

Universidade Federal de Minas Gerais
Escola de Engenharia
Programa de Pós-Graduação em Engenharia Elétrica

**Controle por Aprendizagem Iterativa aplicado a um
modelo de ventilador mecânico para pequenos animais**

Adler Fonseca de Castro

Orientador: Prof. Leonardo Antônio Borges Tôrres
Coorientador: Prof. Alessandro Beda

26 de agosto de 2019

Universidade Federal de Minas Gerais
Escola de Engenharia
Programa de Pós-Graduação em Engenharia Elétrica

Controle por Aprendizagem Iterativa aplicado a um modelo de ventilador mecânico para pequenos animais

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais como requisito parcial para obtenção do título de Mestre em Engenharia Elétrica

Adler Fonseca de Castro

Orientador: Prof. Leonardo Antônio Borges Tôres
Coorientador: Prof. Alessandro Beda

26 de agosto de 2019

C355c

Castro, Adler Fonseca de.

Controle por aprendizagem iterativa aplicado a um modelo de ventilador mecânico para pequenos animais [recurso eletrônico] / Adler Fonseca de Castro. – 2019.

1 recurso online (92 f. : il., color.) : pdf.

Orientador: Leonardo Antônio Borges Tôres.

Coorientador: Alessandro Beda.

Dissertação (mestrado) Universidade Federal de Minas Gerais, Escola de Engenharia.

Anexos: f. 95-97.

Bibliografia: f. 93-94.

Exigências do sistema: Adobe Acrobat Reader.

1. Engenharia elétrica - Teses. 2. Sistemas não lineares - Teses. 3. Respiradores (Medicina) - Teses. 4. Simulação (Computadores) - Teses. 5. Respiração artificial - Teses. I. Tôres, Leonardo Antônio Borges. II. Beda, Alessandro. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV. Título.

CDU: 621.3(043)

"Controle por Aprendizagem Iterativa aplicado a um modelo de ventilador mecânico para pequenos animais"

Ádler Fonseca de Castro

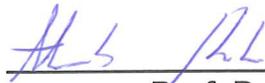
Dissertação de Mestrado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Mestre em Engenharia Elétrica.

Aprovada em 24 de junho de 2019.

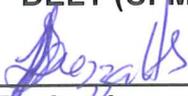
Por:



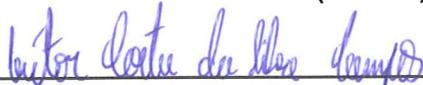
Prof. Dr. Leonardo Antônio Borges Tôrres
DELT (UFMG) - Orientador



Prof. Dr. Alessandro Beda
DELT (UFMG) - Coorientador



Prof. Dr. Luciano Antonio Frezzatto Santos
DELT (UFMG)



Prof. Dr. Víctor Costa da Silva Campos
DELT (UFMG)

Agradecimentos

Agradeço ao meu orientador Leonardo Tôres pela paciência, confiança, e suporte ao longo de todo o mestrado. Nos momentos em que não conseguia progredir, às vezes uma pequena conversa ou email já eram fonte de incentivo e motivação.

Agradeço ao professor Alessandro Beda por ter se disposto a revisar o texto, dar sugestões relevantes e ajudar na melhora desta dissertação.

Agradeço a meus pais, Aloísio e Marina, por tudo que têm feito por mim desde que nasci. Agradeço pelos pequenos gestos de carinho e preocupação que me fazem lembrar que sempre posso contar com vocês.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelo apoio financeiro.

Agradeço ainda a Deus, por tudo.

Resumo

Ventiladores mecânicos são máquinas usadas para assistir a respiração e são amplamente utilizados em pesquisas envolvendo doenças respiratórias. No entanto, a maior parte das opções comerciais disponíveis para pequenos animais possuem limitações em rastrear perfis ventilatórios de pressão ou vazão de ar. Controle por Aprendizagem Iterativa ou *Iterative Learning Control* (ILC) é uma técnica de controle que visa melhorar o desempenho de sistemas com tarefas repetitivas através do aprendizado com execuções anteriores. Este estudo propõe avaliar estratégias de controle com ILC para o problema de rastreamento de perfis associados a modos de ventilação artificial.

Utilizamos estratégias baseadas em ILC, em um controlador PI tradicional e em uma combinação de ambos. Os sistemas de controle foram projetados com base em funções de transferência, obtidas a partir de um modelo simplificado de um ventilador mecânico linearizado por realimentação. O desempenho dos sistemas foi avaliado por meio de simulações, onde incorporamos cenários hipotéticos com vazamentos, ruído na medição e incertezas paramétricas na linearização.

Tendo em consideração que desvios do perfil de referência podem causar danos ao pulmão do paciente, ILC sozinho se mostrou inadequado para ventiladores nas iterações iniciais. A arquitetura com ILC e PI combinou as vantagens das duas estratégias, com um desempenho inicial aceitável, mas que melhora a cada iteração. Essa estratégia com PI e ILC permaneceu robusta para os diversos tipos de perturbações testados, especialmente nos cenários com vazamentos periódicos.

Palavras-chave: Controle por Aprendizagem Iterativa, Iterative Learning Control, Ventilação mecânica, Controle em malha fechada, Sistemas biomédicos, Controle repetitivo, Sistemas não-lineares, Linearização por realimentação.

Abstract

Mechanical ventilators are machines used to assist breathing and are widely used in research involving respiratory diseases. However, most commercial options available for small animals have limitations when tracking ventilatory profiles such as desired air pressure or flow. Iterative Learning Control (ILC) is a control technique that aims to improve performance of systems with repetitive tasks by learning from previous executions. This study proposes ILC control strategies for the problem of tracking profiles associated with ventilation modes.

We use strategies based only in ILC, in a classical PI controller and in a combination of both. The control systems design was based on transfer functions, obtained from a simplified model of a feedback linearized recently proposed ventilator. The systems performance was evaluated with simulations, where we incorporated hypothetical scenarios with leakages, sensor noise and parametric uncertainty in the linearization.

Considering that deviations from the reference profile can harm the patient's lung, ILC alone was shown to be inadequate for mechanical ventilators in the initial iterations. The architecture with ILC and PI combined the benefits of both strategies, with an acceptable initial performance that still improves with each iteration. This strategy with PI and ILC remained robust for all the types of disturbances tested, especially in the scenarios with periodic leakage.

Keywords: Iterative learning control, Mechanical ventilation, Feedback control, Biomedical systems, Process control, Repetitive control, Feedback linearization.

Lista de ilustrações

Figura 1.1 – Sinais de pressão e vazão nas vias aéreas de um rato traqueostomizado sob ventilação artificial	22
Figura 2.1 – Circuito pneumático do ventilador mecânico.	24
Figura 2.2 – Equivalente elétrico do circuito do ventilador.	27
Figura 3.1 – Ilustração de um sistema controlado por ILC.	32
Figura 3.2 – Arquitetura de controle com ILC e controle em malha fechada tradicional. . .	38
Figura 4.1 – Circuito inspiratório simplificado do ventilador.	45
Figura 4.2 – Circuito inspiratório linearizado do ventilador.	46
Figura 4.3 – Comparação das respostas das funções de transferências $G_1(z)$ (b) e $G_2(z)$ (c) com o sistema original com linearização por realimentação para a entrada $v(t)$ (a).	48
Figura 4.4 – Comparação da vazão de saída das três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.	53
Figura 4.5 – Comparação do volume fornecido de ar nas três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.	53
Figura 4.6 – Erro RMS da vazão para as duas arquiteturas com ILC - Modelo linear - Modo controlado a volume.	54
Figura 4.7 – Erro RMS do volume para as duas arquiteturas com ILC - Modelo linear - Modo controlado a volume.	54
Figura 4.8 – Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.	55
Figura 4.9 – Comparação das saídas das três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a pressão.	56
Figura 4.10–Erro RMS para as duas arquiteturas com ILC - Modelo linear - Modo controlado a pressão.	56
Figura 4.11–Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a pressão.	57
Figura 4.12–Comparação da vazão de saída nas três arquiteturas de controle após 10 iterações - Modo controlado a volume.	59
Figura 4.13–Comparação da vazão de saída nas três arquiteturas de controle após 100 iterações - Modo controlado a volume.	59

Figura 4.14–Comparação do volume fornecido de ar nas três arquiteturas de controle após 10 iterações - Modo controlado a volume.	60
Figura 4.15–Comparação do volume fornecido de ar nas três arquiteturas de controle após 100 iterações - Modo controlado a volume.	60
Figura 4.16–Erro RMS da vazão para as duas arquiteturas com ILC - Modo controlado a volume.	61
Figura 4.17–Erro RMS do volume para as duas arquiteturas com ILC - Modo controlado a volume.	61
Figura 4.18–Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações - Modo controlado a volume.	62
Figura 4.19–Comparação das saídas das três arquiteturas de controle após 10 iterações - Modo controlado a pressão.	63
Figura 4.20–Comparação das saídas das três arquiteturas de controle após 100 iterações - Modo controlado a pressão.	64
Figura 4.21–Erro RMS para as duas arquiteturas com ILC - Modo controlado a pressão.	64
Figura 4.22–Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações - Modo controlado a pressão.	65
Figura 4.23–Circuito do ventilador com vazamento hipotético.	65
Figura 4.24–Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento periódico - Modo controlado a volume.	66
Figura 4.25–Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento não-periódico - Modo controlado a volume.	67
Figura 4.26–Comparação da saída do sistema controlado pelo PI com o controlado por PI + [ILC atualizado a cada 5 iterações] para o cenário com vazamento não-periódico - Modo controlado a volume.	67
Figura 4.27–Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com vazamento periódico - Modo controlado a pressão.	68
Figura 4.28–Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento não-periódico - Modo controlado a pressão.	69
Figura 4.29–Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com ruído na medição de q_0 e P_2 - Modo controlado a volume.	70
Figura 4.30–Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com ruído na medição de P_0 e P_2 - Modo controlado a pressão.	70
Figura 4.31–Comparação das saídas do sistema controlado por PI para algumas estimativas de P_1 - Modo controlado a volume.	72

Figura 4.32–Comparação das saídas do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.	72
Figura 4.33–Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.	73
Figura 4.34–Comparação das saídas do sistema controlado por PI para algumas estimativas de R_1 - Modo controlado a volume.	73
Figura 4.35–Comparação das saídas do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a volume.	74
Figura 4.36–Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.	74
Figura 4.37–Comparação das saídas do sistema controlado por PI para grandes erros de estimativas de P_1 e R_1 - Modo controlado a volume.	75
Figura 4.38–Comparação das saídas do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a volume.	75
Figura 4.39–Erro RMS do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a volume.	76
Figura 4.40–Comparação da saída do sistema controlado por PI para algumas estimativas de P_1 - Modo controlado a pressão.	77
Figura 4.41–Comparação da saída do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a pressão.	77
Figura 4.42–Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a pressão.	78
Figura 4.43–Comparação da saída do sistema controlado por PI para algumas estimativas de R_1 - Modo controlado a pressão.	78
Figura 4.44–Comparação da saída do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a pressão.	79
Figura 4.45–Erro RMS do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a pressão.	79
Figura 4.46–Comparação da saída do sistema controlado por PI para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.	80
Figura 4.47–Comparação da saída do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.	80
Figura 4.48–Erro RMS do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.	81
Figura 4.49–(a) Magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{1000}(e^{i\theta})$, (b) quadrado da magnitude de $Q(e^{i\theta})$	82
Figura 4.50–(a) Magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{1000}(e^{i\theta})$ após filtragem, (b) magnitude da resposta em frequência de $1 - e^{i\theta} L(e^{i\theta}) \bar{G}_{1000}(e^{i\theta})$ após filtragem	83

Figura 4.51–Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 1000$ Hz	84
Figura 4.52–Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 1000$ Hz	84
Figura 4.53–(a) Magnitude da resposta em frequência de $1 - e^{i\theta} \tilde{G}_{2d_{1000}}(e^{i\theta})$, (b) quadrado da magnitude de $Q(e^{i\theta})$	86
Figura 4.54–(a) Magnitude da resposta em frequência de $1 - e^{i\theta} \tilde{G}_{2d_{1000}}(e^{i\theta})$ após filtragem, (b) magnitude da resposta em frequência de $1 - e^{i\theta} L(e^{i\theta}) \tilde{G}_{2d_{1000}}(e^{i\theta})$ após filtragem	86
Figura 4.55–Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 1000$ Hz	87
Figura 4.56–Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 1000$ Hz	87
Figura 4.57–Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 100$ Hz	88
Figura 4.58–Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 100$ Hz	89
Figura 4.59–Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 100$ Hz	90
Figura 4.60–Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 100$ Hz	90

Lista de abreviaturas e siglas

ILC	<i>Iterative Learning Control</i> (Controle por Aprendizagem Iterativa)
PEEP	<i>Positive End Expiratory Pressure</i> (Pressão Positiva Expiratória Final)
BIBO	<i>Bounded Input Bounded Output</i> (Entrada Limitada Saída Limitada)
SISO	<i>Single Input Single Output</i> (Uma Entrada Uma Saída)
MIMO	<i>Multiple Inputs Multiple Outputs</i> (Múltiplas Entradas Múltiplas Saídas)
LIT	Linear e invariante no tempo
AS	<i>Asymptotically Stable</i> (Assintoticamente Estável)
RMS	<i>Root Mean Square</i> (Erro Quadrático Médio)
NMC-ILC	<i>Nonlinear Monotonically Convergent Iterative Learning Control</i> (Controle por Aprendizagem Iterativa Não-Linear Monotonicamente Convergente)
PWM	<i>Pulse Width Modulation</i> (Modulação por Largura de Pulso)
PI	Proporcional-Integral
DC	<i>Direct Current</i> (Corrente Contínua)
TEB	<i>Total Error Banda</i> (Banda de Erro Total)

Sumário

1	INTRODUÇÃO	21
1.1	Motivação	21
1.2	Objetivos do trabalho	21
1.3	Organização do texto	22
2	O VENTILADOR MECÂNICO	23
2.1	Modos de ventilação	23
2.2	Descrição da arquitetura	24
2.2.1	Comandos de válvula	26
2.3	Modelagem matemática	26
3	CONTROLE POR APRENDIZAGEM ITERATIVA	29
3.1	Estratégia de Controle por Aprendizagem Iterativa - Visão Geral	29
3.2	ILC linear	30
3.2.1	Sistema Iterativo Linear e Invariante no Tempo	30
3.2.2	Análise de Estabilidade	34
3.2.3	Robustez	38
3.2.4	Projeto	40
3.2.5	Formulação Alternativa	41
3.3	Breve revisão sobre ILC não-linear	43
4	ESTRATÉGIA ILC APLICADA AO VENTILADOR	45
4.1	Simplificação do sistema	45
4.2	Linearização por realimentação	46
4.2.1	Validação	47
4.2.2	Presença de incertezas	48
4.3	Configuração das simulações	50
4.4	Resultados	51
4.4.1	Simulações lineares	52
4.4.1.1	Modo controlado a volume	52
4.4.1.2	Modo controlado a pressão	55
4.4.2	Simulações no ventilador original linearizado	57
4.4.2.1	Modo controlado a volume	57
4.4.2.2	Modo controlado a pressão	62
4.4.3	Simulações com vazamento	65
4.4.3.1	Modo controlado a Volume	66

4.4.3.2	Modo controlado a pressão	68
4.4.4	Ruído na medição	69
4.4.5	Incertezas paramétricas	71
4.4.5.1	Modo controlado a volume	71
4.4.5.2	Modo controlado a pressão	76
4.4.6	Variação na frequência de amostragem	81
4.4.6.1	Modo controlado a volume, $F_s = 1000$ Hz	81
4.4.6.2	Modo controlado a pressão, $F_s = 1000$ Hz	85
4.4.6.3	Modo controlado a volume, $F_s = 100$ Hz	88
4.4.6.4	Modo controlado a pressão, $F_s = 100$ Hz	89
5	CONCLUSÃO	91
5.1	Apanhado geral do que foi feito	91
5.2	Principais resultados e limitações do estudo	92
5.3	Propostas de continuidade	93
	REFERÊNCIAS	95
	APÊNDICES	97
	APÊNDICE A – PARÂMETROS DE SIMULAÇÃO	99

1 Introdução

Ventiladores mecânicos são máquinas usadas para substituir ou auxiliar a respiração espontânea, melhorando a ventilação e reduzindo o trabalho dos músculos respiratórios. Existe uma tendência em se usar pequenos animais em experimentos científicos envolvendo doenças respiratórias. Nesses estudos são necessários ventiladores mecânicos. No entanto, a maior parte das opções disponíveis comercialmente apresenta limitações no rastreamento de perfis ventilatórios [1]. Em [2], propõe-se uma arquitetura de um ventilador mecânico por pressão positiva que visa superar algumas dessas limitações.

Controle por Aprendizagem Iterativa ou *Iterative Learning Control* (ILC), por outro lado, é uma técnica de controle baseada na noção de que o desempenho de um sistema que opera repetitivamente pode ser melhorado por meio de aprendizado com informações das execuções anteriores [3]. ILC coleta amostras do erro de rastreamento produzido por uma entrada de controle durante uma iteração, para então calcular a ação de controle para próxima iteração.

1.1 Motivação

Um bom ventilador mecânico requer um rastreamento preciso de determinados perfis de pressão ou vazão de ar. Esses são baseados nos ciclos de inspiração e expiração, que se repetem enquanto o paciente estiver respirando. A Figura 1.1 mostra alguns exemplos de sinais de pressão e vazão de ar usados em animais pequenos. Um contexto em que se executa a mesma tarefa várias vezes, com as mesmas condições iniciais, é um cenário bastante apropriado para aplicação de ILC [3], [4], [5].

Em 2015, um trabalho relacionado [4], usou uma estratégia ILC para controlar um ventilador mecânico. O presente estudo gerou como contribuição o artigo [7], cujo valor científico está no novo cenário de aplicação de ILC, que considera a nova topologia eletromecânica e de acionamento de um ventilador mecânico de pequenos animais [2].

1.2 Objetivos do trabalho

Tendo em vista o exposto acima, este trabalho tem como objetivos:

1. Propor estratégias de controle baseadas em ILC para o ventilador mecânico proposto em [2].
2. Comparar o desempenho de uma arquitetura de controle usando um controlador clássico com uma baseada somente em ILC, e com outra híbrida.

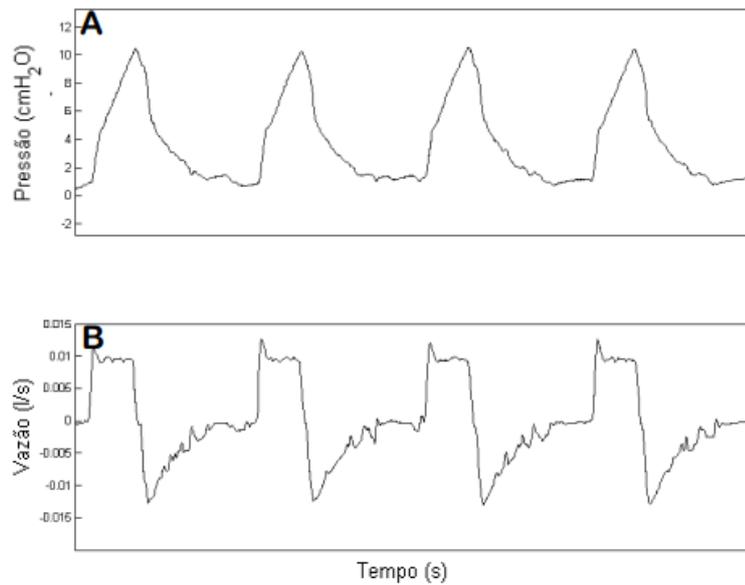


Figura 1.1 – Sinais de pressão (A) e vazão (B) nas vias aéreas de um rato traqueostomizado sob ventilação artificial. Fonte: [6]

3. Avaliar a robustez a incertezas e a perturbações da arquitetura baseada em ILC e controle em malha fechada.

1.3 Organização do texto

Organizamos o trabalho da seguinte forma: no capítulo 2 fizemos uma breve descrição do ventilador mecânico proposto em [2], para permitir ao leitor entender os aspectos essenciais de seu funcionamento e modelagem. No capítulo 3 apresentamos os conceitos e formulação matemática da técnica de *Iterative Learning Control*, com revisão de seu estado da arte. No capítulo 4 descrevemos como foi feito o projeto dos sistemas de controle com ILC e avaliamos o desempenho das estratégias consideradas por meio de simulação. No capítulo 5 concluímos o trabalho com um apanhado geral do que foi feito, principais resultados e propostas de continuidade.

2 O ventilador mecânico

Nosso sistema completo é formado pelo ventilador mecânico (a planta) em conjunto com o sistema de controle. A arquitetura de ventilador que usamos neste trabalho foi proposta em [2], e o objetivo deste capítulo é descrevê-la de forma sucinta para que o leitor possa entender como foi feita sua modelagem. Exceto onde explicitamente houver citação de outras fontes, o conteúdo deste capítulo está baseado em [2].

2.1 Modos de ventilação

A inspiração natural ocorre por pressão negativa, onde a contração do diafragma em conjunto com o trabalho dos músculos intercostais fazem com que a caixa torácica se expanda e que a pressão no pulmão fique menor que a atmosférica. Essa diferença de pressão impulsiona a entrada de ar e faz com que o volume pulmonar aumente [8]. O volume normal de ar inspirado é chamado de volume corrente.

Seguindo as ideias adotadas em [2], podemos definir um modo de ventilação como o modo de funcionamento de um ventilador mecânico em que se estabelece como a respiração será iniciada, como será finalizada, e como se dará o controle da entrega dos gases respiratórios.

O ventilador alvo deste trabalho implementa dois modos de ventilação por pressão positiva. Na ventilação por pressão positiva, a inspiração ocorre pelo fornecimento de um volume de ar ao pulmão do paciente através do estabelecimento de uma pressão de entrada maior do que aquela no interior do pulmão. Em seguida, a expiração é realizada de forma passiva, através da diferença entre a pressão pulmonar e atmosférica.

Os modos de ventilação que o sistema permite são baseados em pressão ou volume de ar. No modo controlado a pressão, a vazão de entrada de ar é gerada pela pressão relativa a atmosférica que é aumentada pelo ventilador de acordo com o perfil de referência dado por

$$P_{\text{ref}} = \min(P_{\text{PEEP}} + \beta t, P_{\text{max}}) \quad \text{e} \quad 0 < t < t_{\text{insp}}, \quad (2.1)$$

onde β é a inclinação desejada para a rampa de pressão, t_{insp} é o tempo de duração da inspiração, t é o tempo decorrido a partir do início da inspiração, P_{max} é a pressão máxima permitida e P_{PEEP} é a pressão positiva expiratória final.

No modo controlado a volume, um volume corrente pré-determinado V_T é fornecido ao paciente através da integração da vazão de entrada de ar, a qual deve ser controlada para seguir o perfil dado por

$$q_{\text{ref}} = \begin{cases} \gamma, & \text{se } \gamma t < V_T \text{ e } 0 < t < t_{\text{insp}} \\ 0, & \text{caso contrário.} \end{cases} \quad (2.2)$$

onde γ é a vazão desejada durante a inspiração.

2.2 Descrição da arquitetura

A Figura 2.1 mostra o circuito pneumático do ventilador mecânico considerado neste trabalho, acoplado ao paciente. A principal inovação da arquitetura está no uso de válvulas solenóides de alta velocidade de abertura e fechamento, em conjunto com filtros mecânicos pneumáticos. Esta arquitetura se contrapõe a arquitetura usual baseada em êmbolos de cilindros pneumáticos que são movidos por motores elétricos. Os elementos do sistema ventilador mecânico e paciente são:

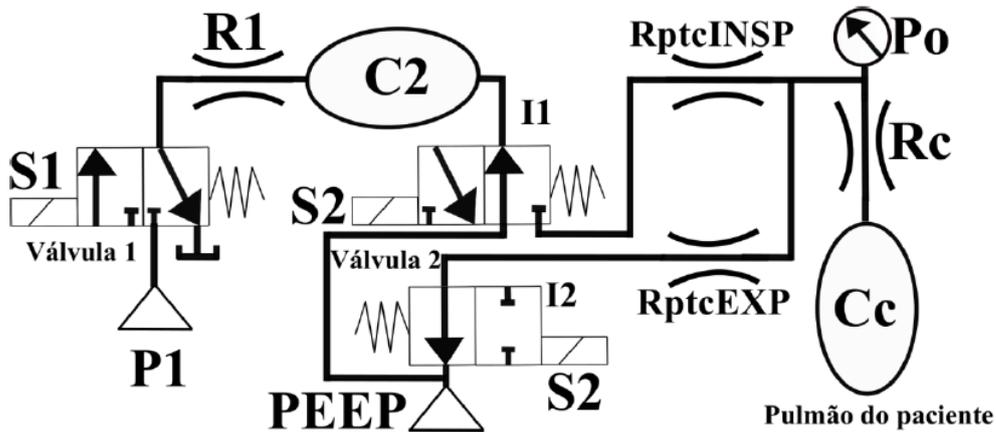


Figura 2.1 – Circuito pneumático do ventilador mecânico. Extraído de [2].

- **Fontes de pressão**

As fontes de pressão no circuito são P1, um cilindro que fornece uma pressão regulada constante, e PEEP, uma coluna de água. PEEP (Positive End Expiratory Pressure) é uma pressão positiva fornecida pelo ventilador ao final da expiração. É usada para evitar que os alvéolos pulmonares se fechem completamente.

- **Reservatórios de pressão**

Os reservatórios de pressão no circuito são C2 e Cc. No caso do pulmão, Cc representa a complacência pulmonar. A complacência é definida como a razão entre uma variação de volume (ΔV) devido a uma variação de pressão (ΔP) [8]:

$$\text{Complacência} = \frac{\Delta V}{\Delta P}.$$

C2 é a "complacência aparente" (equivalente a uma complacência devido a compressibilidade do ar) de um reservatório com paredes rígidas e volume fixo. É proporcional ao volume e inversamente proporcional à temperatura do reservatório.

- **Resistências pneumáticas**

As resistências pneumáticas no circuito são R_1 , R_c , $R_{ptcINSP}$ e R_{ptcEXP} . Uma resistência pneumática é um elemento que se opõe à vazão do fluido provocando uma queda de pressão. A relação entre a perda de carga (ΔP), vazão (Q) e resistência pneumática (R) é dada por:

$$\Delta P = RQ.$$

Essa equação é um modelo simplificado de resistência que assume escoamento laminar e incompressibilidade dos gases. Modelos mais precisos não são lineares.

R_1 é uma resistência pneumática projetada para atuar juntamente com o reservatório de pressão C2 como um filtro passa-baixas pneumático (atenuação de oscilações de pressão de alta frequência). R_c representa a resistência natural das vias aéreas do paciente. $R_{ptcINSP}$ e R_{ptcEXP} são pneumotacógrafos unicapilares responsáveis pela medição da vazão de ar, respectivamente durante a inspiração e expiração. Pneumotacógrafos medem a vazão por meio da queda de pressão através de um tubo capilar com resistência pneumática conhecida.

- **Transdutor de pressão**

O circuito possui um transdutor de pressão manométrica que mede P_0 , a pressão de saída do ventilador e de entrada nas vias aéreas do paciente.

- **Válvulas solenóides**

O circuito possui duas válvulas solenóides *on-off*, S1 e S2. A válvula de 3 vias e 2 posições S1 é o elemento de atuação responsável pelo controle da pressão ou vazão de ar durante o período de tempo correspondente à inspiração.

A válvula S2 é responsável pela mudança entre as etapas de inspiração e expiração do ciclo respiratório. Ela representa a composição de uma válvula de 3 vias e 2 posições I1 com uma válvula de 2 vias e 2 posições I2 comandadas pelo mesmo sinal.

O tempo de resposta de uma válvula (T_{resp}) é definido como o intervalo de tempo a partir da aplicação do comando elétrico para mudança de posição até que a pressão a jusante da válvula atinja 90% da pressão a montante da mesma, caso se trate de abertura; ou 10% da pressão a montante, caso se trate de uma manobra de fechamento.

- **Conexões**

As conexões entre os componentes são feitas com tubulações projetadas para ocupar o menor espaço possível. Uma cânula em forma de Y faz a ligação entre o circuito inspiratório, o expiratório e o paciente.

2.2.1 Comandos de válvula

Os sinais de comandos das válvulas $S_1(t) \in \{0,1\}$ e $S_2(t) \in \{0,1\}$ são definidos como 0 quando a válvula está fechada e 1 quando aberta. Como já dito, a vazão de ar é controlada durante a inspiração e ocorre passivamente durante a expiração. Como consequência, S2 é comandada para permanecer aberta na inspiração e fechada na expiração. S1 permanece fechada na expiração e é controlada durante a inspiração para fornecer o perfil de vazão de ar desejado. A Tabela 2.1 sumariza os comandos das válvulas.

Tabela 2.1 – Comandos das válvulas

Estágio	Modo	$S_1(t)$	$S_2(t)$
Inspiração	1	1	1
	2	0	1
Expiração	3	0	0

Considerando as variáveis $x_{v1}(t)$ e $x_{v2}(t)$ para representar a abertura da válvula, pode-se adotar o modelo de um atraso puro em relação ao sinal de comando $S_k(t)$:

$$x_{vk}(t) = S_k(t - t_0), \quad k = 1,2;$$

onde $t_0 = 0,45T_{\text{resp}}$ é um tempo morto de 45% do tempo de resposta da válvula. As resistências das válvulas são $Rv_{1\text{max}}$ e $Rv_{2\text{max}}$, quando elas estão fechadas, e $Rv_{1\text{min}}$ e $Rv_{2\text{min}}$ quando estão abertas.

2.3 Modelagem matemática

O sistema na Figura 2.1 pode ser modelado pelo equivalente elétrico mostrado na Figura 2.2, de modo que a vazão e a pressão são representadas como tensão e corrente elétricas. Resistências pneumáticas, reservatórios e fontes de pressão se tornam, respectivamente, resistores elétricos, capacitores e fontes de tensão. As válvulas solenóides são modeladas como resistores cujas resistências podem ser alteradas abruptamente entre seus valores máximos e mínimos.

Na Figura 2.2, P_{v1} é a pressão na saída da válvula S1, P_2 é a pressão na saída do reservatório pneumático intermediário C2, e P_c é a pressão nos pulmões do paciente.

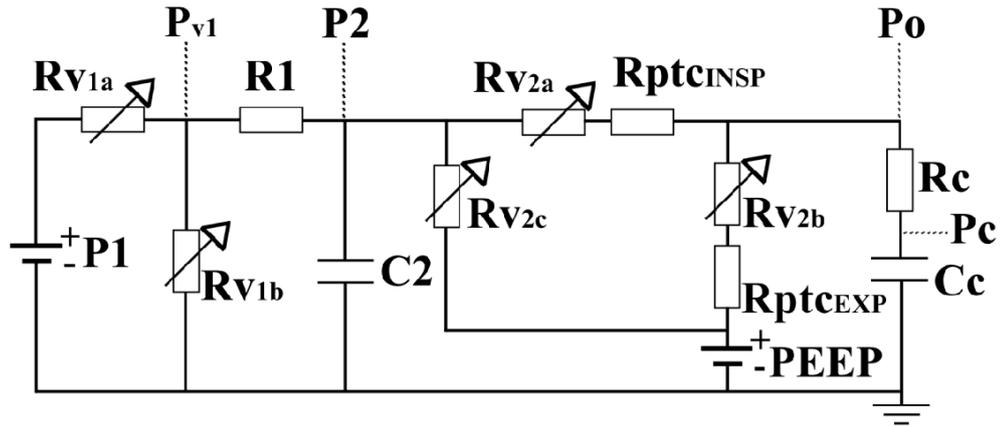


Figura 2.2 – Equivalente elétrico do circuito do ventilador. Extraído de [2].

A partir do circuito equivalente elétrico podemos obter equações diferenciais que representam a dinâmica do circuito pneumático. Tomando as pressões nos capacitores P_2 e P_c como variáveis de estado e $S_1(t)$ e $S_2(t)$ como entradas de controle, temos¹:

$$\begin{bmatrix} \frac{dP_2}{dt} \\ \frac{dP_c}{dt} \end{bmatrix} = \begin{bmatrix} \frac{P_{v1} - P_2}{R_1 C_2} + \frac{P_{PEEP} - P_2}{R_{v2c} C_2} + \frac{P_0 - P_2}{(R_{v2a} + R_{ptc_{insp}}) C_2} \\ \frac{P_0 - P_c}{R_c C_c} \end{bmatrix} \quad (2.3)$$

onde

$$P_0 = \alpha_1(R_{v2a}, R_{v2b}, R_c)P_{PEEP} + \alpha_2(R_{v2a}, R_{v2b}, R_c)P_2 + \alpha_3(R_{v2a}, R_{v2b}, R_c)P_c,$$

$$P_{v1} = \beta_1(R_{v1a}, R_{v1b}, R_1)P_1 + \beta_2(R_{v1a}, R_{v1b}, R_1)P_2,$$

sendo $\alpha_1, \alpha_2, \alpha_3, \beta_1$ e β_2 funções das resistências do circuito, dadas por

¹ O índice temporal foi omitido para facilitar a visualização das equações. Considera-se que, pelo contexto, não será problema para o leitor distinguir quais variáveis são constantes e quais podem variar ao longo do tempo.

$$\alpha_1(Rv_{2a}, Rv_{2b}, R_c) = \frac{(Rv_{2a} + Rptc_{\text{insp}})R_c}{\varphi},$$

$$\alpha_2(Rv_{2a}, Rv_{2b}, R_c) = \frac{(Rv_{2b} + Rptc_{\text{exp}})R_c}{\varphi},$$

$$\alpha_3(Rv_{2a}, Rv_{2b}, R_c) = \frac{(Rv_{2a} + Rptc_{\text{insp}})(Rv_{2b} + Rptc_{\text{exp}})}{\varphi},$$

$$\beta_1(Rv_{1a}, Rv_{1b}, R_1) = \frac{Rv_{1b}R_1}{Rv_{1b}R_1 + Rv_{1a}R_1 + Rv_{1a}Rv_{1b}},$$

$$\beta_2(Rv_{1a}, Rv_{1b}, R_1) = \frac{Rv_{1b}Rv_{1a}}{Rv_{1b}R_1 + Rv_{1a}R_1 + Rv_{1a}Rv_{1b}},$$

com

$$\varphi = (Rv_{2b} + Rptc_{\text{exp}} + Rv_{2a} + Rptc_{\text{insp}})R_c + (Rv_{2a} + Rptc_{\text{insp}})(Rv_{2b} + Rptc_{\text{exp}}).$$

As resistências variáveis são controladas pela abertura das válvulas solenóides $x_{vk}(t)$, $k = 1, 2$:

$$Rv_{1a} = (1 - x_{v1})Rv_{1\text{max}} + x_{v1}Rv_{1\text{min}},$$

$$Rv_{1b} = x_{v1}Rv_{1\text{max}} + (1 - x_{v1})Rv_{1\text{min}},$$

$$Rv_{2a} = (1 - x_{v2})Rv_{2\text{max}} + x_{v2}Rv_{2\text{min}},$$

$$Rv_{2b} = Rv_{2c} = x_{v2}Rv_{2\text{max}} + (1 - x_{v2})Rv_{2\text{min}}.$$

Podemos considerar duas saídas de interesse: P_0 , medida pelo transdutor de pressão manométrica, e a vazão de ar q_0 no pneumotacógrafo inspiratório, que pode ser aproximada por

$$q_0 = \frac{\Delta P_{ptc}}{Rptc_{\text{insp}}}.$$

A variável controlada do sistema é P_0 no modo de ventilação controlado a pressão, e q_0 no modo controlado a volume.

3 Controle por Aprendizagem Iterativa

3.1 Estratégia de Controle por Aprendizagem Iterativa - Visão Geral

Controle por Aprendizagem Iterativa ou *Iterative Learning Control* (ILC) é uma técnica de controle desenvolvida para sistemas que operam repetidamente sob as mesmas condições iniciais. ILC se baseia na noção de que para tais processos é possível melhorar o desempenho utilizando dados de execuções (períodos, repetições, iterações) passadas. Esse aprendizado é feito com informações contidas nos sinais de erro e ações de controle das iterações anteriores. Com ILC é possível alcançar um desempenho satisfatório, com erro transiente mínimo, mesmo com um nível alto de incerteza no modelo e sob a presença de perturbações que se repetem a cada ciclo [3].

O objetivo de ILC é gerar um sinal de controle que rastreie uma referência específica ou rejeite uma perturbação periódica. ILC pode antecipar e compensar sinais externos através do aprendizado com a repetição. Por aprender com a prática, ILC tende a ser robusto a incertezas no sistema. Frequentemente se projeta ILC assumindo modelos lineares para sistemas não-lineares e se consegue bons resultados [3].

ILC é diferente de outras estratégias de controle baseadas em aprendizado, tais como controle adaptativo, redes neurais e controle repetitivo. Controle adaptativo modifica o controlador, que é um sistema, enquanto ILC modifica a ação de controle, que é um sinal. Do mesmo modo, o aprendizado por redes neurais envolve modificar os parâmetros da própria rede, que também é um sistema. O aprendizado por redes neurais exige uma grande quantidade de dados para treinamento, o que pode dificultar a convergência rápida, enquanto ILC pode convergir em poucas iterações [3]. Controle repetitivo engloba processos que se repetem com condições iniciais diferentes em cada repetição. Desse modo, ILC pode ser visto como um caso particular de controle repetitivo.

O algoritmo coleta amostras do erro e da ação de controle durante uma iteração inteira, para então calcular a ação de controle da próxima iteração [9]. O problema com esse tipo de abordagem é que o desempenho pode se tornar inaceitável durante os primeiros ciclos e o sistema pode ficar vulnerável a perturbações que não se repetem. Consequentemente, na maioria das implementações físicas, controle em malha fechada tradicional deve ser usado em conjunto com ILC [3].

Requisitos de projeto ILC incluem [5], [3], [9]:

1. Estabilidade BIBO (entrada limitada \Rightarrow saída limitada) durante cada iteração.

2. Convergência assintótica do erro de rastreamento para um valor mínimo a medida que $j \rightarrow \infty$.
3. Uma taxa de convergência (monotônica) aceitável para o erro de rastreamento.

3.2 ILC linear

3.2.1 Sistema Iterativo Linear e Invariante no Tempo

A estratégia ILC pode ser mais facilmente apresentada para sistemas em tempo discreto, tendo em vista a necessidade de armazenamento de dados passados, os quais são tipicamente amostrados. Consideremos então um sistema discreto linear e invariante no tempo (LIT) com uma entrada e uma saída (SISO)

$$y_j(k) = G(q)u_j(k) + d_j(k), \quad (3.1)$$

onde k é a variável de tempo discreto, j é o índice da iteração, $u_j(k)$ é a entrada de controle, $y_j(k)$ é a saída do sistema e $d_j(k)$ é um sinal externo. $d_j(k)$ pode englobar perturbações, condições iniciais não nulas e termos de sistemas com realimentação ou ação direta *feedforward*. q é o operador de avanço de tempo, definido como $qx(k) \equiv x(k+1)$. A planta $G(q)$ é uma função racional de q e podemos considerá-la BIBO estável. Caso contrário, é possível estabilizá-la com um controlador em malha fechada e então aplicar ILC ao sistema realimentado equivalente. A transformada-Z de um sistema $X(q)$ é obtida pela substituição de q por z .

Considerando sinais compostos por N amostras a cada iteração, e que o sistema tem grau relativo $m \geq 0$, as sequências de entrada e saída são

$$u_j(k), \quad k \in \{0, 1, \dots, N-1\},$$

$$y_j(k), \quad k \in \{m, m+1, \dots, N+m-1\},$$

$$d_j(k), \quad k \in \{m, m+1, \dots, N+m-1\},$$

já que uma entrada $u_j(k)$ leva um número de amostras igual ao grau relativo da planta para começar a afetar a saída. Do mesmo modo, definimos o sinal de referência a partir da amostra $k = m$ como

$$y_{\text{ref}}(k), \quad k \in \{m, m+1, \dots, N+m-1\}.$$

para que o sinal de erro seja

$$e_j(k) = y_{\text{ref}}(k) - y_j(k).$$

Vamos assumir que o sinal de distúrbio é periódico nas análises deste capítulo, de modo que $d_0(k) = d_1(k) = d_2(k) = \dots = d_j(k)$. Nesse caso, podemos omitir o índice da iteração e escrever apenas $d(k)$ na equação (3.1).

Exemplo. Considere um sistema SISO LIT representado no espaço de estados na forma

$$\mathbf{x}_j(k+1) = \mathbf{A}\mathbf{x}_j(k) + \mathbf{B}u_j(k),$$

$$y_j(k) = \mathbf{C}\mathbf{x}_j(k).$$

Podemos reescrevê-lo no formato de (3.1) com

$$y_j(k) = \underbrace{\mathbf{C}(q\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}}_{G(q)}u_j(k) + \underbrace{\mathbf{C}\mathbf{A}^k\mathbf{x}_j(0)}_{d_j(k)}.$$

Nesse caso, $d_j(k)$ é a resposta natural do sistema à condição inicial $\mathbf{x}_j(0)$. A premissa de que a perturbação é periódica implica em se ter a mesma condição inicial $\mathbf{x}(0)$ a cada iteração.

Um dos algoritmos de aprendizado ILC mais comumente utilizados é [3], [9], [10]:

$$u_{j+1}(k) = Q(q)[u_j(k) + L(q)e_j(k+m)]. \quad (3.2)$$

onde $Q(q)$ e $L(q)$ são funções racionais, representando respostas dinâmicas de filtros lineares, escolhidas como parte do procedimento de projeto ILC. $Q(q)$ é chamada de filtro-Q e $L(q)$ de função de aprendizado. A Figura 3.1 ilustra um sistema controlado por ILC. A cada iteração, o erro é filtrado por L , adicionado à ação de controle passada e filtrado novamente por Q . Essa nova ação de controle é aplicada à planta na próxima iteração.

Do ponto de vista de análise em ILC, é útil representar sinais e sistemas na forma "levantada"

$$\underbrace{\begin{bmatrix} y_j(m) \\ y_j(m+1) \\ \vdots \\ y_j(m+N-1) \end{bmatrix}}_{\mathbf{y}_j} = \underbrace{\begin{bmatrix} g_m & 0 & \dots & 0 \\ g_{m+1} & g_m & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{m+N-1} & g_{m+N-2} & \dots & g_m \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} u_j(0) \\ u_j(1) \\ \vdots \\ u_j(N-1) \end{bmatrix}}_{\mathbf{u}_j} + \underbrace{\begin{bmatrix} d(m) \\ d(m+1) \\ \vdots \\ d(m+N-1) \end{bmatrix}}_{\mathbf{d}}. \quad (3.3)$$

Essa representação é construída empilhando os sinais de entrada e saída em vetores

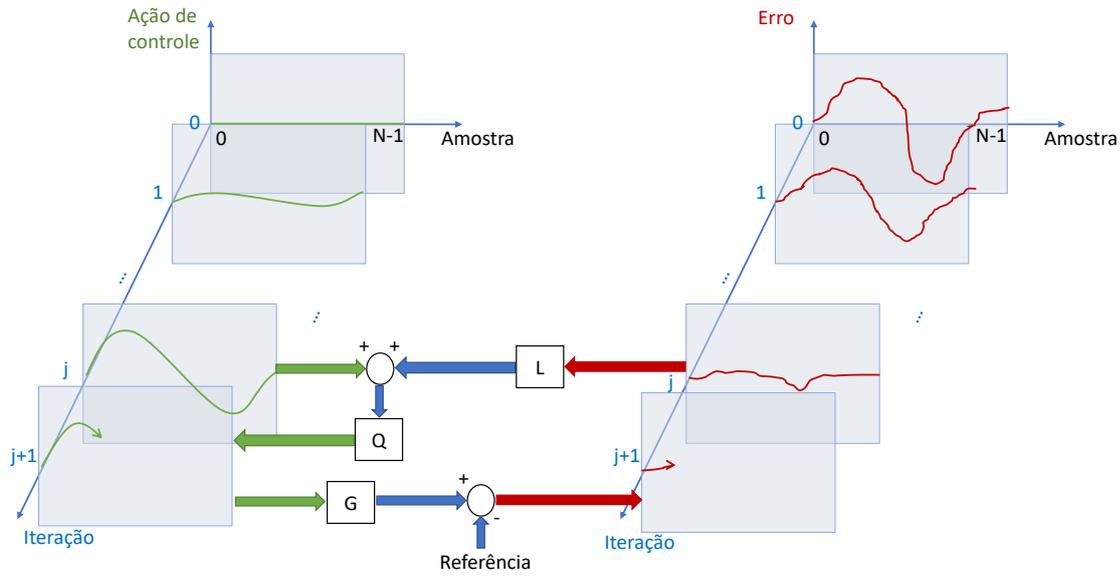


Figura 3.1 – Ilustração de um sistema controlado por ILC. Baseada em [3, Figura 1]

$$\mathbf{y}_j = [y_j(m) \ y_j(m+1) \ \dots \ y_j(m+N-1)]^T,$$

$$\mathbf{u}_j = [u_j(0) \ u_j(1) \ \dots \ u_j(N-1)]^T,$$

$$\mathbf{d} = [d(m) \ d(m+1) \ \dots \ d(m+N-1)]^T,$$

$$\mathbf{y}_{\text{ref}} = [y_{\text{ref}}(m) \ y_{\text{ref}}(m+1) \ \dots \ y_{\text{ref}}(m+N-1)]^T,$$

$$\mathbf{e}_j = [e_j(m) \ e_j(m+1) \ \dots \ e_j(m+N-1)]^T,$$

e tomando a resposta ao impulso do sistema como

$$G(q) = g_m q^{-m} + g_{(m+1)} q^{-(m+1)} + g_{(m+2)} q^{-(m+2)} + \dots,$$

onde os coeficientes g_k são os parâmetros de Markov da planta.

Como exemplo, caso a planta G possua grau relativo $m = 1$, a representação do sistema com resposta ao impulso

$$G(q) = g_1 q^{-1} + g_2 q^{-2} + g_3 q^{-3} + \dots$$

na forma levantada seria

$$\underbrace{\begin{bmatrix} y_j(1) \\ y_j(2) \\ \vdots \\ y_j(N) \end{bmatrix}}_{\mathbf{y}_j} = \underbrace{\begin{bmatrix} g_1 & 0 & \dots & 0 \\ g_2 & g_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_N & g_{N-1} & \dots & g_1 \end{bmatrix}}_{\mathbf{G}} \underbrace{\begin{bmatrix} u_j(0) \\ u_j(1) \\ \vdots \\ u_j(N-1) \end{bmatrix}}_{\mathbf{u}_j} + \underbrace{\begin{bmatrix} d(1) \\ d(2) \\ \vdots \\ d(N) \end{bmatrix}}_{\mathbf{d}},$$

e usáramos o algoritmo ILC

$$u_{j+1}(k) = Q(q)[u_j(k) + L(q)e_j(k+1)]. \quad (3.4)$$

ILC permite o projeto não-causal dos filtros usados em (3.2), no sentido de que podemos usar informações do erro "futuro" de iterações passadas. Desse modo, o filtro-Q e a função de aprendizado podem apresentar respostas ao impulso dadas por

$$Q(q) = \dots + \mathbf{q}_{-2}q^2 + \mathbf{q}_{-1}q^1 + \mathbf{q}_0 + \mathbf{q}_1q^{-1} + \mathbf{q}_2q^{-2} + \dots$$

e

$$L(q) = \dots + l_{-2}q^2 + l_{-1}q^1 + l_0 + l_1q^{-1} + l_2q^{-2} + \dots,$$

respectivamente. Na forma levantada, (3.2) se torna

$$\underbrace{\begin{bmatrix} u_{j+1}(0) \\ u_{j+1}(1) \\ \vdots \\ u_{j+1}(N-1) \end{bmatrix}}_{\mathbf{u}_{j+1}} = \underbrace{\begin{bmatrix} \mathbf{q}_0 & \mathbf{q}_{-1} & \dots & \mathbf{q}_{-(N-1)} \\ \mathbf{q}_1 & \mathbf{q}_0 & \dots & \mathbf{q}_{-(N-2)} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{q}_{N-1} & \mathbf{q}_{N-2} & \dots & \mathbf{q}_0 \end{bmatrix}}_{\mathbf{Q}} \left(\underbrace{\begin{bmatrix} u_j(0) \\ u_j(1) \\ \vdots \\ u_j(N-1) \end{bmatrix}}_{\mathbf{u}_j} \right) + \underbrace{\begin{bmatrix} l_0 & l_{-1} & \dots & l_{-(N-1)} \\ l_1 & l_0 & \dots & l_{-(N-2)} \\ \vdots & \vdots & \ddots & \vdots \\ l_{N-1} & l_{N-2} & \dots & l_0 \end{bmatrix}}_{\mathbf{L}} \underbrace{\begin{bmatrix} e_j(m) \\ e_j(m+1) \\ \vdots \\ e_j(m+N-1) \end{bmatrix}}_{\mathbf{e}_j} \right). \quad (3.5)$$

Se \mathbf{Q} e \mathbf{L} representam funções causais, então $q_{-1} = q_{-2} = \dots = q_{-(N-1)} = 0$ e $l_{-1} = l_{-2} = \dots = l_{-(N-1)} = 0$; como resultado, as matrizes são triangulares inferiores e

Toeplitz (possuem elementos idênticos nas diagonais principais e secundárias). As representações de (3.3) e (3.5) no domínio z são, respectivamente

$$Y_j(z) = G(z)U_j(z) + D(z)$$

e

$$U_{j+1}(z) = Q(z)[U_j(z) + z^m L(z)E(z)].$$

3.2.2 Análise de Estabilidade

Definição 1. O sistema ILC (3.1), (3.2) é *assintoticamente estável (AS)* se existe $\bar{u} \in \mathbb{R}$, $\bar{u} \geq 0$, tal que $|u_j(k)| \leq \bar{u}$ para todo $k \in \{0, 1, \dots, N-1\}$ e $j \in \{0, 1, \dots\}$, e, para todo $k \in \{0, 1, \dots, N-1\}$ existe $u_\infty(k) = \lim_{j \rightarrow \infty} u_j(k)$.

Substituindo $e_j = (\mathbf{y}_{\text{ref}} - \mathbf{y}_j)$ e (3.3) em (3.5), temos

$$\mathbf{u}_{j+1} = \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})\mathbf{u}_j + \mathbf{Q}\mathbf{L}(\mathbf{y}_{\text{ref}} - \mathbf{d}), \quad (3.6)$$

que representa a dinâmica em malha fechada no domínio j das iterações. Seja $\lambda_i(\mathbf{A})$ o i -ésimo autovalor de uma matriz \mathbf{A} , e $\rho(\mathbf{A}) = \max_i |\lambda_i(\mathbf{A})|$ o raio espectral de \mathbf{A} , então:

Teorema 1. (adaptado) [3], [9] *Seja o sistema composto por (3.1), (3.2) onde G é BIBO estável. Esse sistema é AS se, e somente se,*

$$\rho(\mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})) < 1.$$

Esse teorema pode ser interpretado como, para o domínio das iterações, análogo ao caso no domínio do tempo $x(k+1) = \mathbf{A}x(k)$, que é assintoticamente estável se, e somente se, todos os autovalores de \mathbf{A} tiverem magnitude menor que 1 [11]. Vale reforçar que a definição de estabilidade assintótica para ILC é, no entanto, diferente da definição para sistemas no domínio do tempo, que, nesse caso, é estabelecida para a resposta à entrada nula e requer que todo estado inicial convirja para $\mathbf{0}$.

Frequentemente em ILC se avalia o desempenho por meio da comparação do valor assintótico do erro com seu valor inicial para uma dada referência. O erro assintótico é definido como

$$\begin{aligned}
e_\infty(k) &= \lim_{j \rightarrow \infty} e_j(k), \\
&= \lim_{j \rightarrow \infty} (y_{\text{ref}}(k) - G(q)u_j(k) - d(k)), \\
&= y_{\text{ref}}(k) - G(q)u_\infty(k) - d(k).
\end{aligned}$$

A comparação pode ser feita considerando o erro quadrático médio (RMS) a cada iteração j , computado para o conjunto de amostras do vetor e_j . Se o sistema ILC é AS, então a ação de controle assintótica pode ser obtida a partir de (3.6) como

$$\begin{aligned}
\mathbf{u}_\infty &= \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})\mathbf{u}_\infty + \mathbf{Q}\mathbf{L}(\mathbf{y}_{\text{ref}} - \mathbf{d}) \\
\iff [\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})]\mathbf{u}_\infty &= \mathbf{Q}\mathbf{L}(\mathbf{y}_{\text{ref}} - \mathbf{d}) \\
\iff \mathbf{u}_\infty &= [\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})]^{-1}\mathbf{Q}\mathbf{L}(\mathbf{y}_{\text{ref}} - \mathbf{d}),
\end{aligned}$$

e o erro assintótico pode ser obtido na forma levantada como

$$\begin{aligned}
\mathbf{e}_\infty &= \mathbf{y}_{\text{ref}} - \mathbf{G}\mathbf{u}_\infty - \mathbf{d} \\
&= \mathbf{y}_{\text{ref}} - \mathbf{G} \{ [\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})]^{-1}\mathbf{Q}\mathbf{L}(\mathbf{y}_{\text{ref}} - \mathbf{d}) \} - \mathbf{d} \\
&= \{ \mathbf{I} - \mathbf{G}[\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})]^{-1}\mathbf{Q}\mathbf{L} \} (\mathbf{y}_{\text{ref}} - \mathbf{d}).
\end{aligned} \tag{3.7}$$

Muitos algoritmos ILC são projetados para convergir para um erro assintótico nulo, $e_\infty(k) = 0$, $\forall k \in \{m, m+1, \dots, N+m-1\}$, independente da referência ou perturbação periódica, ou seja, precisamos que a igualdade

$$\mathbf{G}[\mathbf{I} - \mathbf{Q}(\mathbf{I} - \mathbf{L}\mathbf{G})]^{-1}\mathbf{Q}\mathbf{L} = \mathbf{I} \tag{3.8}$$

seja satisfeita em (3.7). A seguinte condição é necessária e suficiente para isso:

Teorema 2. [3], [9] *Suponha que $G(q)$ e $L(q)$ não sejam identicamente nulos. Então para o sistema (3.1), (3.2) $e_\infty(k) = 0$ para todo k , para todo $y_{\text{ref}}(k)$ e $d(k)$ periódico, se, e somente se, o sistema é AS e $Q(q) = 1$.*

Demonstração. O erro assintótico (3.7) só existe se o sistema for AS. Como $Q(q) = 1$ é o equivalente a $\mathbf{Q} = \mathbf{I}$ na forma levantada, então, para provar a suficiência basta substituir $\mathbf{Q} = \mathbf{I}$

em (3.8):

$$\begin{aligned}
& G[I - (I - LG)]^{-1}L = I \\
& \iff G(LG)^{-1}L = I \\
& \iff [G(LG)^{-1}L]^{-1} = I^{-1} \\
& \iff L^{-1}LGG^{-1} = I \\
& \iff I = I.
\end{aligned}$$

Para provar a necessidade assumamos que exista $Q \neq I$ tal que (3.8) seja verdadeira. Então,

$$\begin{aligned}
& G[I - Q(I - LG)]^{-1}QL = I \\
& \iff [I - Q(I - LG)]^{-1} = G^{-1}L^{-1}Q^{-1} \\
& \iff I - Q(I - LG) = QLG \\
& \iff I - QLG = Q - QLG \\
& \iff I = Q,
\end{aligned}$$

de modo que $e_\infty = 0$ implica $Q = I$. □

O Teorema 2 afirma que $Q(q) = 1$ é necessário para que se consiga rastreamento perfeito com ILC. Apesar disso, utilizar um filtro- Q diferente de 1, projetado de forma apropriada, pode melhorar o comportamento transiente e a robustez. Transientes de aprendizado inadequados podem levar o sistema a altos níveis de erro entre as iterações antes de convergir. Assim sendo, a convergência monotônica entre o erro inicial e o erro assintótico é uma propriedade desejável.

Seja $\bar{\sigma}(\cdot)$ o valor singular máximo de uma matriz e $\|\cdot\|_2$ a norma euclidiana de um vetor. O seguinte teorema fornece as condições suficientes para convergência monotônica.

Teorema 3. [3] *Se a planta (3.3) com algoritmo (3.5) satisfaz*

$$\eta_1 := \bar{\sigma}(GQ(I - LG)G^{-1}) < 1, \quad (3.9)$$

então

$$\|e_\infty - e_{j+1}\|_2 < \eta_1 \|e_\infty - e_j\|_2, \quad \forall j = 1, 2, \dots$$

A prova formal pode ser encontrada em [12]. O resultado do Teorema 3 pode ser visto de maneira mais intuitiva na equação que governa a convergência do erro

$$\mathbf{e}_\infty - \mathbf{e}_{j+1} = \mathbf{GQ}(\mathbf{I} - \mathbf{LG})\mathbf{G}^{-1}(\mathbf{e}_\infty - \mathbf{e}_{j+1})$$

obtida a partir da manipulação de (3.3), (3.5) e (3.7). η_1 é um limite superior para o transiente de convergência do erro. Quanto mais próximo de zero estiver η_1 , mais rápida tende a ser a taxa de convergência.

Entretanto, a matriz em (3.9), cujo valor singular máximo precisa ser computado, é numericamente mal condicionada. Como alternativa, se $Q(z)$ e $L(z)$ são funções causais, então, no domínio- z , a equação da convergência do erro tem a forma

$$[E_\infty(z) - E_{j+1}(z)] = Q(z)[1 - z^m L(z)G(z)] [E_\infty(z) - E_j(z)]$$

e um limite para o transiente de convergência do erro pode ser obtido no domínio da frequência por [3]

$$\eta_2 := \|Q(z)[1 - z^m L(z)G(z)]\|_\infty, \quad (3.10)$$

com

$$\|\mathbf{e}_\infty - \mathbf{e}_{j+1}\|_2 < \eta_2 \|\mathbf{e}_\infty - \mathbf{e}_j\|_2, \quad \forall j = 1, 2, \dots,$$

onde, para um dado sistema SISO discreto $T(z)$, $\|T(z)\|_\infty$ é definida como $\sup_{\theta \in (-\pi, \pi]} |T(e^{i\theta})|$, sendo $T(e^{i\theta})$ a resposta em frequência do sistema na frequência θ . A prova para esse teorema é similar à prova do Teorema 3.1 em [9] e também pode ser encontrada em [13].

Quando um controlador $C(q)$ é usado em malha fechada em conjunto com ILC, como no esquema mostrado na Figura 3.2, a planta equivalente vista pelo algoritmo é [3]

$$y_j(k) = \underbrace{\frac{G(q)}{1 + G(q)C(q)}}_{G_{eq}(q)} u_j(k) + \underbrace{\frac{G(q)C(q)}{1 + G(q)C(q)}}_{d(k)} y_{\text{ref}}(k), \quad (3.11)$$

e todos os resultados anteriores permanecem válidos para o sistema (3.11) com algoritmo ILC (3.2).

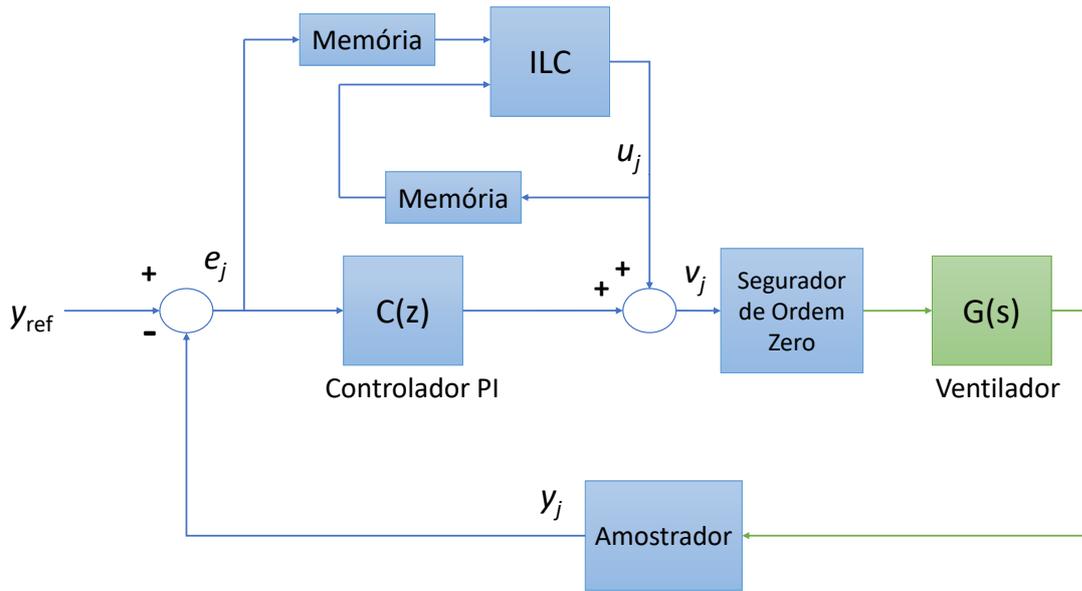


Figura 3.2 – Arquitetura de controle com ILC e controle em malha fechada tradicional.

3.2.3 Robustez

A incerteza na modelagem do processo é um fator que está implícito na formulação de ILC. Se o processo fosse conhecido exatamente, outras estratégias de controle mais diretas poderiam ser utilizadas. Podemos considerar a questão da robustez em ILC em relação à estabilidade assintótica e à monotonicidade da convergência do erro.

Uma questão chave em ILC é se um sistema assintoticamente estável permanece AS sob perturbações na planta. Consideremos um cenário com $L(q)$ causal, e $Q(q) = 1$ para o erro de rastreamento assintótico ser nulo. A condição de estabilidade assintótica colocada no Teorema 1, nesse caso é equivalente a [3]

$$|1 - l_0 g_m| < 1, \quad (3.12)$$

sendo que l_0 e g_m correspondem aos parâmetros de Markov associados às respostas ao impulso de $L(q)$ e $G(q)$, como mostrado em (3.3) e (3.5). Como consequência, esse sistema ILC é AS se, e somente se,

$$\text{sgn}(g_m) = \text{sgn}(l_0) \quad \text{e} \quad l_0 g_m < 2.$$

Esse resultado mostra que ILC pode alcançar erro assintótico nulo com apenas o conhecimento do sinal de g_m e de um limite superior para $|g_m|$. Já que para qualquer valor arbitrariamente grande de $|g_m|$ podemos escolher $|l_0|$ suficientemente pequeno, a estabilidade assintótica de sistemas ILC pode ser robusta a todo tipo de perturbação que não mude o sinal de g_m .

Estabilidade robusta, entretanto, não implica em transientes de aprendizado aceitáveis. Tomemos uma planta incerta

$$G(q) = \hat{G}(q)[1 + W(q)\Delta(q)] \quad (3.13)$$

onde $\hat{G}(q)$ é o modelo nominal da planta, $W(q)$ é conhecido e estável e $\Delta(q)$ é desconhecido e estável com $\|\Delta(z)\|_\infty < 1$. Nesse caso, o seguinte teorema fornece condições para monotonicidade robusta.

Teorema 4. (adaptado de [3]) *Se*

$$|W(e^{i\theta})| < \frac{1 - |Q(e^{i\theta})||1 - e^{i\theta m}L(e^{i\theta})\hat{G}(e^{i\theta})|}{|Q(e^{i\theta})||L(e^{i\theta})\hat{G}(e^{i\theta})|} \quad (3.14)$$

para todo $\theta \in (-\pi, \pi]$, então o sistema ILC (3.1), (3.2), (3.13) é monotonicamente convergente.

Demonstração. De (3.10), o sistema é monotonicamente convergente se

$$\begin{aligned} \eta_2 &= \left\| Q(z)[1 - z^m L(z)G(z)] \right\|_\infty < 1, \\ &= \left\| Q(z) \left[1 - z^m L(z)\hat{G}(z)[1 + W(z)\Delta(z)] \right] \right\|_\infty < 1, \text{ de (3.13)} \\ &= \sup_{\theta} \left| Q(e^{i\theta}) \left[1 - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta}) - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta})W(e^{i\theta})\Delta(e^{i\theta}) \right] \right| < 1, \text{ por definição} \\ &= \sup_{\theta} \left| Q(e^{i\theta}) \left[1 - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta}) \right] - Q(e^{i\theta})e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta})W(e^{i\theta})\Delta(e^{i\theta}) \right| < 1, \\ &\leq \sup_{\theta} \left\{ \left| Q(e^{i\theta}) \left[1 - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta}) \right] \right| + \left| Q(e^{i\theta})e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta})W(e^{i\theta})\Delta(e^{i\theta}) \right| \right\}, \\ &\leq \sup_{\theta} \left\{ \left| Q(e^{i\theta}) \left[1 - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta}) \right] \right| + \left| Q(e^{i\theta})L(e^{i\theta})\hat{G}(e^{i\theta})W(e^{i\theta}) \right| \right\}, \text{ usando } |\Delta(e^{i\theta})| < 1 \\ &\text{ e } |e^{i\theta}| = 1, \quad \forall \theta \in (-\pi, \pi]. \end{aligned}$$

Portanto, se $\left| Q(e^{i\theta}) \left[1 - e^{i\theta m} L(e^{i\theta})\hat{G}(e^{i\theta}) \right] \right| + \left| Q(e^{i\theta})L(e^{i\theta})\hat{G}(e^{i\theta})W(e^{i\theta}) \right| < 1, \quad \forall \theta \in (-\pi, \pi]$

$$\iff |W(e^{i\theta})| < \frac{1 - |Q(e^{i\theta})||1 - e^{i\theta m}L(e^{i\theta})\hat{G}(e^{i\theta})|}{|Q(e^{i\theta})||e^{i\theta}L(e^{i\theta})\hat{G}(e^{i\theta})|}, \quad \forall \theta \in (-\pi, \pi],$$

então a condição (3.10) será satisfeita.

□

Diferente da condição para estabilidade robusta (3.12), que depende apenas dos primeiros parâmetros de Markov, a condição (3.14) depende de $G(q)$, $Q(q)$ e $L(q)$. Examinando (3.14), o modo mais direto de se aumentar a robustez em dada frequência θ é diminuindo o ganho do filtro-Q $|Q(e^{j\theta})|$. No entanto, quanto menor for o ganho do filtro-Q, maior será o erro assintótico de rastreamento nessa faixa de frequências. Temos, portanto, um compromisso entre robustez a incertezas e a diminuição do erro de rastreamento, na escolha de Q [3].

Podemos considerar ainda na análise de robustez os efeitos de ruídos, perturbações não-periódicas e mudança nas condições iniciais entre iterações. Todos esses efeitos podem ser considerados através da adição a (3.1) de um termo $d_j(k)$ que não é constante entre iterações. Nesse caso, o erro não convergirá para o valor assintótico e_∞ . Apesar disso, caso $d_j(k)$ seja limitado, o erro convergirá para uma vizinhança esférica, de raio limitado, em torno de e_∞ . Idealmente, não queremos que nosso algoritmo ILC tente aprender a partir de $d_j(k)$. Se $d_j(k)$ for um processo estocástico estacionário, a taxa de aprendizagem deve ser rápida o suficiente para minimizar a densidade espectral do erro. Ou seja, devemos projetar $Q(q)$ e $L(q)$ para que η_2 em (3.10) seja mínimo na faixa de frequências onde a densidade espectral de $d_j(k)$ é alta. Novamente, a melhor maneira de aumentar a robustez envolve diminuir o ganho do filtro-Q. Nesse caso há um compromisso também entre desempenho nominal e robusto [3].

Uma abordagem que também é possível para lidar com ruídos e perturbações não-periódicas é usar valores médios de algumas iterações para atualizar a ação de controle em (3.2) [3].

3.2.4 Projeto

Para sistemas com grau relativo $m = 1$, a forma mais amplamente utilizada de ILC é o algoritmo com função de aprendizado do tipo PD, escrito como

$$u_{j+1}(k) = u_j(k) + K_p e_j(k+1) + K_d [e_j(k+1) - e_j(k)], \quad (3.15)$$

que é um caso específico de (3.4), com $Q(q) = 1$ e $L(q) = K_p + K_d \frac{q-1}{q}$.

Não há, para ILC, regras semelhantes às de Ziegler-Nichols para controladores PID [3]. Apesar de existirem outros métodos de projeto de $L(q)$, como os baseados em otimização, inversão da planta e \mathcal{H}_∞ para ILC, o mais popular é usar o algoritmo (3.15) com os ganhos K_p e K_d sintonizados por tentativa e erro.

O método de sintonia de (3.15) sugerido e utilizado por [3] consiste em adicionar um filtro-Q passa-baixas e definir, inicialmente, os ganhos de aprendizado e banda de passagem do filtro suficientemente pequenos para que um comportamento transiente estável para o erro seja obtido. Para isso, roda-se o algoritmo ILC por um número razoável de iterações. Em seguida,

umenta-se os ganhos e banda de passagem do filtro, cada um a sua vez, até que uma combinação com bom desempenho seja encontrada. Essa abordagem tem como vantagens a simplicidade e a possibilidade de aplicação em sistemas quando não há um conhecimento profundo do modelo do processo.

Assumindo a disponibilidade de um modelo linear $G(z)$ para a planta, sugerimos como método de projeto:

1. Definir $Q(q) = 1$ para que o erro de rastreamento assintótico seja igual a zero.
- 2a. Escolher a função de aprendizado simplesmente como $L(q) = K$ e utilizar o ganho K que minimiza η_2 em (3.10) por algum método de otimização e para o qual a condição de estabilidade assintótica seja satisfeita.
- 2b. Se $G(z)$ possuir um polo integrador, usar $L(q) = K_d \frac{q-1}{q}$ e adotar o valor de K_d que minimiza η_2 e para o qual a condição de estabilidade assintótica seja satisfeita.
3. Caso $\eta_{2\min} > 1$, projetar um filtro-Q para o qual $|Q(e^{i\theta}) [1 - L(e^{i\theta})G(e^{i\theta})]| < 1$.

Esse método é apenas uma diretriz para sintonizar um algoritmo ILC simples, de maneira que o conhecimento do processo possa ser incorporado de forma benéfica ao projeto. A ideia surgiu a partir da dificuldade de escolher os ganhos da forma feita em [3]. O passo 1 surge do Teorema 2. A concepção dos passos 2a, 2b e 3 poderá ser visualizada mais facilmente no próximo capítulo. Embora tenha sido mencionado na seção 3.2.3 que usar $Q(q) \neq 1$ é importante para um projeto ILC robusto, isso depende do tipo de referência que se deseja rastrear. A técnica de ILC é intrinsecamente robusta, mesmo com $Q(q) = 1$, e o desempenho pode piorar significativamente com o uso de um filtro-Q passa-baixas para sinais de referência com conteúdo espectral em altas frequências, como o degrau.

3.2.5 Formulação Alternativa

Em [9], o algoritmo ILC é proposto na forma

$$u_{j+1} = T_u u_j + T_e (y_{\text{ref}} - y_j),$$

que é equivalente a (3.2). Em [9], o problema é formulado considerando, a princípio, sistemas MIMO e que podem estar no tempo contínuo ou discreto. É desenvolvido um teorema para estabilidade assintótica equivalente ao Teorema 1 e são derivadas equações para ação de controle e erro assintóticos. A equação do erro assintótico é equivalente a (3.7). É afirmado que caso a planta seja conhecida, o melhor controlador ILC não consegue desempenho melhor do que o controlador representado por

$$u^* = T_n^* y_{\text{ref}},$$

onde T_n^* é a melhor aproximação para o modelo inverso do sistema e é definido como a solução do problema

$$T_n^* = \min_{T_n} \|(I - GT_n)y_{\text{ref}}\|.$$

Ao considerar sistemas SISO discretos onde a planta não é bem conhecida, em [9] propõe-se a lei de controle $u^* = Ay_{\text{ref}}$, onde o problema está em encontrar

$$A = H^{-1} = \begin{bmatrix} a_1 & 0 & \dots & 0 \\ a_2 & a_1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a_N & a_{N-1} & \dots & a_1 \end{bmatrix}, \quad (3.16)$$

em que $H(z)$ possui grau relativo $m = 1$, sendo definida como

$$H(z) = z^{m-1}G(z).$$

Os coeficientes de A são calculados após uma iteração e são computados recursivamente por

$$a_i = \frac{1}{y(1)}(u(i) - \sum_{j=1}^{i-1} a_j y(i-j+1)). \quad (3.17)$$

Quando o sistema é de fase não mínima, em [9] usa-se uma abordagem com amostragem multi-taxa para obter uma inversa estável para o sistema. Além disso, em [9] afirma-se que, para problemas envolvendo sistemas LIT, não há a necessidade de se realizar operações repetitivas, já que com uma iteração é possível obter uma lei de controle ótima, de forma que o benefício real de ILC está na sua aplicação na sistemas não-lineares e/ou variantes no tempo.

O método composto por (3.16) e (3.17) faz parte da classe de algoritmos ILC baseados em inversão da planta. Esse tipo de estratégia possui as vantagens de convergência em uma iteração, com erro de rastreamento nulo. Contudo, além das dificuldades decorrentes de se inverter plantas possivelmente de fase não-mínima, os algoritmos baseados em inversão da planta são pouco robustos a incertezas, perturbações e qualquer tipo de mudança no sistema. Para esses casos, geralmente apresentam transientes de aprendizado ruins e não garantem convergência monotônica [3].

3.3 Breve revisão sobre ILC não-linear

O objetivo desta seção é apenas apontar, a título de revisão do estado da arte, alguns trabalhos que utilizam ILC para controlar sistemas não-lineares. Algoritmos ILC não-lineares calculam a ação de controle da próxima iteração, com base em uma função não-linear dos dados coletados do erro de rastreamento e da ação de controle da iteração corrente.

Em [14], propõe-se um método ILC baseado em realimentação de estados que garante rastreamento assintótico de um sinal de referência para a saída. Essa estratégia é restrita a uma classe de sistemas SISO incertos, não-lineares, mas parcialmente linearizáveis por realimentação de estados. [15] propõe um método ILC com resultados semelhantes para uma classe mais geral de sistemas.

Em [16], desenvolve-se uma técnica ILC de contorno adaptativo para uma viga de Euler-Bernoulli (sistema modelado a parâmetros distribuídos) sob perturbações não-periódicas distribuídas e nos contornos levando em conta uma restrição de saturação na entrada de controle.

Em [17], propõe-se o algoritmo ILC não-linear monotonicamente convergente (NMC-ILC). A estratégia, baseada em otimização, lineariza um processo não-linear e garante convergência monotônica através da minimização de um limite superior sobre o erro de rastreamento. Esse limite considera efeitos que surgem de incertezas paramétricas e estruturais. A título de exemplo de um algoritmo ILC não-linear, considere um processo repetitivo descrito pelo seguinte modelo não-linear MIMO no espaço de estados¹:

$$x_j(k+1) = f_{\theta_1}(x_j(k), u_j(k)) \quad (3.18)$$

$$y_j(k) = g_{\theta_2}(x_j(k)).$$

onde as funções não-lineares f e g são parametrizadas por constantes desconhecidas θ_1 e θ_2 . O algoritmo NMC-ILC consiste em resolver o problema de otimização

$$\min_{\mathbf{u}_j} \|\mathbf{e}_{pj}\|_Q + k_f \|\mathbf{u}_j - \mathbf{u}_{(j-1)}\|_Q + k_h \|\mathbf{u}_j - \mathbf{u}_{(j-1)}\|_Q^2, \quad (3.19)$$

com

$$\mathbf{e}_{pj} = \mathbf{y}_{ref} - \mathbf{y}_{jp}$$

onde \mathbf{y}_{jp} é a saída predita pela aproximação linear (baseada em expansão por séries de Taylor) do processo, que é atualizada iterativamente para capturar seu comportamento não-linear local em torno de $\mathbf{u}_{(j-1)}$. k_f e k_h são parâmetros estimados por algoritmos randomizados, que levam

¹ Embora vetores e matrizes tenham sido mostrados em negrito neste capítulo, aqui, especificamente, estão em negrito apenas os supervetores, com todas as amostras de uma iteração empilhadas em cada vetor.

em conta os erros paramétricos e estruturais decorridos da linearização do modelo. A norma de uma matriz é definida como

$$\|M\|_Q = \sqrt{\lambda_{\max}(M^T Q M)}.$$

onde a matriz definida positiva Q em (3.19) tem a função de atribuir pesos distintos às diferentes componentes de e_{jp} .

Caso o sistema descrito por (3.18) satisfaça as seguintes suposições:

1. f e g são funções continuamente diferenciáveis de segunda ordem em relação a x , u , θ_1 , θ_2 .
2. O raio espectral $\rho([\delta f / \delta x]_{(x,u)}) < 1$ para todo par $(x, u) \in \Omega$, onde Ω é um domínio de operação compacto.
3. A condição inicial $x_j(0)$ é a mesma em cada iteração.

Então o domínio factível do problema de otimização (3.19) é não-vazio para $j \in [2, \infty)$, e o erro de rastreamento é limitado e satisfaz $\|e_j\|_Q \leq \|e_{(j-1)}\|_Q$, para $j \in [2, \infty)$.

Nosso sistema (2.3) é não-linear e, portanto, existe um grande número de algoritmos ILC que poderíamos explorar para controlá-lo. Para sistemas não-lineares, ILC não-linear pode oferecer algumas vantagens em relação a (3.2), como maior faixa de operação, robustez ou garantia de convergência. No entanto, algoritmos não-lineares são projetados, em geral, para classes específicas de sistemas, baseados em suposições que muitas vezes na prática podem ser restritivas ou até difíceis de verificar. Além disso, um fator que dificulta o uso de ILC não-linear é a complexidade matemática associada com as provas de estabilidade e convergência para essa classe de leis de controle.

Contudo, como será mostrado no próximo capítulo, o modelo do nosso ventilador é linearizável por realimentação. Isso nos permite considerar o sistema como linear, e aplicar o algoritmo (3.2), com um projeto que possivelmente garante estabilidade assintótica e convergência monotônica com base nos Teoremas 1 e 3. Essa estratégia tem como vantagens a simplicidade e as propriedades de robustez discutidas na seção 3.2.3. A robustez de ILC para sistemas lineares é bem conhecida e uma discussão mais detalhada pode ser encontrada em [5] e [10].

4 Estratégia de Controle por Aprendizagem Iterativa aplicada ao ventilador

4.1 Simplificação do sistema

Já que queremos controlar o sistema ativamente apenas durante a inspiração, podemos considerar um circuito mais simples. A válvula $S2$ está aberta durante toda a inspiração e, portanto, os caminhos que contêm Rv_{2b} e Rv_{2c} podem ser considerados como circuitos abertos para fins práticos.

É possível controlar uma válvula solenóide com PWM (modulação por largura de pulso), de modo que, com uma frequência de chaveamento rápida o suficiente, a tensão média (ou a pressão equivalente) pode ser representada como um sinal continuamente variável [2]. Com base nisso, adotaremos a hipótese simplificadora de que a válvula $S_1(t)$ é continuamente variável.

Além disso, vamos assumir que $Rv_{1a}(t)$ e $Rv_{1b}(t)$ podem ser substituídas por $Rv_1(t)$, dada por

$$Rv_1(t) = (1 - x_{v1}(t))Rv_{1\max} + x_{v1}(t)Rv_{1\min}.$$

Com essas hipóteses, podemos simplificar o circuito inspiratório do ventilador para o mostrado na Figura 4.1.

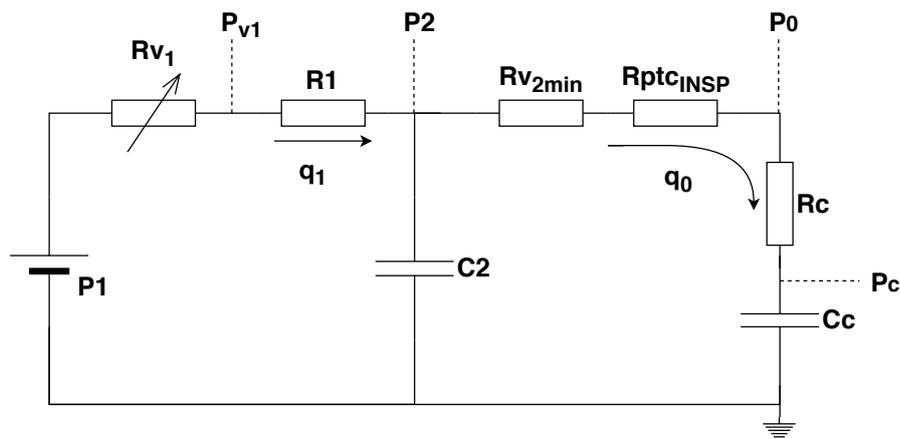


Figura 4.1 – Circuito inspiratório simplificado do ventilador.

Considerando $x_1 = P_2$, $x_2 = P_c$ e $u = Rv_1$, a dinâmica (2.3) do modelo original pode ser simplificada para

$$\begin{bmatrix} \frac{dx_1(t)}{dt} \\ \frac{dx_2(t)}{dt} \end{bmatrix} = \begin{bmatrix} \frac{P_1 - x_1(t)}{C_2(u(t) + R_1)} + \frac{x_2(t) - x_1(t)}{C_2(Rv_{2\min} + R_{ptc_{\text{insp}}} + R_c)} \\ \frac{x_1(t) - x_2(t)}{C_2(Rv_{2\min} + R_{ptc_{\text{insp}}} + R_c)} \end{bmatrix}$$

4.2 Linearização por realimentação

Se assumimos ainda que conseguimos medir ou observar $x_1(t)$, então é possível linearizar a dinâmica acima com uma entrada virtual de controle v , igual à corrente que passa por R_1 :

$$v(t) = q_1(t) = \frac{P_1 - x_1(t)}{u(t) + R_1} \Rightarrow u(t) = \frac{P_1 - x_1(t)}{v(t)} - R_1. \quad (4.1)$$

Ao fazermos isso, no entanto, teremos uma singularidade quando $v = 0$, que corresponderia a uma resistência pneumática infinita. Evitamos esse problema através de um rearranjo, no qual a entrada de controle passa a ser, de fato, a condutância

$$g_{v1}(t) = \frac{1}{u(t) + R_1} = \hat{u}(t) = \frac{v(t)}{P_1 - x_1(t)},$$

com $g_{v1} \in \left[\frac{1}{Rv_{1\max} + R_1}, \frac{1}{Rv_{1\min} + R_1} \right]$. A Figura 4.2 representa o circuito inspiratório linearizado com a fonte de corrente virtual v .

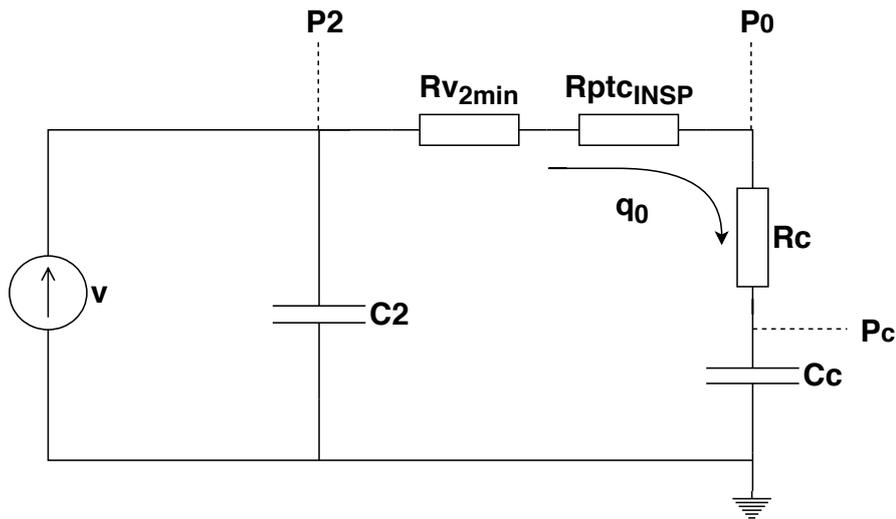


Figura 4.2 – Circuito inspiratório linearizado do ventilador.

Incorporando o tempo de resposta da válvula como um atraso após o comando de $v(t)$, a dinâmica do sistema linearizado se torna

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \frac{1}{R_{2c}} \begin{bmatrix} -1/C_2 & 1/C_2 \\ 1/C_c & -1/C_c \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 1/C_2 \\ 0 \end{bmatrix} v(t - 0,45T_{\text{resp}}) \quad (4.2)$$

$$\begin{bmatrix} q_0(t) \\ p_0(t) \end{bmatrix} = \frac{1}{R_{2c}} \begin{bmatrix} 1 & -1 \\ R_c & Rv_{2\text{min}} + Rptc_{\text{insp}} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix},$$

e, a partir de (4.2), podemos obter

$$G_1(s) = \frac{Q_0(s)}{V(s)} = \frac{C_c}{C_c C_2 R_{2c} s + C_c + C_2} e^{-0,45T_{\text{resp}} s} \quad (4.3a)$$

e

$$G_2(s) = \frac{P_0(s)}{V(s)} = \frac{C_c R_c s + 1}{C_c C_2 R_{2c} s^2 + (C_c + C_2) s} e^{-0,45T_{\text{resp}} s}, \quad (4.3b)$$

com $R_{2c} = Rv_{2\text{min}} + Rptc_{\text{insp}} + R_c$. As funções de transferência $G_1(s)$ e $G_2(s)$ representam, respectivamente, a dinâmica do sistema no modo controlado a volume, e a pressão.

4.2.1 Validação

Para investigar o desempenho ILC, consideramos que as medições do sistema são amostradas e processadas por um computador digital. Esse processo produz uma ação de controle que é mantida constante durante cada intervalo de amostragem por um segurador de ordem zero.

As funções de transferências obtidas a partir da discretização de (4.3a) e (4.3b) com taxa de amostragem $F_s = 500$ Hz são, respectivamente,

$$\bar{G}_1(z) = z^{-1} \frac{3,6418 \times 10^{-3}(z + 0,7986)}{z - 0,9528} \quad (4.4a)$$

e

$$\bar{G}_2(z) = z^{-1} \frac{1,3998 \times 10^{-3}(z - 0,9317)(z + 0,9274)}{(z - 1)(z - 0,9528)}. \quad (4.4b)$$

A Figura 4.3 compara as saídas P_0 e q_0 do sistema (2.3) linearizado por realimentação com as respostas de (4.4a) e (4.4b) à entrada $v(t)$. Para simular (2.3) linearizado, os comandos das válvulas S1 e S2 foram dados por

$$S_1(t) = \frac{Rv_{1\text{max}} - u(t)}{Rv_{1\text{max}} - Rv_{1\text{min}}}$$

e

$$S_2(t) = 1,$$

onde $u(t)$ foi obtida a partir de $v(t)$ em (4.1).

Na Figura 4.3, o primeiro degrau em $v(t)$ excita os sistemas dentro do limite de operação de S1, o segundo degrau é suficiente para saturar a válvula, e o terceiro é aplicado com a válvula S1 já totalmente aberta. Se S1 não estiver saturada, podemos considerar $G_1(z)$ e $G_2(z)$ representações suficientemente boas do sistema original linearizado para usá-las durante o procedimento de projeto do algoritmo ILC.

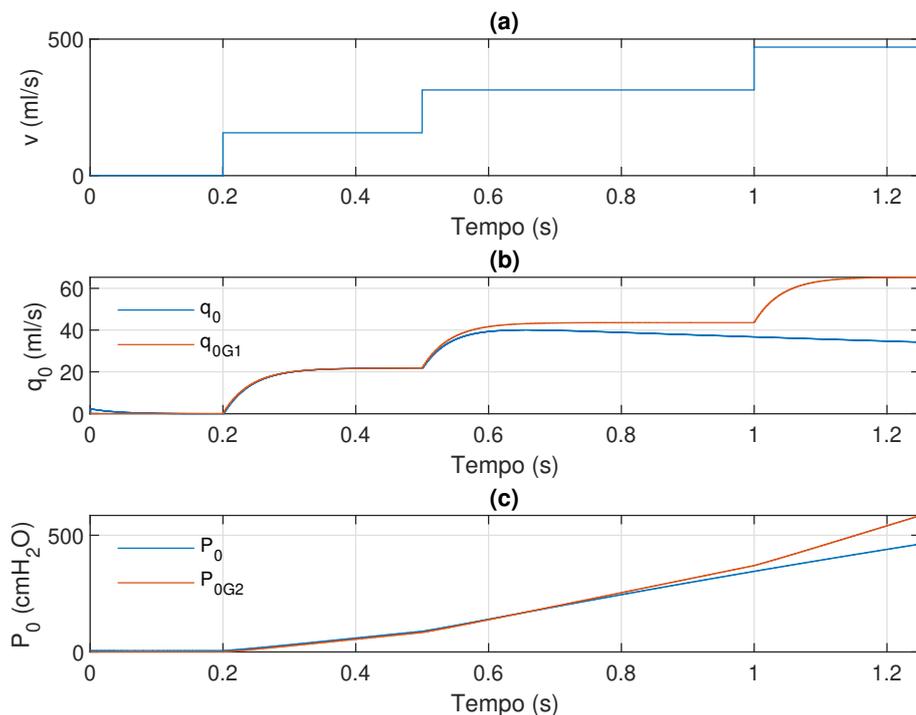


Figura 4.3 – Comparação das respostas das funções de transferências $G_1(z)$ (b) e $G_2(z)$ (c) com o sistema original com linearização por realimentação para a entrada $v(t)$ (a).

4.2.2 Presença de incertezas

A desvantagem de se usar linearização por realimentação é o fato dessa estratégia depender do conhecimento exato do modelo de atuação. Do ponto de vista do sistema real, não é razoável esperar que vamos conhecer exatamente a pressão P_1 , fornecida pelo compressor, ou a resistência pneumática R_1 (que possivelmente varia com o tempo e não representa uma relação linear entre a pressão e a vazão de ar). O que teríamos na realidade seriam apenas estimativas desses parâmetros dadas por

$$\hat{P}_1 = P_1 + \Delta P_1,$$

e

$$\hat{R}_1 = R_1 + \Delta R_1,$$

onde Δx é o erro na estimativa do parâmetro x . Neste caso, a entrada de realimentação linearizante seria

$$v(t - 0,45T_{\text{resp}}) = \frac{\hat{P}_1 - x_1(t)}{u(t) + \hat{R}_1} \Rightarrow u(t) = \frac{\hat{P}_1 - x_1(t)}{v(t - 0,45T_{\text{resp}})} - \hat{R}_1, \quad (4.6)$$

e com essa entrada não cancelaríamos completamente as não-linearidades do sistema. Consequentemente, a dinâmica no espaço de estados ficaria¹

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \frac{1}{R_{2c}} \begin{bmatrix} -1/C_2 & 1/C_2 \\ 1/C_c & -1/C_c \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \frac{1}{C_2} \begin{bmatrix} (1 + \Delta_v(x_1, v)) \\ 0 \end{bmatrix} v(t - 0,45T_{\text{resp}}) \quad (4.7)$$

$$\begin{bmatrix} q_0 \\ p_0 \end{bmatrix} = \frac{1}{R_{2c}} \begin{bmatrix} 1 & -1 \\ R_c & Rv_{2\text{min}} + Rptc_{\text{insp}} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

em que

$$\Delta_v(x_1(t), v(t - 0,45T_{\text{resp}})) = \frac{\Delta P_1 + \Delta R_1 v(t - 0,45T_{\text{resp}})}{\hat{P}_1 - x_1(t) - \Delta R_1 v(t - 0,45T_{\text{resp}})}. \quad (4.8)$$

O fato de Δ_v depender tanto de x_1 quanto de v , faz com que esse termo seja não-linear e que seu valor varie com as condições de operação do sistema. A partir de (4.8), é possível afirmar que, quanto mais próximas as estimativas estiverem dos parâmetros reais, mais próximo a dinâmica (4.7) estará de ser linear. Se $\hat{P}_1 = P_1$ e $\hat{R}_1 = R_1$ então $\Delta_v = 0$ e (4.7) e (4.2) são lineares e equivalentes.

As alterações que as incertezas paramétricas provocam ao atuador virtual do sistema linearizado podem afetar os parâmetros de Markov e, conseqüentemente, as propriedades de estabilidade assintótica e monotonicidade na convergência do erro. É possível estimar o quão distante (4.7) pode estar do modelo linear (4.2) encontrando um limite superior para Δ_v . Substituindo (4.6) em (4.8), temos

¹ Exceto em v , o índice temporal foi omitido para que a equação possa ser melhor visualizada.

$$\begin{aligned}\Delta_v &= \frac{\Delta P_1 + \Delta R_1 \left(\frac{\hat{P}_1 - x_1(t)}{u(t) + \hat{R}_1} \right)}{\hat{P}_1 - x_1(t) - \Delta R_1 \left(\frac{\hat{P}_1 - x_1(t)}{u(t) + \hat{R}_1} \right)} \\ &= \frac{\Delta P_1}{\hat{P}_1 - x_1(t)} \left(1 + \frac{\Delta R_1}{u(t) + R_1} \right) + \frac{\Delta R_1}{u(t) + R_1}.\end{aligned}$$

Se consideramos que a pressão $x_1(t)$ no reservatório C2 é uma fração de P_1 , então $|\Delta_v|$ é máximo para $x_1(t) \cong P_1$ e $u(t) = Rv_{1\min} \approx 0$ e, nesse caso, esperamos ter

$$|\Delta_v| \leq \left(1 + 2 \frac{\Delta R_1}{R_1} \right).$$

Supondo que possamos assegurar que o erro na estimativa de R_1 seja menor que 100%, ou $\Delta R_1 < 1,0R_1$, então teríamos, no pior caso,

$$|\Delta_v|_{\max} = 3.$$

Como

$$\begin{aligned}G_1(s) &= \frac{C_c(1 + \Delta_v)}{C_c C_2 R_{2c} s + C_c + C_2} e^{-0,45T_{\text{resp}}s} = \hat{G}_1(1 + \Delta_v) \quad e \\ G_2(s) &= \frac{(C_c R_c s + 1)(1 + \Delta_v)}{C_c C_2 R_{2c} s^2 + (C_c + C_2)s} e^{-0,45T_{\text{resp}}s} = \hat{G}_2(1 + \Delta_v),\end{aligned}$$

se projetarmos $L(q)$ e $Q(q)$ para que o Teorema 4 seja satisfeito para $|W(q)| = 3$, garantiremos que nosso sistema ILC é monotonicamente robusto para qualquer erro nas estimativas dos parâmetros da linearização desde que $\Delta R_1 < 1,0R_1$. Vale notar, contudo, que essa projeção para Δ_v é conservadora. Podemos relaxar a restrição para $|\Delta_v| \lesssim 2$, se considerarmos o valor máximo de x_1 (≈ 40 cmH₂O) medido nas simulações da seção 4.4 e, para $|\Delta_v| \lesssim 1$, se aumentamos a confiança nos parâmetros, de modo que o erro máximo nas estimativas seja de 50%.

Apesar de que não sintonizaremos os algoritmos ILC pensando em satisfazer a condição de convergência robusta do Teorema 4, algumas simulações serão feitas com o objetivo de avaliar a robustez dos sistemas de controle às incertezas paramétricas.

4.3 Configuração das simulações

As simulações feitas têm o objetivo de comparar o desempenho do controle em malha fechada tradicional com ILC e com ambos combinados. Projetamos um controlador Proporcional-Integral (PI) usando o método do lugar das raízes para conseguir boas características transientes para $G_1(z)$ e $G_2(z)$ em malha fechada. Reforçamos que as medições no sistema são amostradas

e processadas por um controlador digital com taxa $F_s = 500$ Hz. Nosso controlador PI tem a forma

$$C(z) = K_{pPI} + \frac{K_{iPI}}{z - 1},$$

em que os valores projetados de K_{pPIx} e K_{iPIx} ² para cada modo de ventilação estão listados na Tabela A.1.

Na sintonia do algoritmo ILC também usamos as funções de transferência (4.4a) e (4.4b). Escolhemos $Q(q) = 1$, e utilizamos o algoritmo ILC proporcional, com³

$$L(q) = K_x.$$

A sintonia de K_x foi feita através da função *fminbnd* disponível no Matlab para encontrar o valor do ganho que minimiza η_2 dentro de um intervalo especificado. Para a arquitetura que combina PI + ILC, projetamos K_x para a planta equivalente (3.11) vista pelo algoritmo.

A Figura 3.2 ilustra a arquitetura que usamos para controlar o ventilador. Podemos considerar $Q(q) = 0$ para as simulações com apenas o controlador PI, e $C(z) = 0$ para apenas ILC. Todos os parâmetros usados nas simulações estão listados no apêndice A. Aqueles relativos ao ventilador foram extraídos de [2]. As condições iniciais são $x_1(0) = x_2(0) = P_{PEEP}$ e se repetem em todas as iterações.

As simulações foram feitas em ambiente Simulink no MATLAB R2018a, com um *solver* selecionado automaticamente e com um passo de simulação fixo de 1×10^{-4} . O algoritmo ILC foi implementado em um *script* que roda após cada iteração.

4.4 Resultados

No modo controlado a volume, para o sistema que usa apenas ILC, selecionamos $K_{ILCFlow} = 14$ e obtivemos $\eta_2 = 0,9947$. Para a arquitetura que combina malha fechada com ILC, projetamos $K_{PIILCFlow} = 62,95$ e calculamos $\eta_2 = 0,9992$. A vazão desejada é dada por (2.2) com $\gamma = 8$ ml/s e $VT = 2$ ml.

No modo controlado a pressão, sintonizamos $K_{PIILCPressure} = 17,35$ com $\eta_2 = 0,9984$. Para o sistema com apenas ILC, entretanto, não conseguimos encontrar um ganho para o qual a norma do erro é monotonicamente convergente. A razão disso é o polo integrador de $G_2(z)$, que faz com que $|1 - K_{ILCPressure}G_2(e^{i\theta})| \rightarrow \infty$ a medida que $\theta \rightarrow 0$. Se utilizamos a função *fminbnd*, encontramos o valor (errado) de $\eta_2 = 22,4325$, para $K_{ILCPressure} = 12,1$.

² $x \in \{\text{Flow, Pressure}\}$ para diferenciar os ganhos dos controladores PI usados no modo controlado a volume e a pressão.

³ $x \in \{\text{ILC, PIILC}\}$ para diferenciar os ganhos dos sistemas controlados por ILC ou PI + ILC combinados.

Uma solução para esse problema é utilizar o algoritmo ILC derivativo, com

$$L(q) = K_d \frac{q-1}{q}$$

para anular a ação integradora em $G_2(z)$. Com essa função de aprendizado, sintonizamos $K_{dILCPressure} = 270$ por *fminbnd*, com $\eta_2 = 0,9355$. A pressão de referência é dada por (2.1) com $\beta = 150 \text{ cmH}_2\text{O/s}$ e $P_{\max} = 35 \text{ cmH}_2\text{O}$.

Todos os sistemas satisfazem a condição (3.12) e são assintoticamente estáveis. Consideramos cada ciclo de respiração, começando com a expiração, como uma iteração.

4.4.1 Simulações lineares

Para validar a corretude do procedimento realizado, simulamos as três arquiteturas de controle (PI, ILC e PI + ILC) para as funções de transferência (4.4a) e (4.4b).

4.4.1.1 Modo controlado a volume

Para o modo controlado a volume, as Figuras 4.4 e 4.5 comparam, respectivamente, a vazão e o volume de ar fornecidos para 10, 100 e 1000 iterações. As Figuras 4.6 e 4.7 mostram a evolução do erro RMS a cada iteração, também para a vazão e o volume nas arquiteturas que usam ILC. A Figura 4.8 compara o esforço de controle para os três casos, após 10, 100 e 1000 iterações.

É interessante notar que, apesar da Figura 4.6 mostrar que a norma do erro converge de forma monotônica, não temos, necessariamente, uma melhora de desempenho no rastreamento da vazão a cada iteração. Se consideramos critérios de desempenho no domínio do tempo, como tempo de acomodação ou sobre-elevação, a Figura 4.4 mostra que a arquitetura com PI e ILC responde pior após 10 iterações que o PI atuando sozinho inicialmente.

Uma explicação para isso é que o algoritmo ILC proporcional penaliza ou “prioriza” erros maiores, como no momento do degrau. Isso faz com que o tempo de subida chegue a ser próximo de zero já nas iterações iniciais, porém ao custo de oscilações, maior tempo de acomodação e sobre-elevação. Para o sistema real, isso não acontece por causa das limitações no atuador, principalmente a saturação da válvula. O erro de rastreamento da referência de volume de ar fornecido ao pulmão é duas ordens de grandeza menor que o erro de rastreamento da vazão.

Em todas as arquiteturas com ILC, o erro de rastreamento convergiu virtualmente para zero após um certo número de iterações. Do mesmo modo, o esforço de controle convergiu para um mesmo valor final u_∞ nas arquiteturas com ILC. É possível ver na Figura 4.8 que a ação de controle do ILC é igual à do PI no regime permanente. Nos primeiros instantes de tempo após o degrau, entretanto, é mais agressiva, oscilando com uma amplitude aproximadamente 4 vezes maior que a da ação do PI.

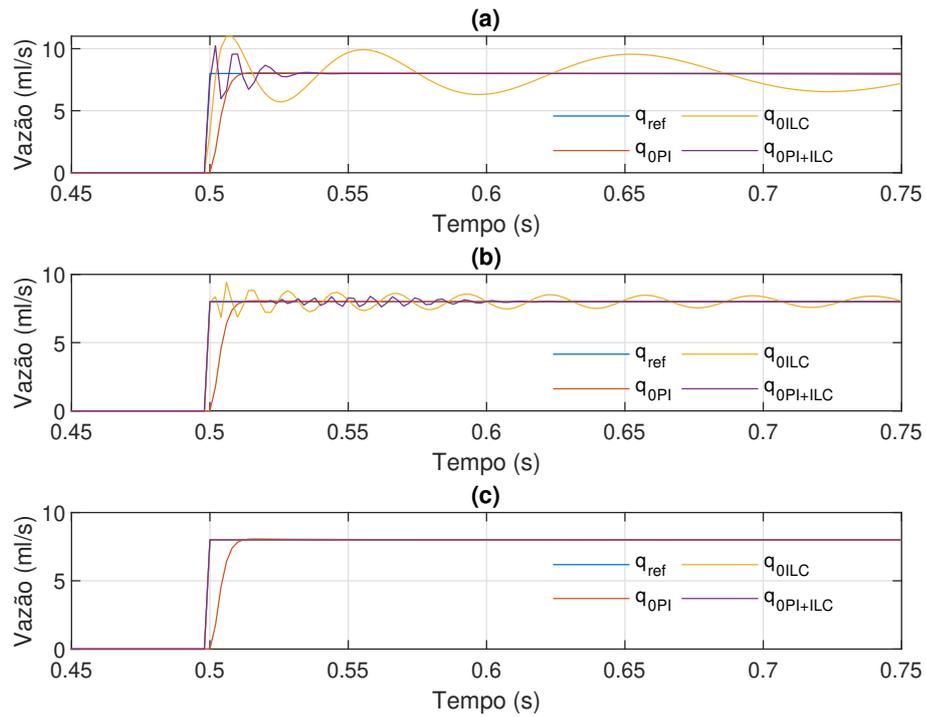


Figura 4.4 – Comparação da vazão de saída das três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.

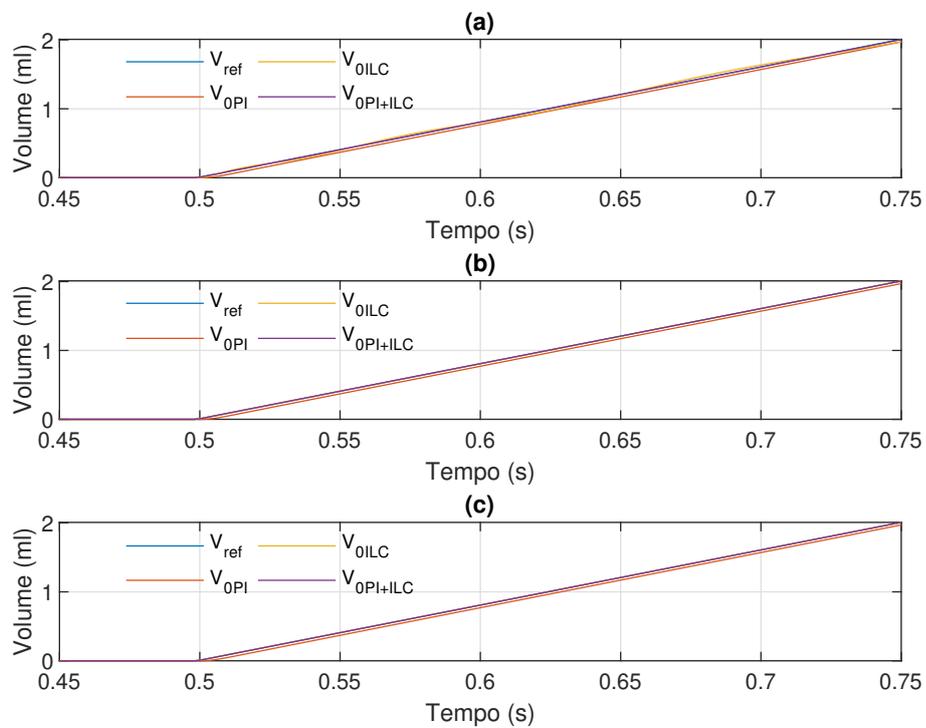


Figura 4.5 – Comparação do volume fornecido de ar nas três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.

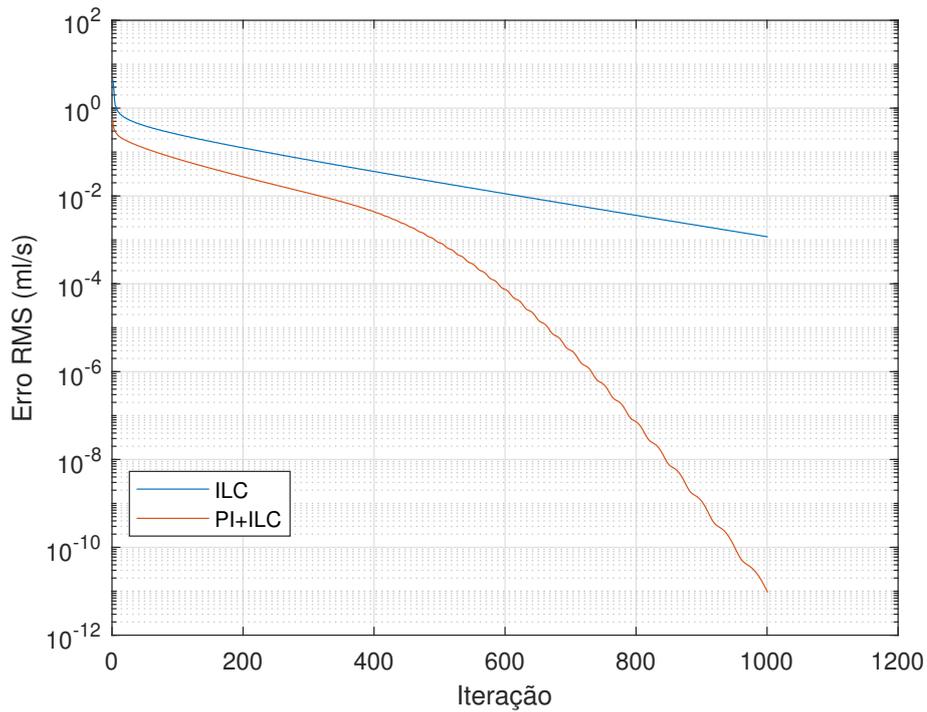


Figura 4.6 – Erro RMS da vazão para as duas arquiteturas com ILC - Modelo linear - Modo controlado a volume.

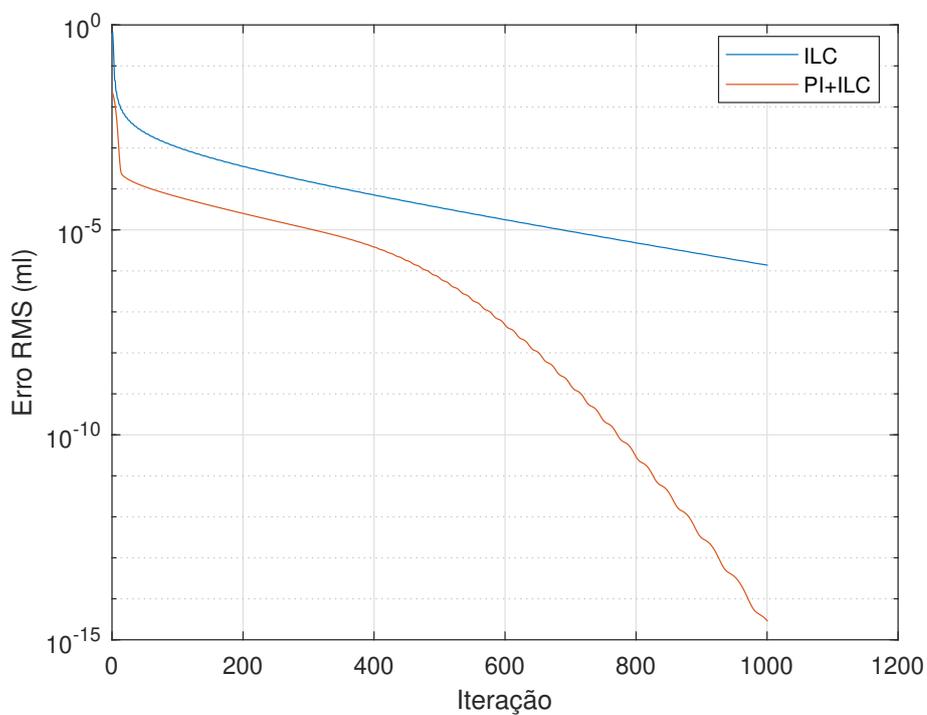


Figura 4.7 – Erro RMS do volume para as duas arquiteturas com ILC - Modelo linear - Modo controlado a volume.

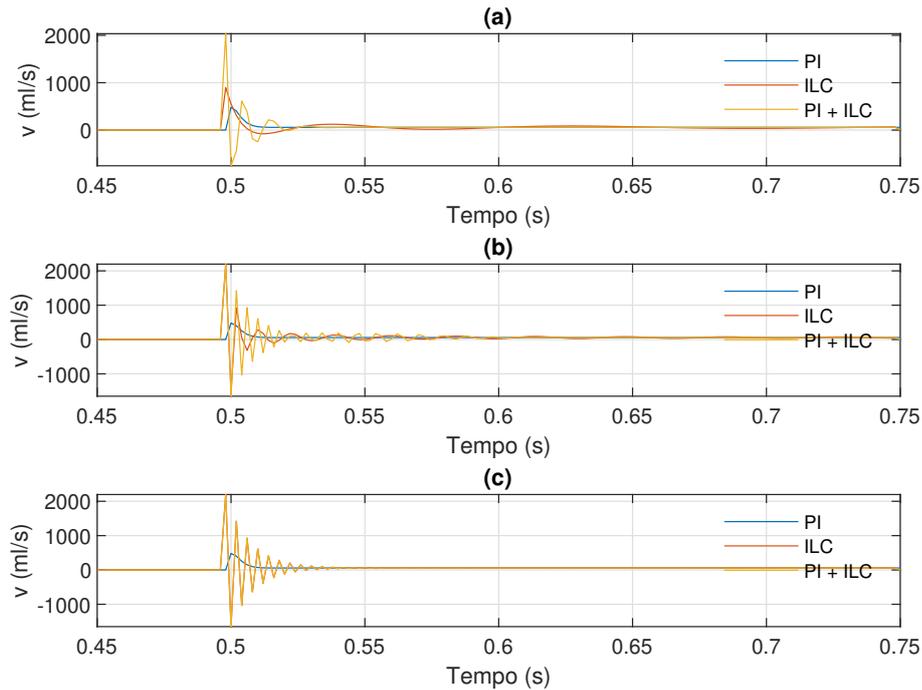


Figura 4.8 – Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a volume.

4.4.1.2 Modo controlado a pressão

As Figuras 4.9, 4.10 e 4.11 mostram, respectivamente, a saída, o erro RMS e o esforço de controle das três arquiteturas no modo controlado a pressão. $P_{ref}(0)$ é não nula e constante até o início da inspiração. Como a função $L(q)$ utilizada para a arquitetura com apenas ILC produz ações de controle em resposta à derivada do erro de rastreamento, existe uma diferença no "valor DC" da saída produzida pelo ILC em relação à referência. A Figura 4.10 mostra que se subtrairmos esse valor DC da curva de referência, essa arquitetura realmente converge de forma monotônica. Para valores do erro RMS menores que cerca de 10^{-15} o MATLAB considera o erro como 0. Para a simulação do sistema original, não teremos que lidar com esse problema, pois $P_{ref}(0) = P_0(0) = P_{PEEP}$.

Para o ILC proporcional, é interessante que, ainda que não haja convergência monotônica, o sistema é assintoticamente estável de acordo com a Definição 1. Isso é refletido no erro RMS como um transiente, onde o erro atinge um valor máximo antes de decair. A arquitetura com PI+ILC convergiu monotonicamente, como projetado. Da mesma maneira que para o modo controlado a volume, o erro de rastreamento converge praticamente para zero após um certo número de iterações em todas as arquiteturas com ILC. A ação de controle dessas arquiteturas é igual à do PI no regime permanente, porém é mais agressiva e oscilatória nas quinas da curva de referência.

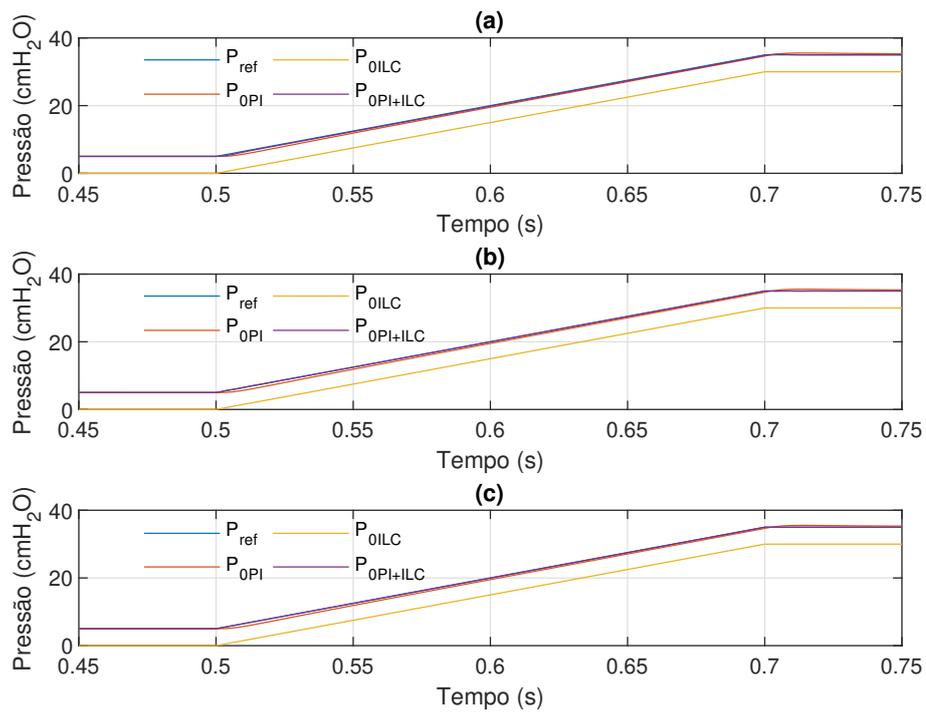


Figura 4.9 – Comparação das saídas das três arquiteturas de controle após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a pressão.

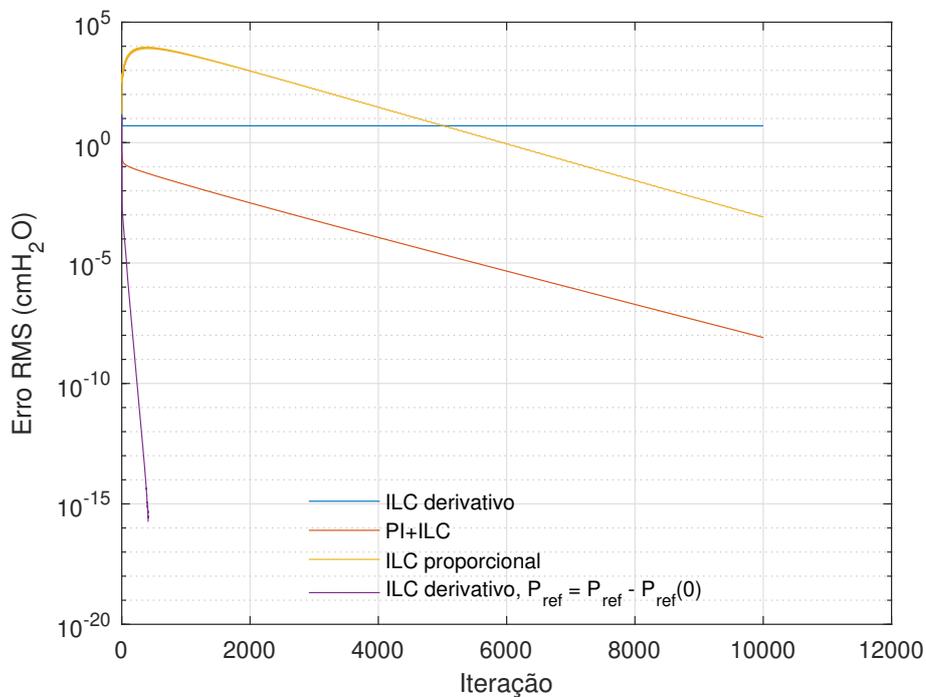


Figura 4.10 – Erro RMS para as duas arquiteturas com ILC - Modelo linear - Modo controlado a pressão.

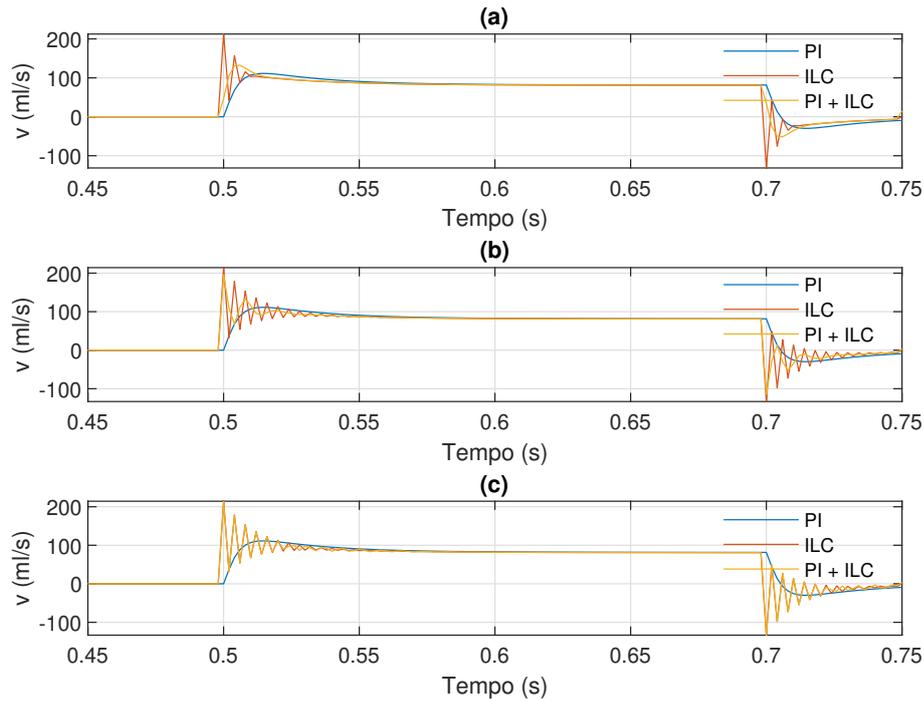


Figura 4.11 – Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações, (c) 1000 iterações - Modelo linear - Modo controlado a pressão.

4.4.2 Simulações no ventilador original linearizado

Validado o procedimento, aplicamos ILC ao modelo não-linear original (2.3) linearizado por realimentação⁴. Assuma-se 100 iterações para os resultados onde o número não for dito de forma explícita.

Nesta seção dedicamos uma figura inteira para cada comparação das saídas, para que os resultados possam ser vistos claramente. Para evitar um excesso de figuras que não seriam tão relevantes, preferimos não simular por mais de 100 iterações. A única arquitetura que ainda pode melhorar seu desempenho de forma significativa após 100 iterações, é a que utiliza apenas ILC, no modo controlado a volume.

4.4.2.1 Modo controlado a volume

Para o modo controlado a volume, as Figuras 4.12 e 4.13 comparam o rastreamento da vazão de ar para os três sistemas, respectivamente, após 10 e 100 iterações. As Figuras 4.14 e 4.15 fazem as mesmas comparações para o volume de ar fornecido ao pulmão. As evoluções dos erros RMS da vazão e do volume, para os sistemas com ILC, são mostradas, respectivamente, nas Figuras 4.16 e 4.17. Para o sistema controlado somente pelo PI, os respectivos erros RMS da vazão e do volume são de 0,6239 ml/s e 0.023 ml. O esforço de controle após 100 iterações é mostrado na Figura 4.18.

⁴ onde a única simplificação em relação a [2] é assumir que S1 é continuamente variável

É possível ver que, como era esperado, ILC sozinho não possui bom desempenho nas iterações iniciais. Por outro lado, a adição de ILC melhorou o desempenho do sistema controlado pelo PI. Na Figura 4.16 vemos que, mesmo com a saturação da válvula S1, a convergência monotônica foi mantida e ainda com menos oscilações que no caso linear da Figura 4.4. Tanto em termos de tempo de acomodação e sobre-elevação, quanto do valor RMS de e_j , a arquitetura com ILC teve performance superior ao controle em malha fechada tradicional no rastreamento da vazão.

Contra-intuitivamente, o erro RMS no rastreamento do volume de ar fornecido aumentou com o passar dos ciclos. A razão disso é que a sobre-elevação produzida pelo PI após o degrau fornece um volume de ar extra que de certo modo "compensa" o atraso na atuação. Apesar disso, o erro para o volume ainda é uma ordem de grandeza menor que o erro para a vazão.

Por mais que tenhamos projetado nosso algoritmo ILC para $e_\infty = 0$ com $Q(q) = 1$, devido à saturação e ao tempo morto na atuação, o menor erro assintótico atingido foi algo próximo de 0,620. O que a Figura 4.18 realmente mostra é a ação de controle comandada, já que Rv_1 saturou durante a subida do degrau nas três abordagens.

ILC produz incrementos na ação de controle com base no erro de rastreamento da iteração passada e, no caso do degrau, é inevitável que não exista nenhum erro, já que o atuador possui as limitações físicas mencionadas. Isso faz com que o algoritmo comande um esforço de controle cada vez maior, ainda que, após certo número de iterações, isso não gere nenhum efeito na prática.

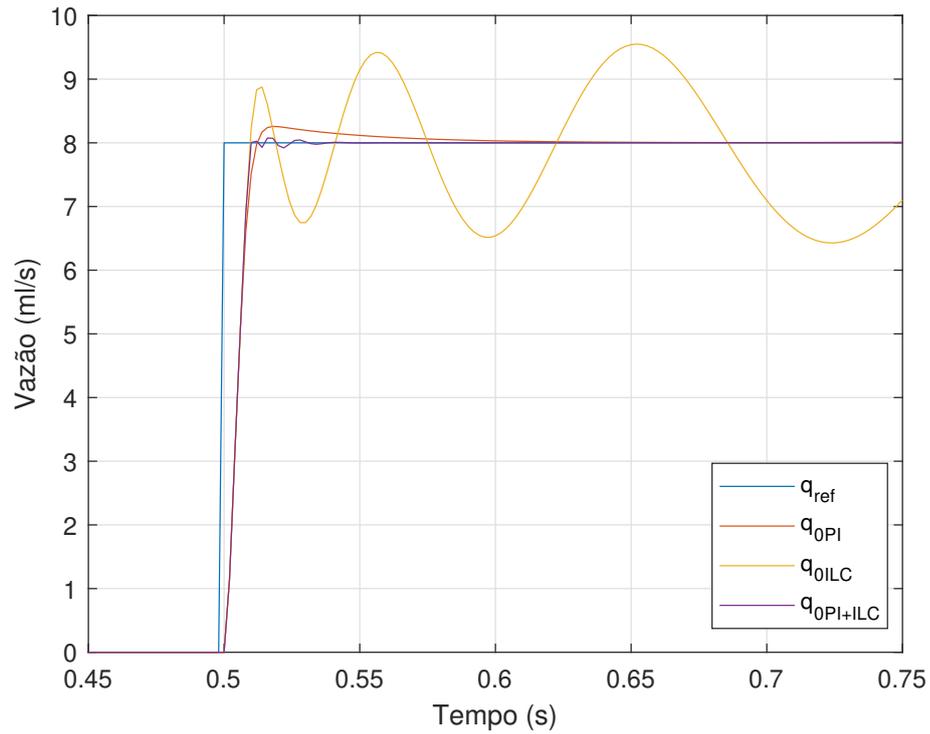


Figura 4.12 – Comparação da vazão de saída nas três arquiteturas de controle após 10 iterações - Modo controlado a volume.

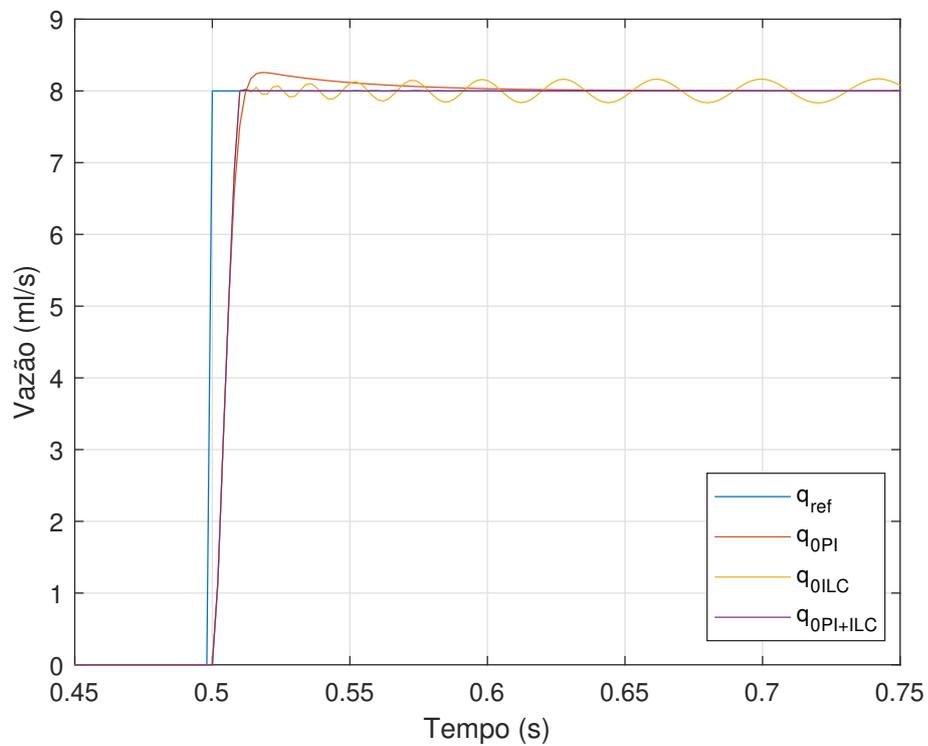


Figura 4.13 – Comparação da vazão de saída nas três arquiteturas de controle após 100 iterações - Modo controlado a volume.

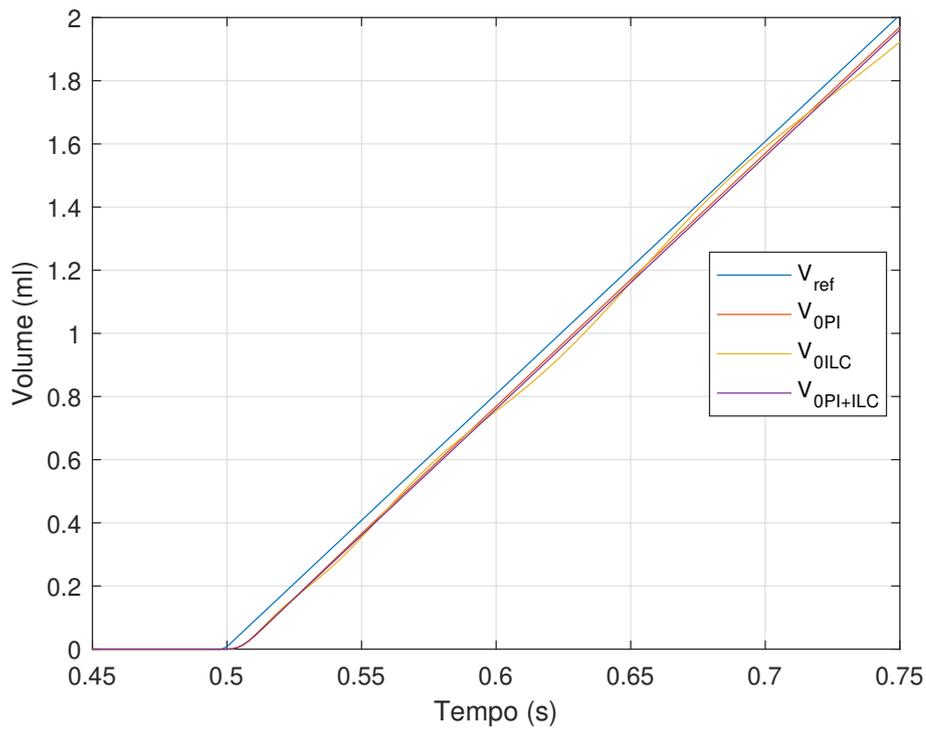


Figura 4.14 – Comparação do volume fornecido de ar nas três arquiteturas de controle após 10 iterações - Modo controlado a volume.

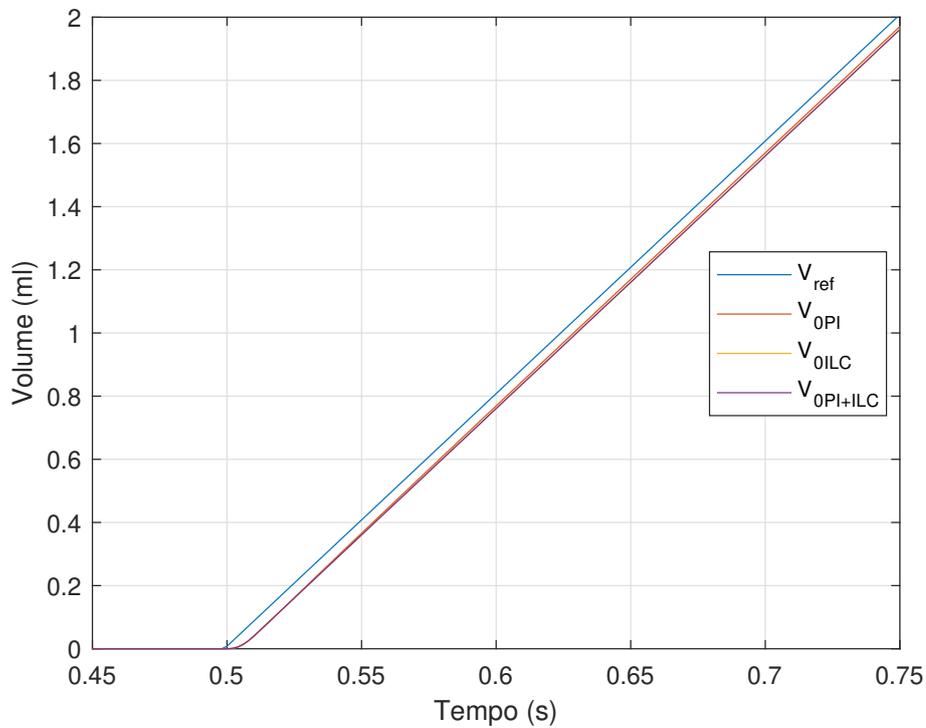


Figura 4.15 – Comparação do volume fornecido de ar nas três arquiteturas de controle após 100 iterações - Modo controlado a volume.

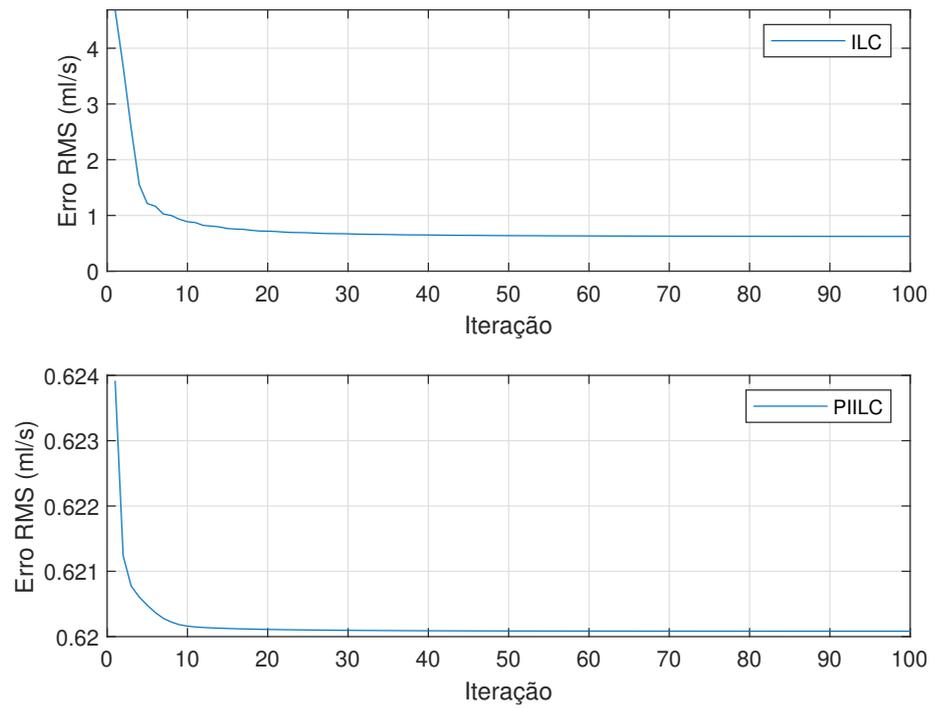


Figura 4.16 – Erro RMS da vazão para as duas arquiteturas com ILC - Modo controlado a volume.

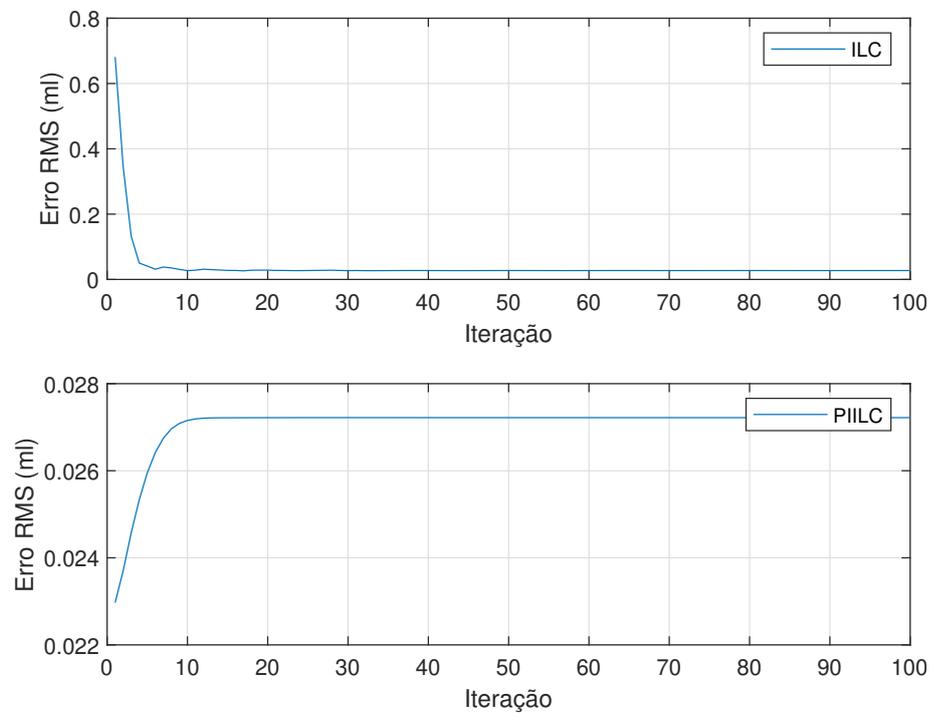


Figura 4.17 – Erro RMS do volume para as duas arquiteturas com ILC - Modo controlado a volume.

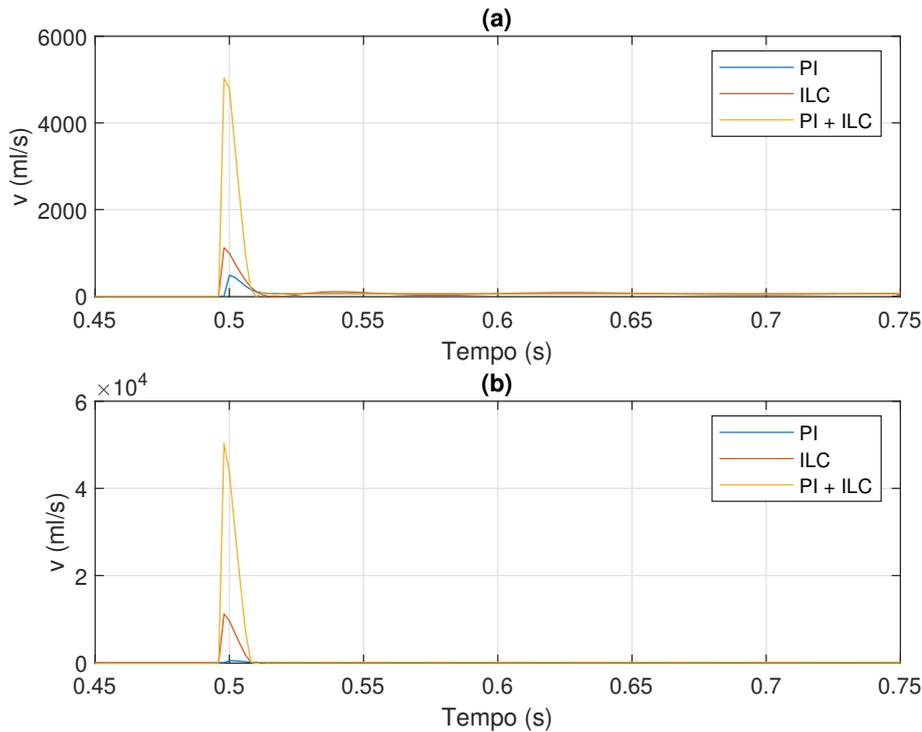


Figura 4.18 – Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações - Modo controlado a volume.

4.4.2.2 Modo controlado a pressão

Para o modo controlado a pressão colocamos nas figuras somente o período inspiratório para facilitar a visualização dos resultados. As Figuras 4.19 e 4.20 comparam o desempenho dos três sistemas, respectivamente, após 10 e 100 iterações. Os erros RMS em cada iteração para os sistemas com ILC são mostrados na Figura 4.21 e para o sistema controlado somente pelo PI o erro RMS é $0,3689 \text{ cmH}_2\text{O}$. O erro no sistema controlado apenas por ILC é alto nas primeiras iterações, porém converge rapidamente para seu valor assintótico. A arquitetura com ILC + PI, ainda que tenha uma taxa de convergência mais lenta, possui erro RMS igual ao do PI inicialmente. O esforço de controle após 100 iterações é mostrado na Figura 4.22.

Embora o laço aberto composto pela planta e pelo PI tenha dois polos integradores, não há tempo suficiente para o sistema chegar em regime permanente. Na prática, podemos considerar que o erro de rastreamento da rampa e_{ramp} pelo PI é de $0,4 \text{ cmH}_2\text{O}$, ao passo que para o PI com ILC é praticamente nulo. A Figura 4.21, contudo, mostra que o erro RMS não convergiu rigorosamente monotonicamente nesse caso.

A saturação do atuador também foi causa de dificuldade para todas as arquiteturas rastrear a referência após $t = 0,7 \text{ s}$. Apesar de que a ação de controle comandada v é bastante negativa a partir desse instante de tempo, se observarmos o circuito da Figura 4.1, vemos que o que realmente acontece é que Rv_1 satura em $Rv_{1\max}$ ($S1$ totalmente fechada). A partir daí, o sistema de controle não tem outra opção a não ser esperar o capacitor $C2$ transferir carga para Cc ,

segundo a dinâmica do circuito RC restante, que é lenta, principalmente devido à capacitância de C2. Em [2], esse problema foi resolvido com uma versão modificada da arquitetura do ventilador, que introduziu um sistema de recuperação de ar e de controle da PEEP.

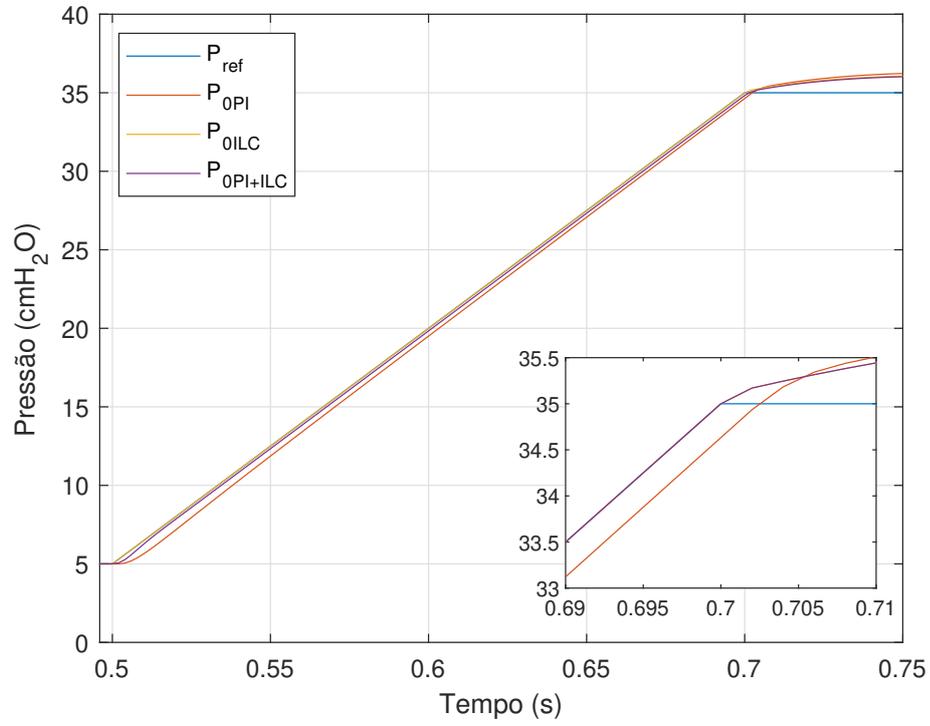


Figura 4.19 – Comparação das saídas das três arquiteturas de controle após 10 iterações - Modo controlado a pressão.

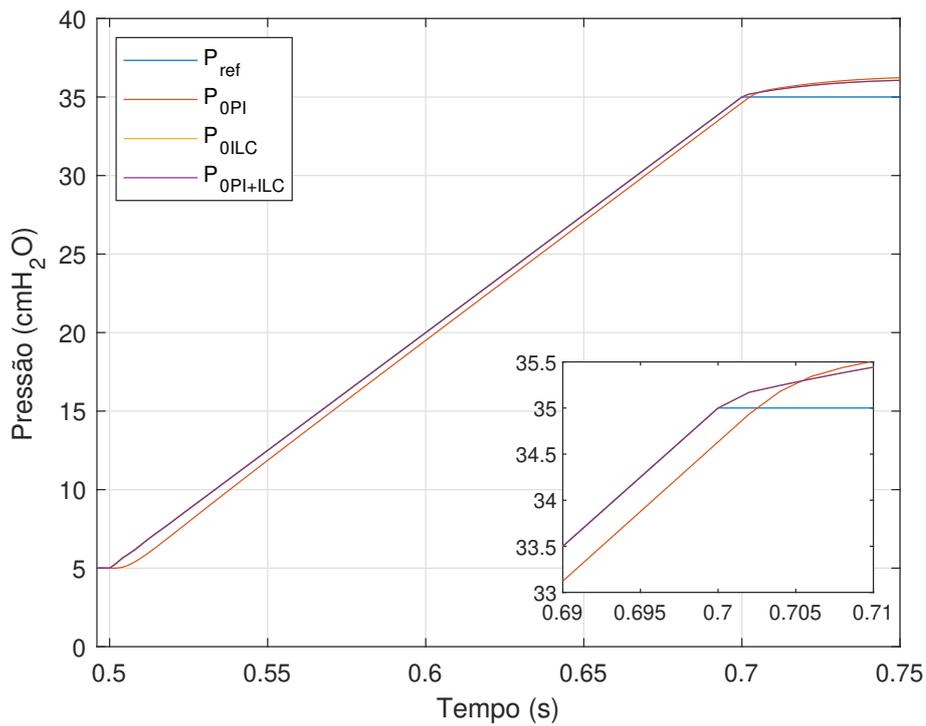


Figura 4.20 – Comparação das saídas das três arquiteturas de controle após 100 iterações - Modo controlado a pressão.

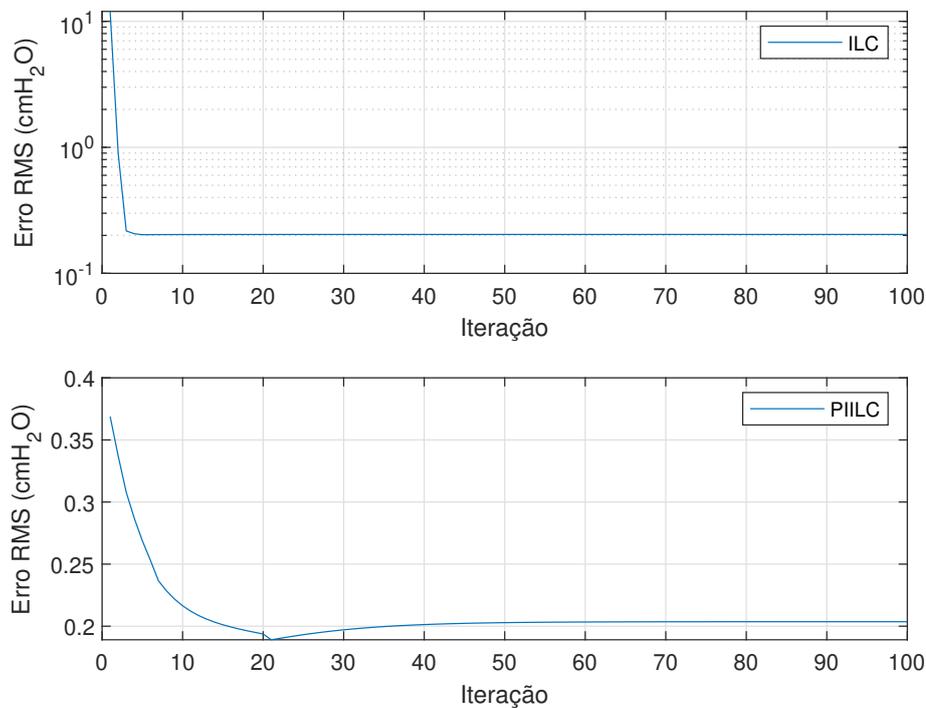


Figura 4.21 – Erro RMS para as duas arquiteturas com ILC - Modo controlado a pressão.

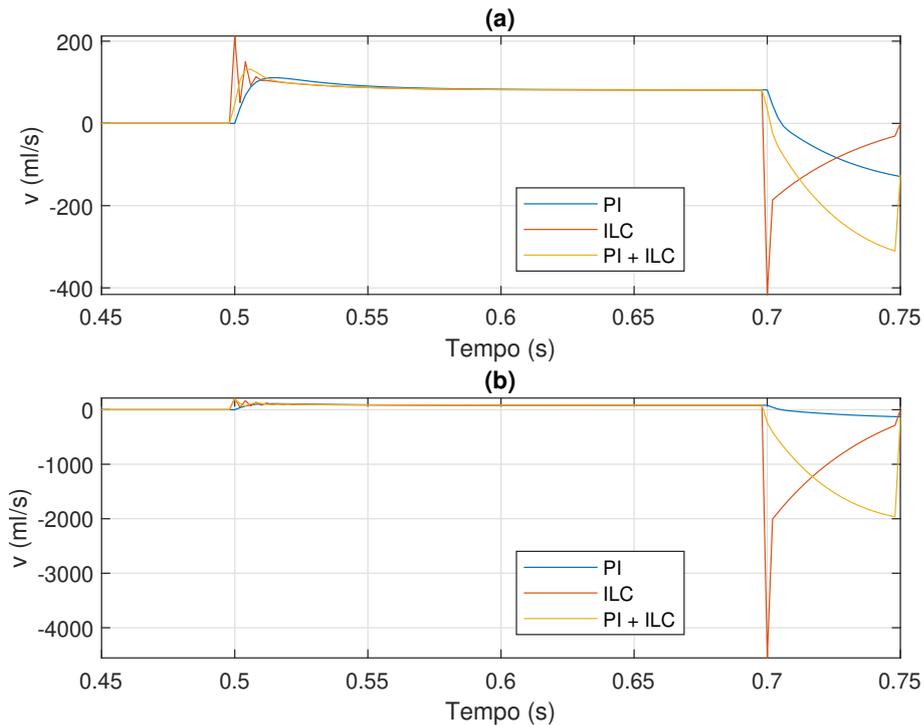


Figura 4.22 – Comparação do esforço de controle para as três arquiteturas após (a) 10 iterações, (b) 100 iterações - Modo controlado a pressão.

4.4.3 Simulações com vazamento

Para avaliar como perturbações de baixa frequência afetam o sistema controlado pela combinação de ILC + PI em comparação com o PI sozinho, imaginamos um vazamento de ar hipotético no reservatório intermediário C2 quando a pressão P_2 for maior que um valor $P_{leakage}$. Esse vazamento foi modelado como uma resistência $R_{leakage}$ em paralelo com C2, como ilustra a Figura 4.23.

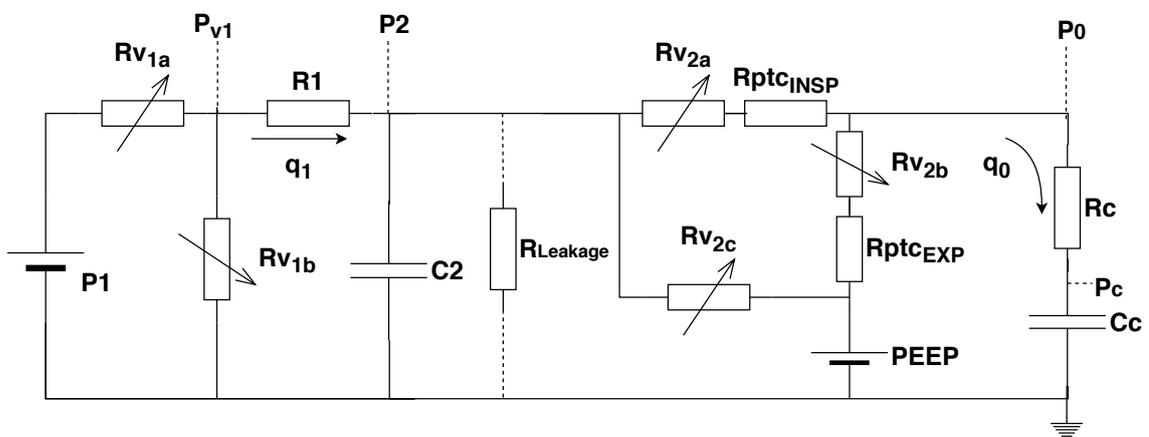


Figura 4.23 – Circuito do ventilador com vazamento hipotético.

Consideramos dois possíveis cenários. Em um deles, o vazamento acontece periodicamente.

mente, com $P_{leakage} = 30 \text{ cmH}_2\text{O}$ e $R_{leakage} = 2R_1$. Em outro, o vazamento acontece de forma aleatória. $P_{leakage}$ e $R_{leakage}$ assumem a cada iteração valores a partir de distribuições uniformes, respectivamente, nos intervalos $(10 \text{ cmH}_2\text{O}, 35 \text{ cmH}_2\text{O})$ e $(0,1R_1, 3R_1)$.

4.4.3.1 Modo controlado a Volume

Para o modo controlado a volume, a Figura 4.24 mostra o caso com vazamento periódico após 100 iterações. Podemos ver que, ainda que a perturbação não afete significativamente a vazão de ar, o controlador PI toma algum tempo para trazer o erro de volta para zero. Isso era esperado, dado que o controle em malha fechada consegue somente reagir a perturbações. Com ILC, no entanto, a ação de controle pode ser ajustada para rejeitar uma perturbação periódica já prevista.

Em relação aos vazamentos não-periódicos, é possível ver na Figura 4.25 que na arquitetura com PI+ILC, somente o controlador PI reage ao vazamento da iteração corrente. ILC, entretanto, produz ações de controle em resposta às perturbações das iterações anteriores. Apesar disso, esses efeitos na ação de controle são mitigados com o passar das iterações, de modo que, apenas a iteração anterior prejudica o desempenho do ILC de forma significativa. Fizemos alguns experimentos tomando a média do erro de 2, 3 e 5 iterações antes de atualizar a ação de controle, mas não houve melhora significativa em relação ao algoritmo normal. A Figura 4.26 mostra um dos resultados para o PI com ILC atualizado a cada 5 iterações.

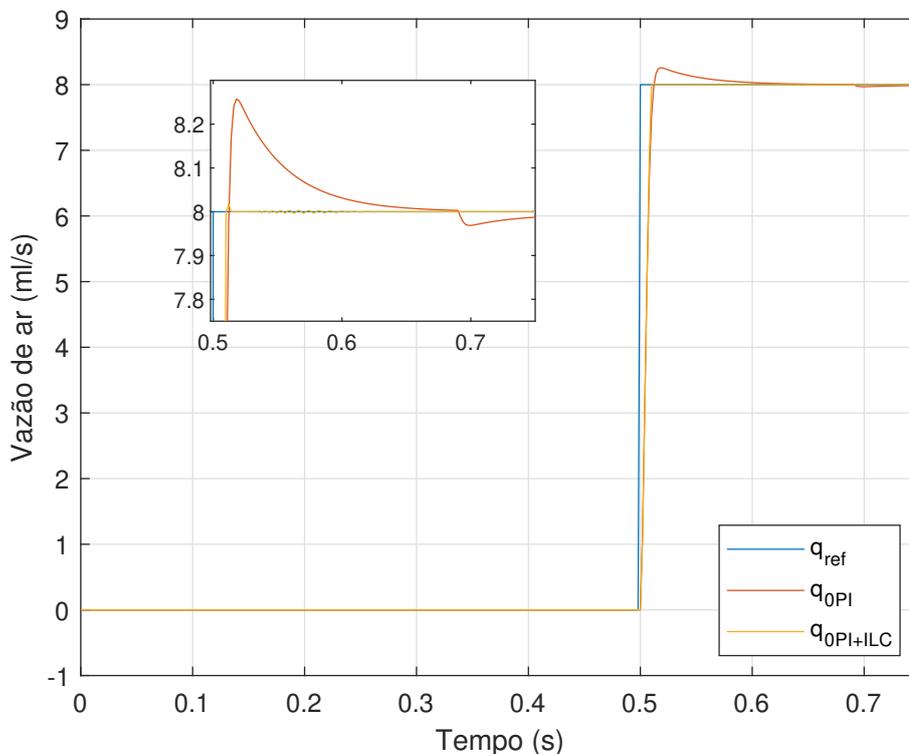


Figura 4.24 – Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento periódico - Modo controlado a volume.

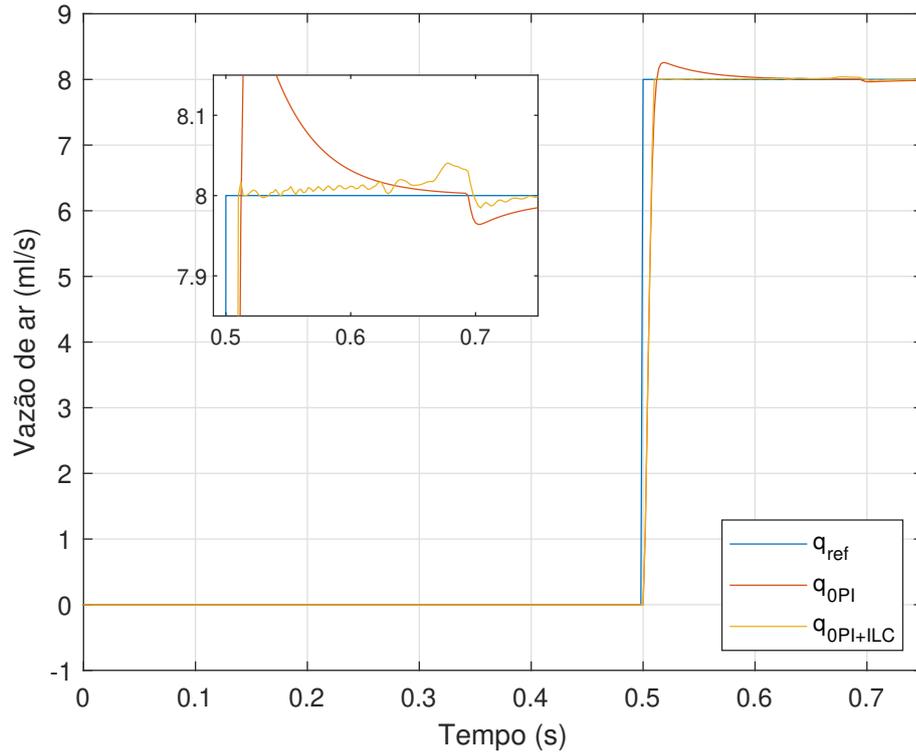


Figura 4.25 – Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento não-periódico - Modo controlado a volume.

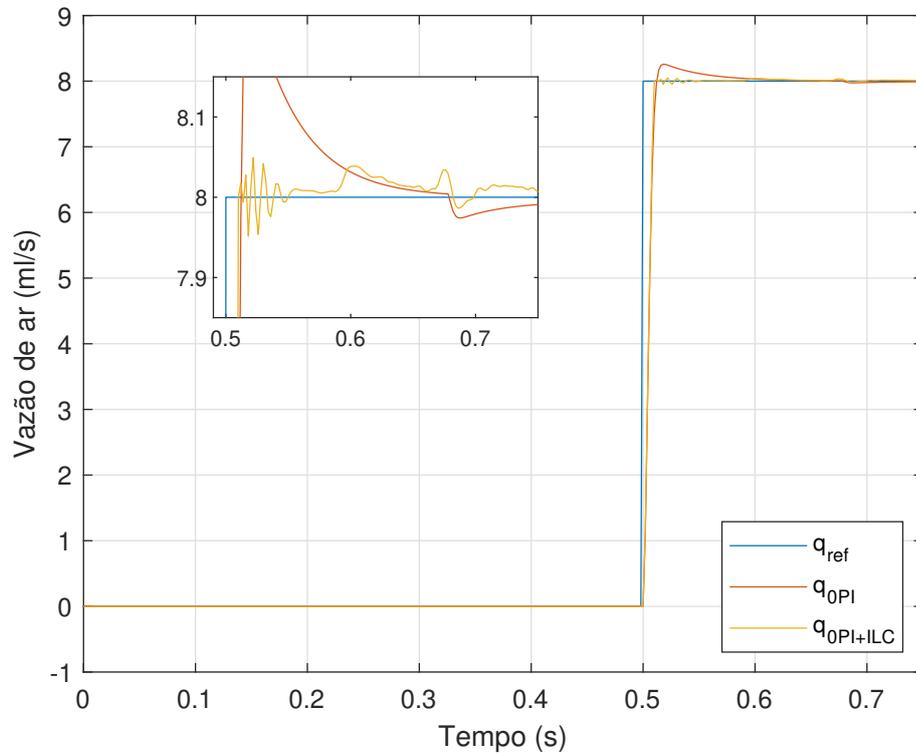


Figura 4.26 – Comparação da saída do sistema controlado pelo PI com o controlado por PI + [ILC atualizado a cada 5 iterações] para o cenário com vazamento não-periódico - Modo controlado a volume.

4.4.3.2 Modo controlado a pressão

Para o modo controlado a pressão, vemos na Figura 4.27 que o vazamento periódico (que ocorre em $P_0(t) = 30 \text{ cmH}_2\text{O}$, em $t \approx 0,67 \text{ s}$) pouco afetou a saída de ambos os sistemas. Pela física do processo, isso está relacionado à capacitância de C2 que faz com que variações em P_2 tenham uma dinâmica lenta e o PI faça a correção antes que isso afete P_0 . O mesmo aconteceu para o cenário com vazamento não-periódico da Figura 4.28.

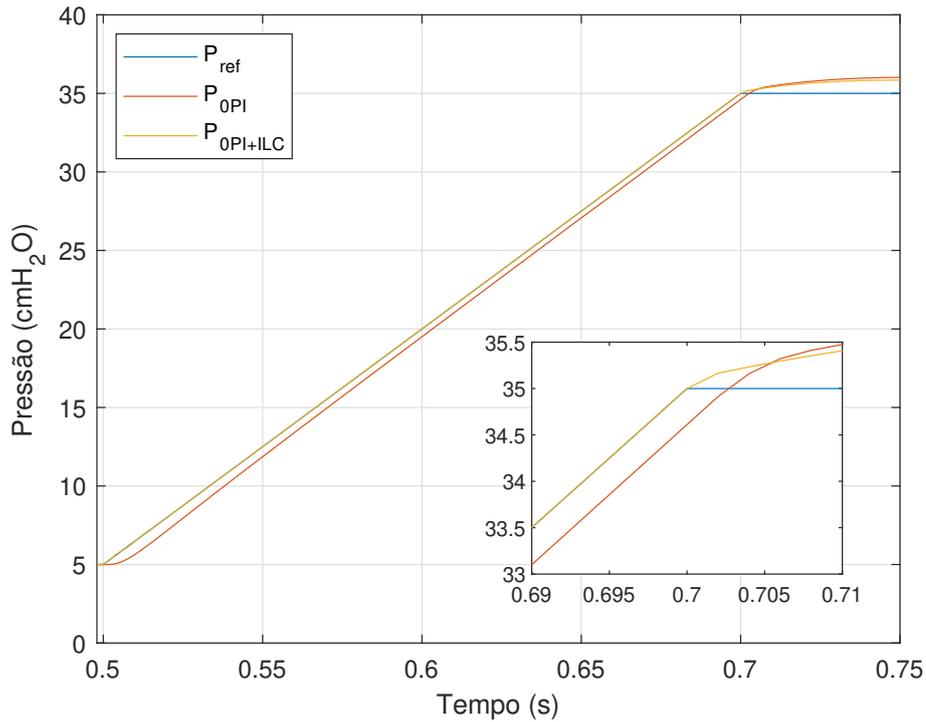


Figura 4.27 – Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com vazamento periódico - Modo controlado a pressão.

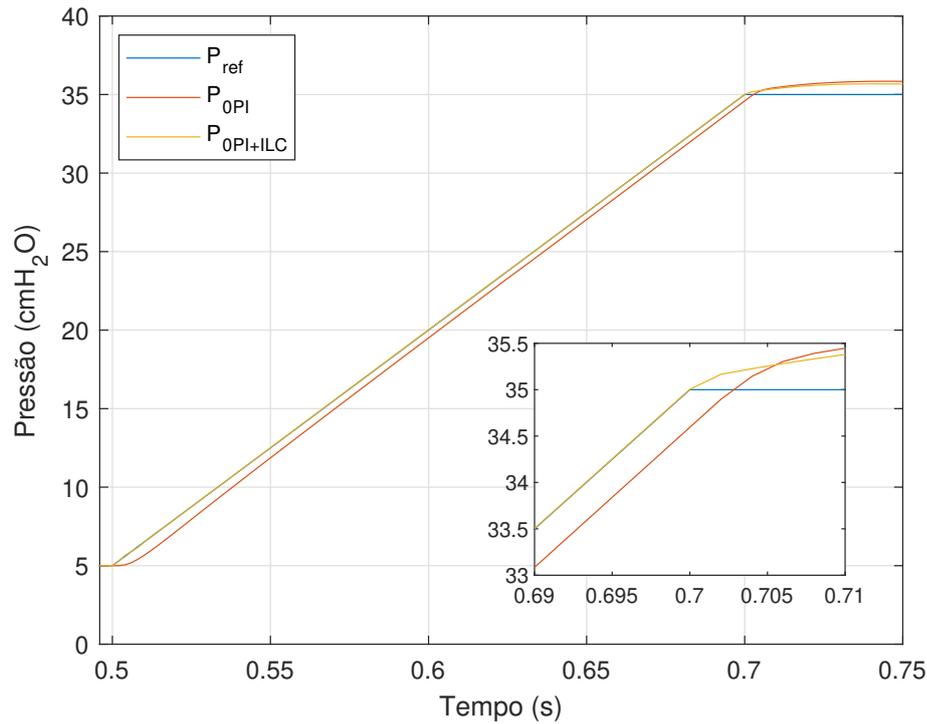


Figura 4.28 – Comparação da saída do sistema controlado pelo PI com o controlado por PI + ILC para o cenário com vazamento não-periódico - Modo controlado a pressão.

4.4.4 Ruído na medição

Como discutimos na seção 3.2.3, os efeitos de ruído na medição podem ser considerados em ILC como um termo $d_j(k)$ não constante entre iterações. Nesta seção simulamos os sistemas controlados pelo PI e PI+ILC para o caso em que existe ruído na medição da saída e de P_2 , usada na linearização por realimentação. O ruído é aditivo e modelado conforme [2], a partir de uma distribuição gaussiana com $\mu = 0$ e $\sigma = (0,5.TEB)$, onde TEB é a *Total Error Band* do transdutor, que representa uma porcentagem do seu fundo de escala. Assumimos que o mesmo modelo de sensor usado para medir P_0 também é usado na medição de P_2 . Os valores da TEB de cada sensor foram extraídos de [2] e estão listados na Tabela A.1. Foram considerados 3% do fundo de escala para a medição de pressão e 3,5% para a medição de q_0 .

As Figuras 4.29 e 4.30 comparam as saídas dos sistemas controlados pelo PI e PI+ILC com ruído nas medições. Ainda que tenhamos utilizado $Q(q) = 1$, a taxa de aprendizado foi rápida o suficiente para evitar que ILC tentasse aprender com o ruído. Embora para o modo controlado a pressão não haja melhora visível de desempenho para a arquitetura de controle com ILC, para o modo controlado a volume, ILC conseguiu evitar a sobre-elevação após o degrau.

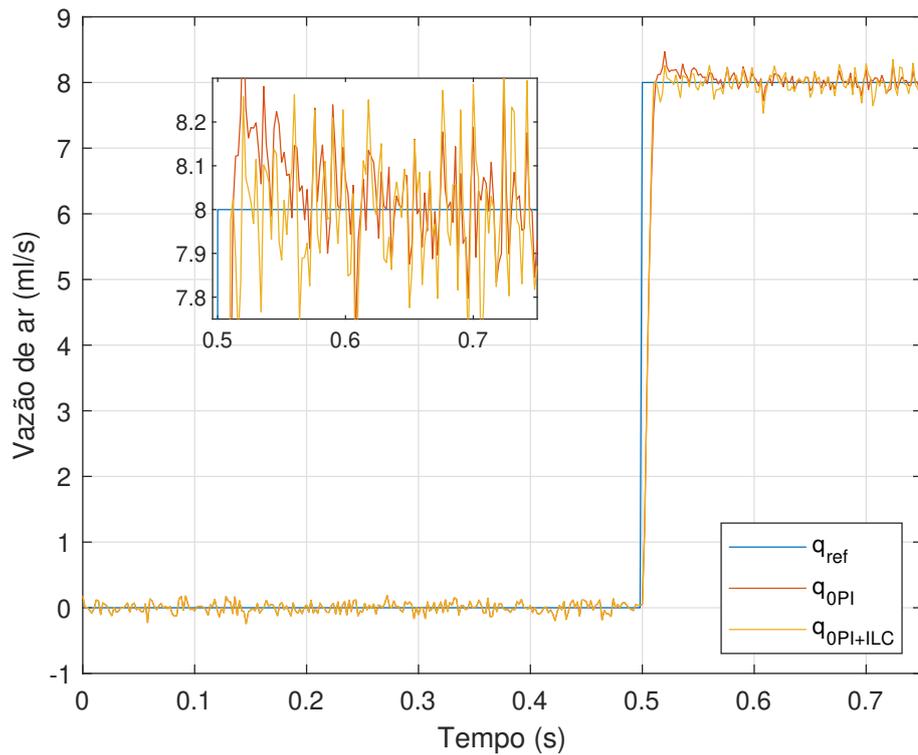


Figura 4.29 – Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com ruído na medição de q_0 e P_2 - Modo controlado a volume.

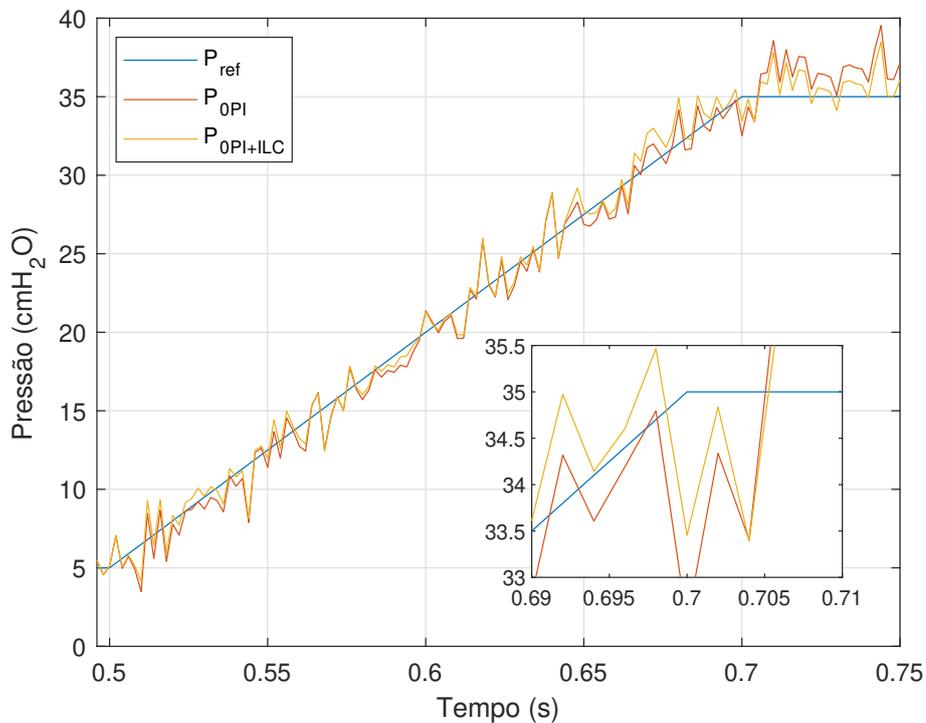


Figura 4.30 – Comparação da saída do sistema controlado pelo PI com o controlador por PI + ILC para o cenário com ruído na medição de P_0 e P_2 - Modo controlado a pressão.

4.4.5 Incertezas paramétricas

Ainda para avaliar a robustez do sistema de controle, consideramos a principal desvantagem da linearização por realimentação que é a sensibilidade a incertezas paramétricas. Como mostramos na seção 4.2.2, pequenos erros na estimativa dos parâmetros utilizados no cálculo da entrada virtual já introduzem não-linearidades no sistema. Vamos avaliar por simulação como diferentes valores de ΔP_1 e ΔR_1 afetam o desempenho dos sistemas controlados pelo PI e pela combinação de PI e ILC.

4.4.5.1 Modo controlado a volume

As Figuras 4.31 e 4.34 mostram como a saída do sistema controlado pelo PI é afetada por erros de estimativa dos parâmetros P_1 e R_1 de $\pm 20\%$ e $\pm 50\%$. Na Figura 4.37 consideramos a combinação de erros de estimativa de $\pm 50\%$ em ambos os parâmetros ao mesmo tempo. Projetamos o PI com uma margem de ganho de 15,1 dB e uma margem de fase de $68,4^\circ$, de modo que temos uma robustez razoável. Mesmo assim, para os casos com $\Delta P_1 = \pm 50\%P_1$ e $\Delta R_1 = \pm 50\%R_1$ tivemos valores de sobre-elevação possivelmente não aceitáveis.

Já as Figuras 4.32, 4.35 e 4.38 mostram a saída do sistema controlado por PI + ILC após 100 iterações para os mesmos cenários das Figuras 4.31, 4.34 e 4.37. A evolução do erro RMS em cada iteração para esses 3 casos é mostrada, respectivamente, nas Figuras 4.33, 4.36 e 4.38.

Como discutimos na seção 3.2.3, geralmente não há problemas em garantir estabilidade assintótica para o ILC. Garantir a convergência monotônica do erro, no entanto, é mais difícil. Considerando o Teorema 4 e a equação do erro assintótico (3.7), existe um compromisso na escolha de $Q(q)$, onde valores próximos a 1 fornecem melhores e_∞ e mais próximos de 0 oferecem mais robustez. Com a sintonia de $Q(q) = 1$, nosso projeto ILC é o menos robusto possível com o algoritmo (3.2). Mesmo assim, exceto para os casos em que $\Delta P_1 = -50\%P_1$, tivemos bons resultados. Para esses casos, a violação da condição de convergência monotônica se manifestou na forma de oscilações de alta amplitude na saída. Eventualmente, com um número maior de iterações ($N \approx 200$), esses casos também convergem para um bom rastreamento.

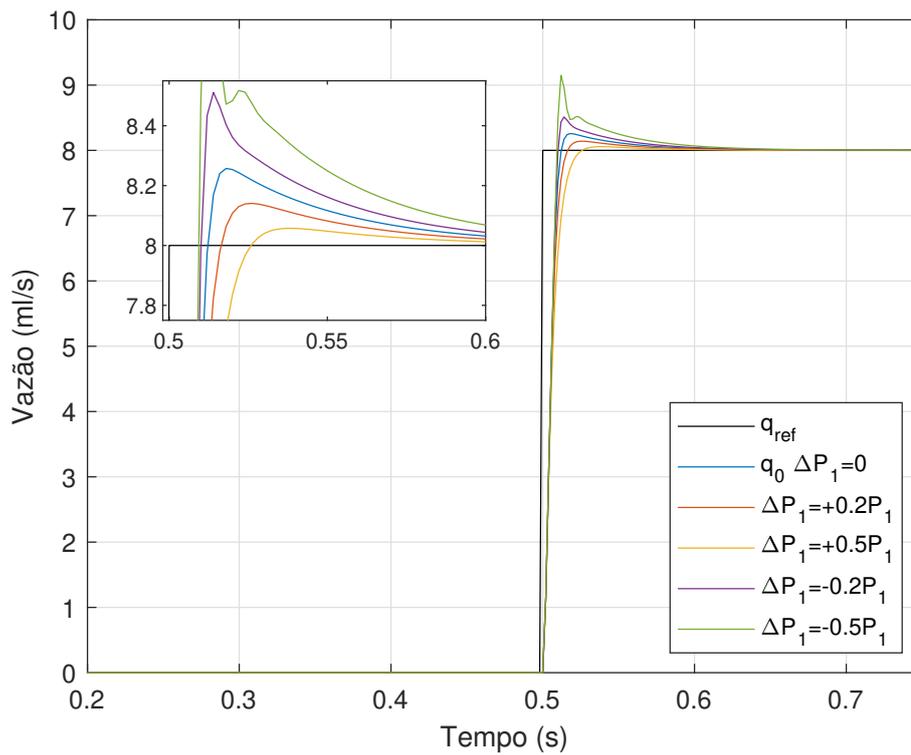


Figura 4.31 – Comparação das saídas do sistema controlado por PI para algumas estimativas de P_1 - Modo controlado a volume.

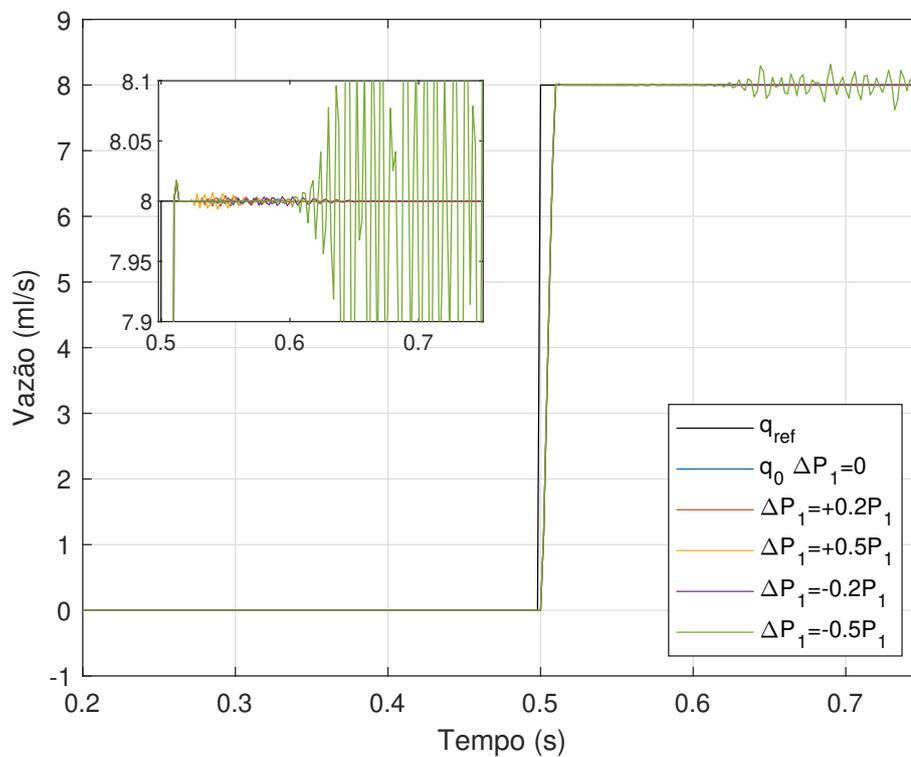


Figura 4.32 – Comparação das saídas do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.

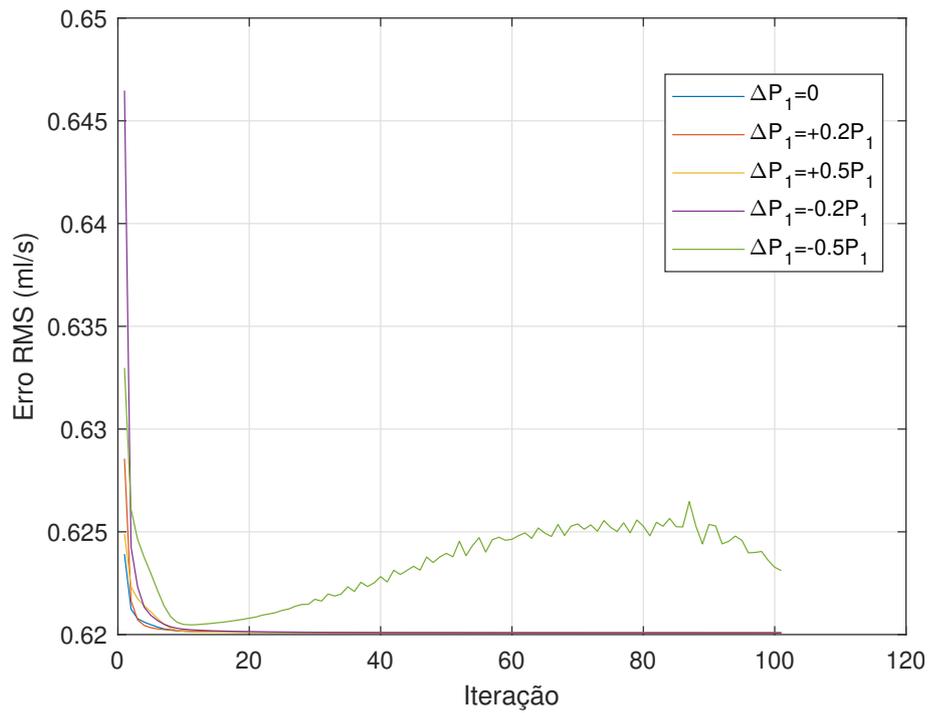


Figura 4.33 – Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.

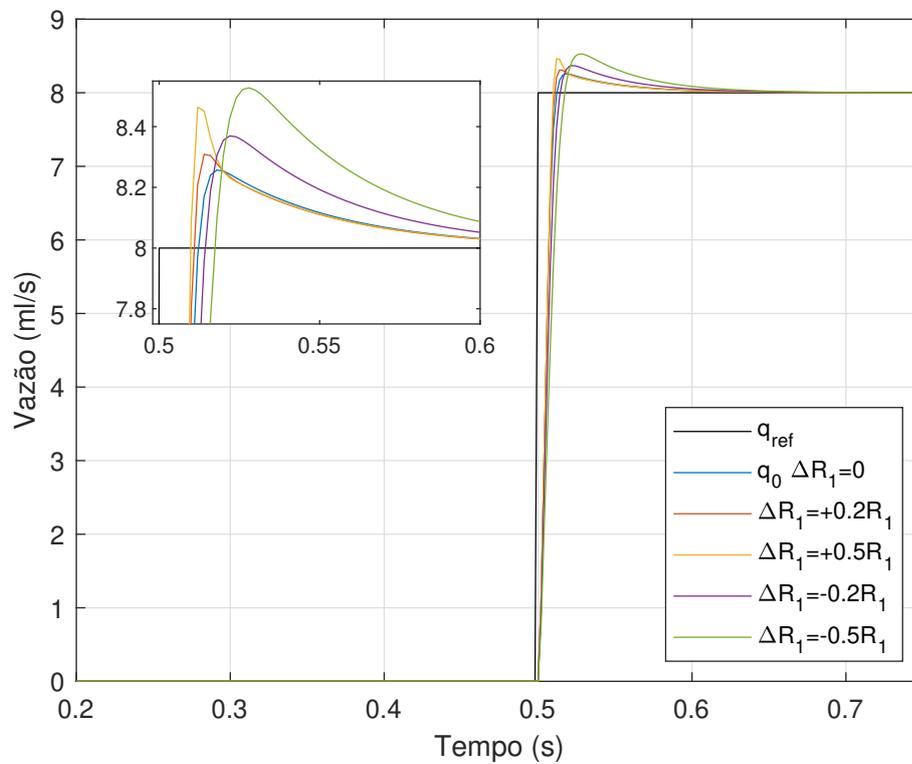


Figura 4.34 – Comparação das saídas do sistema controlado por PI para algumas estimativas de R_1 - Modo controlado a volume.

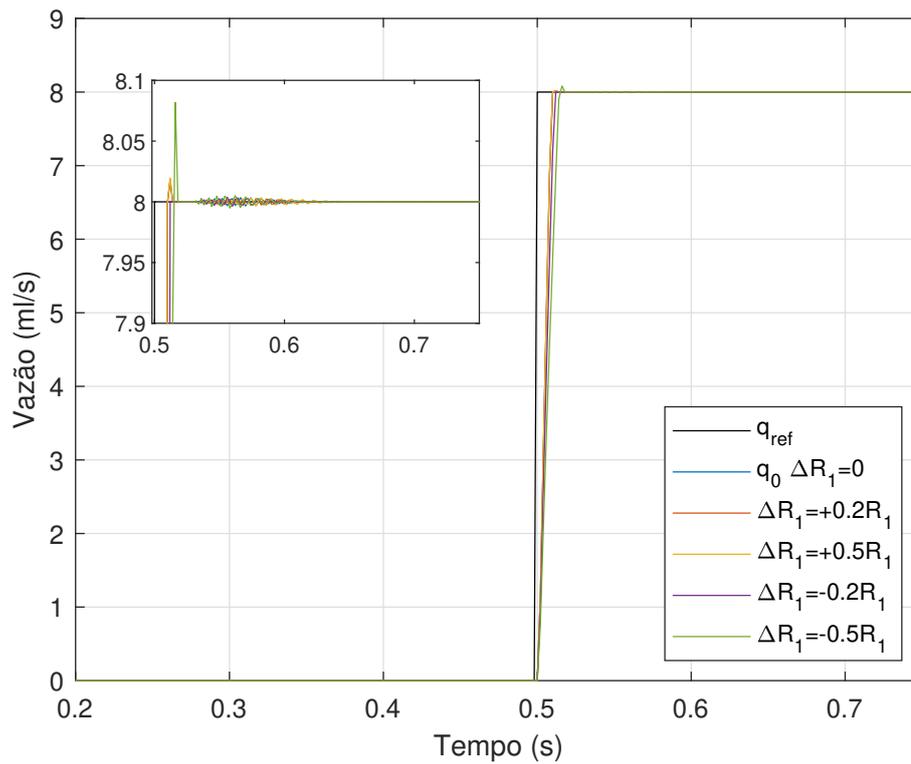


Figura 4.35 – Comparação das saídas do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a volume.

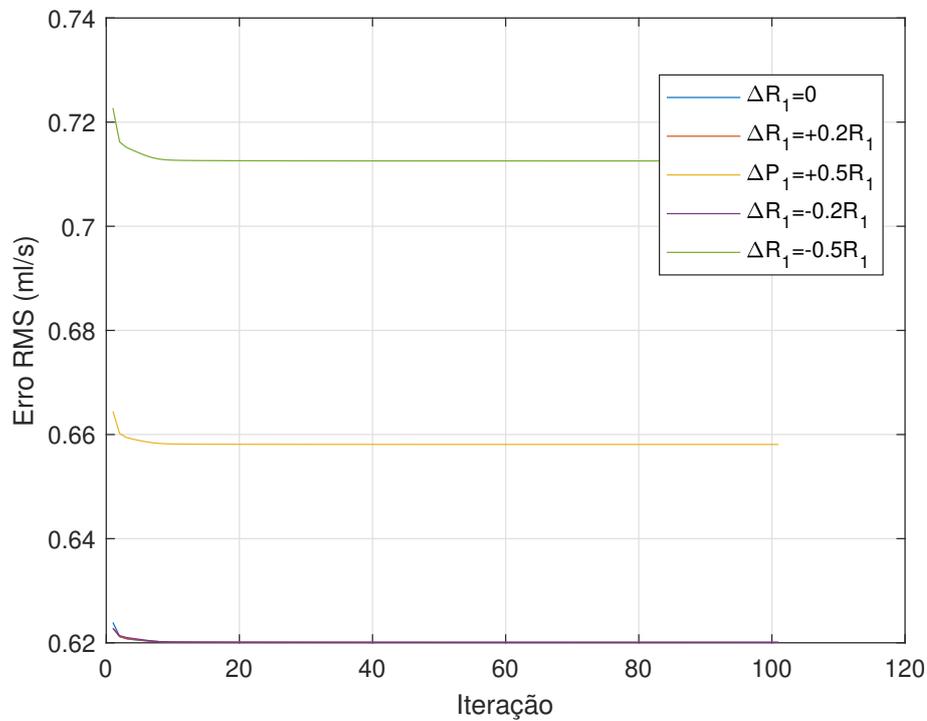


Figura 4.36 – Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a volume.

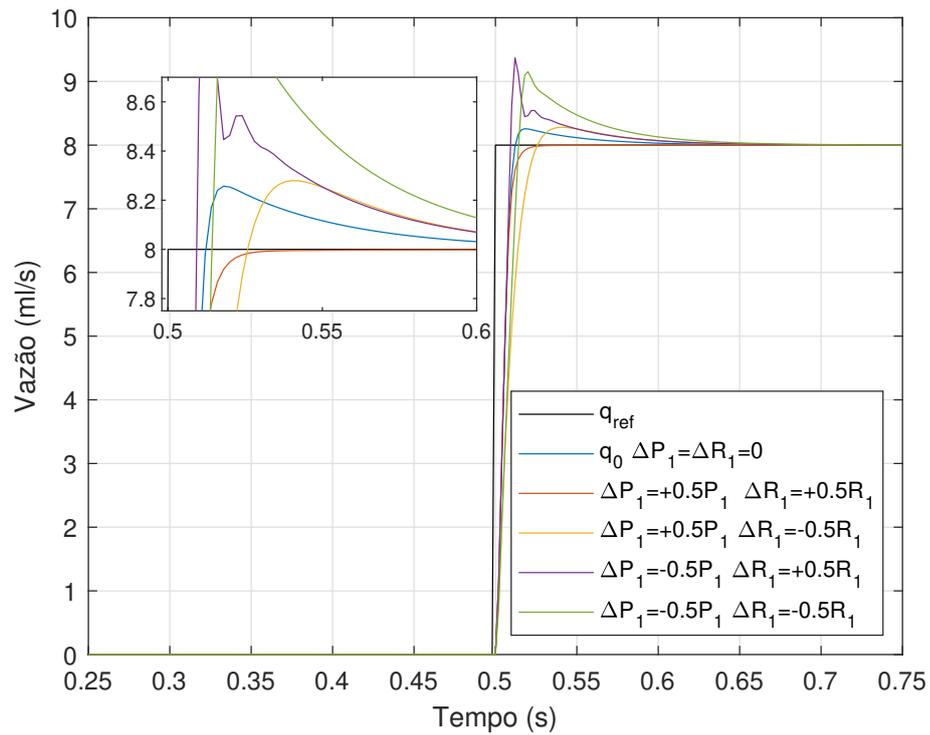


Figura 4.37 – Comparação das saídas do sistema controlado por PI para grandes erros de estimativas de P_1 e R_1 - Modo controlado a volume.

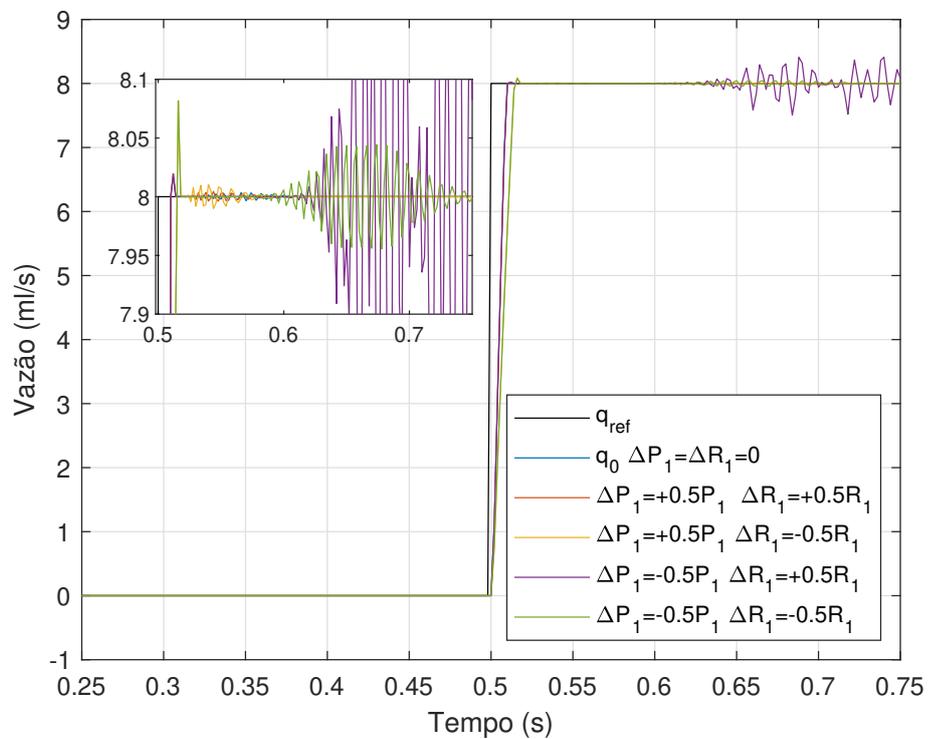


Figura 4.38 – Comparação das saídas do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a volume.

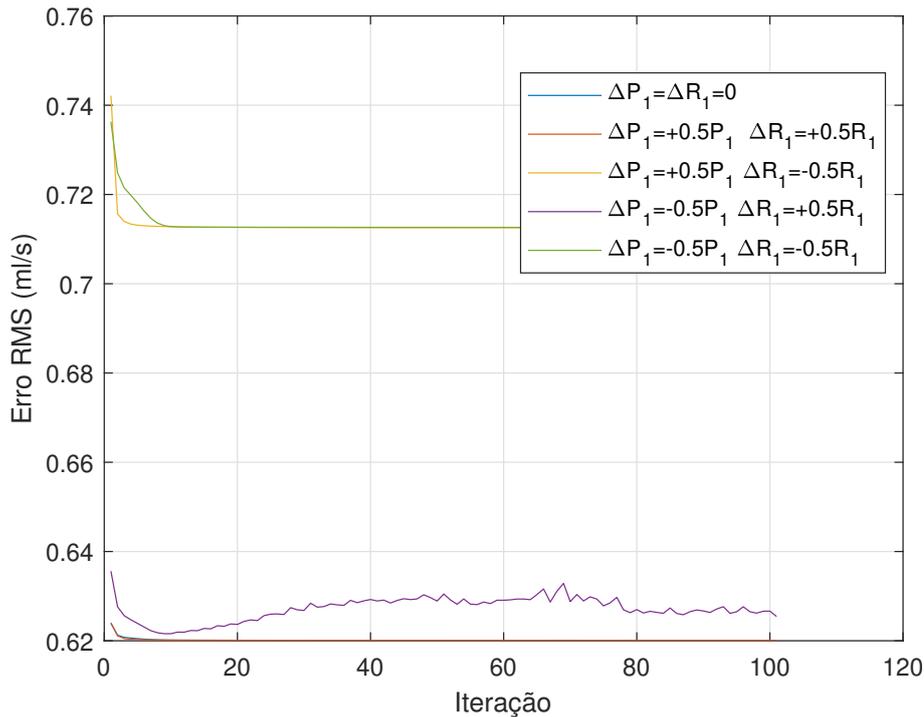


Figura 4.39 – Erro RMS do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a volume.

4.4.5.2 Modo controlado a pressão

As Figuras 4.40 e 4.43 mostram como a saída do sistema controlado pelo PI é afetada por erros de estimativa dos parâmetros P_1 e R_1 de $\pm 20\%$ e $\pm 50\%$. Na Figura 4.46 consideramos a combinação de erros de estimativa de $\pm 50\%$ em ambos os parâmetros ao mesmo tempo. O PI projetado para $\bar{G}_2(z)$ é ainda mais robusto que o usado no modo controlado a volume, com uma margem de ganho de 34,7 dB e uma margem de fase de $75,5^\circ$. Podemos ver nessas figuras que as variações em \hat{P}_1 afetaram relativamente pouco a saída e em \hat{R}_1 menos ainda.

Já as Figuras 4.41, 4.44 e 4.47 mostram a saída do sistema controlado por PI + ILC após 100 iterações nos mesmos cenários. A evolução do erro RMS em cada iteração para cada caso é mostrada, respectivamente, nas Figuras 4.42, 4.45 e 4.48.

Mesmo com $Q(q) = 1$, nosso projeto ILC se manteve robusto e com resultados tão bons quanto as limitações físicas do atuador permitem. Em nenhum caso a condição de convergência monotônica foi violada.

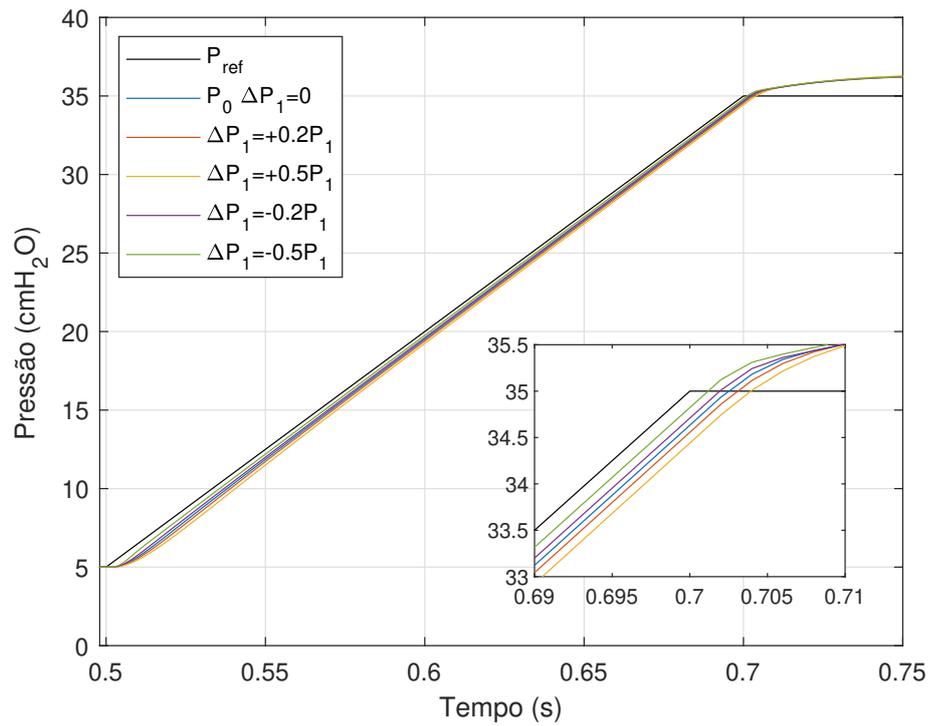


Figura 4.40 – Comparação da saída do sistema controlado por PI para algumas estimativas de P_1 - Modo controlado a pressão.

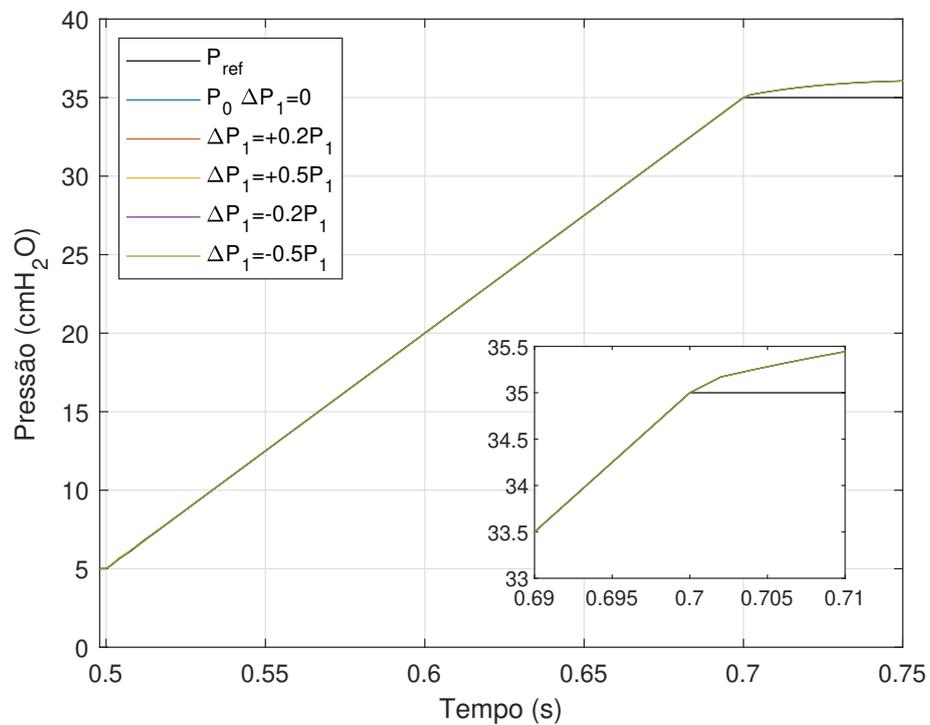


Figura 4.41 – Comparação da saída do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a pressão.

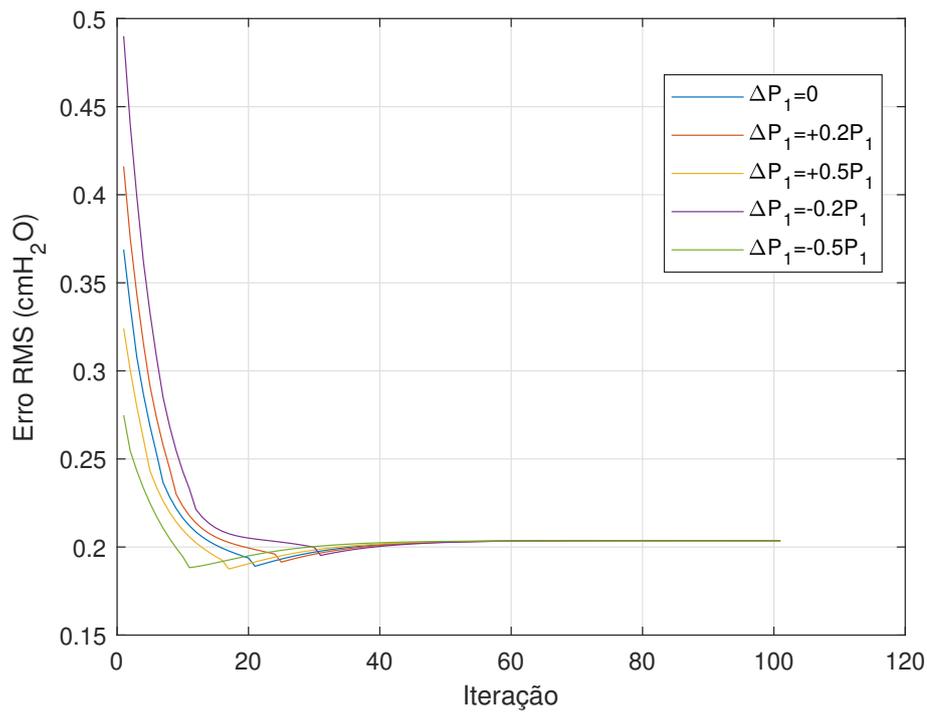


Figura 4.42 – Erro RMS do sistema controlado por PI + ILC para algumas estimativas de P_1 - Modo controlado a pressão.

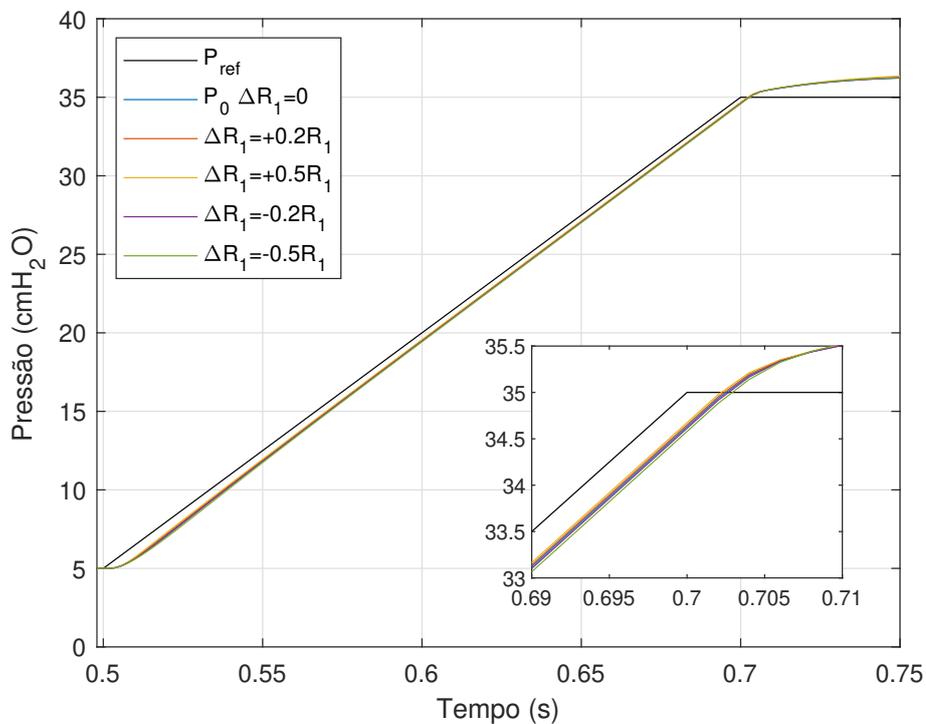


Figura 4.43 – Comparação da saída do sistema controlado por PI para algumas estimativas de R_1 - Modo controlado a pressão.

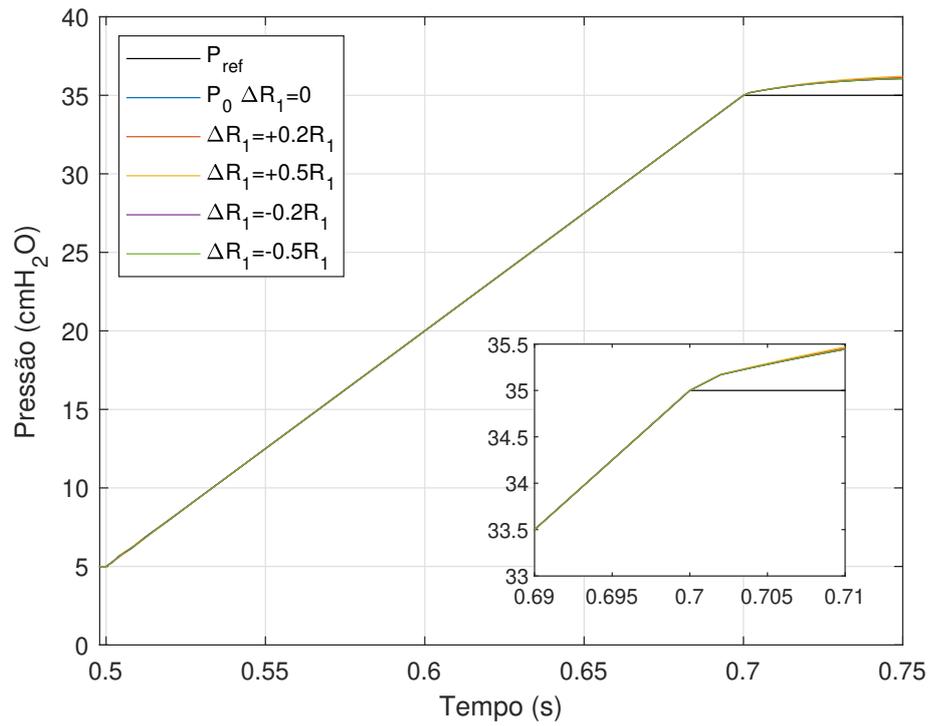


Figura 4.44 – Comparação da saída do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a pressão.

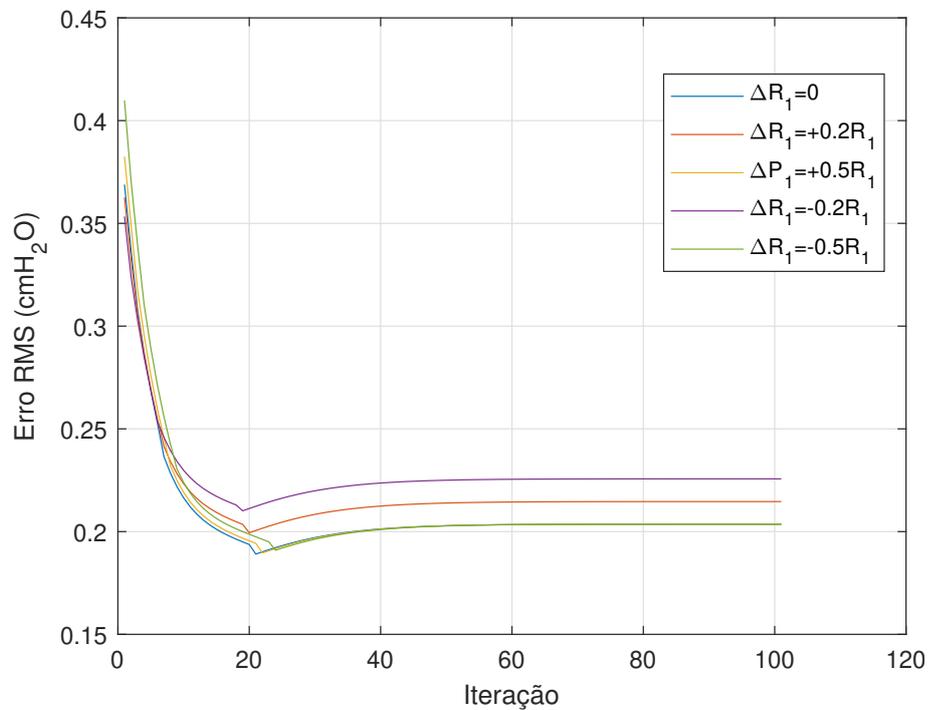


Figura 4.45 – Erro RMS do sistema controlado por PI + ILC para algumas estimativas de R_1 - Modo controlado a pressão.

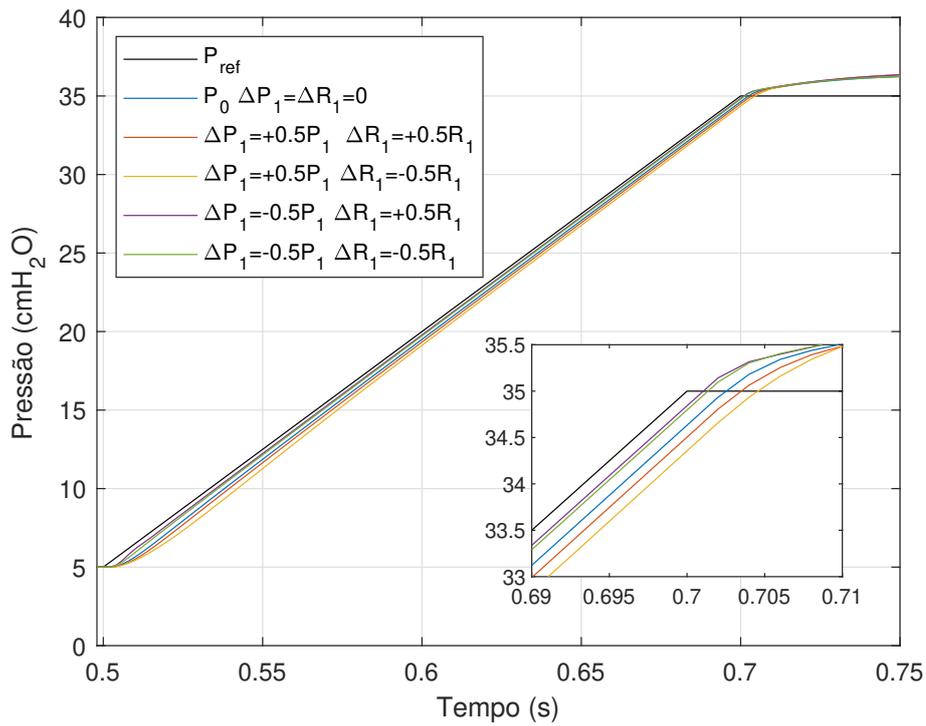


Figura 4.46 – Comparação da saída do sistema controlado por PI para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.

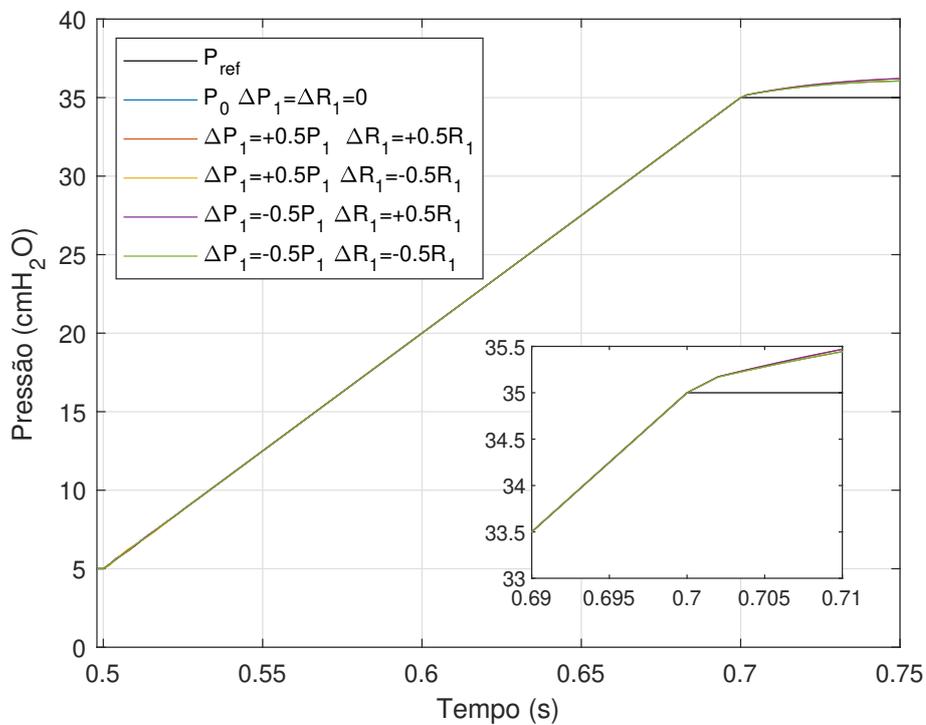


Figura 4.47 – Comparação da saída do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.

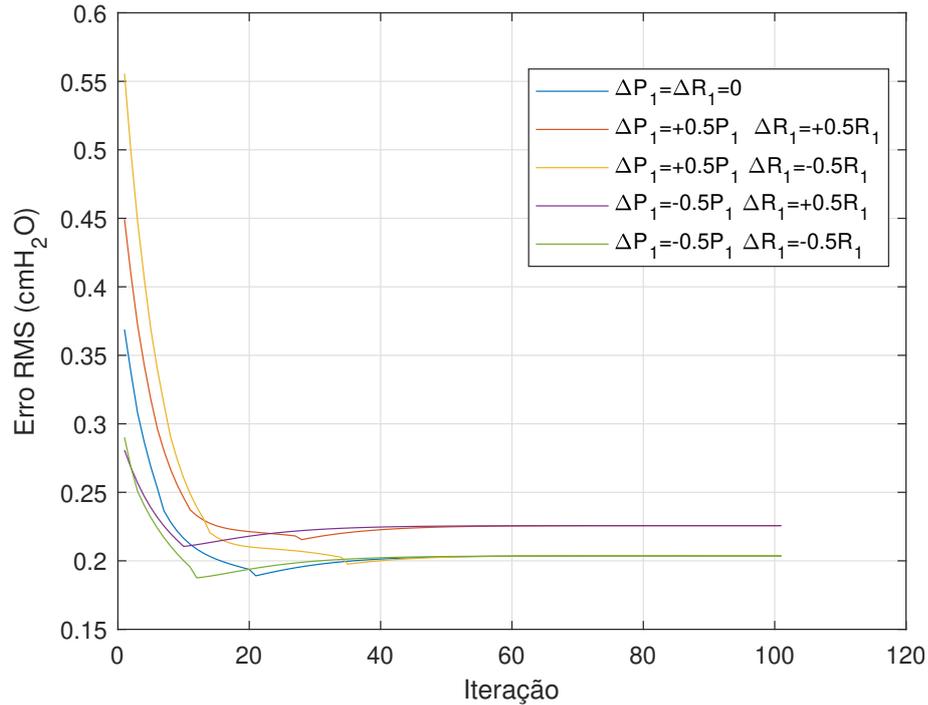


Figura 4.48 – Erro RMS do sistema controlado por PI + ILC para estimativas "extremas" de P_1 e R_1 - Modo controlado a pressão.

4.4.6 Variação na frequência de amostragem

Nas seções anteriores utilizamos $F_s = 500$ Hz como frequência de amostragem. Nesta seção, avaliamos como taxas de amostragem mais rápidas e mais lentas podem afetar a performance de ILC. Também temos a intenção de mostrar como pode ser feito o projeto do filtro-Q, e como ele afeta o rastreamento da saída. Vamos considerar a arquitetura com apenas ILC para exemplificar o que podemos esperar com a variação de F_s .

As magnitudes das respostas em frequência mostradas nesta seção estão em escala linear. Os filtros digitais são do tipo FIR de ordem 25. Utilizamos uma estratégia de filtragem digital de fase-zero com a função *filtfilt* do MATLAB. Essa função evita distorções de fase através da filtragem do sinal de entrada na direção original e, em seguida, na direção reversa. Como resultado, a função de transferência do filtro de fase-zero é igual ao quadrado da magnitude da função de transferência do filtro original [18]. A filtragem de fase-zero só é possível de forma não-causal, porém o fato de termos acesso a todas as amostras dos sinais da iteração anterior permite que usemos esse tipo de estratégia.

4.4.6.1 Modo controlado a volume, $F_s = 1000$ Hz

A função de transferência obtida a partir da discretização de (4.3a) com $F_s = 1000$ Hz é

$$\bar{G}_{1000}(z) = z^{-1} \frac{3,3509 \times 10^{-5}(z + 8,892)}{z - 0,9761}.$$

Para essa função não conseguimos garantir convergência monotônica com $L(q)$ proporcional e $Q(q) = 1$.

Tendo em mente que satisfazer o Teorema 3 equivale a fazer com que a resposta em frequência de $Q(e^{i\theta})[1 - L(e^{i\theta})\bar{G}_{1000}(e^{i\theta})]$ tenha magnitude menor que 1 para $\theta \in [0, \pi]$, podemos alcançar essa condição com o projeto do filtro-Q.

Se tomamos $L(q) = 1$, podemos aplicar o filtro-Q da Figura 4.49b a $|1 - L(e^{i\theta})\bar{G}_{1000}(e^{i\theta})|$ mostrada na Figura 4.49a para que o sistema satisfaça o Teorema 3. Optamos por não utilizar um filtro passa-baixas, pois a atenuação das altas frequências prejudica o rastreamento do degrau. A magnitude da resposta em frequência de $1 - e^{i\theta}\bar{G}_{1000}(e^{i\theta})$ após a filtragem é mostrada na Figura 4.50a e satisfaz a condição de convergência monotônica.

Tomamos $L(q) = K_{ILCFlow1000}$ e fizemos a sintonia desse ganho da maneira descrita na seção 4.3 para o algoritmo com filtro-Q. Projetamos $K_{ILCFlow1000} = 3,65$ e conseguimos $\eta_2 = 0,9804$. A magnitude da resposta em frequência de $1 - K_{ILCFlow1000}\bar{G}_{1000}(e^{i\theta})$ é mostrada na Figura 4.50b.

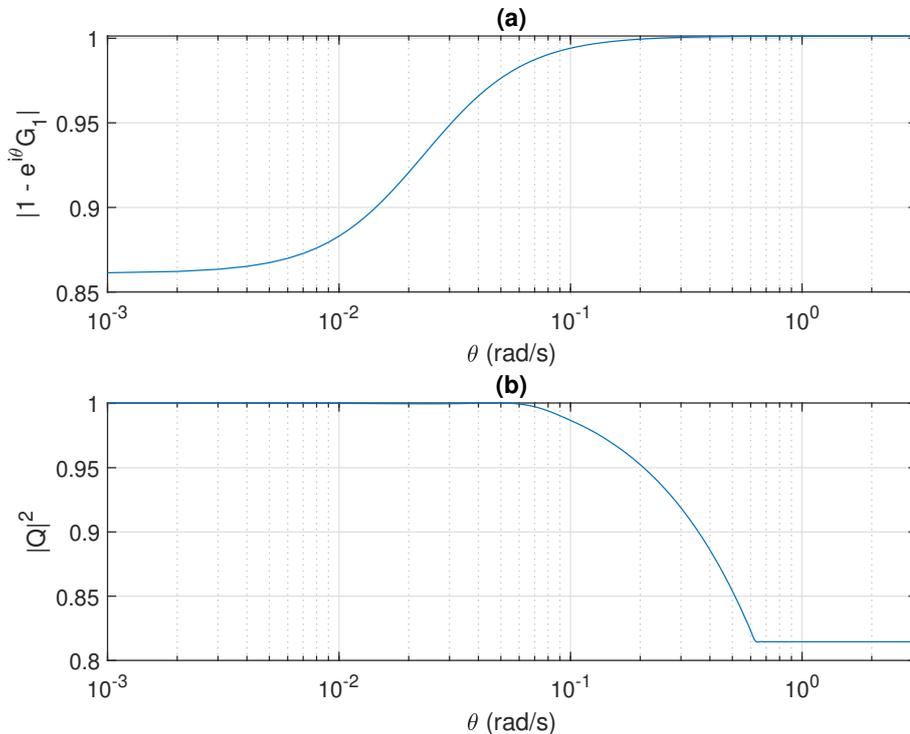


Figura 4.49 – (a) Magnitude da resposta em frequência de $1 - e^{i\theta}\bar{G}_{1000}(e^{i\theta})$, (b) quadrado da magnitude de $Q(e^{i\theta})$

Os resultados e erros RMS tanto para o modelo linear quanto para o sistema original linearizado são mostrados nas Figuras 4.51 e 4.52. O que mais vale notar, é como o desempenho

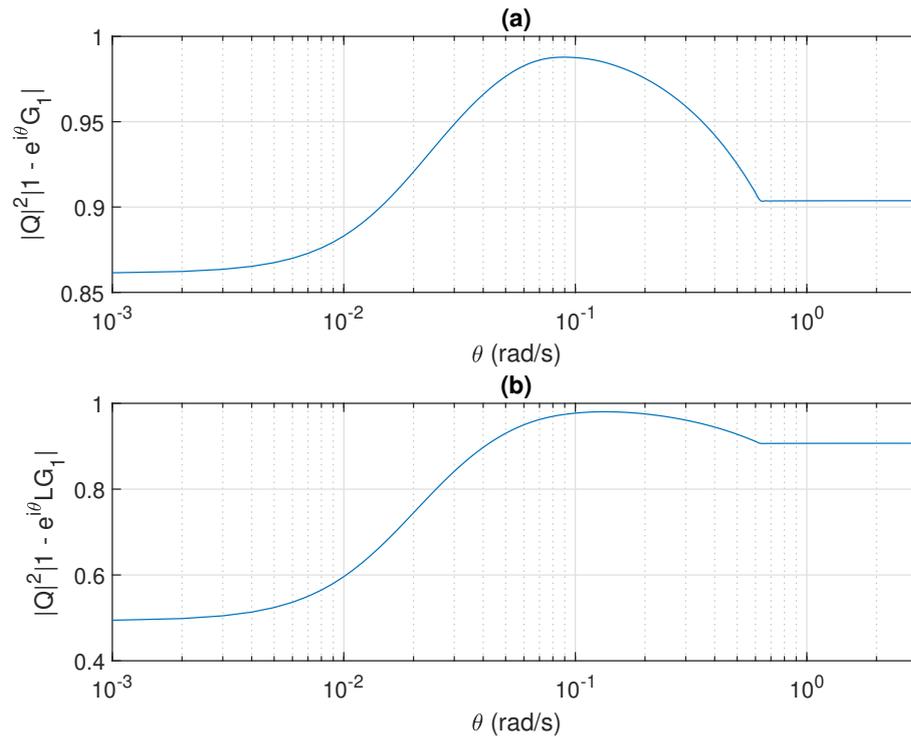


Figura 4.50 – (a) Magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{1,1000}(e^{i\theta})$ após filtragem, (b) magnitude da resposta em frequência de $1 - e^{i\theta} L(e^{i\theta}) \bar{G}_{1,1000}(e^{i\theta})$ após filtragem

do algoritmo ILC com filtro-Q piorou, ainda que, intuitivamente, o esperado era que o aumento da taxa de amostragem melhorasse o rastreamento. Mesmo no caso linear o erro não convergiu para zero, o que era esperado teoricamente para $Q(q) \neq 1$.

O fato da filtragem de fase-zero ser não causal causa a aparente divergência no rastreamento da vazão, antes da quina em 0,75 s. Esse fenômeno também ocorre nas proximidades de 0,5 s, embora isso só seja visível para a resposta linear, já que no modelo original a válvula S2 está fechada durante o período expiratório.

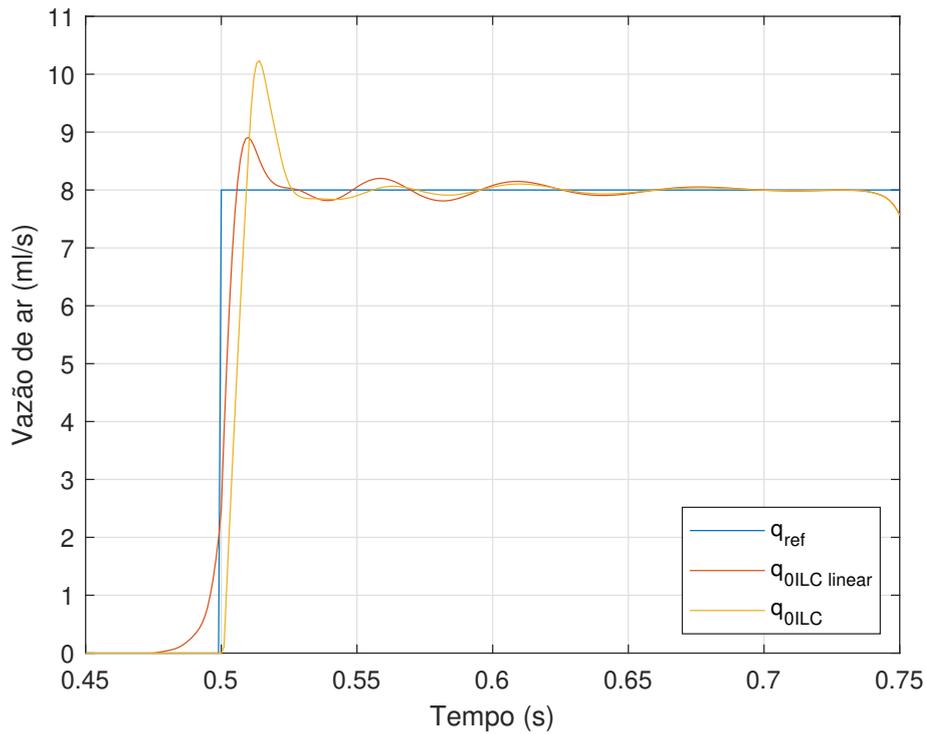


Figura 4.51 – Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 1000$ Hz

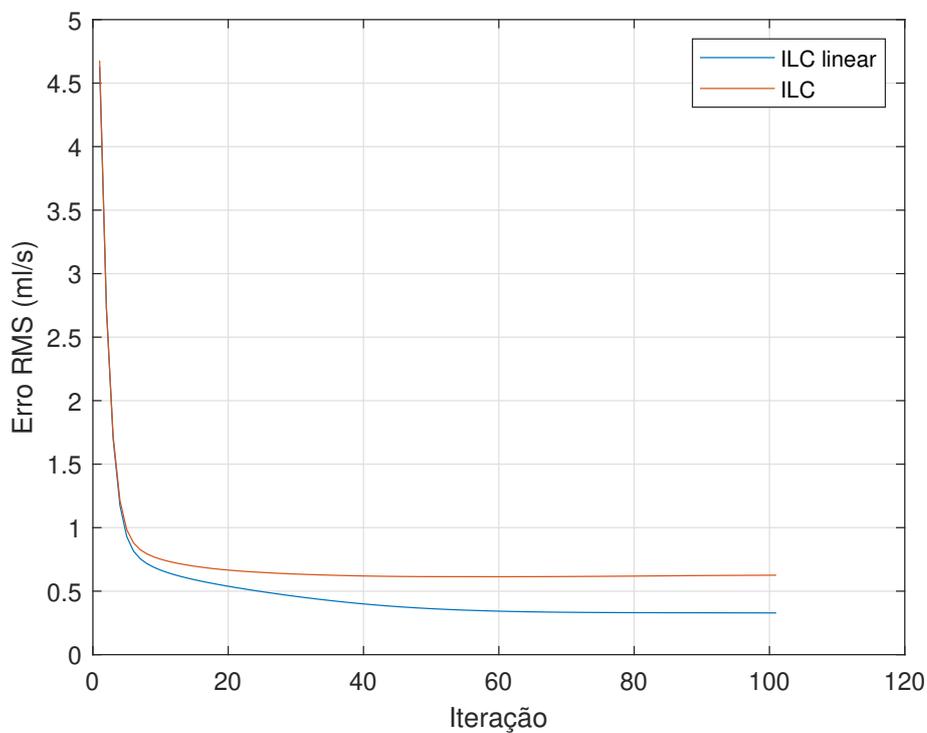


Figura 4.52 – Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 1000$ Hz

4.4.6.2 Modo controlado a pressão, $F_s = 1000$ Hz

A função de transferência obtida a partir da discretização de (4.3b) com $F_s = 1000$ Hz é

$$\bar{G}_{2_{1000}}(z) = z^{-1} \frac{1,2655 \times 10^{-5}(z + 9,05)(z - 0,9653)}{(z - 1)(z - 0,9761)}.$$

Tanto para $L(q)$ proporcional como para o derivativo, não conseguimos garantir convergência monotônica para essa função. Considere

$$\bar{G}_{2d_{1000}}(z) = z^{-2} \frac{1,2655 \times 10^{-5}(z + 9,05)(z - 0,9653)}{z - 0,9761}$$

como o modelo linear do sistema para $L(q)$ derivativo.

Podemos aplicar então o mesmo procedimento de projeto de filtro-Q detalhado na seção anterior. A Figura 4.54a mostra o resultado da aplicação do filtro projetado na Figura 4.53b à magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{2d_{1000}}(e^{i\theta})$, mostrada na Figura 4.53a. Satisfizemos a condição de convergência monotônica com $L(q)$ derivativo, com $K_{\text{dILCPressure}_{1000}} = 200$ e $\eta_2 = 0,9835$. A magnitude da resposta em frequência de $[1 - K_{\text{dILCPressure}_{1000}} \bar{G}_{2d_{1000}}(e^{i\theta})]$ após filtragem é exibida na Figura 4.54b.

Os resultados e erros RMS tanto para o modelo linear quanto para o sistema original linearizado são apresentados nas Figuras 4.55 e 4.56. Embora a piora do desempenho não tenha sido tão acentuada quanto no modo controlado a volume, ainda é possível perceber a dificuldade que o algoritmo com filtro-Q possui para rastrear as quinças da curva de referência.

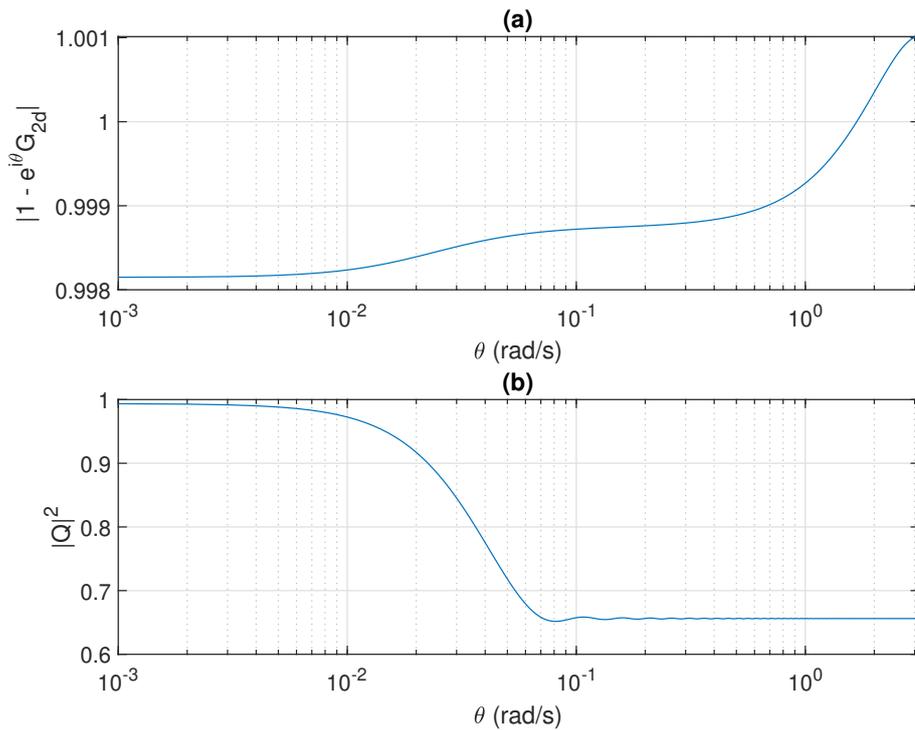


Figura 4.53 – (a) Magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{2d_{1000}}(e^{i\theta})$, (b) quadrado da magnitude de $Q(e^{i\theta})$

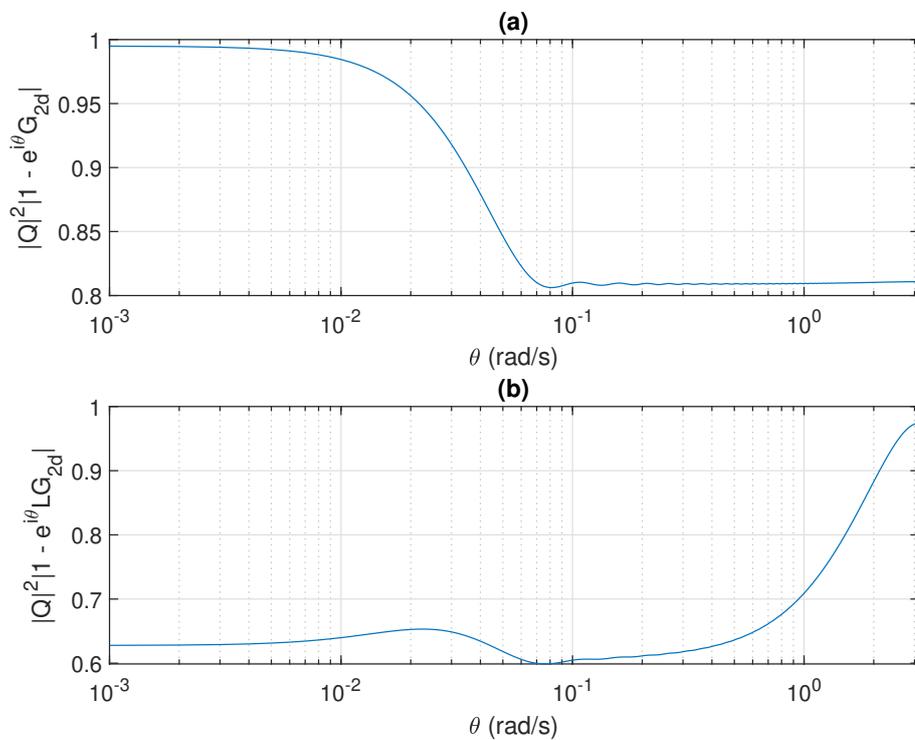


Figura 4.54 – (a) Magnitude da resposta em frequência de $1 - e^{i\theta} \bar{G}_{2d_{1000}}(e^{i\theta})$ após filtragem, (b) magnitude da resposta em frequência de $1 - e^{i\theta} L(e^{i\theta}) \bar{G}_{2d_{1000}}(e^{i\theta})$ após filtragem

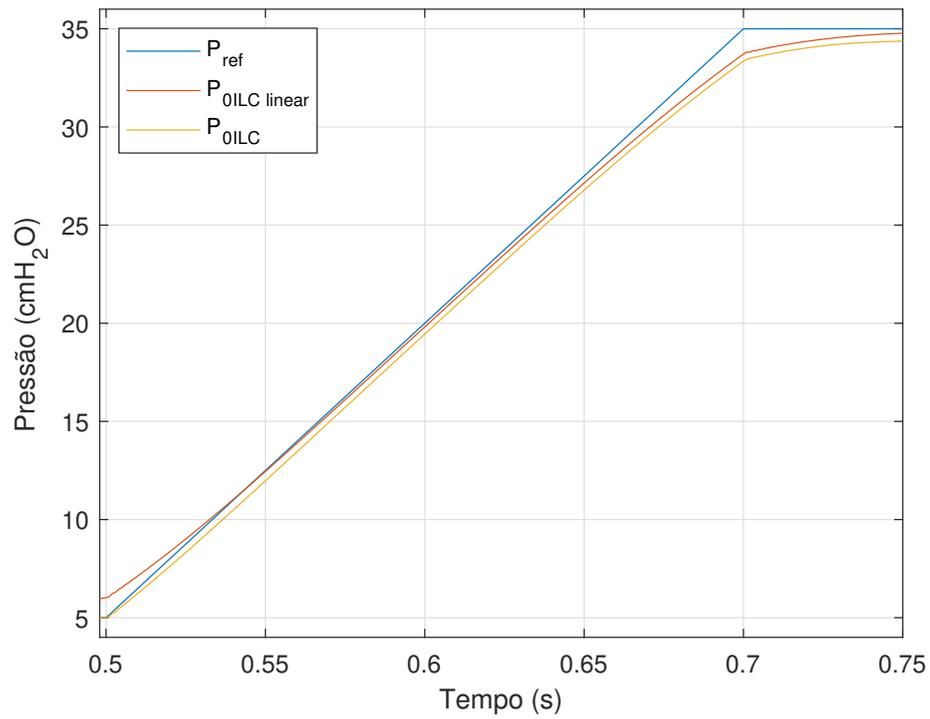


Figura 4.55 – Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 1000$ Hz

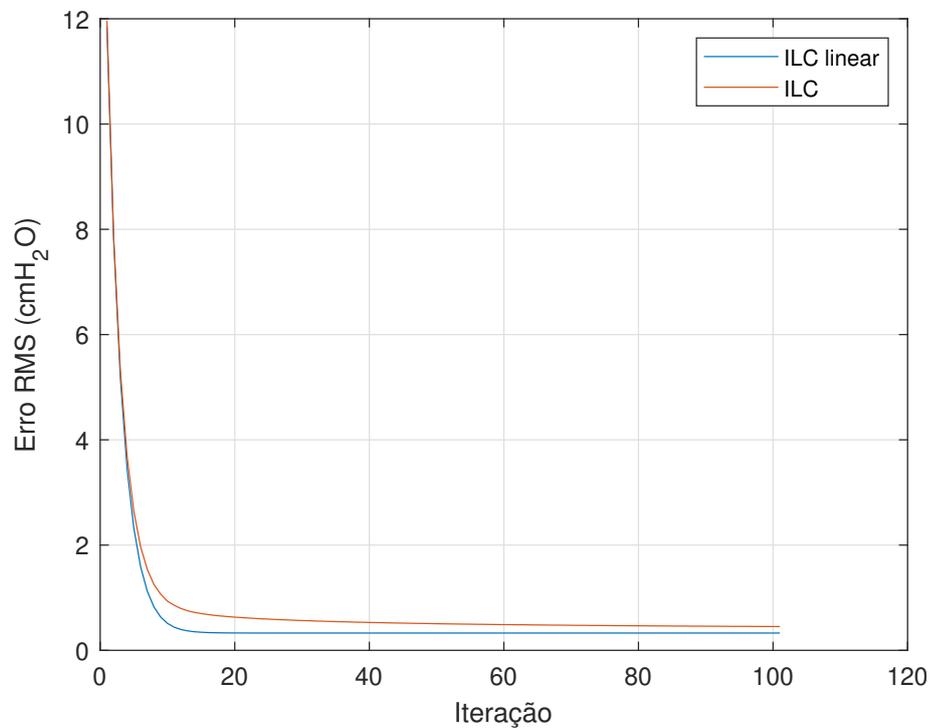


Figura 4.56 – Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 1000$ Hz

4.4.6.3 Modo controlado a volume, $F_s = 100$ Hz

$$\bar{G}_{100}(z) = z^{-1} \frac{2,7407 \times 10^{-2}(z + 0,08747)}{z - 0,7854}.$$

Para essa função conseguimos $\eta_2 = 0,8168$ com $Q(q) = 1$ e $L(q)$ proporcional, com $K_{ILCFlow100} = 13,08$.

Os resultados e erros RMS tanto para o modelo linear quanto para o sistema original linearizado são apresentados nas Figuras 4.57 e 4.58. De todas as simulações que fizemos com a arquitetura com apenas ILC para o modo controlado a volume, essa foi a que teve a taxa de convergência mais rápida. Sem o filtro-Q, ILC no modelo linear converge para erro nulo e ILC aplicado ao sistema original consegue um erro tão baixo quanto possível com as limitações na atuação.

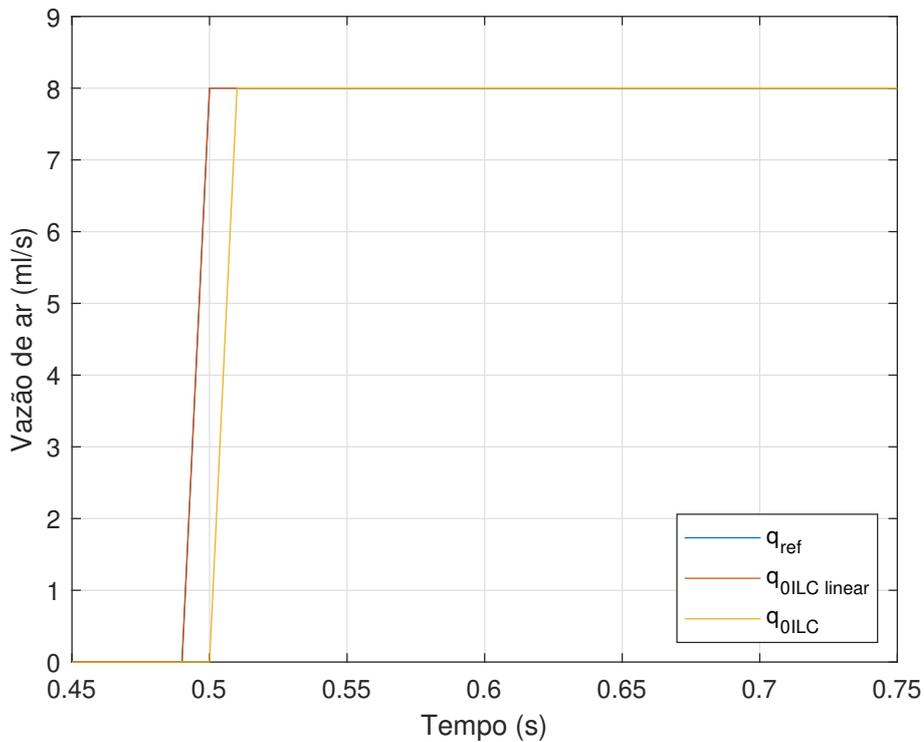


Figura 4.57 – Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 100$ Hz

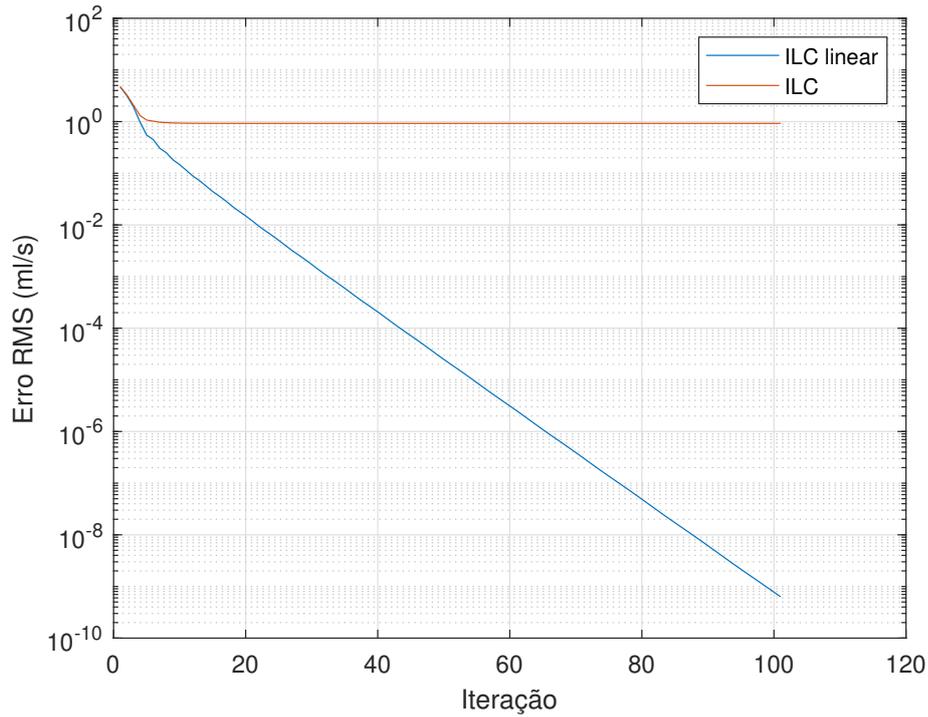


Figura 4.58 – Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a volume, $F_s = 100$ Hz

4.4.6.4 Modo controlado a pressão, $F_s = 100$ Hz

$$\bar{G}_{2_{100}}(z) = z^{-1} \frac{1,2056 \times 10^{-2}(z + 0,1051)(z - 0,7017)}{(z - 1)(z - 0,7854)}.$$

Para essa função conseguimos $\eta_2 = 0,4447$ com $Q(q) = 1$ e $L(q)$ proporcional, com $K_{\text{dILCPressure}_{100}} = 54$.

Os resultados e erros RMS tanto para o modelo linear quanto para o sistema original linearizado são apresentados nas Figuras 4.59 e 4.60. Essa também foi a simulação em que obtivemos a taxa de convergência mais rápida para a arquitetura com apenas ILC no modo controlado a pressão. A convergência do erro de ILC aplicado ao modelo linear converge para um valor virtualmente nulo de erro de rastreamento, e a sobre-elevação após a rampa ocorre devido à saturação da válvula S1.

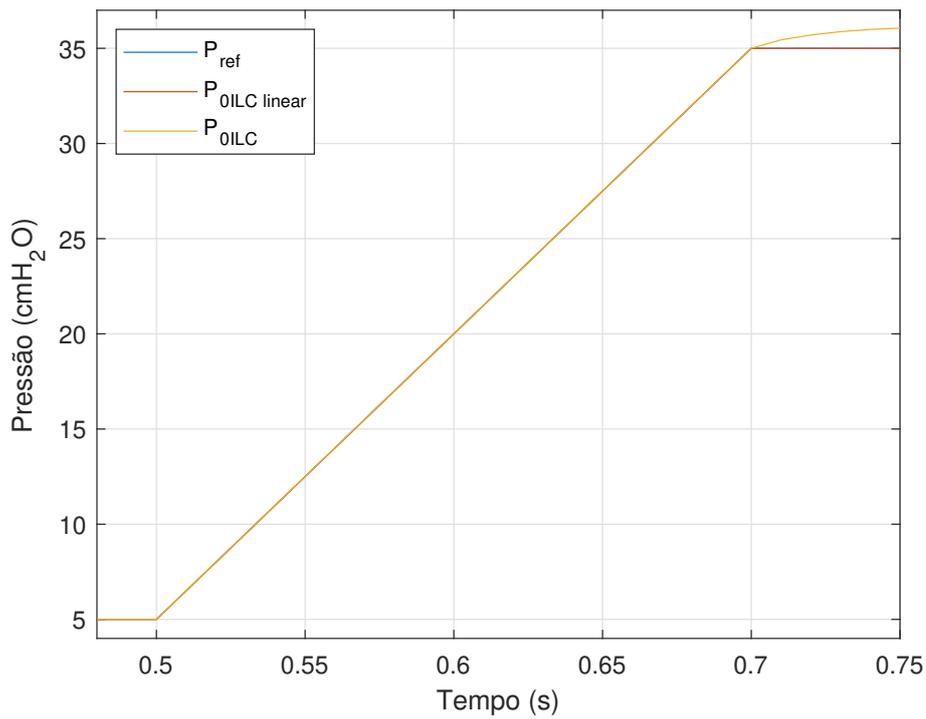


Figura 4.59 – Vazão de saída para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 100$ Hz

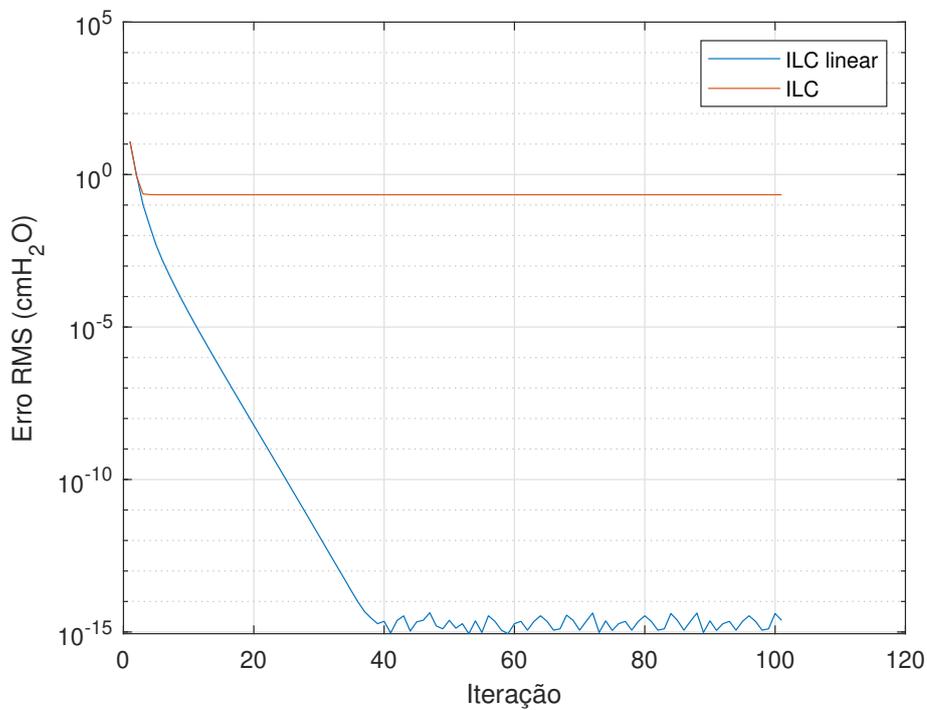


Figura 4.60 – Erro RMS do rastreamento da vazão para as simulações no modelo linear e no sistema original linearizado - Modo controlado a pressão, $F_s = 100$ Hz

5 Conclusão

5.1 Apanhado geral do que foi feito

De modo resumido, neste trabalho introduzimos o problema do ventilador mecânico e como ele se enquadra no contexto de Controle por Aprendizagem Iterativa. Descrevemos de forma sucinta a arquitetura do ventilador proposto em [2], quais são seus modos de ventilação e como o circuito pneumático pode ser modelado por um equivalente elétrico. Havendo detalhado o funcionamento do ventilador, explicamos em que consiste a técnica de ILC e as situações em que funciona bem ou não. Apresentamos então o algoritmo ILC que utilizamos nas simulações e alguns teoremas que nos permitem analisar o desempenho do sistema de controle, seguindo o *framework* conceitual proposto em [3]. Descrevemos também de forma sucinta a formulação de ILC segundo [9] e fizemos uma breve revisão do estado da arte de ILC não-linear.

Em seguida, apresentamos os principais resultados no capítulo 4. Para o procedimento de projeto ILC, assumimos algumas hipóteses que nos permitiram simplificar o circuito do ventilador e linearizá-lo por realimentação de estados. Validamos a linearização e mostramos o porquê, em geral, uma das grandes desvantagens de se utilizar a estratégia de linearização por realimentação é a sensibilidade a incertezas paramétricas. Descrevemos como foi feito o projeto do controlador PI e do algoritmo ILC considerando o modelo linear do sistema. Em seguida, mostramos os resultados das simulações que nos permitem comparar ILC com controle por realimentação ou com uma arquitetura híbrida. Analisamos também, para ambos os modos de controle da ventilação, alguns cenários onde os sistemas de controle propostos estavam expostos a perturbações periódicas e não-periódicas, ruídos na medição e estimativas incertas dos parâmetros usados na linearização por realimentação. Concluimos o capítulo de resultados fazendo uma avaliação de como variar a frequência de amostragem afeta a estratégia baseada apenas em ILC, e ilustramos um procedimento de projeto de filtro-Q.

No Capítulo 1, mencionamos o trabalho de [4] que também usou uma estratégia ILC para controlar um ventilador mecânico. Evidenciamos aqui as principais diferenças entre nosso estudo e [4]:

- Nossa lei de controle ILC é mais simples ao não incluir o termo proporcional ao quadrado do erro.
- Nosso projeto ILC está baseado em garantias formais de estabilidade assintótica e convergência para sistemas lineares conhecidos. [4] garantiu a não-divergência através da desativação de ILC caso a norma do erro cresça entre iterações.

- Nosso sistema é baseado em um ventilador com uma nova tecnologia, acionado por um circuito com válvulas solenóides *on-off* (ainda que S1 tenha sido simplificada como proporcional) e que inclui um reservatório de pressão intermediário. O ventilador descrito em [4] consiste em um soprador centrífugo diretamente ligado às vias aéreas via um tubo terapêutico.
- Nós linearizamos a planta por realimentação e então aplicamos ILC, enquanto [4] aplica ILC diretamente ao sistema não-linear.
- [4] aplicou a estratégia em uma planta experimental, com um pulmão artificial.

5.2 Principais resultados e limitações do estudo

De modo geral, usar somente ILC possui desvantagens em relação ao controle em malha fechada clássico. Ainda que o rastreamento possa ser preciso após certo número de iterações, ILC sozinho não é adequado para aplicações como ventiladores mecânicos, onde desvios do perfil de referência podem danificar o pulmão do paciente. ILC pode permitir um erro inaceitável durante as primeiras execuções e nos deixa sem a robustez do controle em malha fechada.

Apesar disso, a estratégia de ILC combinado com controle em malha fechada apresentou resultados melhores que o PI sozinho em quase todos os casos testados. Se houver garantia de convergência monotônica, essa arquitetura tem o desempenho do PI como ponto de partida, e a cada iteração, o erro tende a diminuir. Quando não há convergência monotônica, o erro pode apresentar um crescimento transiente antes de convergir, de ordens acima do erro inicial. Do ponto de vista prático, ainda que a convergência monotônica possa ser projetada de forma mais robusta (por exemplo, com $Q(q) \neq 1$), seria interessante adicionar um mecanismo de segurança no controle do sistema físico. Poderia ser algo tão simples como desativar o ILC caso $e_{j+1} > e_j$, como feito em [4].

As propriedades de estabilidade assintótica e convergência monotônica, garantidas teoricamente para os modelos lineares, puderam ser estendidas com robustez ao sistema não-linear, mesmo com incertezas nas estimativas dos parâmetros usados na linearização por realimentação ou ruído na medição. As restrições físicas do atuador fazem com que a propriedade de erro de rastreamento assintótico nulo seja aplicável somente para os casos lineares.

A arquitetura com ILC combinado com o PI apresentou bom desempenho na rejeição de perturbações periódicas, conseguindo usar o conhecimento das iterações passadas para atuar com antecedência. Para perturbações não-periódicas, a performance dessa arquitetura é reduzida, porém não a ponto de que seu uso seja inviabilizado. A estratégia com ILC se mostrou tão robusta à existência de ruído na medição quanto o controlador PI.

Em relação à variação na frequência de amostragem, é interessante que quanto menor foi a frequência que utilizamos, mais rápida foi a taxa de convergência obtida para o erro de

rastreamento. Em [2], o melhor desempenho para o sistema amostrado foi com $F_s = 1000$ Hz, e o uso de $F_s = 200$ Hz produziu resultados inaceitáveis nas configurações testadas. Além disso, para as os perfis de referência em degrau e em rampa utilizados, a robustez adicional proveniente do uso de um filtro-Q não compensa o aumento do erro de rastreamento assintótico. Ainda assim, o filtro-Q é melhor que a alternativa de existir a possibilidade de que o erro atinja grandes valores antes de convergir, caso não haja garantia de convergência monotônica.

Uma limitação dos resultados obtidos está nos modelos adotados para as resistências pneumáticas e para a atuação da válvula solenóide. Vale enfatizar novamente que a queda de pressão em uma resistência pneumática é não-linear e varia no tempo. Do mesmo modo, apesar de existirem técnicas que teoricamente permitiriam considerar uma válvula *on-off* como uma resistência continuamente variável, essas exigem uma elevada taxa de comutações por ciclo respiratório. O uso de PWM para modelar características proporcionais a partir da válvula solenóide foi, inclusive, considerado em [2] mas substituído pelo controle por modos deslizantes. O ventilador em questão é projetado para trabalhar com ciclos inspiratórios da ordem de 100 ms, e isso exige o uso de válvulas com tempo de resposta suficientemente baixos. É possível que válvulas comerciais com o tempo de resposta requerido para usarmos PWM não sejam tecnicamente ou financeiramente viáveis.

Com a estratégia de controle por modos deslizantes em [2], no modo controlado a volume ainda ocorre uma oscilação de 1 ml/s no rastreamento do degrau em regime permanente, mesmo nos casos com melhor desempenho. Esse resultado sugere, a princípio, que ainda que não haja válvulas solenóides factíveis com os requisitos para o uso de PWM, poderia ser interessante o uso de válvulas proporcionais controladas por um sinal analógico. Apesar disso, o modelo de ventilador proposto em [2] tinha como principais motivações a simplicidade, versatilidade e baixo custo. A possível melhora no desempenho não justificaria o maior custo, porte e complexidade da arquitetura com a adição de válvulas servo-controladas.

Vale notar ainda que os resultados obtidos em [2] são considerados satisfatórios. ILC é uma técnica que permite rastrear referências com alta precisão e, por mais que esse tipo de performance seja desejável, pequenas oscilações não farão com que a ventilação do paciente seja prejudicada.

5.3 Propostas de continuidade

Propostas de continuidade deste trabalho incluem investigar o desempenho de ILC para modelos mais precisos dos componentes do sistema, considerando tanto a natureza *on-off* da válvula solenóide S1 quanto os aspectos não-lineares das resistências pneumáticas. Como forma de evitar a simplificação da válvula baseada no uso de PWM com alta frequência de chaveamento, uma ideia é simular uma arquitetura com uma malha de controle externa por ILC + PI e uma malha interna com o processo e controle robusto por modos deslizantes. Um dos desafios dessa

abordagem seria a obtenção de um modelo linear aproximado para a malha interna.

Seria relevante também avaliar o desempenho de outros algoritmos, ou até paradigmas, como ILC não-linear, o paradigma de otimização de [5], ILC baseado em inversão da planta e o uso de $L(q)$ dinâmico e não-causal [10]. Além disso, pode-se considerar uma análise mais aprofundada da robustez dos algoritmos ILC.

Referências

- [1] Bornachi, Artur Fonseca, Leonardo Antônio Borges Tôres e Alessandro Beda: *Controle de um sistema de ventilação mecânica para pequenos animais*. Em *Simpósio Brasileiro de Automação Inteligente*, 2015. Citado na página 21.
- [2] Bornachi, Artur Fonseca: *Proposta de uma nova arquitetura de ventilador pulmonar mecânico para pequenos animais*. Tese de Mestrado, Universidade Federal de Minas Gerais, 2016. Citado 12 vezes nas páginas 21, 22, 23, 24, 27, 45, 51, 57, 63, 69, 91 e 93.
- [3] Bristow, Douglas A, Marina Tharayil e Andrew G Alleyne: *A survey of iterative learning control*. *IEEE Control Systems*, 26(3):96–114, 2006. Citado 14 vezes nas páginas 21, 29, 31, 32, 34, 35, 36, 37, 38, 39, 40, 41, 42 e 91.
- [4] Scheel, Mathias, Andreas Berndt e Olaf Simanski: *Iterative Learning Control: An Example for Mechanical Ventilated Patients*. Em *9th IFAC Symposium on Biological and Medical Systems BMS 2015, 31 August-2 September 2015*, volume 48, páginas 523–527, agosto 2015. Citado 3 vezes nas páginas 21, 91 e 92.
- [5] Owens, David H: *Iterative learning control: an optimization paradigm*. Springer, 2015. Citado 4 vezes nas páginas 21, 29, 44 e 94.
- [6] Neto, Alcendino Cândido Jardim: *PEEP-válvula versus PEEP selo d'água: mecânica ventilatória de ratos saudáveis*. Tese de Mestrado, Universidade Federal do Rio de Janeiro, 2011. Citado na página 22.
- [7] Castro, Adler Fonseca de e Leonardo Antônio Borges Tôres: *Iterative Learning Control Applied to a Recently Proposed Mechanical Ventilator Topology*. Em *12th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems, DYCOPS*, 2019. Citado na página 21.
- [8] West, John Burnard: *Respiratory physiology: the essentials*. Lippincott Williams & Wilkins, 2012. Citado 2 vezes nas páginas 23 e 24.
- [9] Moore, Kevin L: *Iterative learning control for deterministic systems*. Springer Science & Business Media, 1993. Citado 8 vezes nas páginas 29, 31, 34, 35, 37, 41, 42 e 91.
- [10] Ahn, Hyo Sung, Kevin L Moore e YangQuan Chen: *Iterative learning control: robustness and monotonic convergence for interval systems*. Springer Science & Business Media, 2007. Citado 3 vezes nas páginas 31, 44 e 94.
- [11] Chen, Chi Tsong: *Linear system theory and design*. Oxford University Press, Inc., 1998. Citado na página 34.

- [12] Norrlöf, Mikael e Svante Gunnarsson: *Time and frequency domain convergence properties in iterative learning control*. International Journal of Control, 75(14):1114–1126, 2002. Citado na página 37.
- [13] De Roover, Dick e Okko H Bosgra: *Synthesis of robust multivariable iterative learning controllers with application to a wafer stage motion system*. international Journal of Control, 73(10):968–979, 2000. Citado na página 37.
- [14] Marino, Riccardo e Patrizio Tomei: *An iterative learning control for a class of partially feedback linearizable systems*. IEEE Transactions on Automatic Control, 54(8):1991–1996, 2009. Citado na página 43.
- [15] Chen, Pengnian e Xiangbin Liu: *Repetitive learning control for a class of partially linearizable uncertain nonlinear systems*. Automatica, 85:397–404, 2017. Citado na página 43.
- [16] He, Wei, Tingting Meng, Deqing Huang e Xuefang Li: *Adaptive boundary iterative learning control for an Euler–Bernoulli beam system with input constraint*. IEEE transactions on neural networks and learning systems, 29(5):1539–1549, 2018. Citado na página 43.
- [17] Lu, Jingyi, Zhixing Cao, Ridong Zhang e Furong Gao: *Nonlinear Monotonically Convergent Iterative Learning Control for Batch Processes*. IEEE Trans. Industrial Electronics, 65(7):5826–5836, 2018. Citado na página 43.
- [18] *Documentação de filtfilt*. <https://www.mathworks.com/help/signal/ref/filtfilt.html>. Acessado em 10 de julho de 2019. Citado na página 81.

Apêndices

APÊNDICE A – Parâmetros de simulação

Tabela A.1 – Parâmetros de simulação

Parâmetro	Valor	Unidade
P_1	2039,2	cmH ₂ O*
R_1	6,4516	cmH ₂ O/ml/s
C_2	0,465	ml/cmH ₂ O
R_c	0,377	cmH ₂ O/ml/s
C_c	0,075	ml/cmH ₂ O
$Rptc_{insp}$	$0,7R_c$	cmH ₂ O/ml/s
$Rptc_{exp}$	$0,2R_c$	cmH ₂ O/ml/s
Rv_{1min}	$1,062 \times 10^{-4}$	cmH ₂ O/ml/s
Rv_{1max}	5000	cmH ₂ O/ml/s
Rv_{2min}	$1,062 \times 10^{-4}$	cmH ₂ O/ml/s
Rv_{2max}	5000	cmH ₂ O/ml/s
VT	2	ml
T_{breath}	0,75	s
P_{PEEP}	5	cmH ₂ O
t_{resp}	2×10^{-3}	s
T_{insp}	$T_{breath}/3$	s
T_{exp}	$2T_{breath}/3$	s
$\gamma (q_{ref})$	8	ml/s
$\beta (P_{ref})$	150	cmH ₂ O/s
P_{max}	35	cmH ₂ O
F_s	500	Hz
K_{pFlow}	60,335	ml/s/ml/s
K_{iFlow}	3,0571	ml/s/ml
$K_{pPressure}$	123,9	ml/s/ml/s
$K_{iPressure}$	0,6814	ml/s/ml
$K_{ILCFlow}$	14	ml/s/ml/s
$K_{PIILCFlow}$	62,95	ml/s/ml/s
$K_{dILCPressure}$	270	ml/s/ml/s
$K_{PIILCPressure}$	17,35	ml/s/ml/s
N	376	
$P_{leakage}$	30	cmH ₂ O
$R_{leakage}$	$2R_1$	cmH ₂ O/ml/s
TEB_{Flow}	0,1658	ml/s
$TEB_{Pressure}$	2,248	cmH ₂ O

* 1 cmH₂O = 98,0665 Pa.