. Novamente, a matriz  $Q_k$  foi mantida constante para simplificar o problema de otimização. Neste caso, a variável de decisão do problema de otimização é  $\mathbf{x} = (R_k)$ , com limites superior e inferior iguais a 10000 e 1, respectivamente. Escolheu-se horizontes  $N_p =$



Figura 17 – Experimento em malha aberta - SOPDT



Fonte: Autoria própria.

Tabela 2 – Parâmetros de sintonia e métricas - SOPDT

$\tau$	$N_p$	$N_c$	$Q_k$	$R_k$	RMSE	IAE	TV
2	80	4	1	75.11	0.3629	462.1189	16.6516
3.5	80	4	1	3630.26	0.3537	608.8256	5.2044
5.7	80	4	1	8287.99	0.8787	625.9690	4.1018

Fonte: Autoria própria.

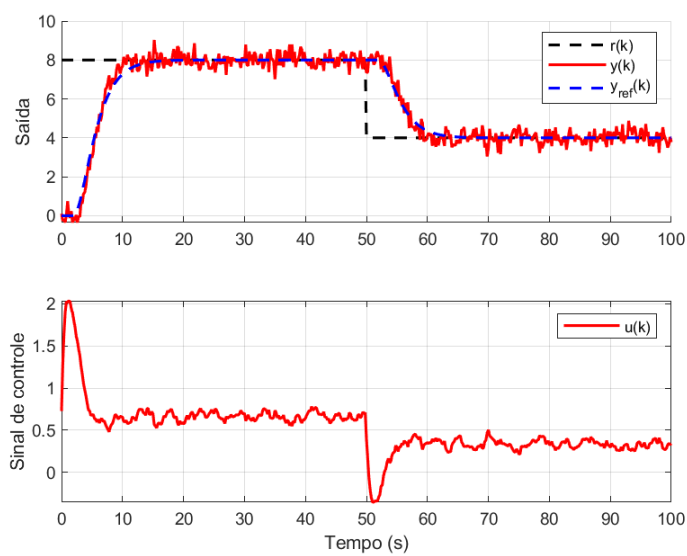
80 e  $N_c = 4$ . A Tab. 2 traz os parâmetros de sintonia obtidos para cada caso, assim como as métricas de desempenho do sistema de controle. As Fig. 18, Fig. 19 e Fig. 20 mostram as respostas obtidas em malha fechada para  $\tau = 2$ ,  $\tau = 3.5$  e  $\tau = 5.7$ , respectivamente.

Os resultados obtidos mostram que para as constantes de tempo  $\tau = 2$  e  $\tau = 3.5$  a resposta em malha fechada aproximou-se consideravelmente da trajetória desejada. Contudo, para o caso em que  $\tau = 5.7$ , a dinâmica obtida pelo sistema não conseguiu ser tão lenta quanto o pretendido pelo método de sintonia. Isto se deve ao fato de que, quanto maior o valor de  $R_k$ , mais lenta a resposta. Entretanto, valores excessivamente grandes de  $R_k$  acabam por induzir a ganhos do controlador  $K_e$  e  $K_w$  muito pequenos de acordo com (2.27) e (2.28). Assim, aumentar o valor de  $R_k$  indefinidamente não mostrou melhoria significativa no RMSE em relação a trajetória de referência.

### 5.1.3 Caso multivariável: modelo 2x2 fracionador de petróleo pesado

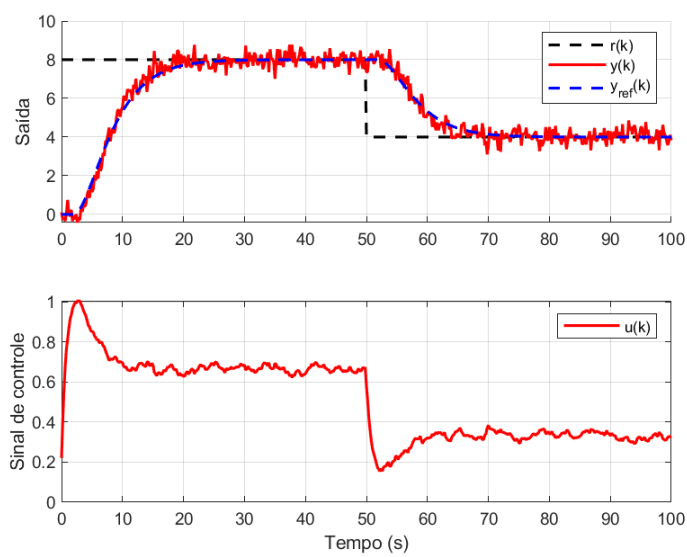
Para realizar a avaliação do controlador, escolheu-se aplicá-lo a um modelo simplificado de um fracionador de petróleo. Este modelo considera um sistema 2x2 com os *Top*

Figura 18 – Avaliação em malha fechada:  $\tau = 2$  - SOPDT

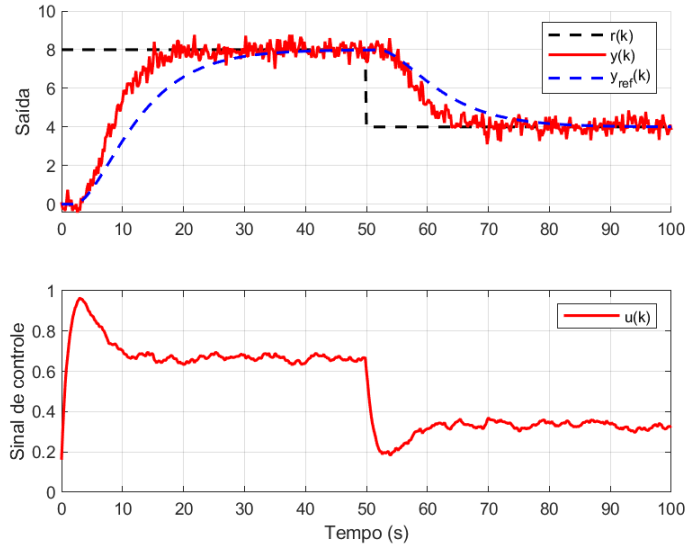


Fonte: Autoria própria.

Figura 19 – Avaliação em malha fechada:  $\tau = 3.5$  - SOPDT



Fonte: Autoria própria.

Figura 20 – Avaliação em malha fechada:  $\tau = 5.7$  - SOPDT

Fonte: Autoria própria.

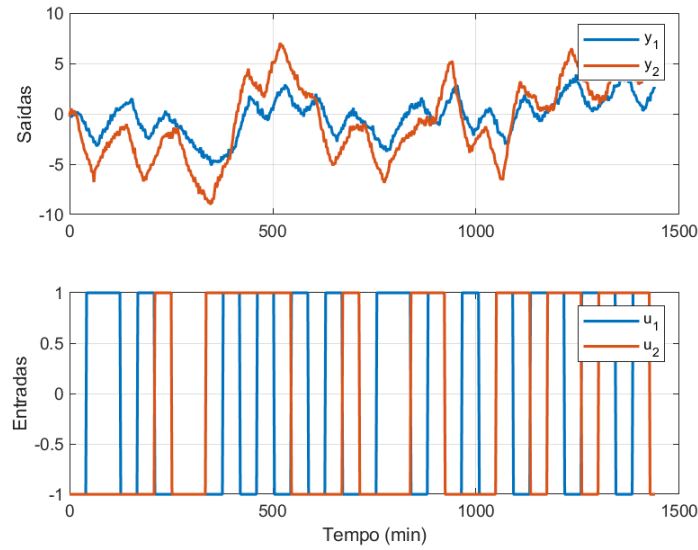
e *Side Draws* como variáveis de controle e os *Top* e *Side End Points* como as saídas do processo. O modelo nominal do fracionador é um sistema  $5 \times 7$ , e mais detalhes sobre ele podem ser encontrados em (PRETT; GARCÍA, 2013). Neste caso, o modelo é:

$$G(s) = \begin{pmatrix} G_{1,1}(s) & G_{1,2}(s) \\ G_{2,1}(s) & G_{2,2}(s) \end{pmatrix} = \begin{pmatrix} \frac{4.05}{50s+1} e^{-27s} & \frac{5.39}{50s+1} e^{-18s} \\ \frac{1.77}{60s+1} e^{-28s} & \frac{5.72}{60s+1} e^{-14s} \end{pmatrix} \quad (5.8)$$

onde  $G_{i,j}(s)$  representa a função de transferência em relação a entrada  $i$  e à saída  $j$ , cujas constantes de tempo estão expressas em minutos. Um experimento em malha aberta foi realizado para colher os dados necessários para calcular as matrizes de predição, apresentado na Fig. 21, com tempo de amostragem  $T_s = 2$ min. A potência do sinal de ruído do tipo ruído branco foi de 0.02.

Para analisar o desempenho do processo em malha fechada, o controlador foi sintonizado a partir do método proposto. Para isto, foram utilizadas duas funções de referência do tipo FOPDT (5.4), sendo uma para cada saída do processo. Neste caso, os parâmetros de referência utilizados foram  $K = 1$ ,  $\theta_1 = 27$ ,  $\theta_2 = 14$  e as constantes de tempo foram 40 e 25 para a  $y_1$  e 50 e 32 para  $y_2$ . Neste caso, as matrizes  $Q_k = \text{diag}(q_1, q_2)$  e  $R_k = \text{diag}(r_1, r_2)$  têm dimensão  $2 \times 2$ . Conforme mencionado na Seção 3.3.2, o elemento  $q_2$  da matriz  $Q_k$  foi mantido constante para simplificar o problema de otimização. Assim, a variável de decisão é  $\mathbf{x} = (r_1, r_2, q_1)$ , com limites superior e inferior iguais a  $(6000, 6000, 20)$  e  $(1, 1, 1)$ , respectivamente. Os horizontes de predição e controle empregados foram  $N_p = 70$  e  $N_c = 3$ . É importante destacar que, ao contrário dos casos univariáveis já apresentados, no caso multivariável é necessário aplicar o método da solução de compromisso para obter as matrizes  $Q_k$  e  $R_k$  que ponderem os objetivos relacionados as funções de referência.

Figura 21 – Experimento em malha aberta - Fracionador de petróleo pesado 2x2



Fonte: Autoria própria.

Tabela 3 – Parâmetros de sintonia e métricas - Fracionador de petróleo pesado 2x2

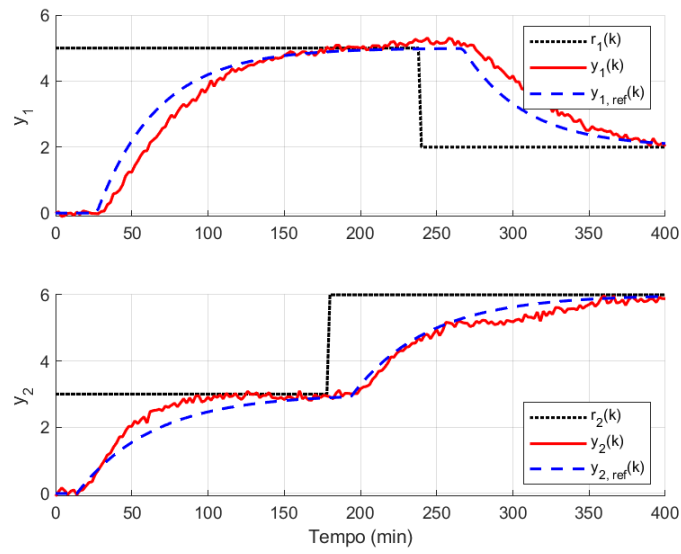
$\tau_1$	$\tau_2$	$N_p$	$N_c$	$Q_k$	$R_k$	RMSE	IAE	TV
40	50	70	3	$\begin{pmatrix} 3.27 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 5924.58 & 0 \\ 0 & 983.57 \end{pmatrix}$	$\begin{pmatrix} 0.4097 \\ 0.2843 \end{pmatrix}$	$\begin{pmatrix} 325.57 \\ 190.28 \end{pmatrix}$	$\begin{pmatrix} 3.9932 \\ 4.7401 \end{pmatrix}$
25	32	70	3	$\begin{pmatrix} 8.31 & 0 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 4333.25 & 0 \\ 0 & 4622.45 \end{pmatrix}$	$\begin{pmatrix} 0.4974 \\ 0.7772 \end{pmatrix}$	$\begin{pmatrix} 283.63 \\ 230.69 \end{pmatrix}$	$\begin{pmatrix} 4.8306 \\ 4.0181 \end{pmatrix}$

Fonte: Autoria própria.

A Tab. 3 traz os parâmetros de sintonia obtidos para cada caso, assim como as métricas de desempenho do sistema de controle. As Fig. ?? e Fig. ?? apresentam as respostas obtidas em malha fechada para  $(\tau_1 = 40, \tau_2 = 50)$  e  $(\tau_1 = 25, \tau_2 = 32)$ , respectivamente.

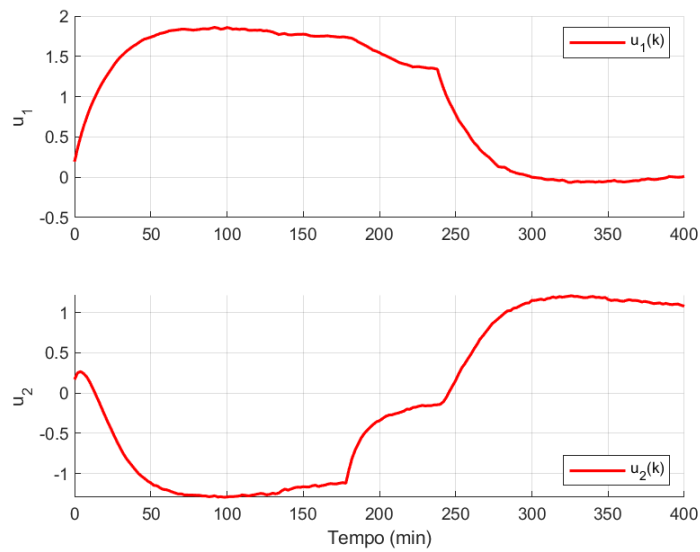
De acordo com as métricas apresentadas na Tab. 3 e as dinâmicas exibidas nas Fig. 22, 23 para o primeiro caso e 24, 25 para o segundo caso, verifica-se que em ambos os casos os resultados obtidos garantem que o método de sintonia é capaz de garantir que o sistema em malha fechada tenha uma dinâmica próxima das trajetórias desejadas. Contudo, é possível observar que para o caso em que deseja-se uma resposta mais rápida, a diferença entre  $y(k)$  e  $y_{ref}(k)$  aumenta consideravelmente. Além disto, a sintonia para uma dinâmica mais agressiva resultou em um sistema com maior sensibilidade a mudanças no *set point* da malha cruzada, como pode ser observado no efeito que a mudança variação do sinal de referência  $r_1(k)$  na saída em malha fechada  $y_2(k)$ . Por fim, as Fig. 22, 23, 24 e 25 mostram que o controlador é capaz de prever os efeitos das mudanças no sinal de referência da

Figura 22 – Resposta em malha fechada:  $\tau_1 = 40$ ,  $\tau_2 = 50$  - Fracionador de petróleo pesado 2x2



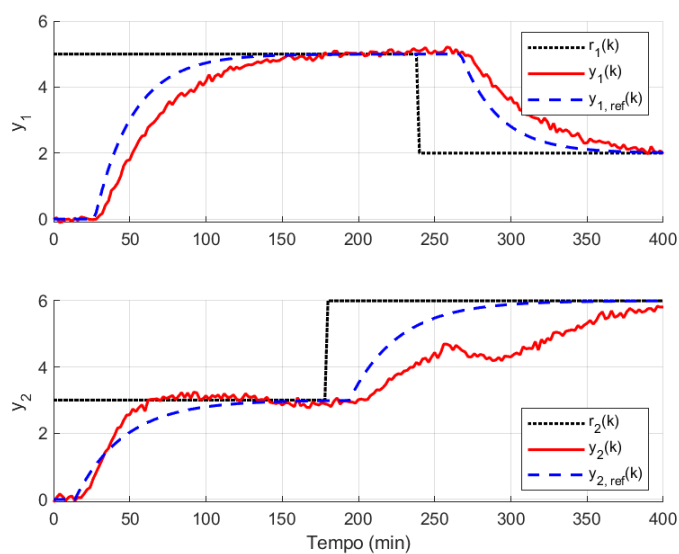
Fonte: Autoria própria.

Figura 23 – Sinais de controle:  $\tau_1 = 40$ ,  $\tau_2 = 50$  - Fracionador de petróleo pesado 2x2



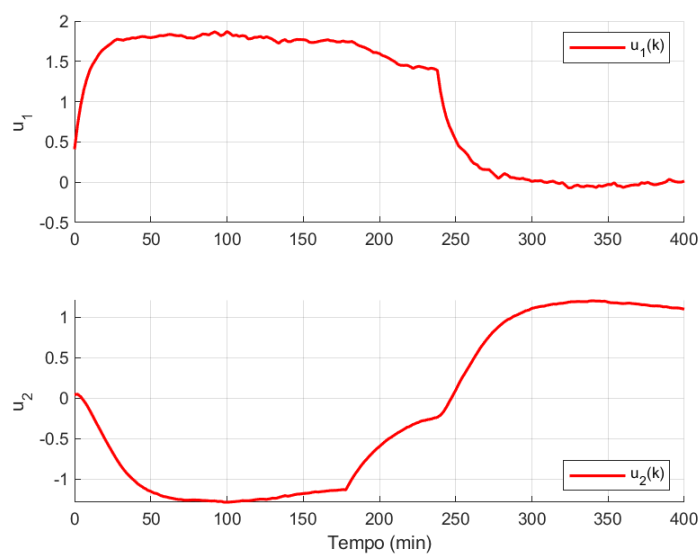
Fonte: Autoria própria.

Figura 24 – Resposta em malha fechada:  $\tau_1 = 25$ ,  $\tau_2 = 32$  - Fracionador de petróleo pesado 2x2



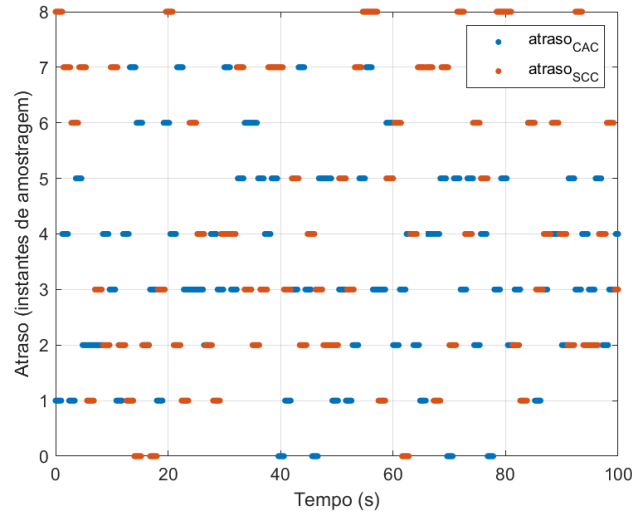
Fonte: Autoria própria.

Figura 25 – Sinais de controle:  $\tau_1 = 25$ ,  $\tau_2 = 32$  - Fracionador de petróleo pesado 2x2



Fonte: Autoria própria.

Figura 26 – Distribuição dos atrasos ao longo do tempo: NCS - FOPDT



Fonte: Autoria própria.

outra malha de controle e já adequa o sinal de controle enviado para o processo de modo a mitigar os efeitos da malha cruzada.

## 5.2 SPC em rede

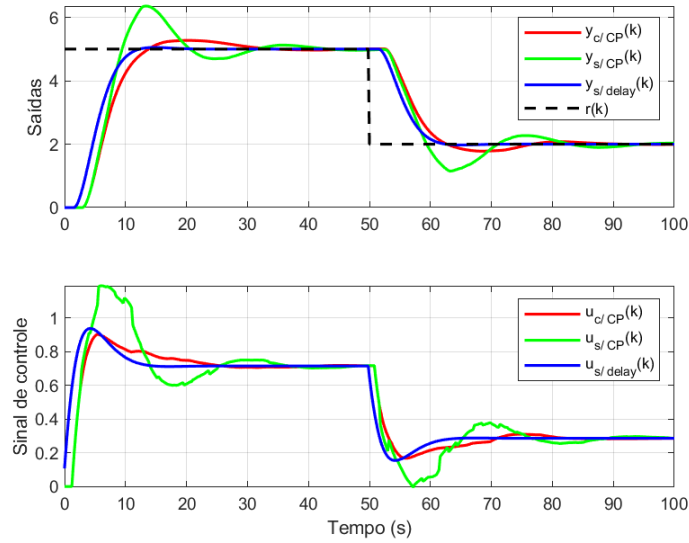
### 5.2.1 Caso univariável: modelo FOPDT

Para avaliar a efetividade da solução proposta no Capítulo 4, escolheu-se aplicá-lo ao modelo utilizado na Seção 5.1.1. Para tanto, manteve-se os parâmetros de sintonias obtidos para a trajetória de referência com  $\tau = 4$ ,  $R_k = 1912.14$  e  $Q_k = 1$ . Contudo, conforme já mencionado anteriormente, para a aplicação do SPC no contexto de controle em rede, é necessário ajustar o tamanho do horizonte de controle. Neste caso,  $N_c = N_p = M = 50$ .

Foram aplicados atrasos de comunicação nos canais de comunicação direta (CAC) e de realimentação (SCC), que foram simulados utilizando sinais aleatórios de distribuição uniforme. A Fig. 26 mostra os atrasos em instantes de amostragem ao longo do tempo de simulação.

As respostas em malha fechada submetidas aos atrasos da Fig. 26 sem a estratégia de compensação  $y_{s/CP}(k)$  e com a estratégia de compensação  $y_{c/CP}(k)$  são comparadas com o caso sem atraso de comunicação  $y_{s/delay}(k)$ . Os resultados são apresentados na Fig. 27, enquanto que a Tab. 4 mostra as métricas de desempenho em todos os casos. A partir deles, é notável que as estratégias de compensação de atraso em ambos os canais melhoram significativamente a qualidade da resposta em malha fechada, com grande diminuição do sobressinal e da oscilação em torno do *set point*. Além disso, o menor TV encontrado

Figura 27 – Resposta em malha fechada: NCS - FOPDT



Fonte: Autoria própria.

no caso com compensação indica que o atuador sofre menor desgaste devido a variação do sinal de controle, quando comparado ao caso sem compensação. Por fim, é possível observar que a dinâmica em malha fechada com atrasos aproxima-se da dinâmica sem atrasos quando há compensação dos atrasos por parte do controlador.

Tabela 4 – Métricas - SPC/NCS: FOPDT

Situação	IAE	TV
Sem atraso	229.33	1.7523
Com atraso, sem compensação	360.92	3.7960
Com atraso, com compensação	317.03	1.8516

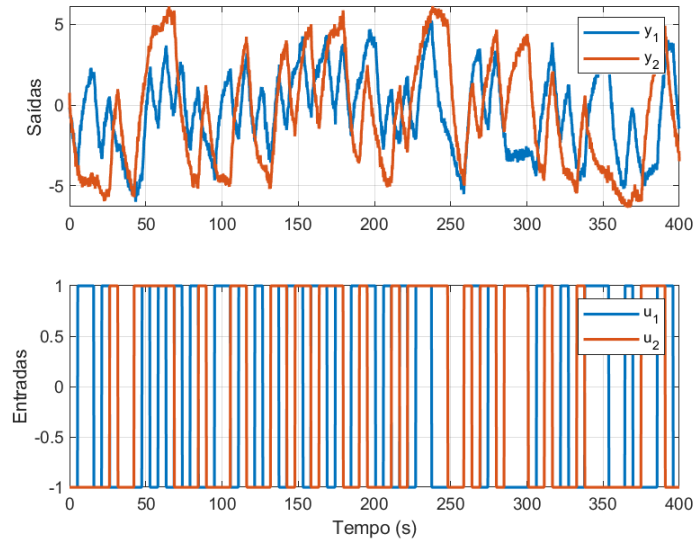
Fonte: Autoria própria

### 5.2.2 Caso multivariável: 2x2 FOPDT

De modo a verificar a efetividade do algoritmo adaptado para o SPC em rede, foi avaliado seu desempenho em um caso multivariável. Contudo, o modelo definido para o fracionador de petróleo pesado na Seção 5.1.3 apresenta constantes de tempo muito lentas, da ordem de dezenas de minutos, com tempo de amostragem igual a dois minutos. Deste modo, uma análise de atrasos de comunicação tem pouca aplicabilidade prática, visto que tais atrasos tem ordem de grandeza consideravelmente menor do que o intervalo de amostragem e não teria impacto relevante na dinâmica em malha fechada. Por este motivo, optou-se por adotar um modelo genérico multivariável 2x2 FOPDT com dinâmica mais



Figura 28 – Experimento em malha aberta - 2x2 FOPDT



Fonte: Autoria própria.

rápida:

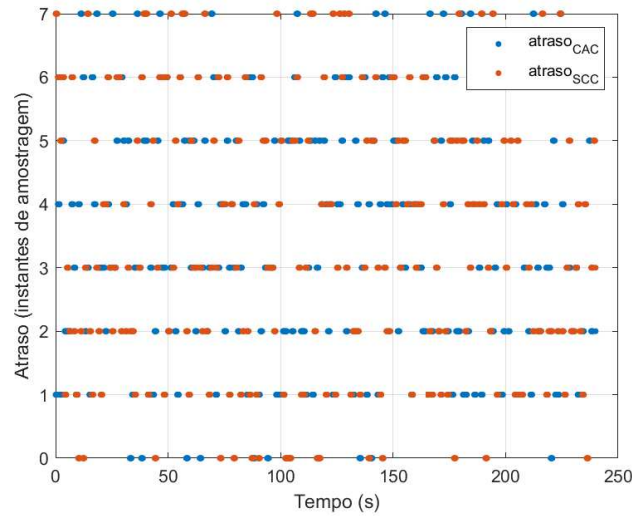
$$G(s) = \begin{pmatrix} \frac{4.5}{5s+1} & \frac{1.2}{7.5s+1} e^{-2.4s} \\ \frac{1.8}{8s+1} e^{-3s} & \frac{5.7}{6s+1} \end{pmatrix} \quad (5.9)$$

com tempo de amostragem  $T_s = 0.2s$ . Os dados necessários para se calcular as matrizes de predição foram obtidos a partir da simulação de um experimento em malha aberta com a aplicação de um PRBS à entrada do modelo (5.9), conforme mostra na Fig. 28. O período de amostragem escolhido foi de  $T_s = 0.2s$ . A potência do sinal de ruído do tipo ruído branco foi de 0.01.

Para avaliar o desempenho do sistema de controle em malha fechada, o controlador foi sintonizado a partir do método já apresentado. Para tanto, foram utilizadas duas funções de referência do tipo primeira ordem sem atraso, sendo uma para cada saída do processo. Neste caso, os parâmetros de referência utilizados foram  $K = 1$  e as constantes de tempo foram  $\tau_1 = 6.5$  e  $\tau_2 = 8$  para as saídas  $y_1$  e  $y_2$  respectivamente. Neste caso, os parâmetros de sintonia a serem obtidos pelo problema MOO são as matrizes  $Q_k = \text{diag}(q_1, q_2)$  e  $R_k = \text{diag}(r_1, r_2)$ , mantendo elemento  $q_2$  da matriz  $Q_k$  mantido constante para simplificar o processo de otimização. A variável de decisão é  $\mathbf{x} = (r_1, r_2, q_1)$ , com limites superior e inferior iguais a  $(8000, 8000, 20)$  e  $(2000, 2000, 1)$ , respectivamente. Como o controlador será utilizado em rede, o horizonte de controle deve ser igual ao horizontes de predição. Para o caso aqui exposto,  $N_p = N_c = M = 65$ . As matrizes são obtidas a partir do método da solução de compromisso:

$$Q_k = \begin{pmatrix} 1.28 & 0 \\ 0 & 1 \end{pmatrix}, R_k = \begin{pmatrix} 2337.28 & 0 \\ 0 & 7308.40 \end{pmatrix}$$

Figura 29 – Distribuição dos atrasos ao longo do tempo: NCS - 2x2 FOPDT



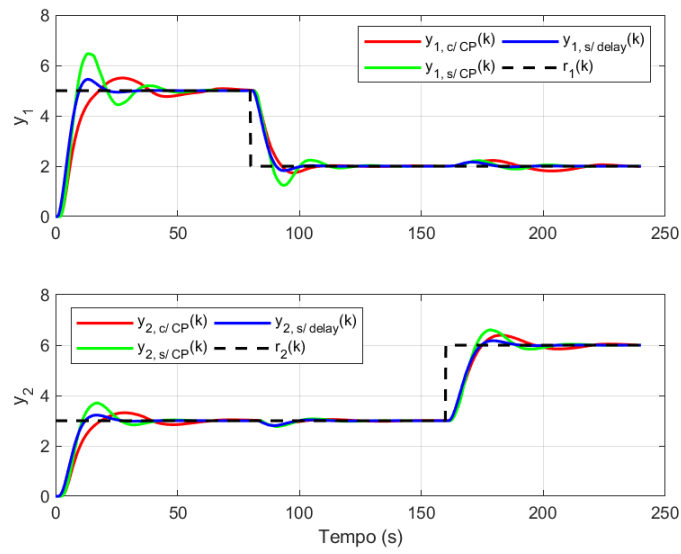
Fonte: Autoria própria.

A avaliação em malha fechada foi realizada a partir da aplicação de atrasos de comunicação nos canais de comunicação direta (CAC) e de realimentação (SCC), que foram simulados utilizando sinais aleatórios de distribuição uniforme. A Fig. 29 mostra os atrasos em instantes de amostragem ao longo do tempo de simulação.

As respostas em malha fechada do modelo (5.9) submetidas aos atrasos apresentados na Fig. 29 sem a estratégia de compensação  $y_{s/CP}(k)$  e com a estratégia de compensação  $y_{c/CP}(k)$  são comparadas com o caso sem atraso de comunicação  $y_{s/delay}(k)$ , conforme mostrada nas Fig. 30 e Fig. 31

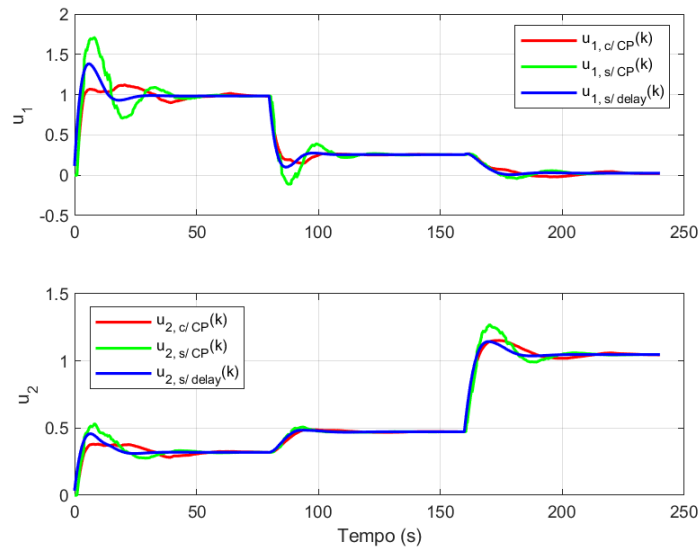
A Tab. 5 mostra as métricas de desempenho para todos os casos. A partir delas e das dinâmicas mostrada nas figuras, é possível observar que as estratégias de compensação de atraso em ambos os canais melhoram a qualidade da resposta em malha fechada. Para ambas as saídas, houve uma diminuição do sobressinal. A dinâmica com a compensação ficou mais lenta e menos agressiva, conforme indicado pelo aumento do IAE e pela diminuição do TV. Esta perda de performance é algo tolerável, visto que a resposta sem compensação apresenta um sobressinal acima de 50% para a  $y_1$ , que em aplicações reais é uma característica bastante indesejável. Além disto, houve o menor TV indica que o atuador sofreria menor desgaste devido a variação do sinal de controle, quando comparado ao caso sem compensação.

Figura 30 – Resposta em malha fechada: NCS - 2x2 FOPDT



Fonte: Autoria própria.

Figura 31 – Sinais de controle em malha fechada: NCS - 2x2 FOPDT



Fonte: Autoria própria.

Tabela 5 – Métricas - SPC/NCS: 2x2 FOPDT

<b>Situação</b>	<b>IAE <math>y_1(k)</math></b>	<b>TV <math>u_1(k)</math></b>	<b>IAE <math>y_2(k)</math></b>	<b>TV <math>u_2(k)</math></b>
Sem atraso	246.89	3.1627	242.04	1.5555
Com atraso, sem compensação	381.63	6.9166	323.48	2.6888
Com atraso, com compensação	398.7066	2.9838	346.67	1.6493

Fonte: Autoria própria

## 6 Conclusões

Esse trabalho teve por objetivo propor um método de controle que fosse capaz de lidar com atrasos intermitentes de comunicação e perdas de pacotes de dados, que são alguns dos principais problemas encontrados para sistemas de controle em rede e são gargalos para sua utilização na indústria. Foram identificadas algumas soluções na literatura, mas, as mais consolidadas limitam-se à aplicações com controle preditivo baseado em modelo.

Neste contexto, foi apresentado o controle preditivo por subespaços, que é capaz de simplificar projetos de sistemas de controle, dado que retira a necessidade de obter um modelo do processo. Além disso, foi mostrado o potencial de uso do SPC em contextos de comunicação em rede devido a sua capacidade de prever a dinâmica do processo e de calcular o sinal de controle para instantes futuros. Foram propostas algumas alterações no controlador, adaptando algumas soluções encontradas na literatura MPC para o contexto SPC.

Nos resultados para a aplicação em rede, foi observado que o emprego do compensador e de preditor para lidar com os atrasos na malha direta e na malha de realimentação, respectivamente, impõe uma dinâmica em malha fechada próxima daquela que é observada no caso em que não há atraso de comunicação. Além disso, foi observado que, caso o sistema de controle não seja adequado para lidar com os atrasos, a resposta em malha fechada acaba tornando-se bastante oscilatória, com elevado sobressinal e uma variação expressiva no sinal de controle, algo que é indesejado.

Adicionalmente, foi identificada na literatura a ausência de métodos de sintonia apropriados para o SPC. Deste modo, este trabalho também contribui com um método de sintonia de controladores preditivos por subespaços. O algoritmo SPC extrai a maioria das informações relevantes para o controlador por meio de dados operacionais do processo. Além disto, devido ao método de sintonia desenvolvido, é possível sintonizar o controlador de modo a obter uma resposta em malha fechada que atenda os requisitos desejados, expressos por meio de constantes de tempo. O método de sintonia apresentado teve seus resultados publicados nos trabalhos (MACIEL; BARROS; LIMA, 2021) e (MACIEL; LIMA, 2022), com aplicações do SPC para o controle de posição de um aeropêndulo e de temperatura para uma plataforma termoelétrica, respectivamente.

Alguns desafios para a implementação de controladores preditivos por subespaços são oriundas da qualidade dos dados disponíveis para o projetista. Além disso, para processos instáveis, esta metodologia é mais difícil de se aplicar devido a impossibilidade de realizar experimentos de malha aberta para a obtenção de dados. Entretanto, uma

possibilidade para solucionar este problema é a utilização de técnicas de identificação por subespaços em malha fechada, que já estão consolidadas na literatura ([OVERSCHEE; MOOR, 1996](#)).

Como trabalhos futuros, planeja-se adotar uma abordagem SPC com restrições as variáveis do processo, incluindo o ajuste do método de sintonia para contemplar estes casos. Neste contexto, a lei de controle passa a ser resolvida como um problema de otimização a cada instante de amostragem, em geral solucionado por meio de programação quadrática ([WANG, 2009](#)). Além disto, pretende-se realizar a implementação do algoritmo SPC em rede em situações reais, utilizando ferramentas de computação em nuvem e plantas piloto.

## Referências Bibliográficas

- ABB. *Model predictive control technology demystified*. 2021. Disponível em: <<https://new.abb.com/control-systems/features/model-predictive-control-mpc>>. Citado na página 11.
- ALHAJERI, M.; SOROUGH, M. Tuning guidelines for model-predictive control. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 59, n. 10, p. 4177–4191, 2020. Citado 3 vezes nas páginas 16, 20 e 21.
- ANTSAKLIS, P.; BAILLIEUL, J. Special issue on technology of networked control systems. *Proceedings of the IEEE*, IEEE, v. 95, n. 1, p. 5–8, 2007. Citado 2 vezes nas páginas 24 e 25.
- CAMACHO, E. F.; ALBA, C. B. *Model predictive control*. [S.l.]: Springer Science & Business Media, 2013. Citado 2 vezes nas páginas 2 e 10.
- CHENARU, O. et al. Open cloud solution for integrating advanced process control in plant operation. In: IEEE. *2015 23rd Mediterranean Conference on Control and Automation (MED)*. [S.l.], 2015. p. 973–978. Citado na página 1.
- COCK, K. D.; MOOR, B. D.; LEUVEN, K. Subspace identification methods. *Contribution to section*, Citeseer, v. 5, p. 933–979, 2003. Citado na página 28.
- DEB, K. Multi-objective optimization. In: *Search methodologies*. [S.l.]: Springer, 2014. p. 403–449. Citado 3 vezes nas páginas 17, 18 e 19.
- FAVOREEL, W.; MOOR, B. D.; GEVERS, M. Spc: Subspace predictive control. *IFAC Proceedings Volumes*, Elsevier, v. 32, n. 2, p. 4004–4009, 1999. Citado 3 vezes nas páginas 8, 12 e 14.
- GARRIGA, J. L.; SOROUGH, M. Model predictive control tuning methods: A review. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 49, n. 8, p. 3505–3515, 2010. Citado 3 vezes nas páginas 16, 20 e 21.
- HEGAZY, T.; HEFEEDA, M. Industrial automation as a cloud service. *IEEE Transactions on Parallel and Distributed Systems*, IEEE, v. 26, n. 10, p. 2750–2763, 2014. Citado na página 1.
- HUANG, B.; KADALI, R. *Dynamic modeling, predictive control and performance monitoring: a data-driven subspace approach*. [S.l.]: Springer, 2008. Citado 3 vezes nas páginas 2, 12 e 14.
- ISERMANN, R.; MÜNCHHOF, M. *Identification of dynamic systems: an introduction with applications*. [S.l.]: Springer, 2011. v. 85. Citado na página 12.
- JAIMES, A. L. et al. An introduction to multiobjective optimization techniques. *Optimization in Polymer Processing*, UK: Nova Science Pub Inc, p. 29–57, 2009. Citado na página 19.

- KATAYAMA, T. *Subspace methods for system identification*. [S.l.]: Springer Science & Business Media, 2006. Citado na página 9.
- KEESMAN, K. J. *System identification: an introduction*. [S.l.]: Springer Science & Business Media, 2011. 1–12 p. Citado 2 vezes nas páginas 5 e 12.
- LI, S.; XU, L. D.; ZHAO, S. The internet of things: a survey. *Information systems frontiers*, Springer, v. 17, n. 2, p. 243–259, 2015. Citado na página 24.
- LJUNG, L. Characterization of the concept of 'persistently exciting' in the frequency domain. Department of Automatic Control, Lund Institute of Technology (LTH), 1971. Citado na página 7.
- MACIEL, V. R. B.; BARROS, S. N.; LIMA, R. B. Sintonia de controlador preditivo por subespaços baseado em pso aplicado a um aeropêndulo. In: IEEE. *2021 14th IEEE International Conference on Industry Applications (INDUSCON)*. [S.l.], 2021. p. 1490–1496. Citado 3 vezes nas páginas 17, 22 e 47.
- MACIEL, V. R. B.; LIMA, R. Subspace predictive control tuning with multiobjective optimization. *IEEE Latin America Transactions*, IEEE, v. 20, n. 5, p. 763–769, 2022. Citado 4 vezes nas páginas 11, 16, 17 e 47.
- MARLER, R. T.; ARORA, J. S. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, Springer, v. 26, n. 6, p. 369–395, 2004. Citado 3 vezes nas páginas 17, 18 e 19.
- MATHWORKS. *Particle swarm optimization - MATLAB particleswarm*. 2022. Disponível em: <<https://www.mathworks.com/help/gads/particleswarm.html>>. Citado na página 31.
- MORARI, M.; LEE, J. H. Model predictive control: past, present and future. *Computers & Chemical Engineering*, Elsevier, v. 23, n. 4-5, p. 667–682, 1999. Citado na página 10.
- OVERSCHEE, P. V.; MOOR, B. D. *Subspace identification for linear systems: Theory, Implementation, Applications*. [S.l.]: Springer Science & Business Media, 1996. Citado 6 vezes nas páginas 5, 6, 7, 9, 28 e 48.
- PERSIS, C. D.; TESI, P. Formulas for data-driven control: Stabilization, optimality, and robustness. *IEEE Transactions on Automatic Control*, IEEE, v. 65, n. 3, p. 909–924, 2019. Citado na página 12.
- PRETT, D. M.; GARCÍA, C. E. *Fundamental process control: Butterworths series in chemical engineering*. [S.l.]: Butterworth-Heinemann, 2013. Citado na página 37.
- QIN, S. J.; BADGWELL, T. A. A survey of industrial model predictive control technology. *Control engineering practice*, Elsevier, v. 11, n. 7, p. 733–764, 2003. Citado na página 10.
- RICHALET, J. et al. Model predictive heuristic control: Applications to industrial processes. *Automatica*, Elsevier, v. 14, n. 5, p. 413–428, 1978. Citado na página 10.
- ROSE, K.; ELDRIDGE, S.; CHAPIN, L. The internet of things: An overview. *The internet society (ISOC)*, Reston, VA, v. 80, p. 1–50, 2015. Citado na página 24.



- SCHWEIDTMANN, A. M. et al. Machine learning meets continuous flow chemistry: Automated optimization towards the pareto front of multiple objectives. *Chemical Engineering Journal*, Elsevier, v. 352, p. 277–282, 2018. Citado na página 20.
- SEDGHIZADEH, S.; BEHESHTI, S. Data-driven subspace predictive control: Stability and horizon tuning. *Journal of the Franklin Institute*, Elsevier, v. 355, n. 15, p. 7509–7547, 2018. Citado na página 16.
- SHI, L.; XIE, L.; MURRAY, R. M. Kalman filtering over a packet-delaying network: A probabilistic approach. *Automatica*, Elsevier, v. 45, n. 9, p. 2134–2140, 2009. Citado na página 28.
- STATISTA. *IoT market size worldwide 2017-2025*. 2019. Disponível em: <<https://www.statista.com/statistics/976313/global-iot-market-size/>>. Citado na página 24.
- UNBEHAUEN, H.; RAO, G. A review of identification in continuous-time systems. *Annual reviews in Control*, Elsevier, v. 22, p. 145–171, 1998. Citado na página 12.
- VAJPAYEE, V.; MUKHOPADHYAY, S.; TIWARI, A. P. Data-driven subspace predictive control of a nuclear reactor. *IEEE Transactions on Nuclear Science*, IEEE, v. 65, n. 2, p. 666–679, 2017. Citado 3 vezes nas páginas 12, 13 e 14.
- WANG, L. *Model predictive control system design and implementation using MATLAB®*. [S.l.]: Springer Science & Business Media, 2009. Citado 4 vezes nas páginas 2, 10, 11 e 48.
- WILLEMS, J. C. et al. A note on persistency of excitation. *Systems & Control Letters*, Elsevier, v. 54, n. 4, p. 325–329, 2005. Citado na página 12.
- XIA, Y. Cloud control systems. *IEEE/CAA Journal of Automatica Sinica*, IEEE, v. 2, n. 2, p. 134–142, 2015. Citado 3 vezes nas páginas 1, 25 e 27.
- XIA, Y.; FU, M.; LIU, G.-P. *Analysis and synthesis of networked control systems*. [S.l.]: Springer Science & Business Media, 2011. v. 409. Citado 3 vezes nas páginas 26, 28 e 29.
- XIA, Y. et al. Data-driven predictive control for networked control systems. *Information Sciences*, Elsevier, v. 235, p. 45–54, 2013. Citado 2 vezes nas páginas 25 e 27.
- YAMASHITA, A. S. et al. Reference trajectory tuning of model predictive control. *Control Engineering Practice*, Elsevier, v. 50, p. 1–11, 2016. Citado 2 vezes nas páginas 17 e 21.