



JOAO VIVAS CISALPINO

**CONTROLE DE TRAJETÓRIA DE IMPRESSORAS 3D
UTILIZANDO ALGORITMO ITERATIVO E PROGRAMAÇÃO
NÃO LINEAR**

LAVRAS - MG

2023

JOAO VIVAS CISALPINO

**CONTROLE DE TRAJETÓRIA DE IMPRESSORAS 3D UTILIZANDO
ALGORITMO ITERATIVO E PROGRAMAÇÃO NÃO LINEAR**

Monografia apresentada à Universidade Federal
de Lavras, como parte das exigências Curso de
Engenharia Mecânica, para a obtenção do título
de Bacharel.

Prof. Dr. Wander Gustavo Rocha Vieira
Orientador

LAVRAS - MG

2023

JOAO VIVAS CISALPINO

**CONTROLE DE TRAJETÓRIA DE IMPRESSORAS 3D UTILIZANDO
ALGORITMO ITERATIVO E PROGRAMAÇÃO NÃO LINEAR**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências Curso de Engenharia Mecânica, para a obtenção do título de Bacharel.

APROVADO em 21 de Novembro de 2023.

Prof. Dr. Henrique UFLA

Prof. Dr. Belisario UFLA

Prof. Dr. Wander Gustavo Rocha Vieira
Orientador

**LAVRAS - MG
2023**

RESUMO

A manufatura aditiva, com ênfase na impressão 3D e no método "Fused Deposition Modeling"(FDM), destaca-se como uma tecnologia altamente promissora para a produção de peças complexas em quantidades reduzidas. Ela impulsiona a iterabilidade e a produção descentralizada sob demanda, encontrando aplicação em diversos setores, como engenharia, medicina e a indústria aeroespacial. O objetivo principal do trabalho é investigar e desenvolver uma metodologia para atuação de controle na geração de comandos em impressoras 3D de forma a possibilitar maiores velocidades e garantindo a precisão dimensional das peças produzidas. O desenvolvimento desse controle incorpora um algoritmo iterativo que minimiza os desvios entre o trajeto desejado para a impressora e o trajeto efetivamente percorrido, levando em consideração a dinâmica da impressora. Isso resulta em peças impressas de maior qualidade, permitindo aos usuários selecionar velocidades de impressão mais elevadas sem comprometer a qualidade do produto final, em comparação com a ausência desse método. O algoritmo foi submetido a testes de sensibilidade de parâmetros para avaliar seu desempenho sob diferentes condições. Os resultados desses testes foram satisfatórios e indicaram a capacidade do método de minimizar os desvios da impressora, com base em uma modelagem precisa. Além disso, o método apresenta potencial para expansão, possibilitando a consideração de nuances do modelo em várias regiões da impressora, aprimorando ainda mais sua eficácia.

Palavras-chave: Manufatura aditiva Modelagem por Fused Deposition Modeling FDM Geração de Trajetória Impressão 3D Algoritmos de controle Modelagem dinâmica

ABSTRACT

Additive manufacturing, with an emphasis on 3D printing and the "Fused Deposition Modeling"(FDM) method, stands out as a highly promising technology for the production of complex parts in reduced quantities. It drives iterability and decentralized on-demand production, finding applications in various sectors, such as engineering, medicine, and the aerospace industry. The primary goal of this work is to investigate and develop a methodology for control action in generating commands in 3D printers to enable higher speeds while ensuring the dimensional accuracy of the produced parts. The development of this control incorporates an iterative algorithm that minimizes deviations between the desired path for the printer and the path actually traveled, taking into account the printer's dynamics. This results in higher quality printed parts, allowing users to select higher printing speeds without compromising the quality of the final product compared to the absence of this method. The algorithm underwent parameter sensitivity tests to assess its performance under different conditions. The results of these tests were satisfactory and indicated the method's capability to minimize printer deviations based on accurate modeling. Furthermore, the method has potential for expansion, allowing for the consideration of nuances in the model in various printer regions, further enhancing its effectiveness.

Keywords: Additive manufacturing Fused Deposition Modeling (FDM) Trajectory generation 3D printing Control algorithms Dynamic modeling

LISTA DE FIGURAS

Figura 1.1 – Fluxograma etapas Impressora 3D FDM	10
Figura 2.1 – Distribuição de uso de MA nas industrias	12
Figura 2.2 – Princípio e processo de impressão para FDM	13
Figura 2.3 – Indicação dos componentes de uma impressora 3D	14
Figura 2.4 – Interface do fatiador PrusaSlicer	15
Figura 2.5 – Exemplo de um arquivo Gcode	17
Figura 2.6 – Perfil de velocidade - Curva trapezoidal de velocidade	18
Figura 2.7 – Comparação da resposta ao degrau e da resposta a escada	20
Figura 2.8 – Ilustração Segmentação Cúbica	21
Figura 3.1 – Fluxograma geral das etapas para o controle de trajetória	23
Figura 3.2 – Curva de velocidade trapezoidal	24
Figura 3.3 – Curva de velocidade trapezoidal	25
Figura 3.4 – Curva de velocidade triangular	26
Figura 3.5 – Modelo simplificado impressora 3D	29
Figura 3.6 – Modelagem de 1 eixo	29
Figura 3.7 – Modelagem dos eixos x e y	30
Figura 3.8 – Fluxograma Controle de Trajetória	32
Figura 3.9 – Movimento base	33
Figura 3.10 – Fluxograma geral com os parâmetros.	34
Figura 4.1 – Caso referência - Comportamento no tempo das velocidades em x e y da ponta e da base	37
Figura 4.2 – Caso referência - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	38
Figura 4.3 – Caso referência - Caminho percorrido x vs y da ponta e da base	38
Figura 4.4 – Caso referência - Num de fun x Viabilidade	39
Figura 4.5 – Caso 1A - Comportamento no tempo das velocidades em x e y da ponta e da base	40
Figura 4.6 – Caso 1B - Comportamento no tempo das velocidades em x e y da ponta e da base	40
Figura 4.7 – Caso 1C - Comportamento no tempo das velocidades em x e y da ponta e da base	41

Figura 4.8 – Caso 1A - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	41
Figura 4.9 – Caso 1C - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	42
Figura 4.10 – Caso 1A - Caminho percorrido x vs y da ponta e da base	42
Figura 4.11 – Caso 1C - Caminho percorrido x vs y da ponta e da base	43
Figura 4.12 – Caso 1 - Num de fun x Viabilidade	43
Figura 4.13 – Caso 2B - Caminho percorrido x vs y da ponta e da base	44
Figura 4.14 – Caso 2C - Caminho percorrido x vs y da ponta e da base	45
Figura 4.15 – Caso 2 - Num de fun x Viabilidade	45
Figura 4.16 – Caso 3A - Comportamento no tempo das velocidades em x e y da ponta e da base	46
Figura 4.17 – Caso 3 - Num de fun x Viabilidade	47
Figura 4.18 – Caso 4A - Comportamento no tempo das velocidades em x e y da ponta e da base	48
Figura 4.19 – Caso 4B - Comportamento no tempo das velocidades em x e y da ponta e da base	48
Figura 4.20 – Caso 4C - Comportamento no tempo das velocidades em x e y da ponta e da base	49
Figura 4.21 – Caso 4A - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	49
Figura 4.22 – Caso 4B - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	50
Figura 4.23 – Caso 4C - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	50
Figura 4.24 – Caso 4 - Num de fun x Viabilidade	51
Figura 4.25 – Caso 5B - Comportamento no tempo das velocidades em x e y da ponta e da base	52
Figura 4.26 – Caso 5B - Comportamento no tempo dos deslocamentos em x e y da ponta e da base	52
Figura 4.27 – Caso 5 - Num de fun x Viabilidade	53

LISTA DE TABELAS

Tabela 3.1 – Critérios de Parada do Algoritmo de Otimização	33
Tabela 3.2 – Valores dos parâmetros utilizados na simulação.	34
Tabela 3.3 – Parâmetros utilizados na análise de sensibilidade.	35
Tabela 3.4 – Especificações do computador	35

SUMÁRIO

1	INTRODUÇÃO	9
2	REFERENCIAL TEÓRICO	11
2.1	Manufatura Aditiva	11
2.2	<i>Fused Deposition Modeling</i>	12
2.2.1	Geração de Comando	16
2.2.2	Geração de trajetória	17
2.2.3	Curvas de velocidade trapezoidal	17
2.2.4	Técnicas de Controle	18
2.2.5	<i>Feedforward</i>	18
2.2.5.1	<i>Input Shaper</i>	19
2.3	Espaço de Estados	20
2.4	Programação Não Linear	20
3	METODOLOGIA	23
3.1	Geração de Trajetória	23
3.1.1	Curva trapezoidal de velocidade	24
3.1.2	Interpolação	27
3.2	Modelagem dinâmica de uma impressora 3D	27
3.2.1	Espaço de estados	30
3.3	Controle de Trajetória	31
3.3.1	Restrições	32
3.3.2	Função de Otimização	33
3.4	Simulação Computacional e Análise de Dados	33
3.4.1	Análise de Sensibilidade	34
4	RESULTADOS E DISCUSSÃO	36
4.1	Resultados da Simulação Referência	36
4.2	Análise de Sensibilidade dos Resultados	39
4.2.1	Caso 1 - Variação da frequência	39
4.2.2	Caso 2 - Variação do coeficiente de amortecimento	44
4.2.3	Caso 3 - Variação na aceleração	46
4.2.4	Caso 4 - Variação dos passos de tempo	47
4.2.5	Caso 5 - Variação da velocidade	51

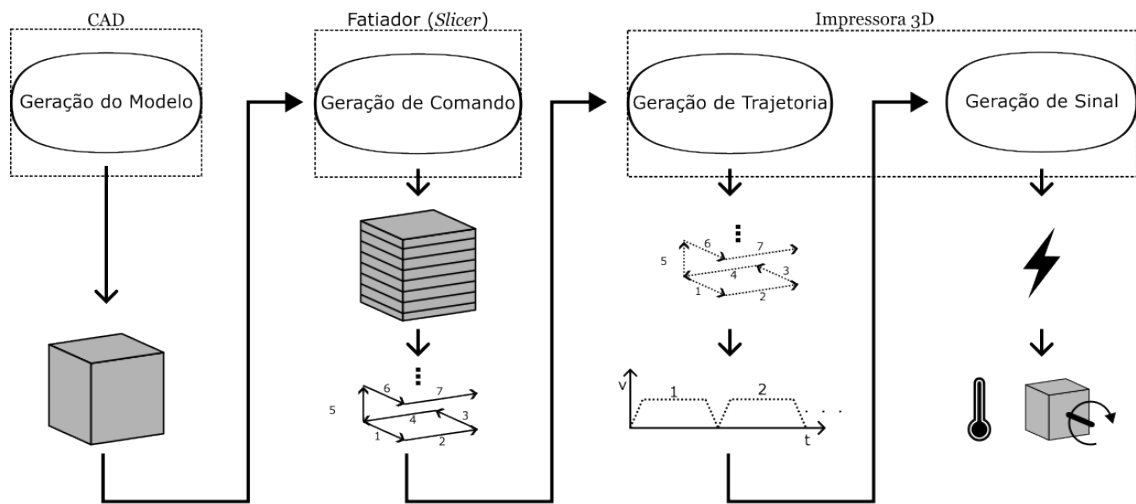
4.3	Discussão Integrada dos Resultados	53
4.3.1	Influência dos Parâmetros Variáveis	53
4.3.2	Comportamento e Convergência do Sistema	54
4.3.3	Considerações futuras	54
5	CONCLUSÃO	55
	REFERÊNCIAS	56

1 INTRODUÇÃO

A manufatura aditiva emerge como uma tecnologia altamente promissora para a produção de peças e componentes em diversas áreas, incluindo engenharia, medicina e a indústria aeroespacial. Suas características distintivas viabilizam a fabricação de peças complexas em pequenas quantidades, promovendo uma notável iterabilidade, bem como suportando a produção descentralizada sob demanda. Nesse contexto, a impressão 3D, em particular o método de *Fused Deposition Modeling* (FDM), ganha destaque crescente, encontrando aplicações variadas nos setores aeroespacial, automobilístico e prototipagem rápida, ao mesmo tempo que se torna mais acessível e disseminada.

A modelagem digital desempenha um papel essencial no processo de impressão 3D, trabalhando em conjunto com ferramentas como o *Computer Aided Design* (CAD). Essa parceria possibilita a criação de modelos tridimensionais altamente precisos, que podem ser compartilhados e reproduzidos de forma descentralizada. Quando se trata de imprimir um desses modelos, a preparação é realizada através de um software de fatiamento, conhecido como *slicer*. O *slicer* divide o modelo em camadas e gera os comandos necessários para a impressora 3D, em geral esses comandos são representados pelo Gcode nas impressoras FDM. A impressora, por sua vez, interpreta esses comandos para determinar como proceder e quando executar cada ação, por fim atuando os motores e outros dispositivos, como resistências elétricas para o aquecimento. Na Figura 1.1 esse processo é exemplificado para uma impressora FDM através de um fluxograma. É importante notar que entre a interpretação e a execução desses comandos existem diversos processos intermediários que exercem influência direta sobre a qualidade e a velocidade da impressão, sendo um deles a geração de trajetória, onde é construído o comportamento ao longo do tempo a ser executado pelos atuadores e dispositivos, por exemplo movimentos e mudanças de temperatura.

Figura 1.1 – Fluxograma etapas Impressora 3D FDM



No entanto, uma das limitações significativas da impressão 3D, especialmente do tipo FDM, é o tempo de impressão, que ainda restringe o tamanho das peças produzidas em um período razoável. Frequentemente, é necessário utilizar camadas e linhas mais grossas para compensar esse aspecto, diminuindo a habilidade de se reproduzir detalhes menores. Diante disso, existe uma procura por maneiras de se imprimir mais rapidamente, sem comprometer a qualidade.

Assim, é relevante explorar técnicas que permitam alcançar capacidades superiores de qualidade e velocidade de impressão, flexibilizando a tecnologia e ampliando sua aplicação comercial viável.

Este trabalho tem como objetivo geral investigar e desenvolver uma metodologia para atuação de controle na geração de comandos em impressoras 3D utilizando o método FDM de forma a possibilitar maiores velocidades e garantindo a precisão dimensional das peças produzidas. E os seguintes objetivos específicos:

- Desenvolver um algoritmo iterativo que possa ser integrado ao sistema de controle de impressoras 3D para minimizar os desvios entre o percurso desejado e o efetivamente percorrido, levando em consideração a dinâmica da impressora.
- Simular o comportamento da impressora 3D com o novo algoritmo para avaliar o comportamento do método em relação aos parâmetros controlados.

2 REFERENCIAL TEÓRICO

2.1 Manufatura Aditiva

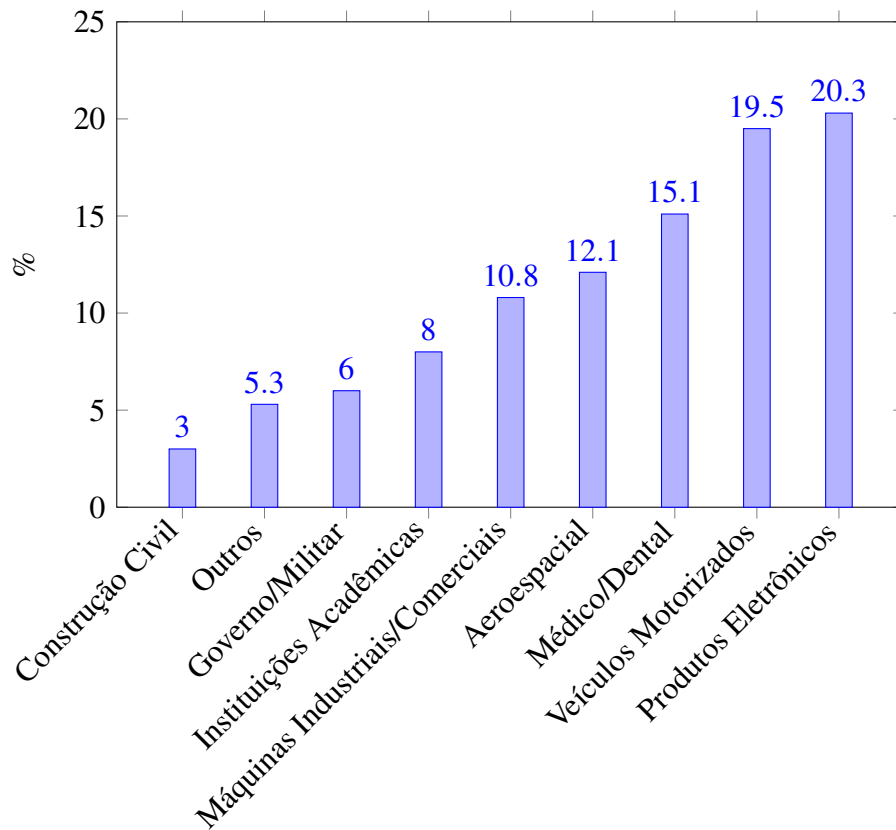
O princípio fundamental da manufatura aditiva (MA) consiste em fabricar um modelo tridimensional de forma integrada, dispensando a necessidade de planejar as operações de maneira individual. Este processo começa com um modelo tridimensional digital, frequentemente desenvolvido via *Computer Aided Design* (CAD), cujas especificações são interpretadas por um software fatiador. Este software ajusta parâmetros e gera instruções essenciais para a máquina de manufatura aditiva fabricar o modelo físico. Estas instruções, variando conforme a tecnologia e o modelo específico da impressora, são comumente transmitidas através do Gcode. O Gcode é uma linguagem de programação que direciona a impressora 3D sobre como construir o objeto, incluindo informações sobre movimentos, velocidades e temperaturas. Além do Gcode, informações adicionais como arquivos de imagens podem ser utilizadas para complementar o processo.

Uma das características principais da MA é a rapidez na qual é possível criar protótipos diretamente de modelos digitais, por conta disso, em um contexto de desenvolvimento de produto, o termo prototipagem rápida era utilizado. Entretanto, conforme a MA foi se aperfeiçoando era perceptível a capacidade dessas tecnologias não só se aterem à produção de protótipos, mas também de peças utilizadas em produtos finais. Além disso, o termo prototipagem rápida não considerava o princípio básico que unia essas tecnologias e assim o termo manufatura aditiva foi apresentado e adotado pela *American Society for Testing and Materials* (ASTM) (GIBSON *et al.*, 2015).

Atualmente, existe uma grande variedade de tecnologias e processos de manufatura aditiva. Os métodos de impressão 3D variam na maneira como depositam o material, como extrusão, sinterização a laser e estereolitografia. Eles também diferem nos princípios físicos que utilizam, como fusão, cura por luz e aglutinação. Além disso, os materiais que podem ser utilizados incluem plásticos, resinas, metais e cerâmicas. Como mencionado anteriormente, um dos métodos de manufatura aditiva mais populares é a tecnologia FDM, entretanto existem diversas outras tecnologias que tem crescido muito em popularidade como as tecnologias baseadas na cura seletiva de resinas, *stereolithography* (SLA) e *Masked stereolithography Apparatus* (MSLA), além de outras tecnologias menos acessíveis, mas com aplicações em diversas indústrias, como por exemplo *selective laser melting* (SLM) e *Selective laser Sintering* (SLS)

(BIKAS; STAVROPOULOS; CHRYSSOLOURIS, 2016). Na figura 2.1 podemos observar a distribuição do uso de tecnologias MA por tipo de indústria.

Figura 2.1 – Distribuição de uso de MA nas indústrias



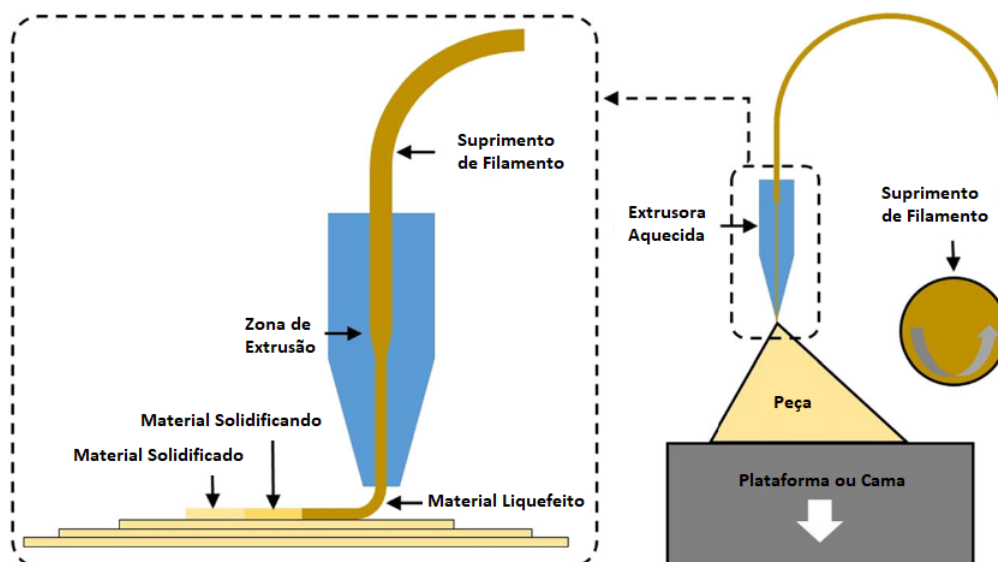
Fonte: Adaptado de BIKAS; STAVROPOULOS; CHRYSSOLOURIS, 2016

2.2 Fused Deposition Modeling

Fused Deposition Modeling (FDM) ou *Fused Filament Fabrication* (FFF) é uma das tecnologias MA mais populares como mencionado anteriormente. Ela se consiste por depositar material através de um processo onde um filamento de material é forçado dentro de uma câmara através, geralmente, de rolos dentados onde em uma região específica esse material é liquefeito. Por conta da pressão criada pelo filamento adentrando a câmara dentro do extrusor, ainda no estado sólido, o material liquefeito é extrudado através de um bocal, comumente fabricado de bronze. Então, o filamento liquefeito é depositado em uma plataforma de forma a percorrer a trajetória desejada utilizando mecanismos movidos de forma controlada, geralmente por motores de passos. O processo é repetido camada por camada, de forma que elas estejam apoiadas

por camadas anteriores e a primeira camada continue fixa na plataforma ou cama, até que o processo finalize (TURNER; STRONG; GOLD, 2014). A Figura 2.2 ilustra este processo.

Figura 2.2 – Princípio e processo de impressão para FDM



Fonte: Adaptado de BIKAS; STAVROPOULOS; CHRYSSOLOURIS, 2016

Este procedimento meticuloso exige a colaboração de vários componentes essenciais da impressora 3D. Estes componentes incluem:

Extrusora: A extrusora é responsável por derreter o filamento de material termoplástico e extrudá-lo em forma de filamento derretido. Ela consiste em um bico aquecido (*hotend*) que funde o material e um motor que empurra o filamento através do bico. Alguns modelos mais avançados podem ter extrusoras duplas para imprimir com materiais diferentes ou suportes solúveis.

Mesa de impressão: A mesa de impressão é a superfície onde o objeto está sendo construído. Ela é aquecida em muitas impressoras FDM para ajudar a aderência do material à superfície. Além disso, algumas mesas de impressão têm características especiais, como superfícies texturizadas ou magnéticas, para facilitar a aderência e a remoção do objeto após a conclusão.

Plataforma de construção: A plataforma de construção é o suporte físico onde a mesa de impressão é montada. Ela pode ser ajustada em altura para nivelar a superfície de impressão e garantir que a primeira camada do objeto seja depositada com precisão.

Motor de movimento: Impressoras 3D FDM possuem motores de movimento que controlam a posição da extrusora e da mesa de impressão ao longo dos eixos X, Y e Z. Geralmente

são motores de passo e seus movimentos de rotação são geralmente convertidos em movimentos lineares através de correias ou parafusos de rosca trapezoidal.

Filamento: O filamento é o material de alimentação para a impressora 3D. Ele é um longo fio de plástico termoplástico que é inserido na extrusora e derretido durante o processo de impressão. Os filamentos vêm em várias cores e tipos de material, dependendo do objeto a ser impresso.

Circuitos eletrônicos: O principal circuito eletrônico em uma impressora 3D é a placa-mãe, que recebe comandos do software, em geral no formato Gcode, e os traduz em movimentos dos motores, controle de temperatura da extrusora e da mesa de impressão, velocidade dos ventiladores entre outros acessórios. A impressora também pode apresentar um painel de controle, disponibilizando uma tela de exibição e controles para operação manual.

A Figura 2.3 indica alguns destes componentes e a Figura 2.4 mostra a interface de um fatiador.

Figura 2.3 – Indicação dos componentes de uma impressora 3D

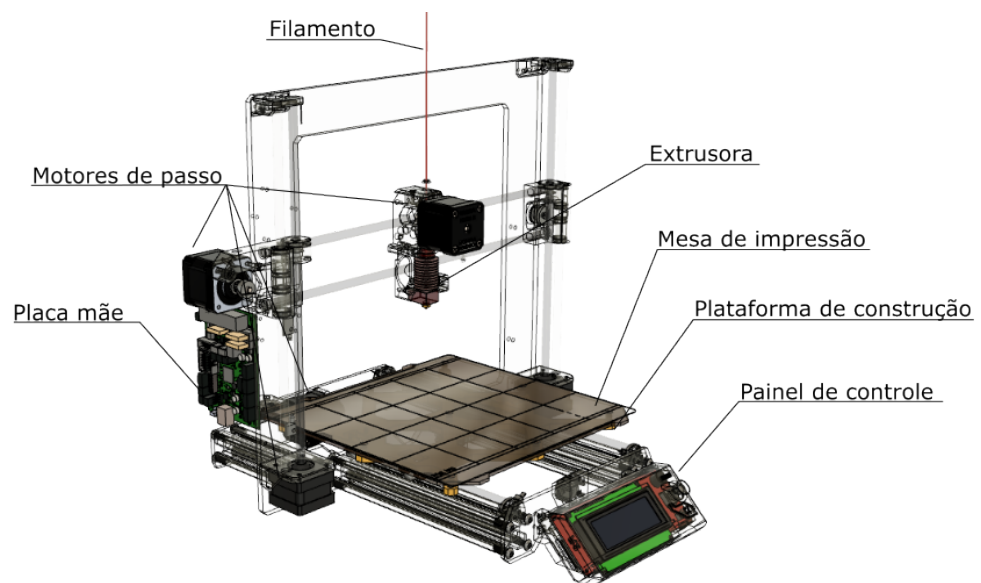
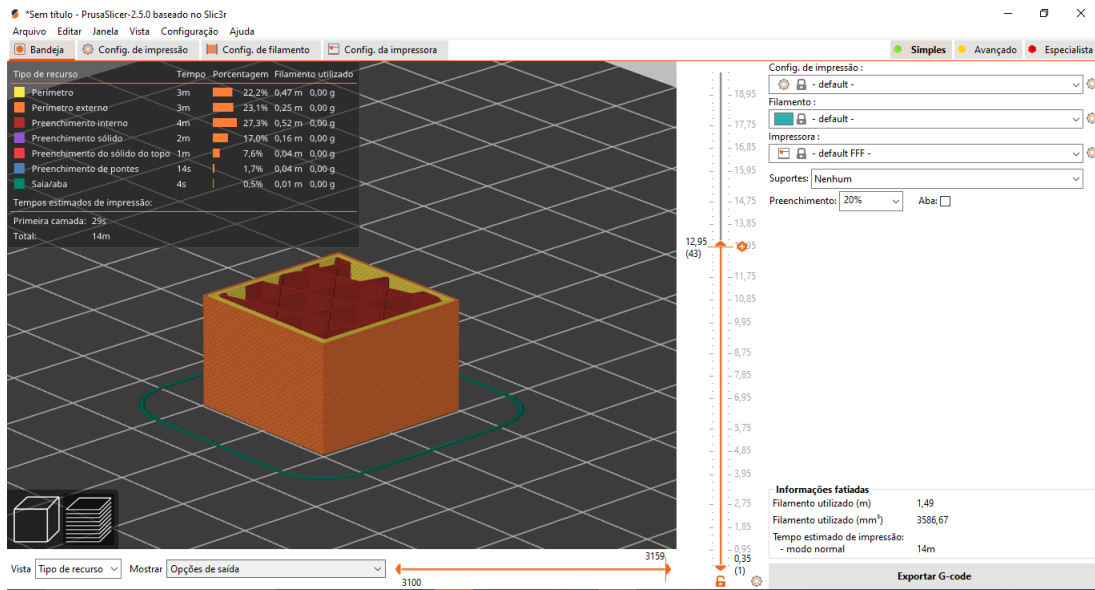


Figura 2.4 – Interface do fatiador PrusaSlicer



Podemos separar, de maneira simplificada, o *software* de impressoras 3D FDM em algumas etapas: fatiamento (*slicing*) ou geração de comando, geração de trajetória e controle ou controle de trajetória. A etapa de geração de comando, realizado pelo fatiador, onde o volume do modelo é dividido em camadas, definindo um percurso através de uma sequência de comandos. Esses comandos controlam os movimentos a serem realizados, definição de configurações, o controle da temperatura e outros dispositivos. Já na etapa de geração de trajetória, os comandos criados pelo fatiador (*slicer*) na etapa anterior são interpretadas, entretanto esses comandos não descrevem em detalhes. Portanto, é necessário a definição do comportamento que a impressora deve tomar para cumprir o comando interpretado, no caso de um comando de movimento por exemplo precisamos construir a trajetória dos eixos para poder realizar o comando com sucesso, assim chamamos esta etapa de geração de trajetória. Podemos, antes de prosseguir para a próxima etapa, utilizar de técnicas de controle para melhorar a performance da execução dos comandos, por exemplo utilizar os sensores de temperatura para modificar a taxa de aquecimento do bico extrusor, ou utilizar essas técnicas de controle para alterar a trajetória construída na etapa anterior de maneira a contribuir para a qualidade da impressão. Por fim, temos a etapa de geração de sinais, onde a impressora precisa converter as abstrações definidas sobre seus dispositivos em sinais que são responsáveis em controlá-los. Por exemplo, converter as características do movimento de um eixo em uma série de pulsos elétricos que controlam os motores de passo.

2.2.1 Geração de Comando

O Gcode (Código G) é uma linguagem de programação usada em impressoras 3D e máquinas CNC para controlar o movimento e as ações da máquina durante o processo de fabricação. Ele é muito utilizado pelos fatiadores para representar os comandos para a impressora 3D. O Gcode é composto por uma série de comandos textuais, cada um com um formato específico. Aqui estão alguns elementos-chave da estrutura típica de um comando G-code:

- **Prefixo (Código G):** Todo comando G-code começa com a letra 'G', que indica que é um comando de movimento ou função.
- **Número do Comando:** Após o 'G', segue um número que identifica o tipo específico de comando. Por exemplo, 'G0' é frequentemente usado para mover rapidamente a cabeça de impressão para uma posição, enquanto 'G1' é usado para movimentos de impressão lineares.
- **Parâmetros:** Após o número do comando, podem seguir-se parâmetros adicionais. Esses parâmetros variam dependendo do comando, mas podem incluir coordenadas de posicionamento (X, Y, Z), velocidades de movimento, taxas de alimentação, temperaturas e outros valores relevantes.
- **Comentários:** O G-code também pode incluir comentários precedidos por um ponto e vírgula (;) ou entre parênteses (). Os comentários não afetam a execução do programa, mas ajudam a documentar o código para facilitar a compreensão.
- **Fim de Linha:** Cada comando G-code é normalmente concluído com um caractere de fim de linha, como o retorno de carro ('\n') ou a combinação de retorno de carro e nova linha ('\r\n'), dependendo do sistema.

Figura 2.5 – Exemplo de um arquivo Gcode

```

G1 X100.0 E12.5 F1000.0 ; intro line
G92 E0.0
M221 S95
G21 ; set units to millimeters
G90 ; use absolute coordinates
M82 ; use absolute distances for extrusion
G92 E0
; Filament gcode
M107
;LAYER_CHANGE
;Z:0.12
;HEIGHT:0.12
G1 E-.5 F2400
G92 E0
G1 Z.12 F6000
G1 X54.465 Y89.348
G1 E.5 F1800
;TYPE:Skirt/Brim
;WIDTH:0.5
G1 F1500
G1 X55.343 Y88.639 E.5304
G1 X56.517 Y87.853 E.56846
G1 X57.508 Y87.311 E.59889

```

2.2.2 Geração de trajetória

A geração de trajetória é o processo que define o comportamento dos eixos da impressora ao longo do tempo com base nos comandos interpretados, incluindo além dos eixos de movimento, eixos de temperatura, potência dos ventiladores e de taxa de extrusão por exemplo. Esses diferentes componentes precisam coordenados e seu comportamento necessita de ser definido para que seja possível converter essas abstrações no mundo físico posteriormente. No caso dos eixos de movimento, uma estratégia para se construir esse comportamento são as curvas de velocidade trapezoidal (YU *et al.*, 2020).

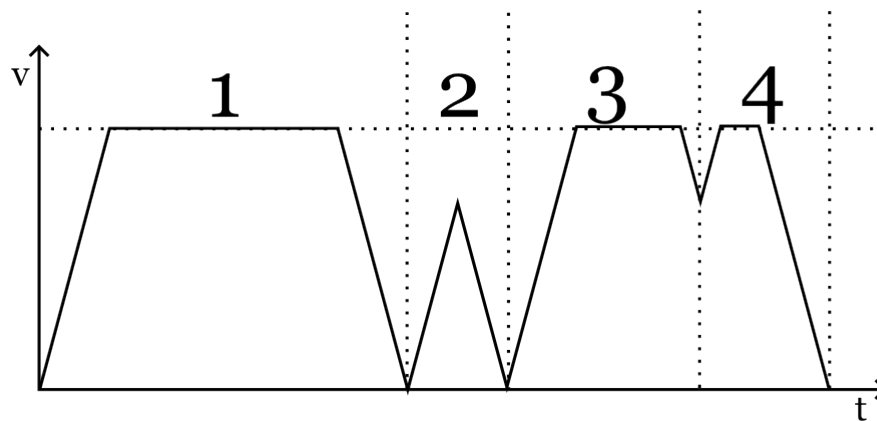
2.2.3 Curvas de velocidade trapezoidal

A construção das curvas de velocidade trapezoidal se da no objetivo de se deslocar de uma posição e velocidade inicial para uma posição e velocidade final, considerando uma velocidade máxima. Assim, se utiliza uma configuração de aceleração armazenada na impressora a fim de definir o comportamento das variações de velocidade. Vale lembrar que os comandos de movimento podem possuir atuação de múltiplos eixos e dado o objetivo das condições iniciais e finais de cada eixo se darem ao mesmo tempo, é utilizado o modulo do vetor de velocidade, composto pelas velocidades de cada eixo.

A partir dessas características, ao construir o perfil de velocidade em uma situação onde se tenha tempo suficiente para atingir a velocidade máxima, com base na aceleração definida pela impressora, o perfil de velocidade tem uma aparência trapezoidal (seções 1, 3 e 4), justificando o nome da estratégia. Entretanto, caso não seja possível alcançar a velocidade máxima

dentro do tempo e deslocamento possível para o movimento, o perfil de velocidade passa a possuir uma aparência triangular (seção 2), causada pelo fato de não existir mais uma seção onde é mantida a velocidade máxima (YU *et al.*, 2020; KLIPPER, 2017). A Figura 2.6 mostra um perfil de velocidade de uma sequência de quatro movimentos, apresentando seções trapezoidais e triangulares.

Figura 2.6 – Perfil de velocidade - Curva trapezoidal de velocidade



2.2.4 Técnicas de Controle

2.2.5 *Feedforward*

Nos métodos de controle para impressoras 3D *Fused Deposition Modeling* (FDM), o controle *feedforward* se consolida como a estratégia mais eficaz para os eixos de movimento, especialmente quando se pondera as restrições orçamentárias típicas de impressoras 3D padrão.

O controle *feedforward* é uma abordagem utilizada em sistemas automáticos, destinada a antecipar e corrigir possíveis perturbações que possam interferir em um sistema. Ele atua proativamente, antes que tais perturbações comprometam a saída prevista. Essa técnica é especialmente útil em sistemas onde é possível prever com exatidão as perturbações.

No universo das impressoras 3D, que são tipicamente menos susceptíveis a interferências externas, é factível prever o comportamento dinâmico da máquina com precisão, baseando-se em um modelo apropriado, e sem a necessidade de sensores de alto custo. Assim, o controle *feedforward* emerge como uma solução prática para otimizar o desempenho de impressoras já em operação, exigindo alterações físicas mínimas e simples ajustes no *software*.

Contudo, a implementação do controle *feedforward* em impressoras 3D apresenta desafios. Entre eles, a complexidade em estabelecer um modelo fidedigno, a demanda por capacidade computacional significativa e a obrigação de simular o processo de impressão do começo ao fim, considerando a influência do estado inicial (RAMANI; EDOIMIOYA; OKWUDIRE, 2020; DUAN; YOON; OKWUDIRE, 2018)

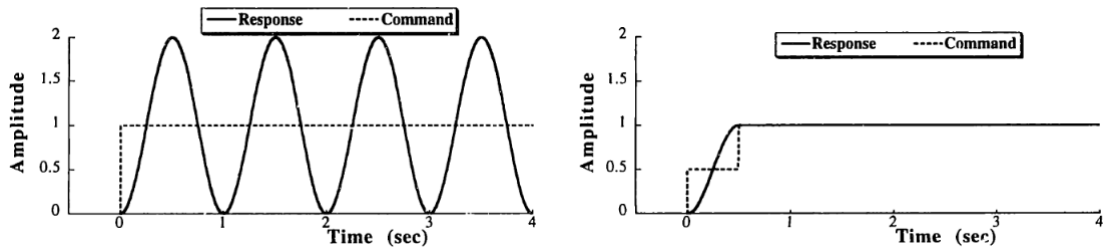
2.2.5.1 *Input Shaper*

Ao entender a trajetória pretendida e as peculiaridades do sistema, podemos deduzir uma sequência de comandos que ajustam o comando de referência com base nessas características. O objetivo é alinhar a trajetória final o máximo possível ao comando de referência. No entanto, em vez de processar todo o comando inicial, podemos adaptá-lo em tempo real usando um filtro especializado. O *Input Shaper* é uma abordagem exemplar desse tipo de filtro, onde diversos *Shapers* são formulados considerando variados propósitos e limitações.

O *Input Shaping* é uma metodologia de controle criada para atenuar ou anular vibrações indesejadas em sistemas mecânicos ou estruturas manipuladas por atuadores, como em robôs, veículos e equipamentos industriais. Essa estratégia aprimora a sequência de pulsos de entrada, concebendo-os de modo a suprimir a ativação das frequências naturais de ressonância do sistema. Assim, o *Input Shaping* desempenha um papel crucial na diminuição da resposta vibratória, conferindo maior precisão e estabilidade ao sistema durante a realização de movimentos ou operações.

O valor do *Input Shaping* é evidente em sistemas de controle de alta exatidão, como na robótica, onde a supressão de vibrações é fundamental para a excelência operacional e o posicionamento acurado. Utilizando o entendimento das especificidades do sistema e da trajetória almejada, essa técnica ajusta os comandos de entrada para evitar a ativação das frequências vibratórias intrínsecas, originando sistemas de controle mais robustos e eficientes. (SINGHOSE, 1997). A Figura 2.7 ilustra uma comparação entre as respostas ao estímulo de degrau e a função escalonada aplicada pelo *shaper*.

Figura 2.7 – Comparação da resposta ao degrau e da resposta a escada



Fonte: SINGHOSE, 1997

2.3 Espaço de Estados

Os sistemas dinâmicos podem ser descritos através de uma formulação chamada de espaço de estados, que tem como objetivo expressar modelos de equações diferenciais parciais (EDP) ou ordinárias (EDO) de ordem superior como um conjunto de EDPs ou EDOs de primeira ordem. Na equação 2.1 podemos observar uma EDO de segunda ordem representando um sistema massa mola simples, logo abaixo (2.2) a mesma equação representada na formulação de espaço de estados (HAMILTON, 1994).

$$m\ddot{x} + c\dot{x} + kx = f(t) \quad (2.1)$$

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ k/m & c/m \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} f(t) \quad (2.2)$$

2.4 Programação Não Linear

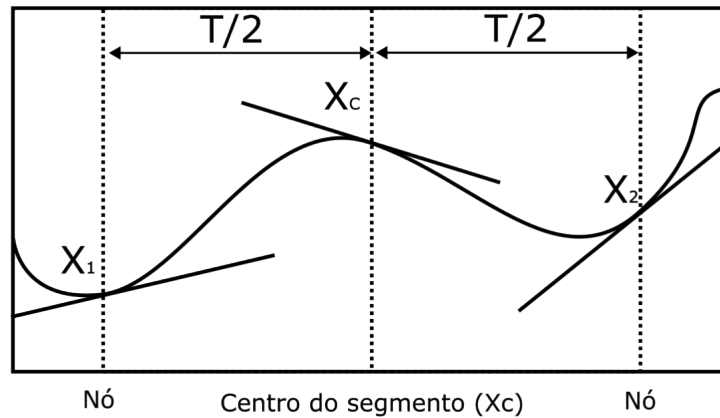
HARGRAVES; PARIS (1987) descreve um algoritmo para a solução numérica direta de problemas de controle ótimo. Este algoritmo emprega uma abordagem que utiliza polinômios cúbicos para representar as variáveis de estado. Adicionalmente, recorre à interpolação linear para tratar as variáveis de controle. Esse enfoque converte efetivamente o problema de controle ótimo em um problema de programação matemática.

Uma das principais vantagens desse método é sua facilidade de implementação e sua capacidade de lidar com uma ampla gama de problemas de otimização de trajetória. Isso inclui a consideração de restrições de caminho, estados descontínuos e desigualdades de controle.

O método alcança sua aproximação das soluções das equações diferenciais através da subdivisão de cada estado na matriz de espaço de estados em segmentos. Cada um desses segmentos é representado por polinômios de terceiro grau.

Os valores de estado são então selecionados de maneira a garantir que a curva resultante da concatenação desses polinômios seja contínua, ou seja, o valor da função e de sua derivada precisa ser igual para ambos polinômios nas conexões como observado na figura 2.8

Figura 2.8 – Ilustração Segmentação Cúbica



Fonte: HARGRAVES; PARIS, 1987

O procedimento base pode ser aplicado pelos seguintes passos.

A equação 2.3 avalia o estado no centro do segmento, onde x representa o estado, T representa o comprimento do segmento e f_i representa o valor da função avaliado em x_i . O subscrito c representa o centro do segmento.

$$x_c = \frac{x_1 + x_2}{2} + T \frac{f_1 - f_2}{8} \quad (2.3)$$

Da mesma maneira sua derivada é apresentada na equação 2.4.

$$\dot{x}_c = -3 \frac{x_1 + x_2}{2T} + \frac{f_1 + f_2}{4} \quad (2.4)$$

A equação 2.5 define então o valor do defeito no centro do segmento.

$$\Delta = f_c - \dot{x}_c \quad (2.5)$$

Considerando também que a entrada do sistema pode ser avaliada de forma aproximada no centro do segmento através da equação 2.6.

$$u_c = \frac{u_1 + u_2}{2} \quad (2.6)$$

Os valores de estado agora podem ser alterados de maneira que o defeito tenda a zero.

3 METODOLOGIA

Neste trabalho, a abordagem metodológica para o estudo do controle de trajetória em impressoras 3D, aplicando algoritmos iterativos e técnicas de programação não linear, é estruturada em duas etapas fundamentais. A primeira etapa envolve a elaboração de um modelo computacional que representa o comportamento mecânico da impressora 3D, integrando o método avançado de controle de trajetória baseado em programação não linear proposto neste estudo. A última fase é dedicada à realização de simulações computacionais, as quais são utilizadas para avaliar o desempenho do método de controle, empregando uma análise de sensibilidade em relação a uma gama de variáveis significativas.

O processo metodológico é visualizado de forma esquemática no fluxograma abaixo (Figura 3.1), o qual esclarece as etapas consecutivas desde a geração do comando até a geração dos sinais de controle, enfatizando a aplicação do modelo desenvolvido na fase de controle de trajetória.

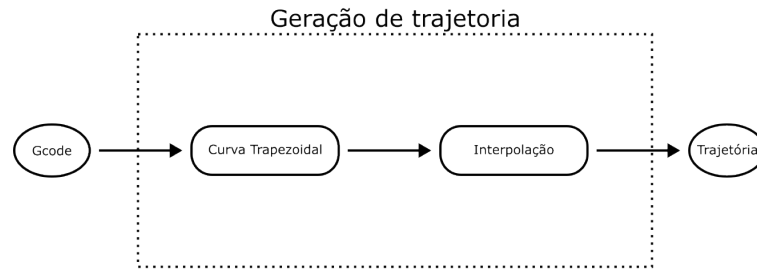
Figura 3.1 – Fluxograma geral das etapas para o controle de trajetória



3.1 Geração de Trajetória

A fase de geração de trajetória inicia-se com a análise do Gcode, que fornece as coordenadas e velocidades de destino dos movimentos. Neste processo, são considerados exclusivamente os comandos G1, que indicam movimentos lineares, e são extraídas as informações referentes aos eixos X, Y e à taxa de avanço (*feedrate* - F). É importante notar que a taxa de avanço, usualmente expressa em milímetros por minuto nos arquivos Gcode gerados por fatiadores, é convertida para milímetros por segundo.

Figura 3.2 – Curva de velocidade trapezoidal



3.1.1 Curva trapezoidal de velocidade

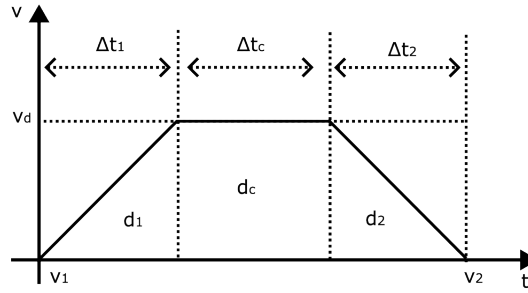
A próxima etapa envolve a elaboração da curva trapezoidal de velocidade. Esta etapa se baseia em dados de entrada como deslocamento, velocidades iniciais e finais, e a velocidade almejada. A execução desta velocidade desejada é avaliada através do cálculo da velocidade de pico v_p . Tal velocidade é obtida pela interseção das trajetórias de aceleração e desaceleração, que iniciam nas velocidades iniciais e finais respectivamente, e considerando que a área sob a curva deve ser equivalente ao deslocamento requerido. A fórmula para calcular v_p é expressa pela Equação 3.1:

$$v_p = \sqrt{\frac{(v_1^2 + v_2^2)}{2} + ad} \quad (3.1)$$

Nesta equação, v_1 e v_2 representam as velocidades iniciais e finais, a denota a aceleração definida na impressora, e d corresponde ao deslocamento.

A comparação entre a velocidade de pico e a velocidade desejada, esta última estabelecida pelo *feedrate* no Gcode, é crucial para definir se a trajetória do movimento adotará um perfil trapezoidal ou triangular de velocidade. A configuração do perfil depende da relação entre a velocidade de pico e a velocidade desejada: caso a primeira seja superior, o movimento será estruturado em três segmentos distintos, conforme ilustrado na Figura 3.3.

Figura 3.3 – Curva de velocidade trapezoidal



Os segmentos de deslocamento d_1 , d_2 , e d_c correspondem às fases de aceleração, velocidade constante e desaceleração do movimento, respectivamente, e devem totalizar d , o deslocamento total necessário. As variações temporais Δt_1 , Δt_2 , e Δt_c representam as durações de cada fase, baseadas nas velocidades inicial (v_1) e final (v_2), e na velocidade desejada (v_d). Os deslocamentos parciais são determinados pelas equações abaixo (Equação 3.2, 3.3 e 3.4), que levam em conta a aceleração a :

$$d_1 = \frac{(v_d^2 - v_1^2)}{(2a)} \quad (3.2)$$

$$d_2 = \frac{(v_2^2 - v_d^2)}{(2a)} \quad (3.3)$$

$$d_c = d - (d_1 + d_2) \quad (3.4)$$

Os intervalos de tempo para as fases de aceleração, velocidade constante e desaceleração são calculados conforme (Equação 3.5, 3.6 e 3.7):

$$\Delta t_1 = \frac{(v_d - v_1)}{a} \quad (3.5)$$

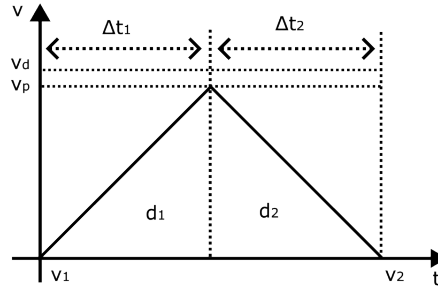
$$\Delta t_2 = \frac{(v_2 - v_d)}{a} \quad (3.6)$$

$$\Delta t_c = \frac{d_c}{v_d} \quad (3.7)$$

Quando a velocidade de pico (v_p) é inferior à velocidade desejada (v_d), o perfil da trajetória de movimento assume uma forma triangular, em vez de trapezoidal. Esta condição implica

que a velocidade desejada não é atingida durante o comando e, por conseguinte, o movimento é caracterizado por uma aceleração constante seguida de uma desaceleração constante, sem fase de velocidade constante. A Figura 3.4 ilustra este perfil de velocidade triangular.

Figura 3.4 – Curva de velocidade triangular



Neste cenário, os segmentos de deslocamento d_1 e d_2 representam, respectivamente, as etapas de aceleração até a velocidade de pico e a desaceleração de volta à velocidade final. Os valores de d_1 e d_2 são calculados pelas seguintes equações (Equação 3.8, 3.9), que incorporam a aceleração (a) e as velocidades inicial (v_1) e final (v_2):

$$d_1 = \frac{(v_p^2 - v_1^2)}{(2a)} \quad (3.8)$$

$$d_2 = \frac{(v_2^2 - v_p^2)}{(2a)} \quad (3.9)$$

É possível calcular os intervalos de tempo dessas fases, através das Equações 3.10 e 3.11.

$$t_1 = \frac{(v_p - v_1)}{a} \quad (3.10)$$

$$t_2 = \frac{(v_2 - v_p)}{a} \quad (3.11)$$

Onde os tempos t_1 e t_2 representam, respectivamente, o tempo necessário para acelerar de v_1 a v_p e para desacelerar de v_p a v_2 .

Esses passos transformam a sequência de comandos movimentos do Gcode em uma trajetória com pontos com informações do deslocamento, velocidade, tempo, definidos nos nós

onde ha alteração na aceleração, estabelecendo o comportamento dos movimentos de x e y no tempo.

3.1.2 Interpolação

A interpolação é um passo crucial para refinar a trajetória de movimento na impressão 3D. Esta fase trabalha sobre a trajetória definida para cada eixo na etapa anterior, empregando uma função de interpolação que gera pontos intermediários. Esses pontos são criados com base em um intervalo de tempo previamente definido, melhorando significativamente a resolução da trajetória.

Para subdividir esses intervalos de maneira eficaz, a Equação 3.12 é utilizada para calcular o número de passos de interpolação necessários:

$$N = \lceil \frac{\Delta t}{\Delta p} - 1 \rceil \quad (3.12)$$

Esta fórmula determina o número N de passos a serem tomados dentro de um dado intervalo de tempo Δt , com cada passo tendo um período Δp . Após a divisão dos intervalos, a Equação 3.13 calcula o tempo restante no último passo de interpolação (Δt_f):

$$\Delta t_f = \Delta t - \Delta p N \quad (3.13)$$

Finalmente, com base nesses passos de tempo determinados (Δt_i) anexando Δt_f à lista de passos Δp de tamanho N , a Equação 3.14 é empregada para calcular o deslocamento correspondente a cada passo (Δd_i), utilizando a aceleração do segmento a ser interpolado (a_s) e a velocidade inicial do segmento (v_s):

$$\Delta d_i = \Delta v_s \Delta t_i + \frac{a_s \Delta t_i^2}{2} \quad (3.14)$$

Esses cálculos são fundamentais para garantir que a trajetória seja suficientemente detalhada, permitindo que a fase de controle da trajetória seja executada com sucesso.

3.2 Modelagem dinâmica de uma impressora 3D

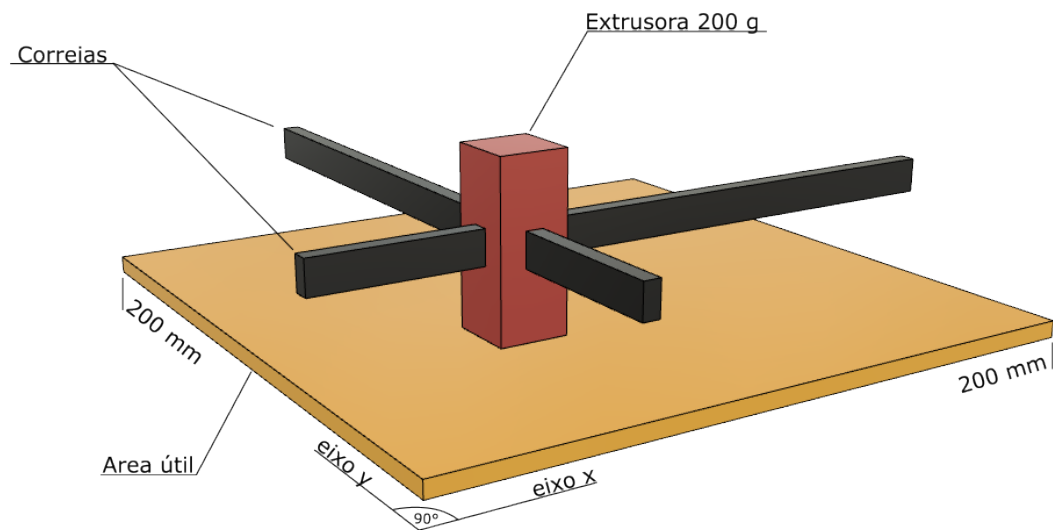
A modelagem do sistema mecânico da impressora 3D é um passo crucial para a implementação eficaz do método proposto. Essa modelagem não só facilita a compreensão do

comportamento da impressora mas também é fundamental para definir as restrições necessárias na etapa subsequente de controle de trajetória, restrições essas que aplicam as equações de movimento e condições de contorno definidas no modelo.

É fundamental enfatizar a importância de um modelo representativo. A eficácia do método proposto depende diretamente da acurácia com que o modelo simula o comportamento real da impressora 3D. Neste estudo, consideramos as seguintes características principais do sistema (Figura 3.5):

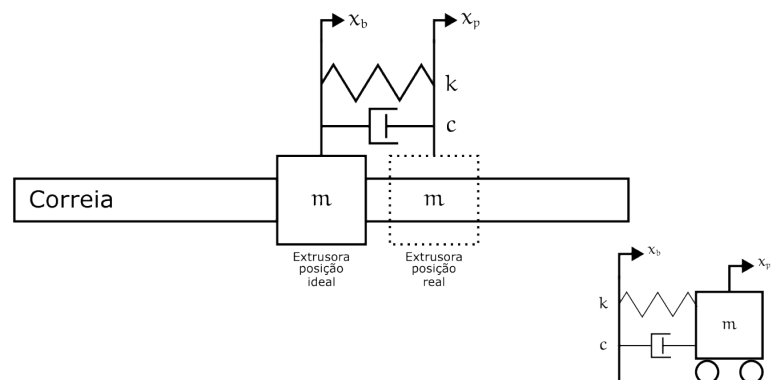
- **Influência da Correia:** A correia é o componente chave responsável por introduzir desvios nas trajetórias de impressão. Ela age como uma combinação de mola e amortecedor, afetando a dinâmica do movimento.
- **Modelagem do Conjunto Bico Injetor e Extrusora:** Este conjunto é tratado como um corpo rígido uniforme, simplificando sua representação geométrica. Sua massa é considerada como 200g, o que influencia a dinâmica do movimento.
- **Dimensões da Mesa de Impressão:** A área útil da mesa de impressão é de 200 mm x 200 mm, definindo o espaço de trabalho disponível.
- **Configuração Cartesiana:** A impressora opera em um sistema cartesiano, com eixos ortogonais, o que simplifica a análise de movimento.
- **Independência dos Eixos:** Cada eixo da impressora opera independentemente dos outros, permitindo uma análise mais simplificada das dinâmicas individuais.
- **Condições Iniciais de Movimento:** Assume-se que todos os movimentos da impressora iniciam a partir do estado de repouso.

Figura 3.5 – Modelo simplificado impressora 3D



Com base nesses parâmetros, definimos duas posições-chave para análise em cada eixo. A primeira é a posição ideal (x_b) ou posição da base, que representa o ponto desejado pelo usuário, assumindo um sistema sem flexibilidade ou perdas. A segunda é a posição real (x_p) ou a posição da ponta, que leva em conta as forças inerciais e a flexibilidade introduzida pela correia. Este modelo é ilustrado na figura 3.6.

Figura 3.6 – Modelagem de 1 eixo

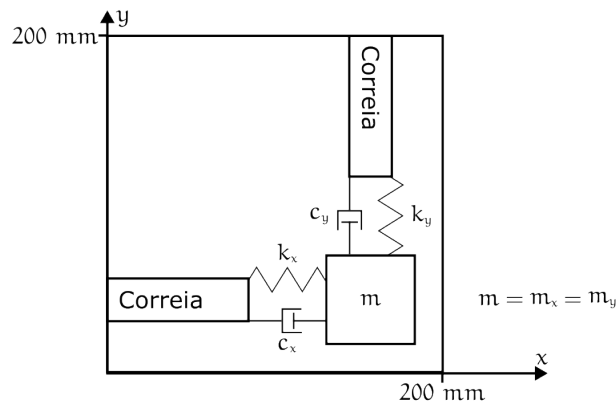


As equações de movimento para a impressora são descritas a seguir:

$$\begin{aligned}
 m\ddot{x}_p + c(\dot{x}_p - \dot{x}_b) + k(x_p - x_b) &= 0 \\
 \ddot{x}_p &= -\frac{c}{m}\dot{x}_p - \frac{k}{m}x_p + \frac{c}{m}\dot{x}_b + \frac{k}{m}x_b
 \end{aligned}
 \tag{3.15}$$

Nestas equações, m representa a massa do conjunto bico injetor e extrusora, c é a constante de amortecimento da correia, e k é a constante da mola equivalente da correia. As variáveis x_p e x_b correspondem, respectivamente, às posições da ponta e da base do componente em movimento. Essas equações fundamentam o modelo dinâmico que empregamos para simular e otimizar a trajetória de impressão na impressora 3D. Na Figura 3.7 é representada a composição dos eixos x e y utilizado neste estudo, sendo considerada a aplicação das Equação 3.15 para cada um dos eixos de maneira análoga, podendo identificar o eixo através dos subíndices x e y , ou no caso das posições o eixo y é identificado por y_p e y_b (posição da ponta do eixo y e posição da base do eixo y respectivamente).

Figura 3.7 – Modelagem dos eixos x e y



3.2.1 Espaço de estados

A formulação do espaço de estados é adotada neste estudo para simplificar as operações e a solução do sistema dinâmico da impressora 3D. Esta abordagem é eficaz pois transforma uma equação diferencial de ordem superior em um sistema de equações diferenciais de primeira ordem, mas com um número maior de equações. Esta metodologia facilita o entendimento e a manipulação das dinâmicas do sistema.

O modelo dinâmico no espaço de estados é representado na seguinte forma (Equação 3.16):

$$\dot{x} = A * x + B * u \quad (3.16)$$

Nesta equação, \dot{x} representa o vetor de estados derivados, x é o vetor de estados, A é a matriz do sistema que define a relação entre os estados atuais e suas taxas de mudança, u é o vetor de entradas externas, e B é a matriz de controle que relaciona as entradas com os estados.

Baseado na equação de movimento 3.15, análogos ao eixo y, expandimos as matrizes e vetores para representar com precisão a dinâmica do sistema no espaço de estados, conforme apresentado na equação 3.17:

$$\begin{bmatrix} \dot{x}_p \\ \ddot{x}_p \\ \dot{y}_p \\ \ddot{y}_p \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_x}{m_x} & -\frac{c_x}{m_x} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{k_y}{m_y} & -\frac{c_y}{m_y} \end{bmatrix} \begin{bmatrix} x_p \\ \dot{x}_p \\ y_p \\ \dot{y}_p \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{k_x}{m_x} & \frac{c_x}{m_x} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{k_y}{m_y} & \frac{c_y}{m_y} \end{bmatrix} \begin{bmatrix} x_b \\ \dot{x}_b \\ y_b \\ \dot{y}_b \end{bmatrix} \quad (3.17)$$

Nesta equação, x_p e y_p são as posições reais (da ponta) nos eixos X e Y, respectivamente, enquanto x_b e y_b são as posições ideais (da base). As variáveis \dot{x}_p e \dot{y}_p representam a primeira derivada do tempo das posições nos eixos X e Y, indicando a velocidade e aceleração. k_x e k_y denotam as constantes elásticas das correias nos eixos X e Y, enquanto c_x e c_y representam as constantes de amortecimento dessas correias. m_x e m_y são as massas associadas aos conjuntos de bico injetor e extrusora nos respectivos eixos.

3.3 Controle de Trajetória

O Controle de Trajetória desempenha um papel essencial em aperfeiçoar a precisão dos movimentos na impressora 3D. A escolha das estratégias de controle é crucial para maximizar a eficiência operacional. No escopo deste estudo, a técnica de controle adotada se baseia no modelo estabelecido anteriormente, onde aplicamos uma abordagem *feedforward* à trajetória gerada na fase de construção da trajetória.

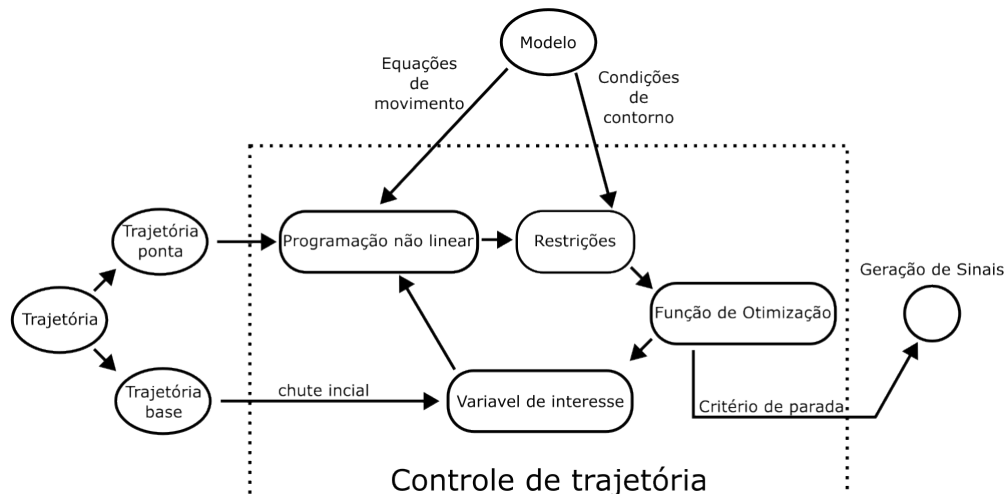
A metodologia de controle em foco procura resolver as equações de movimento e atender às condições de contorno estipuladas pela modelagem da impressora. O ajuste é feito na

trajetória da base do sistema, ajustando-a de forma que a saída do vetor de estados corresponda à trajetória da ponta projetada.

Utiliza-se uma função de otimização iterativa para refinar a trajetória da base, que é a principal variável de interesse. Este refinamento é feito minimizando um conjunto de restrições derivadas das equações de movimento e das condições de contorno. A iteração prossegue até que um critério de parada estabelecido seja alcançado, sugerindo que a trajetória base modificada satisfaz a trajetória da ponta almejada.

Este método assegura uma resposta proativa às dinâmicas da impressora, alinhando-se à trajetória planejada e, conseqüentemente, elevando a acuidade dos movimentos da impressora 3D. A Figura 3.8 ilustra o esquema do método de controle proposto.

Figura 3.8 – Fluxograma Controle de Trajetória



3.3.1 Restrições

A formulação do conjunto de restrições é um componente crucial do método, pois é através dele que as equações de movimento são implementadas. A função de otimização adota um algoritmo para minimizar essas restrições, ou seja, para que se aproximem tanto quanto possível de zero.

As condições de contorno são aplicadas estabelecendo que tanto a posição quanto a velocidade sejam zero no instante inicial e que a velocidade também seja zero ao final do percurso. As equações de movimento são incorporadas nas restrições através do método de programação não linear descrito na Seção 2.4. Com base no modelo em espaço de estados especificado na Seção 3.2. Utilizamos as Equações 2.3, 2.4, 2.6 e 2.5 para criar um vetor de defeitos, como descrito no Capítulo 2.4. A minimização destas diferenças faz com que os polinômios cúbicos

dos segmentos da curva se aproximem das soluções das equações de movimento, que são então integradas ao vetor de restrições.

3.3.2 Função de Otimização

Os limites da variável de interesse, que neste caso são determinados pela área de trabalho da impressora, também são definidos para que a extrusora não ultrapasse os limites da base de impressão, estabelecendo-se entre o mínimo de 0 e o máximo de 200 mm para os eixos x e y.

A função de otimização adota o algoritmo interior-point, eficaz em resolver problemas de otimização não-linear com restrições. Este método é preferível para grandes conjuntos de dados, por ser mais rápido e numericamente estável do que abordagens tradicionais. Os critérios de parada para o algoritmo são apresentados na Tabela 3.1.

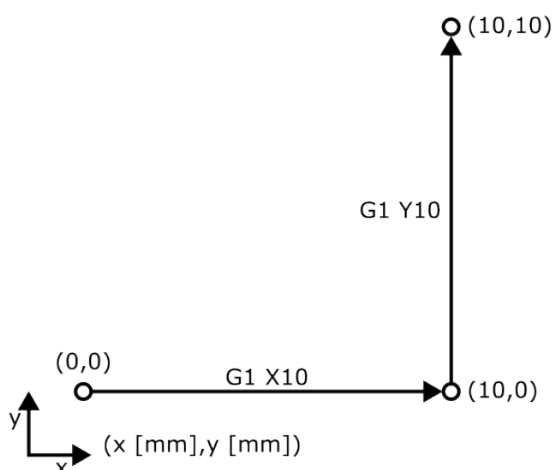
Tabela 3.1 – Critérios de Parada do Algoritmo de Otimização

Opção	Valor
Máximo de Iterações	100000
Diferença Mínima entre Iterações	0.0001

3.4 Simulação Computacional e Análise de Dados

Nesta seção, descrevemos o processo de simulação para testar o método proposto. Iniciamos com a simulação de dois movimentos lineares: um deslocamento de 10 milímetros no eixo x seguido de um movimento similar no eixo y, partindo da posição inicial (0,0) e em estado de repouso. A Figura 3.9 ilustra estes movimentos.

Figura 3.9 – Movimento base



Selecionamos parâmetros chave para o controle da simulação, apresentados na Figura 3.10, e seus respectivos valores são listados na Tabela 3.2.

Figura 3.10 – Fluxograma geral com os parâmetros.

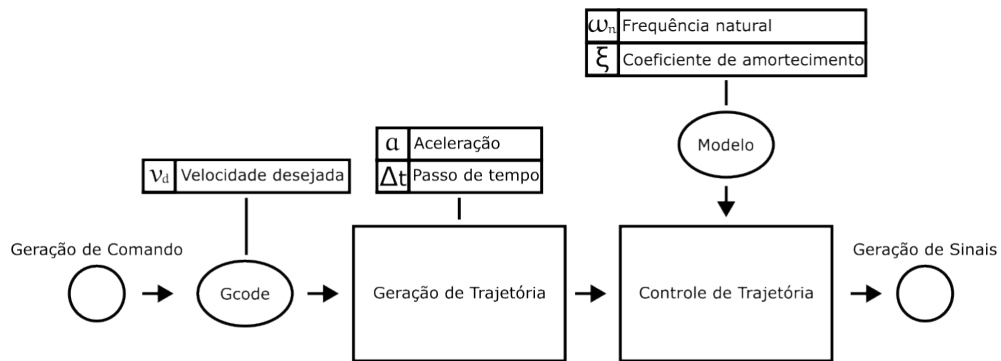


Tabela 3.2 – Valores dos parâmetros utilizados na simulação.

Parâmetro	Valor	Unidade
Frequência	100	rad/s
Coeficiente de amortecimento	0,5	-
Aceleração base	5000	mm/s^2
Passo de tempo	0,005	s
Velocidade desejada	100	mm/s

Avaliamos a trajetória resultante focando em três aspectos: a viabilidade, o número de iterações e a dimensão do vetor de variáveis de interesse. A viabilidade é medida pelo maior desvio em relação às restrições do modelo, o que nos ajuda a compreender a capacidade da simulação de atingir uma solução que respeite todas as restrições estabelecidas, além de indicar o padrão de convergência ao observar a evolução da viabilidade ao longo das iterações. O tamanho do vetor de variáveis também é considerado, dada a sua relevância para o desempenho computacional da simulação.

3.4.1 Análise de Sensibilidade

Realizamos uma análise de sensibilidade para examinar o comportamento do método frente a variações nos parâmetros listados na Tabela 3.2. Alteramos cada parâmetro individu-

almente em 2 ou 3 simulações distintas, conforme detalhado na Tabela 3.3. Cada simulação é identificada por um número e uma letra, indicando o parâmetro modificado e o valor aplicado.

Tabela 3.3 – Parâmetros utilizados na análise de sensibilidade.

Caso	Parâmetro	Valor A	Valor B	Valor C	Unidade
1	Frequência	50	200	500	rad/s
2	Coeficiente de amortecimento	0	0,25	1	-
3		1000	10000	-	mm/s^2
4	Passo de tempo	0,01	0,001	0,0002	s
5	Velocidade desejada	50	200	-	mm/s

As simulações foram executadas em um computador com as especificações listadas na Tabela 3.4.

Tabela 3.4 – Especificações do computador

Processador	Intel I7-5500U 2.40GHz
Memoria	8,00 GB
Placa de vídeo	Nvidia Geforce 920M
Sistema	64 bits

4 RESULTADOS E DISCUSSÃO

Este capítulo detalha os resultados alcançados por meio da análise de sensibilidade aplicada às simulações no estudo do controle de trajetória em impressoras 3D. Essa análise é essencial para avaliar a eficiência do método de controle proposto, considerando sua resposta diante de diferentes condições e parâmetros operacionais. Os resultados destacam as características de resposta do método, que dão indicações de sua viabilidade e possíveis limitações. A apresentação e interpretação dos dados têm como objetivo enfatizar o desempenho do método e oferecer percepções valiosas para futuros avanços e aplicações práticas na área de controle de trajetória para a manufatura aditiva.

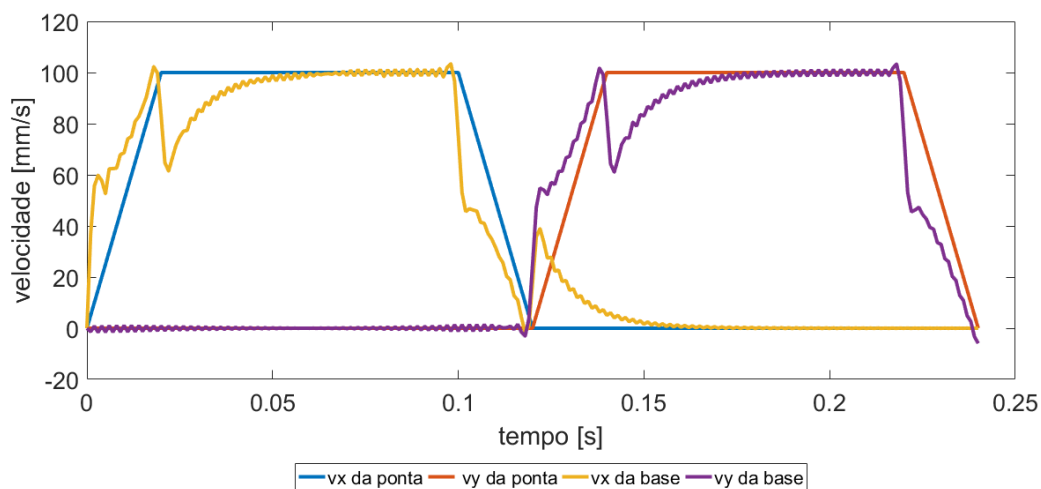
Prosseguindo, a análise dos resultados inicia-se com a avaliação da simulação de referência, estabelecendo um ponto de comparação para as demais simulações. Em seguida, detalhamos o impacto e as implicações de cada um dos cinco parâmetros escolhidos na análise de sensibilidade. Estes parâmetros foram criteriosamente selecionados com o objetivo de explorar diferentes aspectos do método de controle e sua resposta em variados cenários operacionais. A investigação destes parâmetros oferece uma visão abrangente sobre como cada elemento influencia o desempenho geral do sistema, permitindo uma compreensão mais profunda e um refinamento do método proposto.

4.1 Resultados da Simulação Referência

Esta seção aborda os resultados obtidos na simulação de referência, e estabelece a referência para a análise de sensibilidade. As análises são quebradas em diferentes focos dado os diferentes gráficos dos resultados. Os seguintes focos de análise são estabelecidos:

- **Análise de Velocidades (Figura 4.1):** Este gráfico exhibe as variações das velocidades ao longo do tempo nos eixos X e Y. Ele demonstra a resposta dinâmica tanto da ponta do manipulador quanto da base, conforme a metodologia proposta.

Figura 4.1 – Caso referência - Comportamento no tempo das velocidades em x e y da ponta e da base

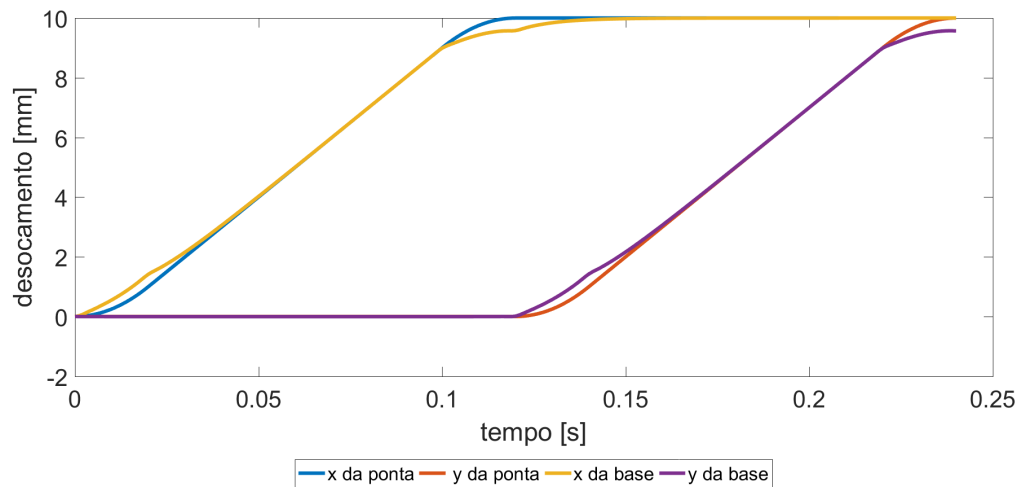


Com os parâmetros de referência a trajetória obtida é trapezoidal e é definida pelos parâmetros de velocidade desejada, aceleração e pelas distâncias dos movimentos, como comentado no trabalho.

As linhas amarela e roxa apresentam as curvas de velocidade da base dos eixos x e y respectivamente, estas foram obtidas através da aplicação do método proposto como solução das restrições para a trajetória desejada da ponta, representada pelas linhas azul e laranja.

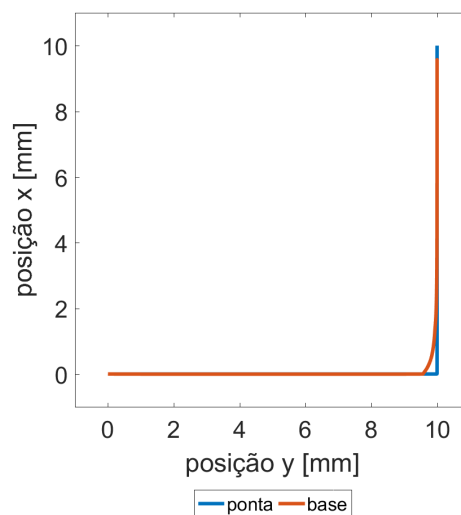
- Deslocamentos Temporais (Figura 4.2): Aqui, os deslocamentos nos eixos X e Y são plotados contra o tempo. Esses resultados refletem diretamente a precisão e eficiência do sistema simulado, conforme os critérios estabelecidos na metodologia.

Figura 4.2 – Caso referência - Comportamento no tempo dos deslocamentos em x e y da ponta e da base



- Trajetória Espacial (Figura 4.3): Este gráfico ilustra o caminho percorrido nos planos X e Y, comparando a trajetória da ponta com a trajetória de referência. A aderência ao caminho planejado, conforme descrito na metodologia, é fundamental para avaliar a acurácia do sistema.

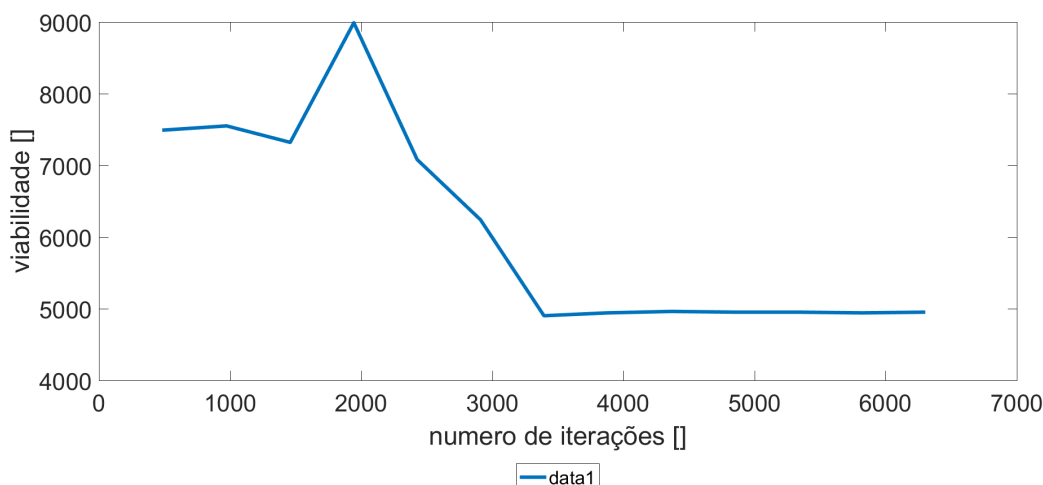
Figura 4.3 – Caso referência - Caminho percorrido x vs y da ponta e da base



- Análise da viabilidade (Figura 4.4): Este gráfico exibe a evolução das iterações durante a simulação. Cada ponto no gráfico representa uma iteração distinta. O eixo X mostra

o número de avaliações da função objetivo e das restrições, enquanto o eixo Y reflete o valor de viabilidade, que indica o grau de cumprimento das restrições mais desafiadoras, conforme discutido na metodologia.

Figura 4.4 – Caso referência - Num de fun x Viabilidade



Observa-se uma tendência clara de convergência ao longo das iterações, um indicativo de que a simulação está alcançando resultados bem-sucedidos. Notavelmente, os valores iniciais de viabilidade se aproximam de $7,5 \times 10^{-3}$, um padrão consistente observado em vários casos. Além disso, o tempo total de simulação para esta análise foi de aproximadamente 89,8 segundos. Os vetores de posição empregados na simulação contavam com 241 elementos. Este gráfico é fundamental para entender a eficácia da simulação em atender às restrições estabelecidas e a capacidade do sistema de convergir para uma solução viável em um tempo eficiente.

4.2 Análise de Sensibilidade dos Resultados

4.2.1 Caso 1 - Variação da frequência

Este caso investiga como a variação da frequência natural afeta o modelo dinâmico do sistema. A frequência natural, sendo um parâmetro crítico, influencia diretamente o comportamento da planta do modelo, com implicações notáveis nas amplitudes dos desvios e na necessidade de compensação.

Observamos que sistemas com maior rigidez, caracterizados por frequências naturais mais altas, exibem menores amplitudes de desvio e uma reduzida necessidade de compensação. Esta tendência é claramente ilustrada ao comparar as Figuras 4.5, 4.6 e 4.7, que mostram o comportamento das velocidades em x e y para diferentes configurações de frequência.

Figura 4.5 – Caso 1A - Comportamento no tempo das velocidades em x e y da ponta e da base

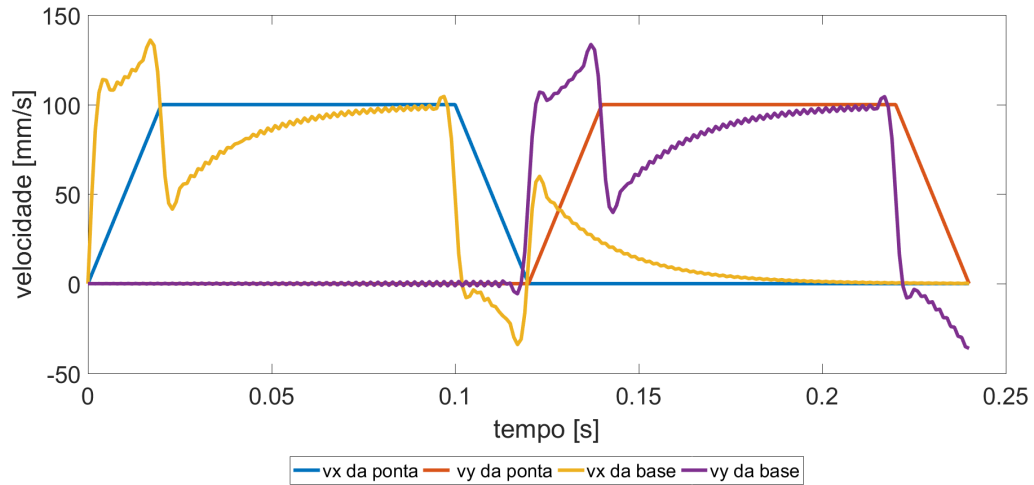


Figura 4.6 – Caso 1B - Comportamento no tempo das velocidades em x e y da ponta e da base

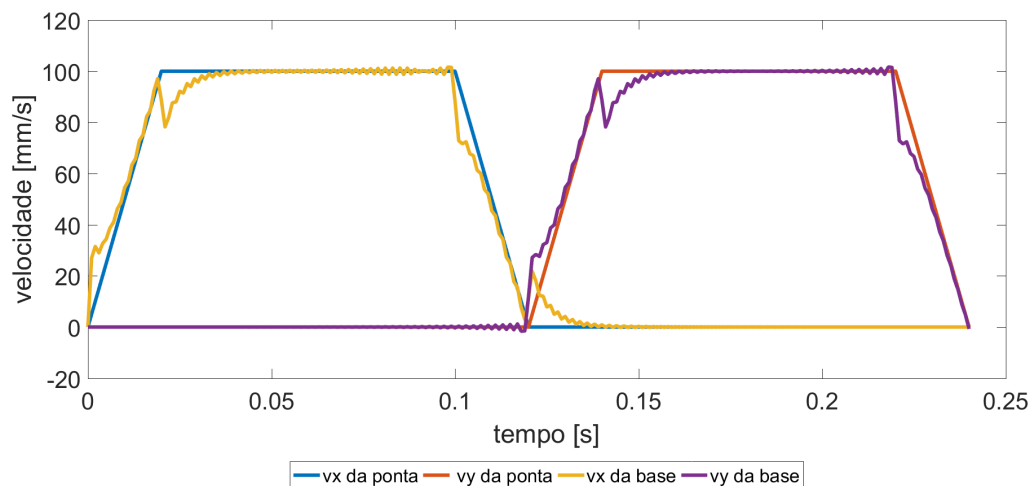
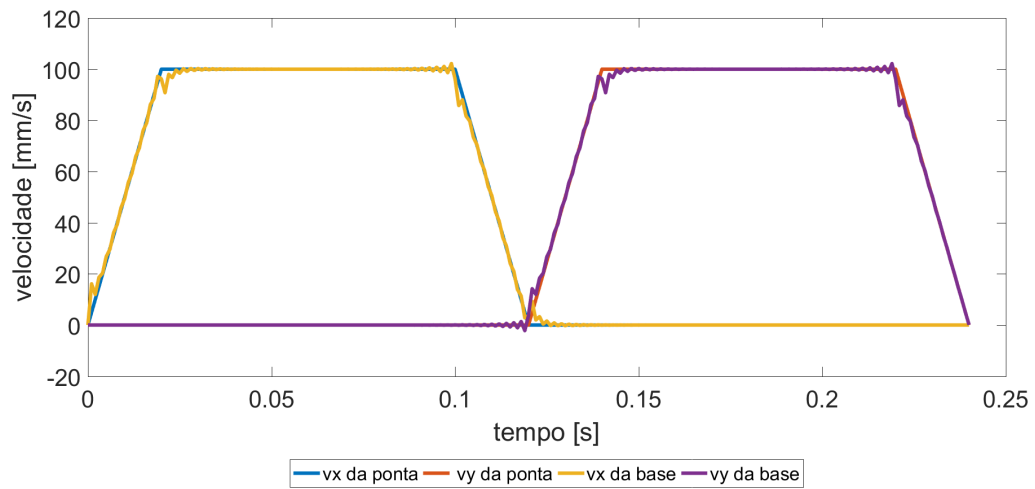


Figura 4.7 – Caso 1C - Comportamento no tempo das velocidades em x e y da ponta e da base



A análise é aprofundada ao examinar as Figuras 4.8, 4.9, 4.10 e 4.11, que revelam as diferenças no comportamento do deslocamento e no caminho percorrido pela ponta do sistema nos extremos do caso (A e C). Estes resultados indicam que variações na frequência natural não apenas afetam as propriedades dinâmicas, mas também têm implicações diretas no controle e precisão do sistema.

Figura 4.8 – Caso 1A - Comportamento no tempo dos deslocamentos em x e y da ponta e da base

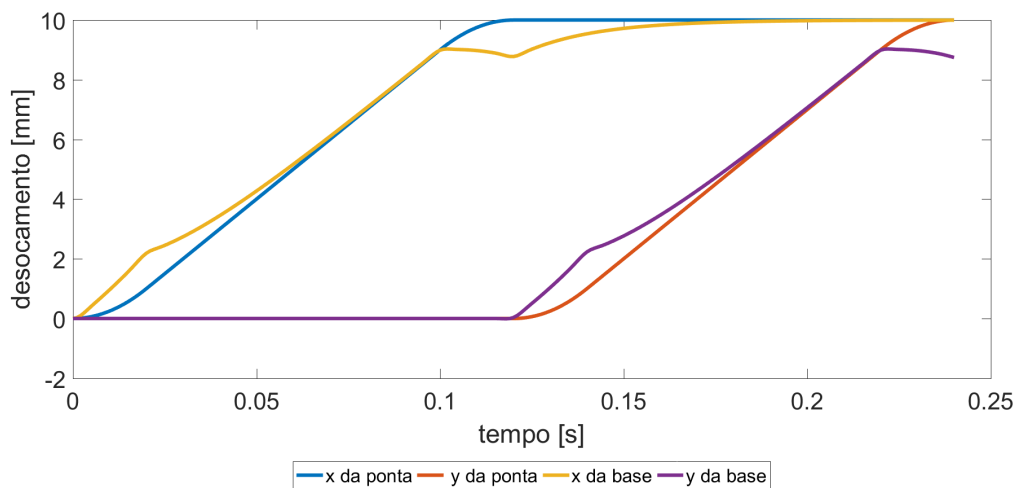


Figura 4.9 – Caso 1C - Comportamento no tempo dos deslocamentos em x e y da ponta e da base

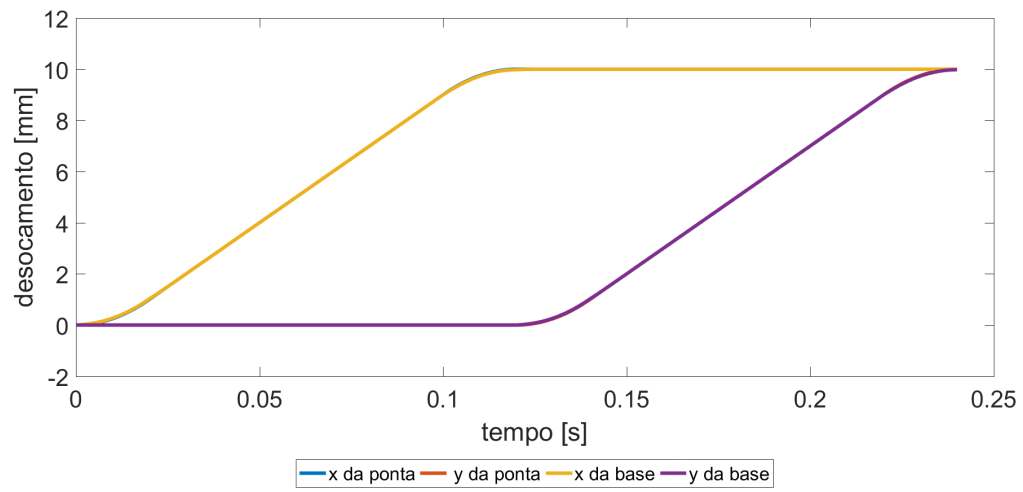


Figura 4.10 – Caso 1A - Caminho percorrido x vs y da ponta e da base

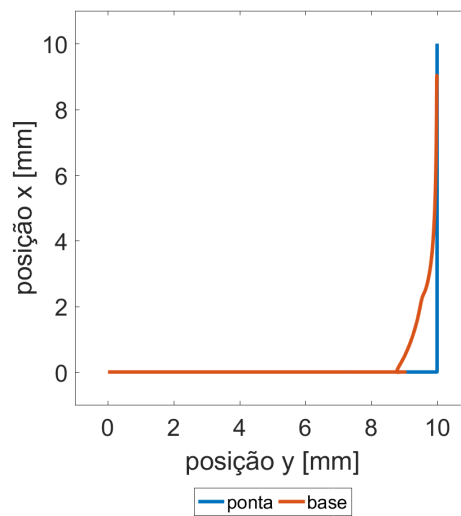
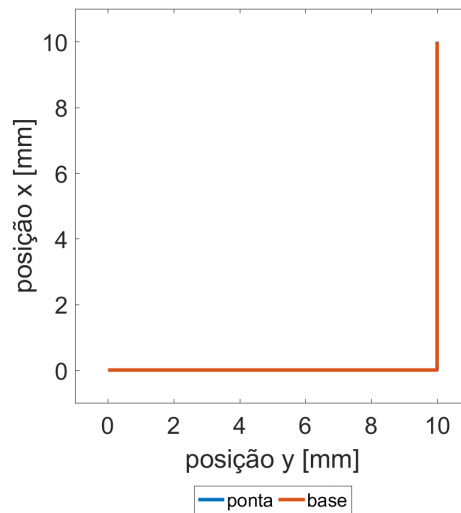
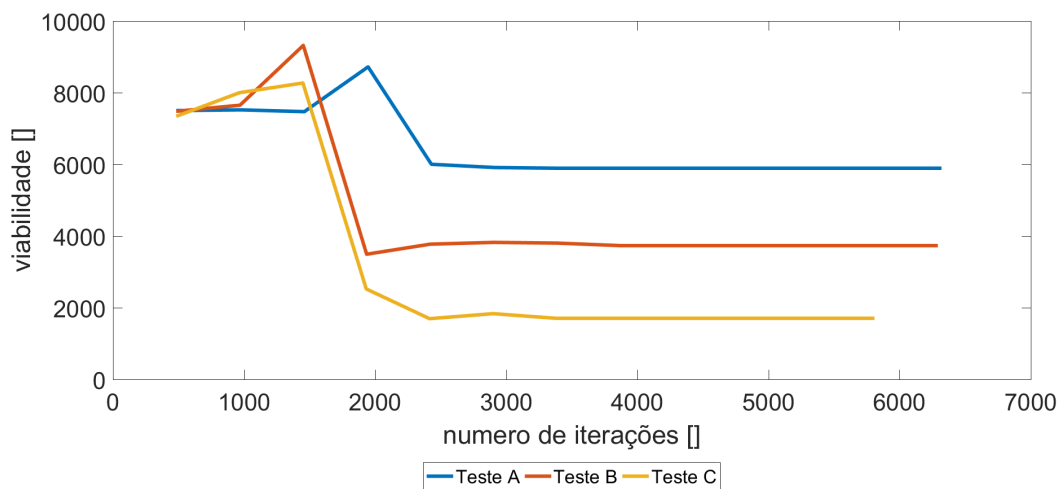


Figura 4.11 – Caso 1C - Caminho percorrido x vs y da ponta e da base



Além disso, a Figura 4.12 fornece uma visão geral da viabilidade do sistema sob diferentes configurações de frequência natural. Observamos um padrão de convergência similar em todos os casos, mas com uma viabilidade menor e, conseqüentemente, menores violações das restrições impostas nos casos de maior rigidez. Isso reflete um equilíbrio mais eficiente entre precisão e esforço computacional.

Figura 4.12 – Caso 1 - Num de fun x Viabilidade



Importante também é o impacto dessas variações na frequência natural no tempo de simulação. Observamos que os tempos de simulação variaram de 96,76 segundos no Caso A até 80,77 segundos no Caso C, com o Caso B intermediário, em 89,04 segundos. Esta tendência sugere que sistemas com maior rigidez (frequências naturais mais altas) não apenas melhoram a eficácia em termos de cumprimento das restrições, mas também podem levar a uma simulação mais rápida, apesar de todos os casos utilizarem vetores de posição com 241 elementos. Esta observação é crítica para aplicações práticas, onde a eficiência computacional e a rapidez na resposta são essenciais.

4.2.2 Caso 2 - Variação do coeficiente de amortecimento

Neste caso, investigamos o impacto de diferentes coeficientes de amortecimento no comportamento do sistema. As Figuras 4.13 e 4.14 são cruciais para esta análise, pois ilustram as trajetórias espaciais para as variações B e C do coeficiente de amortecimento. Estes resultados são contrastados com a Figura 4.3, que representa um coeficiente intermediário, oferecendo uma comparação direta entre os extremos e o caso base.

Figura 4.13 – Caso 2B - Caminho percorrido x vs y da ponta e da base

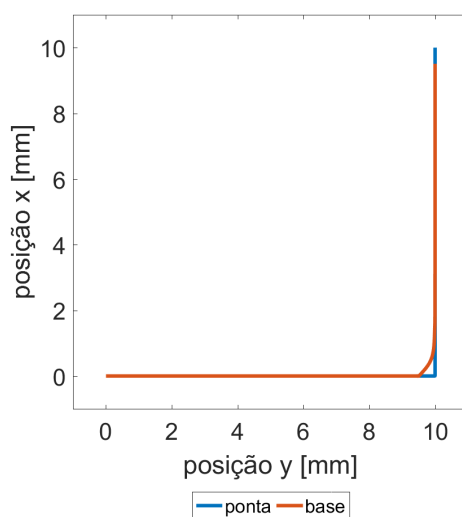
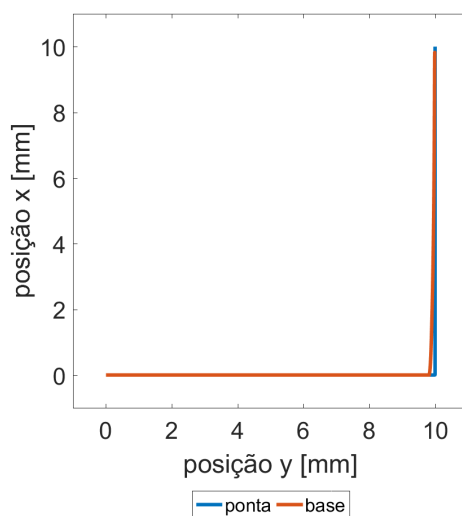
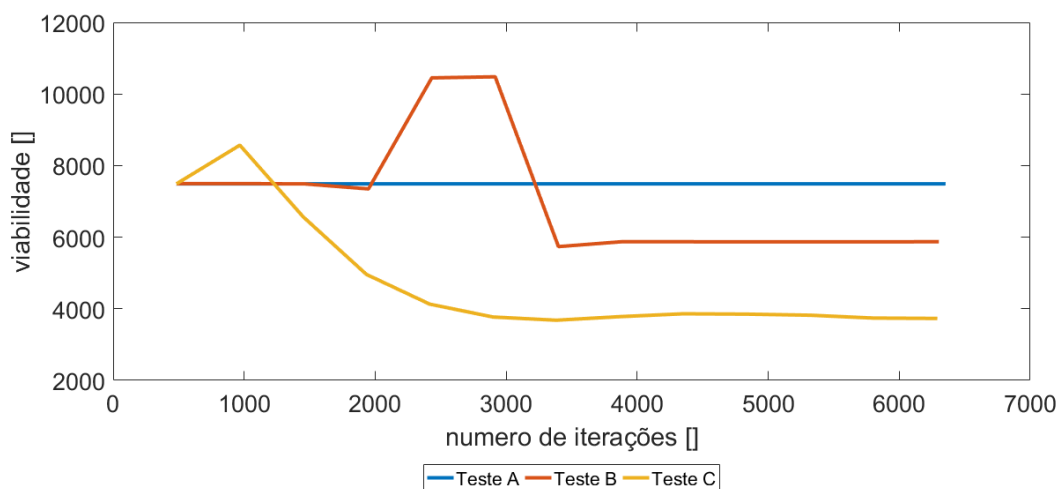


Figura 4.14 – Caso 2C - Caminho percorrido x vs y da ponta e da base



Curiosamente, o Caso A, que não conseguiu convergir e apresentou valores de entrada similares aos outros casos, serve como um exemplo notável de como variações no coeficiente de amortecimento podem levar a comportamentos divergentes. Este fenômeno é evidenciado na Figura 4.15, onde observamos uma tendência de convergência para valores de viabilidade menores à medida que o coeficiente de amortecimento aumenta. Isso sugere uma maior facilidade em compensar desvios quando o coeficiente é elevado.

Figura 4.15 – Caso 2 - Num de fun x Viabilidade

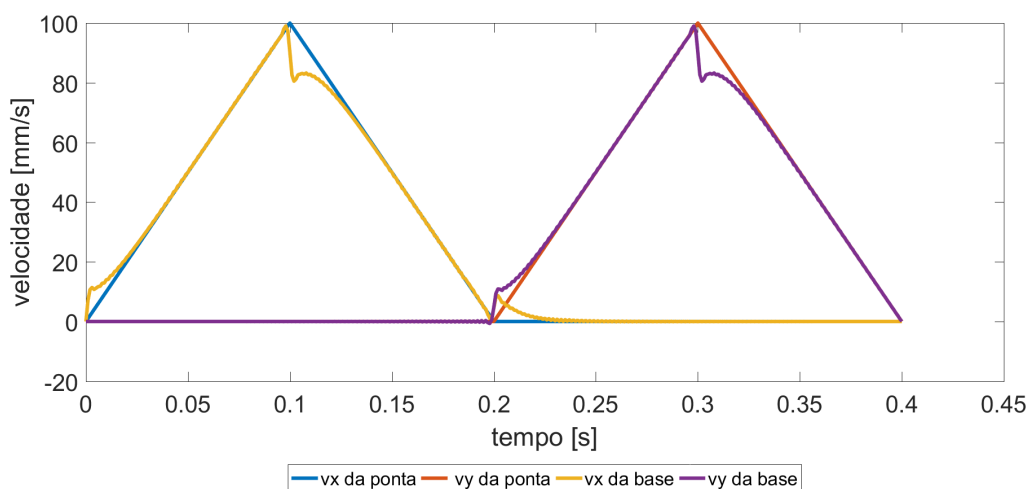


Em termos de desempenho computacional, todos os casos mantiveram uma consistência, com 241 elementos nos vetores de posição. Os tempos de simulação foram de 69, 89 e 52 segundos para os Casos 2A, 2B e 2C, respectivamente. Estes resultados indicam que, apesar das variações no coeficiente de amortecimento, a simulação manteve uma eficiência computacional comparável entre os diferentes casos.

4.2.3 Caso 3 - Variação na aceleração

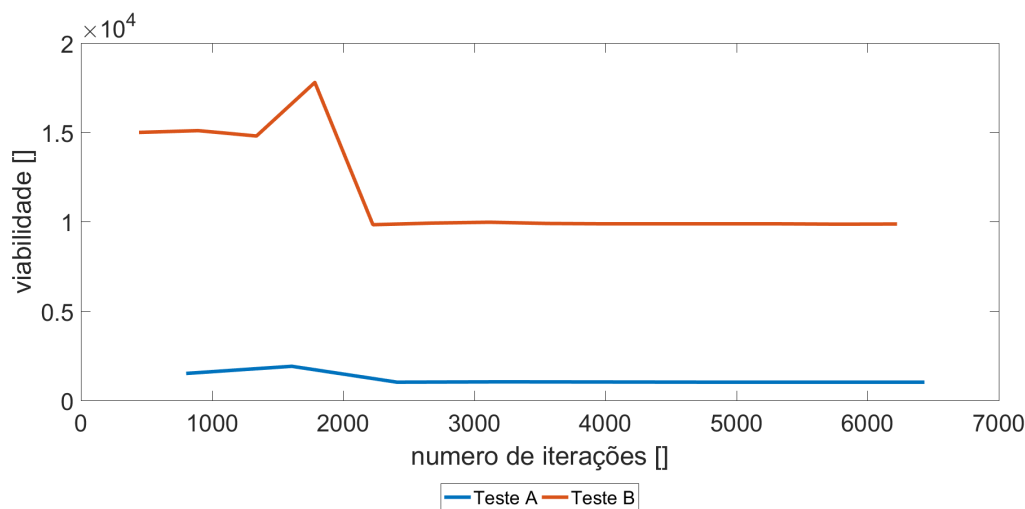
Neste caso, focamos na influência da variação da aceleração na geração de comandos do sistema. A Figura 4.16 é central nesta análise, pois ela revela uma forma distinta na curva de velocidade. Com a redução da aceleração, a velocidade não atingiu o patamar desejado, resultando em uma curva com formato triangular. Este comportamento destaca a sensibilidade do sistema às mudanças na aceleração e como isso afeta a capacidade de atingir velocidades alvo.

Figura 4.16 – Caso 3A - Comportamento no tempo das velocidades em x e y da ponta e da base



A Figura 4.17 fornece uma visão adicional sobre as dificuldades encontradas no Caso A. Ao contrário do Caso B, que mostrou um padrão de resposta semelhante aos casos anteriores, o Caso A enfrentou desafios significativos, como evidenciado por um aumento nos valores de viabilidade. Esta discrepância sugere que as variações na aceleração têm um impacto considerável na capacidade do sistema de satisfazer as restrições impostas.

Figura 4.17 – Caso 3 - Num de fun x Viabilidade



Uma possível explicação para as dificuldades observadas no Caso A pode ser encontrada no tamanho dos vetores de posição utilizados. No Caso A, o vetor de posições tinha 401 elementos e a simulação levou 195 segundos para ser concluída. Em contraste, o Caso B utilizou um vetor de 221 elementos e completou a simulação em 78 segundos. Estes resultados indicam que ajustes na aceleração podem ter implicações significativas não apenas no desempenho do sistema, mas também na eficiência computacional da simulação.

4.2.4 Caso 4 - Variação dos passos de tempo

Este caso destaca como diferentes resoluções na malha de tempo afetam o comportamento do sistema. As Figuras 4.18, 4.19 e 4.20 são fundamentais para entender a influência deste parâmetro nas oscilações observadas nos gráficos de velocidade. Cada uma destas figuras representa um cenário distinto, evidenciando como a resolução da malha de tempo pode alterar as dinâmicas de velocidade do sistema.

Figura 4.18 – Caso 4A - Comportamento no tempo das velocidades em x e y da ponta e da base

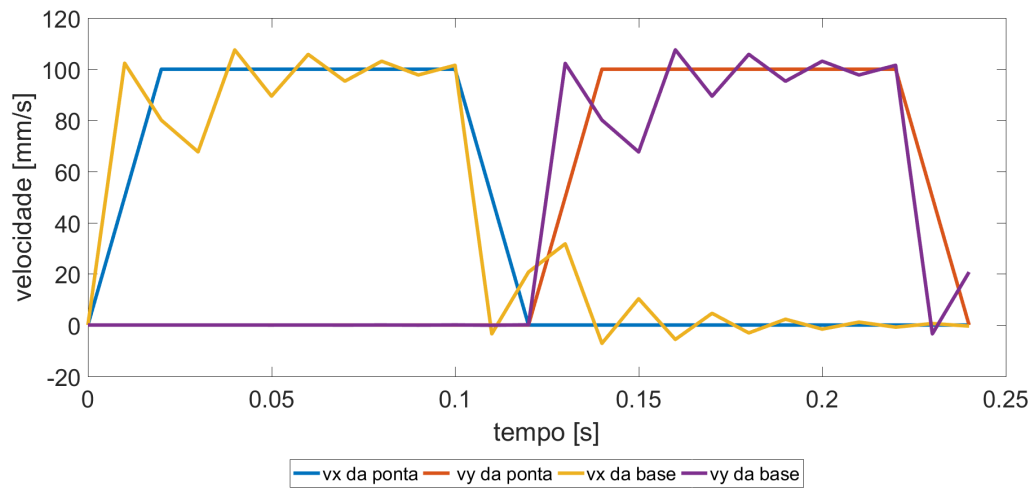


Figura 4.19 – Caso 4B - Comportamento no tempo das velocidades em x e y da ponta e da base

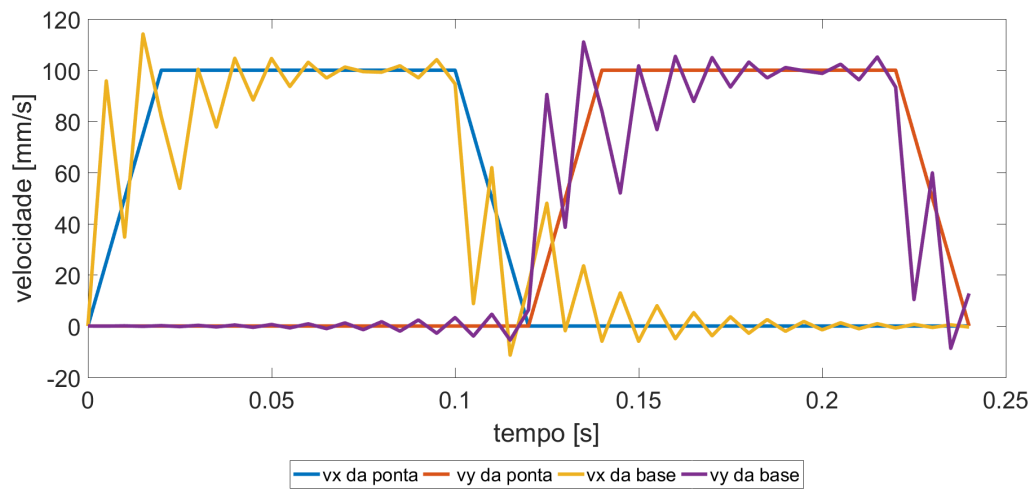
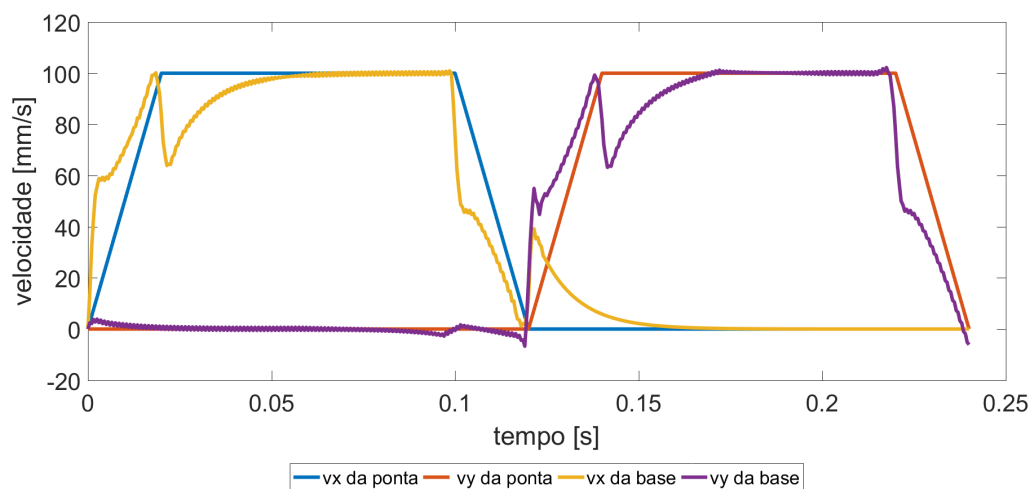


Figura 4.20 – Caso 4C - Comportamento no tempo das velocidades em x e y da ponta e da base



Por outro lado, ao examinar os gráficos de deslocamento nas Figuras 4.21, 4.22 e 4.23, notamos que as diferenças são menos pronunciadas. Isso sugere que, embora a resolução da malha de tempo tenha um impacto significativo na velocidade, seu efeito nos deslocamentos é mais sutil.

Figura 4.21 – Caso 4A - Comportamento no tempo dos deslocamentos em x e y da ponta e da base

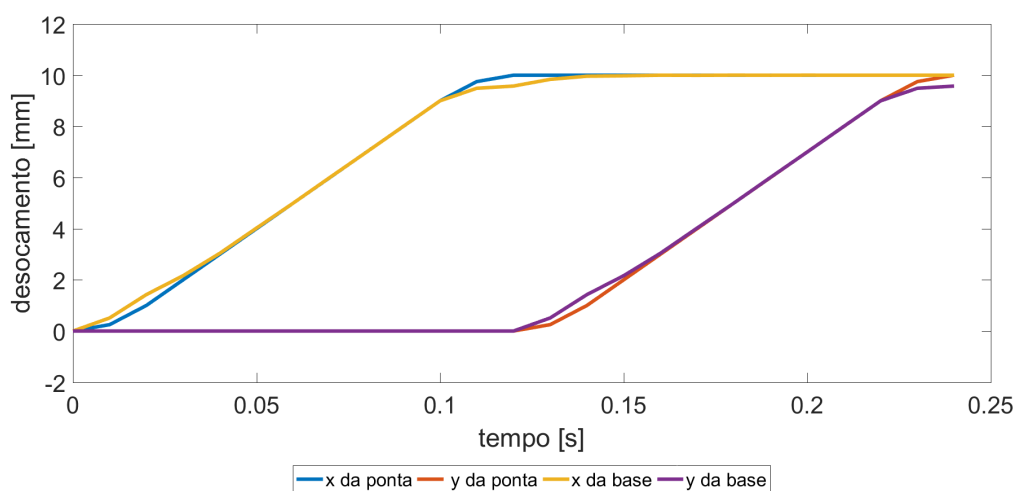


Figura 4.22 – Caso 4B - Comportamento no tempo dos deslocamentos em x e y da ponta e da base

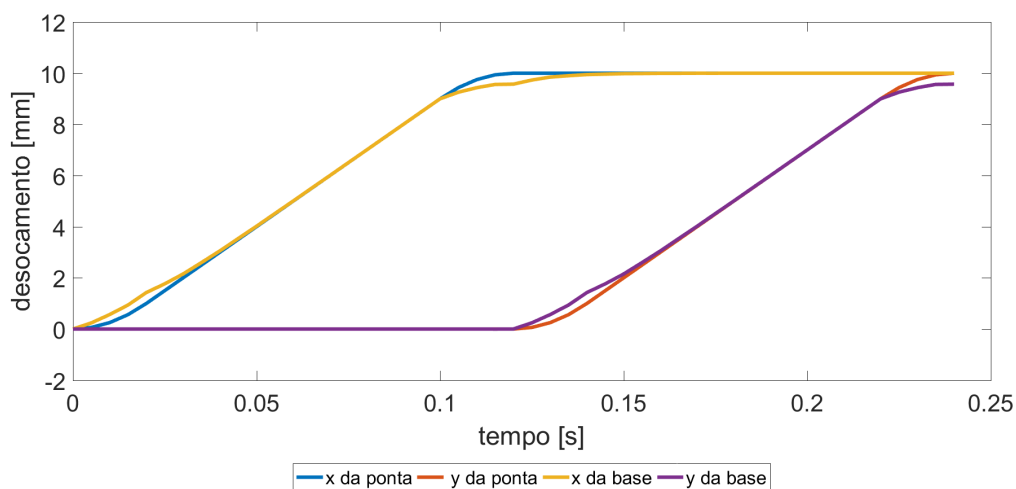
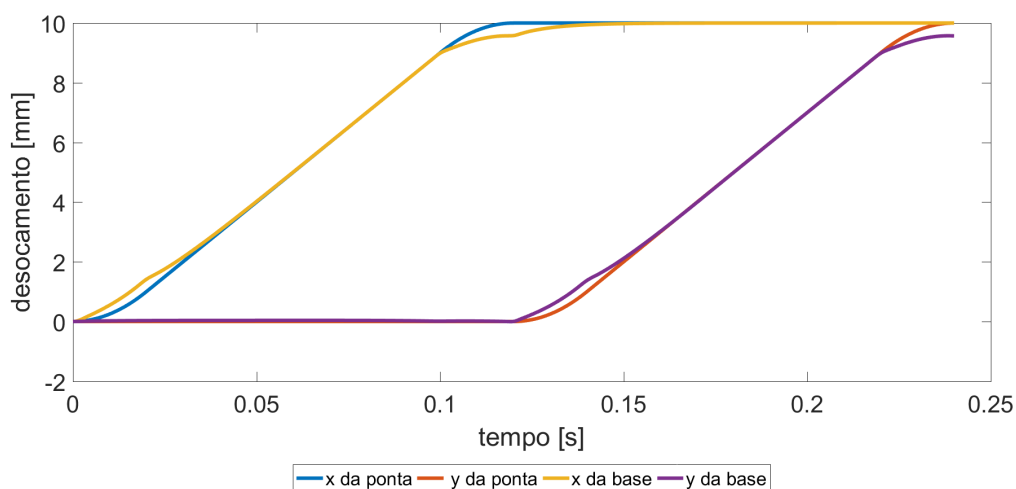
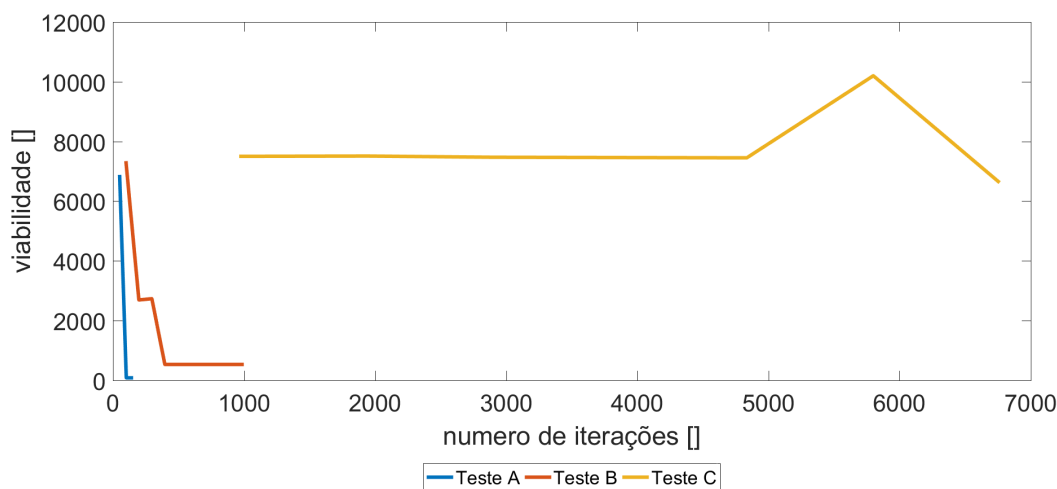


Figura 4.23 – Caso 4C - Comportamento no tempo dos deslocamentos em x e y da ponta e da base



Além disso, a Figura 4.24 revela o impacto dessas variações na viabilidade da simulação, sugerindo uma relação entre a resolução da malha de tempo e a eficácia do sistema em alcançar a convergência. Os tempos de simulação variaram consideravelmente entre os casos, sendo 0,7 segundos para o Caso A, 3 segundos para o Caso B e 279 segundos para o Caso C. Este padrão de tempo está diretamente relacionado ao tamanho dos vetores de posição, que foram de 25, 49 e 481 elementos, respectivamente, para os Casos A, B e C.

Figura 4.24 – Caso 4 - Num de fun x Viabilidade



Podemos notar também o impacto na velocidade e facilidade de se convergir, podendo ser observado na Figura 4.24, assim como os tempos de simulação 0,7s, 3s e 279s para os Casos A, B e C respectivamente. Além do tamanho dos vetores que está diretamente relacionado, respectivamente em 25, 49 e 481 para os Casos A, B e C.

4.2.5 Caso 5 - Variação da velocidade

Este caso foca na análise dos impactos causados por diferentes velocidades desejadas, ajustadas através do Gcode de entrada. Investigamos como as velocidades mais baixas e mais altas influenciam a dinâmica do sistema, tanto em termos de desempenho quanto de eficiência computacional.

No Caso A, onde a velocidade desejada é mais baixa, notamos um impacto significativo no tamanho dos vetores e no tempo de simulação. Devido à menor velocidade, o sistema leva mais tempo para completar o percurso, resultando em uma malha mais densa de pontos, dada a resolução de dt constante. Isso se traduz em um tempo de simulação mais longo de 219 segundos e 421 elementos nos vetores de posição.

Em contraste, o Caso B, com uma velocidade desejada mais alta, apresentou um tempo de simulação substancialmente menor de 60 segundos e 181 elementos nos vetores de posição. Interessante notar que, apesar da maior velocidade desejada e da manutenção da mesma aceleração máxima na geração de comando, a curva de velocidade neste caso assemelha-se à observada no Caso 3A, como mostrado na Figura 4.25. A correspondente curva de desloca-

mento é ilustrada na Figura 4.26, destacando as diferenças no comportamento do sistema sob essas condições.

Figura 4.25 – Caso 5B - Comportamento no tempo das velocidades em x e y da ponta e da base

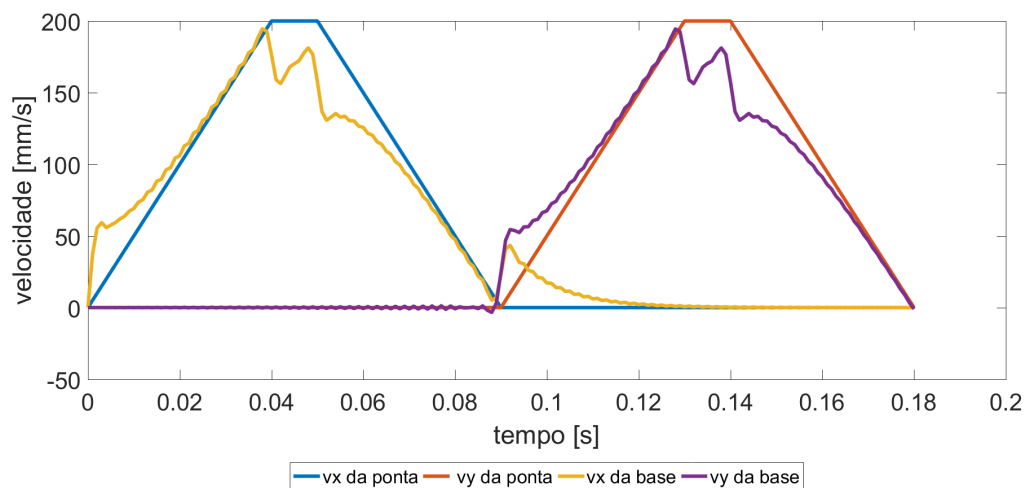
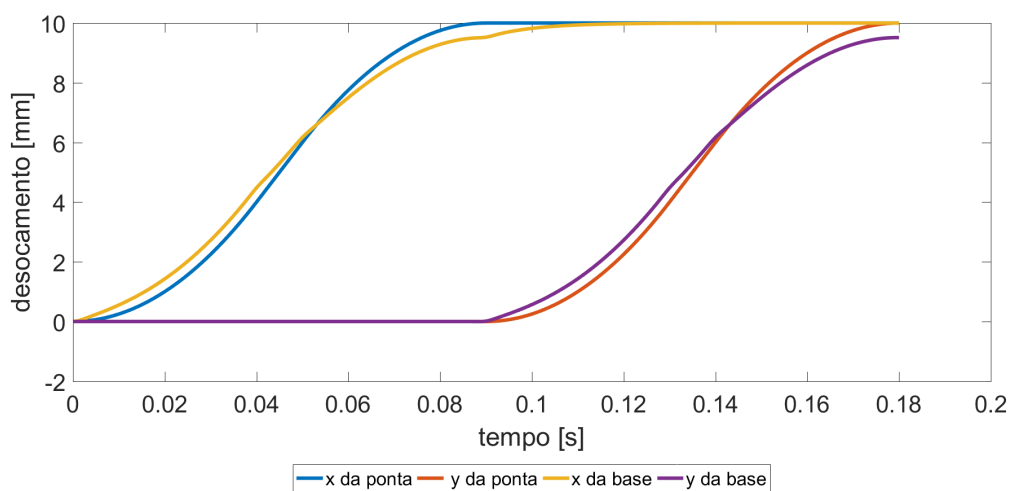
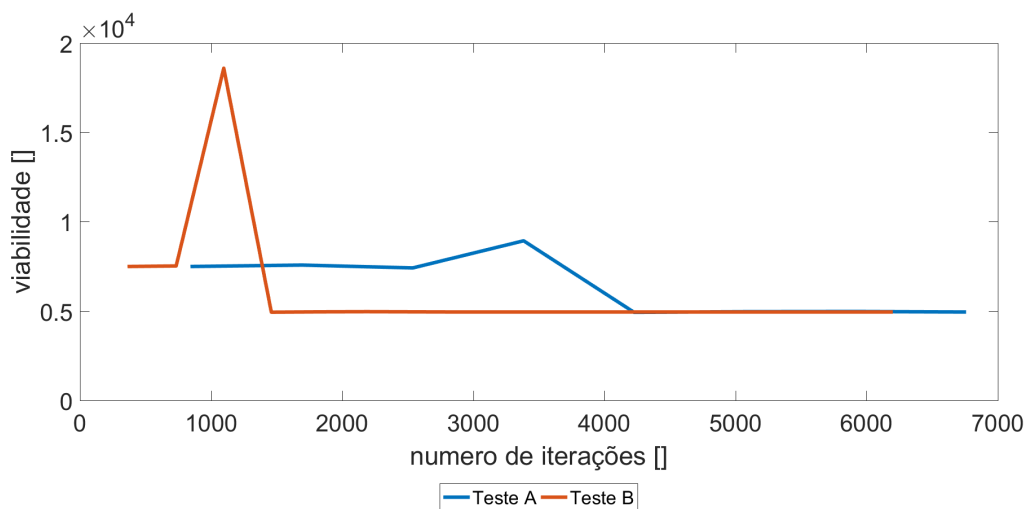


Figura 4.26 – Caso 5B - Comportamento no tempo dos deslocamentos em x e y da ponta e da base



Além disso, a Figura 4.27 permite uma comparação do padrão de convergência em diferentes velocidades. Observa-se uma variação na viabilidade e na capacidade de o sistema atender às restrições impostas, variando de acordo com a velocidade configurada.

Figura 4.27 – Caso 5 - Num de fun x Viabilidade



4.3 Discussão Integrada dos Resultados

Esta seção oferece uma análise integrada dos resultados obtidos nos diferentes casos de estudo, considerando as variáveis-chave e suas influências no comportamento do sistema simulado. A discussão foca em identificar padrões comuns, divergências e as implicações práticas dos resultados obtidos.

4.3.1 Influência dos Parâmetros Variáveis

A análise de sensibilidade realizada nos Casos 1 a 5 revelou que variações em parâmetros específicos, como frequência natural, coeficiente de amortecimento, aceleração, resolução de malha de tempo e velocidade desejada, têm impactos significativos e distintos no comportamento do sistema.

No Caso 1, observou-se que a variação da frequência natural afeta diretamente a amplitude dos desvios e a necessidade de compensação. O Caso 2 mostrou que diferentes coeficientes de amortecimento podem levar a variações notáveis na trajetória e na viabilidade do sistema. No Caso 3, a alteração da aceleração demonstrou como esse parâmetro pode influenciar a forma da curva de velocidade e a eficiência da simulação. O Caso 4 destacou a importância da resolução da malha de tempo, especialmente em relação às oscilações de velocidade. Por fim, o Caso 5 ilustrou como a velocidade desejada afeta tanto o tempo de simulação quanto o tamanho dos vetores de posição.

4.3.2 Comportamento e Convergência do Sistema

Em todos os casos, observou-se uma tendência de convergência para soluções viáveis, embora com variações na velocidade e facilidade desta convergência. Notavelmente, os casos com maiores desafios em termos de convergência e viabilidade foram aqueles com maiores variações nos parâmetros estudados. Isso sugere que, enquanto o sistema é robusto e adaptável a uma ampla gama de condições, existem limites críticos além dos quais a eficácia do sistema começa a diminuir.

4.3.3 Considerações futuras

Uma possível abordagem a ser explorada utilizando a ideia do método deste trabalho é a sobreposição de algoritmos, onde um método referenciado em uma planta do sistema poderia buscar remover uma parcela das vibrações, atuando de forma estagiada, com a participação de um método como *InputShaping* para atacar as vibrações remanescentes.

Uma possibilidade que o tipo de método abordado neste trabalho oferece é a capacidade de otimizar os parâmetros da planta para uma determinada posição. Assim, oferecendo a capacidade de se ajustar em grande nível de detalhe as peculiaridades do sistema, podendo até construir a malha utilizando sensores, semelhantemente a rotinas de configuração de *InputShaping* que amostram o comportamento em frequência no ponto central da impressora. Considerando também que a utilização desse tipo de malha, teria pouco impacto computacional.

5 CONCLUSÃO

Este estudo investigou a geração de comandos em impressoras 3D utilizando o método de *Fused Deposition Modeling* (FDM), com o objetivo principal de desenvolver metodologias que permitam velocidades de movimentação mais rápidas e garantam a precisão dimensional das peças produzidas. Através de um referencial teórico sólido e uma metodologia robusta, este trabalho abordou propostas relevantes sobre o controle de trajetória em impressoras 3D.

A análise de sensibilidade realizada nos diversos casos de estudo destacou como variações em parâmetros específicos, como frequência natural, coeficiente de amortecimento, aceleração, resolução de malha de tempo e velocidade desejada, impactam significativamente o comportamento do sistema simulado. Foi evidenciado que, apesar da robustez e adaptabilidade do sistema a uma ampla gama de condições, existem limites críticos nos quais a eficácia do sistema começa a diminuir.

Os resultados obtidos demonstraram a eficiência das técnicas e metodologias empregadas, fornecendo uma base sólida para a melhoria contínua do processo de manufatura aditiva por FDM. Este estudo representa um passo importante no aprimoramento da manufatura aditiva, oferecendo subsídios teóricos e metodológicos para a geração de comandos mais eficientes e precisos.

Em termos de aplicações práticas, o trabalho fornece importantes contribuições para a produção de peças com maior precisão dimensional e redução do tempo de fabricação, sendo um incentivo para pesquisas adicionais e avanços na área.

Por fim, este trabalho não apenas alcançou seus objetivos propostos, mas também abriu caminho para futuras investigações e melhorias na área de controle de trajetória para impressoras 3D. Espera-se que as contribuições deste estudo inspirem novas pesquisas e inovações, contribuindo para o avanço da manufatura aditiva como um todo.

REFERÊNCIAS

BIKAS, H.; STAVROPOULOS, P.; CHRYSSOLOURIS, G. Additive manufacturing methods and modelling approaches: a critical review. **The International Journal of Advanced Manufacturing Technology**, Springer, v. 83, p. 389–405, 2016.

DUAN, M.; YOON, D.; OKWUDIRE, C. E. A limited-preview filtered b-spline approach to tracking control—with application to vibration-induced error compensation of a 3d printer. **Mechatronics**, Elsevier, v. 56, p. 287–296, 2018.

GIBSON, I. *et al.* Applications for additive manufacture. **Additive Manufacturing Technologies: 3D Printing, Rapid Prototyping, and Direct Digital Manufacturing**, Springer, p. 451–474, 2015.

HAMILTON, J. D. State-space models. **Handbook of econometrics**, Elsevier, v. 4, p. 3039–3080, 1994.

HARGRAVES, C. R.; PARIS, S. W. Direct trajectory optimization using nonlinear programming and collocation. **Journal of guidance, control, and dynamics**, v. 10, n. 4, p. 338–342, 1987.

KLIPPER. **Klipper Kinematics Documentation**. 2017. Disponível em: <<https://www.klipper3d.org/Kinematics.html>>.

RAMANI, K. S.; EDOIMIOYA, N.; OKWUDIRE, C. E. A robust filtered basis functions approach for feedforward tracking control—with application to a vibration-prone 3-d printer. **IEEE/ASME Transactions on Mechatronics**, IEEE, v. 25, n. 5, p. 2556–2564, 2020.

SINGHOSE, W. E. **Command generation for flexible systems**. Tese (Doutorado) — Massachusetts Institute of Technology, 1997.

TURNER, B. N.; STRONG, R.; GOLD, S. A. A review of melt extrusion additive manufacturing processes: I. process design and modeling. **Rapid prototyping journal**, Emerald Group Publishing Limited, v. 20, n. 3, p. 192–204, 2014.

YU, K. *et al.* Application of the five-phase s-curve velocity model on fdm three-dimensional printer. In: IEEE. **2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)**. [S.l.], 2020. p. 1365–1371.