

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO



Mathias Moura Keller

**Projeto de exoesqueleto passivo para
reabilitação muscular com uso de molas não
lineares**

Projeto de Graduação

Projeto de Graduação apresentado ao Departamento de Engenharia
Mecânica da PUC-Rio

Orientador: Marco Antonio Meggiolaro

Rio de Janeiro
Novembro de 2023

AGRADECIMENTOS

Primeiramente, gostaria de agradecer a minha família pelo apoio incondicional dado a mim. Estiveram sempre me dando todo o apoio possível para que conseguisse realizar meus objetivos, por mais difíceis que fossem.

Agradeço ao meu orientador, Marco Antonio Meggiolaro, que, além de me orientar e apoiar durante minhas dificuldades, também confiou em meu potencial e capacidade de desenvolver este projeto.

Uma menção especial à equipe Riobotz, pois sem ela este trabalho não teria sido concluído. Além de ter sido uma parte importante da minha jornada acadêmica, também foi de muita importância para a conclusão deste trabalho.

Gostaria de mencionar e agradecer o apoio dos professores Adrian Giassone e Pedro Paulo Almeida Silva por seus conselhos e suporte no desenvolvimento do projeto.

Aos meus amigos Luiz Fernando, Ariel, Pedro, Thiago, Iago e Paulo não há palavras que possam descrever a importância de sua presença durante a vida acadêmica e fora dela. Acredito que não poderia ter chegado aqui se não fossem vocês.

Não poderia deixar de agradecer a Mariene Campos Cardoso Ribeiro. Apesar de não estar mais entre nós, sinto a necessidade de eternizar a importância que ela teve em minha vida e no meu desenvolvimento como pessoa. Seus conselhos, seu apoio e sua presença foram e sempre serão a base para quem eu desejo ser em meu futuro.

RESUMO

Projeto de exoesqueleto passivo para reabilitação muscular com uso de molas não lineares

O presente trabalho se propõe a gerar e analisar resultados de simulações referentes ao comportamento de molas lineares e não lineares na aplicação em exoesqueletos passivos, visando. Visando validar a viabilidade de utilização de molas não lineares para projetos mais eficientes e confortáveis para os usuários.

Para a realização das simulações foram estudados os comportamentos biomecânicos durante a subida de escadas, por ser um dos movimentos que mais demanda força no movimento do cotidiano.

A simulação foi realizada modelando o exoesqueleto como um braço robótico de 2 graus de liberdade e acoplando uma mola na modelagem da simulação. Assim podendo analisar as variações de energia do sistema em várias configurações e resistências mecânicas de molas.

Para validar a simulação foi feito um teste com um robô humanoide. O teste consistia em prender uma mola em sua perna e analisar o gasto de bateria utilizado ao fazer um movimento repetitivo. Assim podendo chegar à conclusão de que a aplicação de molas tem um efeito de acordo com a simulação realizada.

Ao final, comparando os resultados é possível notar uma aplicação para molas não lineares no sistema, estas apresentaram um comportamento mais atrativo ao serem comparadas com as molas lineares

Palavras chaves: Exoesqueleto passivo. Molas não lineares. Molas lineares. Matlab. Braço robótico. Robô humanoide. Reabilitação muscular

ABSTRACT

Passive Exoskeleton Design for Muscle Rehabilitation Using Non-Linear Springs

The present work aims to generate and analyze simulation results related to the behavior of linear and non-linear springs in the application of passive exoskeletons. The goal is to validate the feasibility of using non-linear springs for more efficient and comfortable designs for users.

For such simulations, biomechanical behaviors during stair climbing were studied, as it is one of the movements that demands the most force in daily activities.

The simulation was conducted by modeling the exoskeleton as a 2-degree-of-freedom robotic arm and coupling a spring in the simulation model. This allowed the analysis of energy variations in the system under various configurations and mechanical resistances of springs.

To validate the simulation, a test was conducted with a humanoid robot. The test involved attaching a spring to its leg and analyzing the battery consumption during a repetitive movement. This led to the conclusion that the application of springs has an effect consistent with the simulation.

In conclusion, comparing the results reveals a potential application for non-linear springs in the system. These exhibited a more attractive behavior when compared to linear springs.

Key-words: Passive Exoskeleton. Non-Linear Springs. Linear Springs. Matlab. Robotic Arm. Humanoid Robot. Muscle Rehabilitation

Sumário

1 Introdução	9
1.1. Motivação e Importância	10
2 Revisão bibliográfica	13
2.1. Exoesqueleto Ativo	13
2.2. Exoesqueleto Passivo	13
2.3. Estado da Arte	14
3 Fundamentação teórica	20
3.1. Modelagem como braço robótico	20
3.2. Equacionamento de molas não lineares	28
3.3. Biomecânica e atividade muscular	31
4 Simulação no Matlab	33
5 Testes no robô humanoide	52
6 Trabalhos futuros	57
6.1. Projeto de sistema não linear de molas	57
6.2. Modelagem considerando o terceiro elo integro	58
6.3. Construção de um protótipo	59
7 CONCLUSÃO	61
8 REFERÊNCIAS BIBLIOGRAFICAS	63
9 ANEXO A – Codido de Matlab para simulação 1	65
10 Anexo B – Função para simulação 1	72
11 Anexo C – Modelagem para primeira simulação	74
12 Anexo D – Codigo para segunda simulação	76

13 Anexo E – Função para segunda simulação	81
14 Anexo F – Modelagem para segunda simulação	83
15 Anexo G – Equações	85

Lista de figuras

Figura 1 - Gráfico de comparação entre mola linear, progressiva e atenuante	29
Figura 2 - Exemplo do movimento feito na primeira simulação	33
Figura 3 - Variação da posição angular de θ_1 e θ_2 durante a simulação.....	37
Figura 4 - Gráfico de comparação entre Torques e força gravitacional.....	38
Figura 5 - Demonstração do posicionamento de d e x	39
Figura 6 – Comparação do torque sem mola com o torque de diferentes valores de x	41
Figura 7 – Comparação dos torques para escolha da mola linear.	42
Figura 8 - Comparação entre a equação estimada e a equivalente para a mola 2 ...	44
Figura 9 - Comparação entre a equação estimada e a equivalente para a mola 3 ...	45
Figura 10 - Comparação entre a equação estimada e a equivalente para a mola 4 .	46
Figura 11 - Comparação do torque das 4 molas não lineares com o da mola linear.	47
Figura 12 - Gráfico de torque sem mola, da mola linear e da segunda mola não linear	48
Figura 13 - (a) Modelagem da primeira equação (b) modelagem da segunda equação.	49
Figura 14 - Exemplo do movimento simulado nesta parte	50
Figura 15 - Comparação dos torques sem mola, com mola linear e não lineares no movimento de levantar a perna.....	50
Figura 16 - Comparação do torque sem mola, com mola linear e com a mola não linear 2.	51
Figura 17 - ROBOTIS BIOLOID GP	52
Figura 18 - (a) início do agachamento (b) meio e final do agachamento	54
Figura 19 - Mola presa no robô humanoide	55
Figura 20 - Exemplo de molas em série.....	57

Lista de tabelas

Tabela 1 - Bateria do humanoide	53
Tabela 2 - Dados do teste sem mola.....	54
Tabela 3 - Dados da mola linear utilizada no humanoide.....	55
Tabela 4 - Dados do teste com molas.....	56

1 Introdução

Exoesqueletos passivos são uma forma inovadora de tecnologia assistiva. Surgiram como uma solução promissora para apoiar a movimentação de pessoas com restrições de mobilidade e para ampliar sua capacidade de força. À medida que a investigação e o desenvolvimento neste campo avançam, tornar-se-á cada vez mais evidente que estes dispositivos podem desempenhar um papel determinante na melhoria da qualidade de vida dos indivíduos que enfrentam limitações de mobilidade devido a lesões, doenças neuromusculares ou cirurgia.

São dispositivos mecânicos projetados para fornecer suporte estrutural e aliviar a tensão muscular sem que haja necessidade de motores ou baterias. Projetados para se adaptar às articulações e movimentos naturais do corpo humano, são ideais para uma variedade de aplicações, incluindo reabilitação muscular das pernas.

Para pacientes submetidos à reabilitação muscular dos membros inferiores, os benefícios são inequívocos. Em primeiro lugar, o uso de um exoesqueleto passivo reduz significativamente a tensão muscular necessária durante a locomoção, permitindo que os pacientes em reabilitação realizem movimentos controlados e repetitivos sem sobrecarregar os músculos em recuperação. Isto não só acelera o processo de recuperação, mas também ajuda a prevenir a fadiga muscular excessiva. O que poderia retardar e / ou complicar o processo de recuperação dos movimentos.

Por esse motivo, os exoesqueletos passivos podem ser incorporados às sessões de fisioterapia, permitindo que os pacientes realizem treinamento de força e amplitude de movimento de forma mais eficaz e segura. Além disso, esses dispositivos servem como suporte confiável e garantem controle e monitoramento dos movimentos durante o tratamento.

Outro benefício importante é o aumento da manobrabilidade. As pessoas em processo de reabilitação normalmente apresentam dificuldade em realizar tarefas simples, como caminhar ou subir escadas. Os exoesqueletos passivos aumentam essa mobilidade, permitindo que essas atividades sejam realizadas com mais facilidade e segurança.

Durante o processo de reabilitação dos músculos das pernas, prevenir quedas é uma preocupação constante e fundamental, já que novas quedas podem ocasionar novas lesões ou até mesmo agravar as já existentes e, por conseguinte, acrescentar complicações ainda maiores ao quadro de dificuldade de recuperação da mobilidade. Dessa forma, um exoesqueleto passivo atua como suporte de segurança, intervindo na manutenção da estabilidade do usuário, o que auxilia na redução da fadiga muscular e na melhora progressiva da mobilidade.

Os exoesqueletos passivos fornecem, assim, uma solução para reduzir a fadiga muscular, melhorar a mobilidade e prevenir quedas., intervindo significativamente na recuperação da qualidade de vida dos pacientes submetidos à reabilitação muscular das pernas, o que os auxilia na recuperação de sua independência motora.

1.1.Motivação e Importância

A pesquisa e desenvolvimento de exoesqueletos tornou-se uma área de estudo cada vez mais importante no mundo da tecnologia assistiva e da reabilitação física. Entre as abordagens e variações destes dispositivos, o exoesqueleto passivo destaca-se por vários motivos. Dois deles são destacados aqui como principais: sua manutenção é simples e possuem a capacidade de melhorar a mobilidade humana.

Além disso, os exoesqueletos passivos têm vantagens significativas em relação aos exoesqueletos ativos já que não dependem de baterias ou motores complexos. Isto significa que, se forem devidamente concebidos e ajustados, podem ser mais fáceis de manter e potencialmente durar mais tempo. Esta facilidade de utilização não só reduz os custos a longo prazo, como também aumenta a viabilidade e a disponibilidade para uma gama mais vasta de utilizadores.

Além disso, estes dispositivos passivos podem ser particularmente benéficos para indivíduos que procuram reabilitação muscular. Pessoas que enfrentam problemas de mobilidade devido a lesões, distúrbios neuromusculares ou acidentes podem utilizar exoesqueletos passivos como uma ferramenta valiosa no processo de recuperação. Ao fornecer suporte mecânico, eles permitem que os pacientes realizem

movimentos repetitivos e controlados, ajudando no fortalecimento dos músculos afetados e na restauração da função física.

Um aspecto que ainda não foi explorado extensamente na área de exoesqueletos passivos é o uso de molas não lineares. Tradicionalmente, esses dispositivos foram concebidos utilizando sistemas de molas lineares, mas a introdução de molas não lineares abre novas e animadoras possibilidades, já que essas molas se adaptam melhor aos movimentos naturais do corpo humano e proporcionam uma experiência de movimento mais natural

O estudo da aplicação de molas não lineares em exoesqueletos passivos é uma área de pesquisa promissora que poderá levar a avanços significativos na eficiência desses dispositivos. Isso inclui maior eficiência energética, redução do desgaste das articulações e uma experiência de usuário mais confortável.

Um ponto importante sobre exoesqueletos passivos é que podem ser integrados às sessões de fisioterapia, permitindo que os pacientes realizem exercícios de força e amplitude de movimento de forma mais eficaz e segura. Esses dispositivos atuam como suporte confiável, garantindo controle e monitoramento dos movimentos durante a terapia.

Outro benefício significativo é a melhoria da mobilidade. Pacientes em reabilitação diversas vezes enfrentam problemas para realizar tarefas simples, como caminhar ou subir escadas. Os exoesqueletos passivos proporcionam uma melhoria significativa nesta mobilidade, permitindo que estas atividades sejam realizadas com maior facilidade e confiança.

A prevenção de quedas é uma preocupação fundamental nos casos de reabilitação muscular das pernas, pois as quedas podem causar mais lesões e complicações. Os exoesqueletos passivos podem atuar como suporte de segurança, intervindo para manter a estabilidade do usuário e prevenir quedas prejudiciais.

Ao oferecer uma solução que alivia a fadiga muscular, melhorando a mobilidade e prevenindo quedas, os exoesqueletos passivos têm o potencial de melhorar significativamente a qualidade de vida dos pacientes em reabilitação muscular das pernas, ajudando-os a recuperar a independência e a funcionalidade.

Em suma, pode-se afirmar que a motivação para pesquisar exoesqueletos passivos e molas não lineares é encontrar soluções mais acessíveis, eficientes e versáteis para melhorar a mobilidade humana, ajudando àqueles que enfrentam os

desafios da reabilitação muscular a recuperarem a independência e a qualidade de vida.

2 Revisão bibliográfica

2.1.Exoesqueleto Ativo

De acordo com os métodos de potência usuais, os exoesqueletos podem ser classificados em exoesqueletos ativos e exoesqueletos passivos.

Os exoesqueletos ativos são caracterizados pelo uso de motores elétricos ou hidráulicos para sua movimentação, com a utilização de baterias ou fontes externas de energia (ZHOU *et al.*, 2020). Apesar de ser uma tecnologia efetiva em reabilitação, eles possuem a limitação de que o paciente não o controla completamente, se movimentando apenas em direções predeterminadas (ZHOU *et al.*, 2020).

2.2.Exoesqueleto Passivo

Nos exoesqueletos passivos não há a necessidade de uma fonte de energia externa para o seu movimento, de forma que estes movimentos estão sob o controle de quem o usa (ZHOU *et al.*, 2020). De forma geral, os exoesqueletos passivos são mais leves, mais baratos e os pacientes se sentem mais seguros do que nos exoesqueletos ativos (ZHOU *et al.*, 2020).

Este tipo de exoesqueleto possui molas em seu projeto por duas razões: para armazenar e retornar energia a cada passo e para balancear o peso dos membros ou do corpo humano (ZHOU *et al.*, 2020).

A maioria dos exoesqueletos passivos utilizam molas para compensar a gravidade da carga, já que elas possuem a vantagem de ser leves e fáceis de vestir. No entanto, são necessários instrumentos auxiliares, como, por exemplo, molas do lado exterior e isto aumenta a complexidade estrutural do esqueleto e o risco de ferimentos (ZHOU *et al.*, 2020).

2.3.Estado da Arte

A pesquisa de Zhou *et al.* (2020) buscou desenvolver um exoesqueleto passivo para os membros inferiores com juntas nos joelhos e quadril para auxiliar a locomoção de pessoas com disfunções motoras nestes membros. Neste projeto as molas estão localizadas no interior das juntas e a força gerada por elas é convertida em torque nas juntas dos joelhos e quadril por engrenagens, como pode ser visto na figura X.

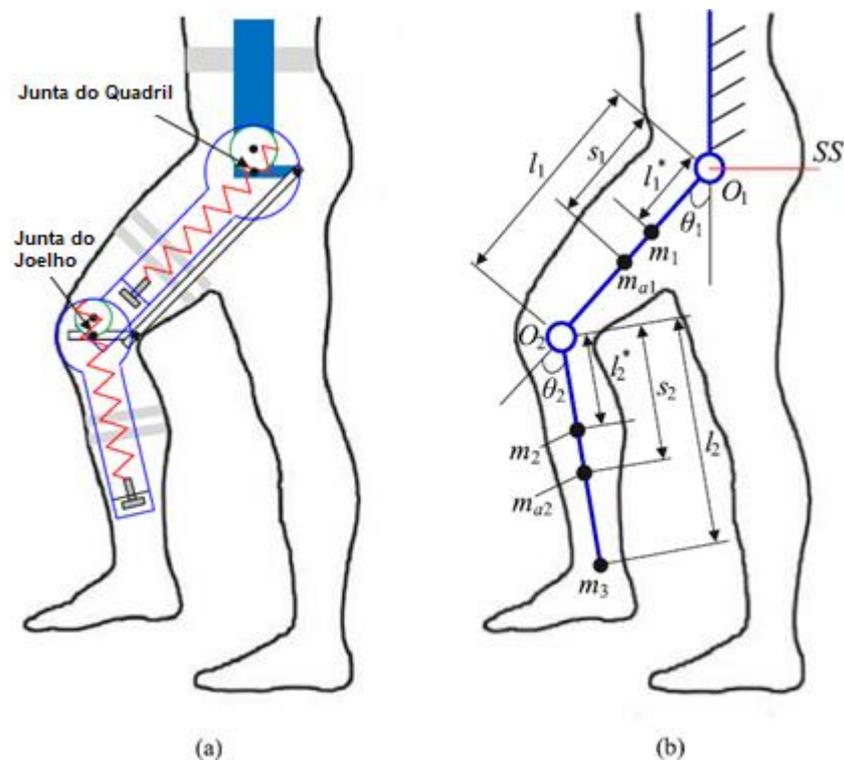


Figura 1. Modelo da perna de exoesqueleto humano (a) componentes básicos do exoesqueleto (b) termos e parâmetros.

O princípio do mecanismo do exoesqueleto proposto por Zhou *et al.* (2020) é mostrado na figura 1, onde é possível constatar que é composto por duas engrenagens, duas molas de compressão idênticas, uma mola tensora, um mecanismo de ajuste e uma ligação de saída.

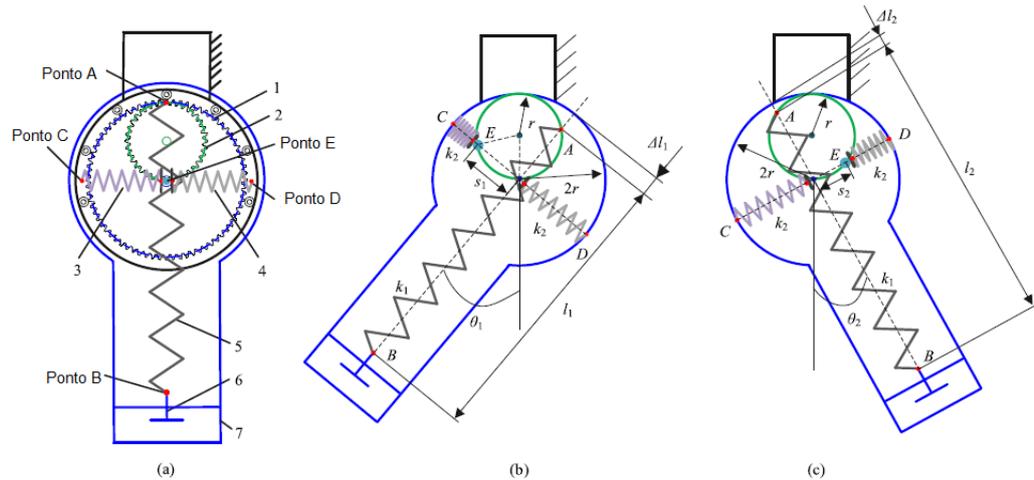


Figura 2. Princípio básico do projeto: (a) posição inicial; (b) rotação em sentido horário; (c) rotação em sentido anti-horário.

No ponto inicial, as molas estão livres e estão entre o ponto onde está fixado o pino e nos pontos C e D, que são fixos à engrenagem, a linha de centro da mola tensora passa através do centro de rotação das engrenagens no ponto A fixo ao pino até o ponto B que está fixo ao mecanismo de ajuste. A engrenagem maior e o mecanismo de ajuste estão montados na ligação de saída.

Foi desenvolvido um modelo cinemático deste mecanismo que foi otimizado e simulado em MATLAB para diversos cenários, além de se ter desenvolvido um protótipo e realizado experimentos com ele, obtendo-se resultados satisfatórios.

O exoesqueleto passivo proposto por Zhang, Tian e Guo (2009) baseou-se no veículo de reabilitação unidirecional mostrado na figura 3, cuja movimentação pode ser dividida em uma fase de apoio e uma fase de balanço.

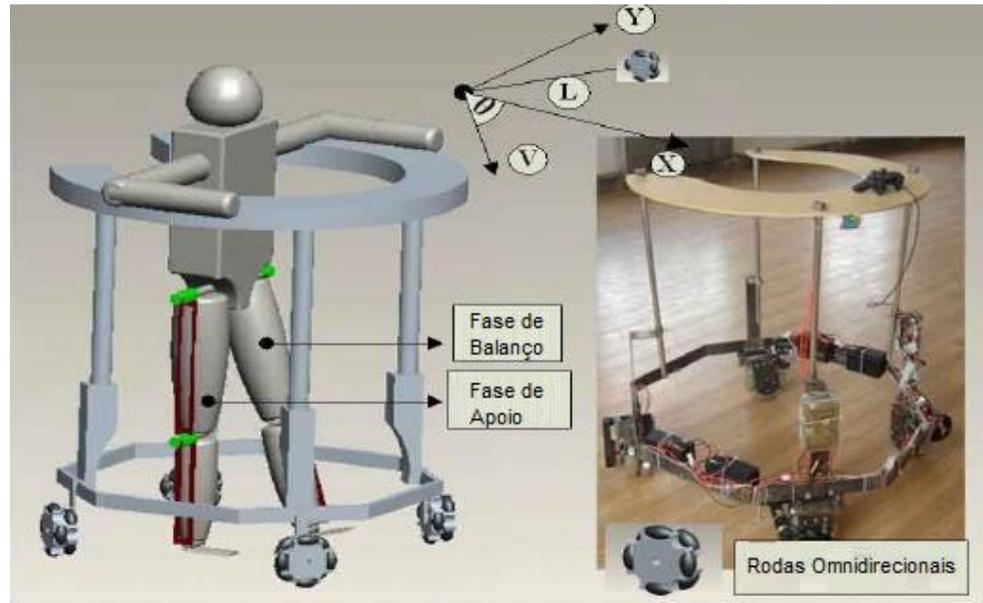


Figura 3. Veículos de reabilitação omnidirecionais.

O modelo cinemático foi desenvolvido levando-se em conta que na fase de apoio o ângulo da junta dos joelhos é praticamente fixo e o ângulo do quadril oscila do valor máximo ao valor mínimo, enquanto na fase de balanço, a junta do joelho gira do valor mínimo até o ponto fixo inicial. Este tipo de veículo é direcionado por quatro motores controlados por computador, permitindo que se movimente, andando em qualquer direção sem necessidade de giro.

O exoesqueleto aqui proposto utiliza molas nos joelhos e quadril para transmitir energia do veículo diretamente às juntas, possuindo também potenciômetros na junta para a medição do ângulo durante o andar e acelerômetros para controle e cálculo dos parâmetros biomecânicos da perna.

Foram realizadas simulações em MACCEPA, pelas quais foi encontrado o valor de torque de $7 \pm 0,5 \text{ N} \cdot \text{m}$ que minimiza o esforço físico. Foi desenvolvido um protótipo do exoesqueleto e foram coletados dados sobre seu desempenho com e sem o auxílio das molas, onde foi verificada uma compatibilidade.

No trabalho de Bartenbach Gort e Riener (2016) foi proposto um exoesqueleto para membros inferiores reconfigurável e modular que pode se adaptar aos usuários e aplicações. Foram desenvolvidos três exemplares para os testes do projeto: um exoesqueleto monoarticulado para apoio ao quadril, um exoesqueleto totalmente articulado para membros inferiores e um sistema de esteira para reabilitação, como mostrado na figura 4.

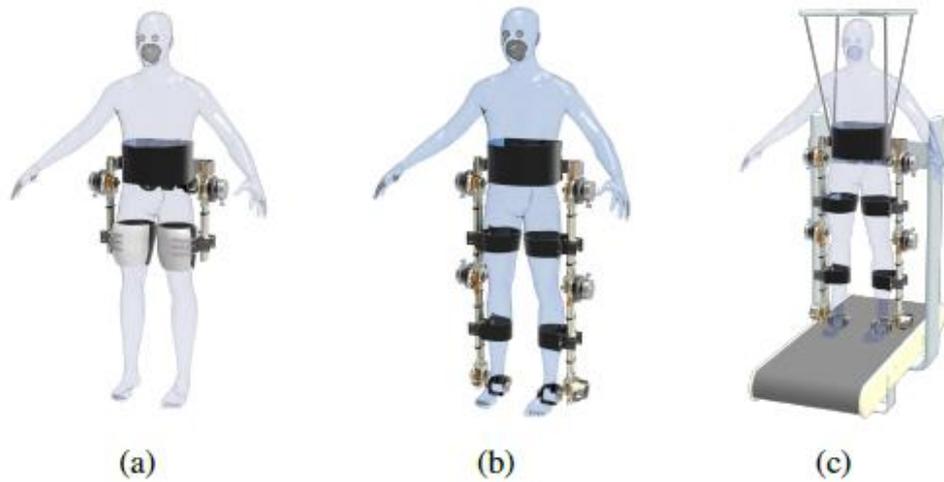


Figura 4. Exoesqueletos propostos: (a) exoesqueleto monoarticulado; (b) exoesqueleto totalmente articulado; (c) configuração proposta para andar estacionário.

O exoesqueleto pode ser ajustado para servir em usuários de 1,52m a 1,85m e larguras de pélvis de 30cm a 41cm, como é mostrado na figura 5.

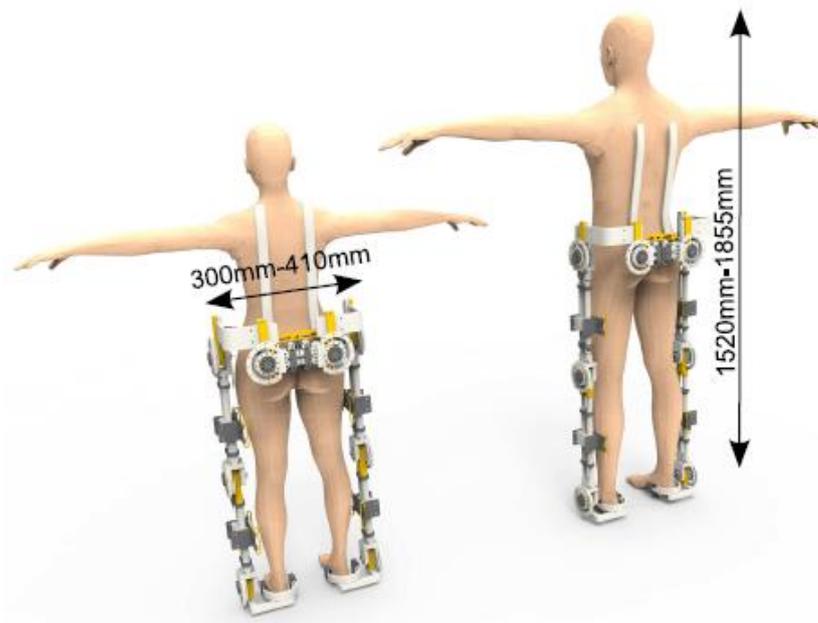


Figura 5. Exoesqueleto completo com dimensões ajustáveis.

Para que o exoesqueleto seja funcional foram inseridos atuadores em cada junta que promoveram 80 N*m de torque junto de duas molas para prover

estabilização da junta. Os resultados dos sistemas propostos indicam que a flexibilidade é um ponto positivo, enquanto a massa adicionada é um ponto negativo.

O exoesqueleto ativo proposto por Pirjade *et al.* (2020) possui atuadores nas juntas feitos de motores elétricos. Para seu projeto foi necessário obter os ângulos de junta para reproduzir o andar humano. Um protótipo foi produzido e testado, onde os dados de ângulo de junta foram coletados, como mostra a figura 6.

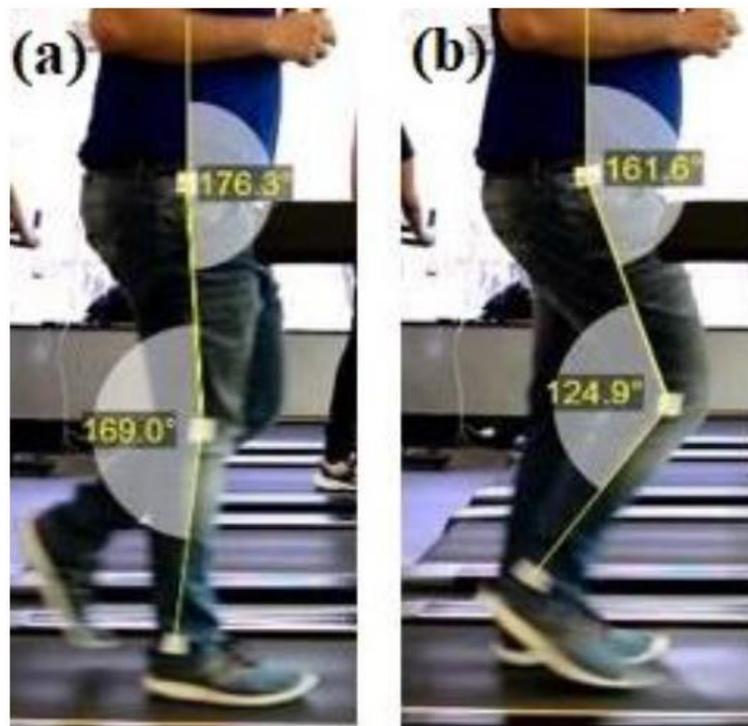


Figura 6. Ângulos de junta no joelho e no quadril em caminhada: (a) posição de pé completamente no chão; (b) posição de meio balanço

Em seguida, foram feitas simulações no OpemSim 4.0 para se obter o torque necessário durante o andar. Em seguida foi realizado o projeto do exoesqueleto de acordo com as especificações de carga e altura definidas em projeto, para então ser simulado em MATLAB Simulink e verificar a diferença entre os torques requeridos e obtidos.



Figura 7. Modelo em CAD do exoesqueleto.

No trabalho de Unluhisarcikli *et al.* (2011) foi desenvolvido um sistema de reabilitação chamado ANdROS, que aplica não só torques resistentes, mas também um torque corretivo na junta do joelho.

Para determinar as especificações de torque e potência do ANdROS, foi aplicada dinâmica inversa e trajetórias de referência de uma amostragem de pessoas saudáveis para definir uma trajetória de referência. A partir desta referência, o controlador de impedância aplica o torque corretivo na junta do joelho.

Os resultados do controle de impedância se mostraram positivos, apresentando uma relação estável e relação dinâmica entre o exoesqueleto e o usuário.

3 Fundamentação teórica

3.1. Modelagem como braço robótico

Primeiro, veremos a estrutura básica de um braço robótico de três elos rotativos em um plano 2D, esta será a base para o entendimento geral dos procedimentos e simulações abordados neste trabalho. O sistema consiste em uma “base” que representa o ponto de ancoragem do primeiro elo, em uma “extremidade” que é o ponto final do terceiro elo e dois “elos intermediários” que conectam esses pontos, formando uma cadeia cinemática.

No processo de modelagem, coordenadas cartesianas (x, y) são utilizadas para representar a posição da extremidade do braço robótico no plano 2D. Além disso, vários parâmetros são necessários para descrever o sistema. Neste trabalho serão utilizados **ângulos de junta** e **comprimento de elos**.

Cada junta permite rotação em relação ao elo anterior e os ângulos de juntas (geralmente chamados de θ_1, θ_2 e θ_3) determinam a configuração do braço robótico em determinado instante. Esses ângulos são medidos em relação a uma posição de referência, comparado ao segmento anterior. Já no comprimento dos elos (geralmente chamados de l_1, l_2 e l_3), são as dimensões que representam o distanciamento entre cada junta do braço robótico sendo essenciais para o cálculo da cinemática do sistema.

A cinemática direta é uma parte fundamental deste modelo, pois permite que a posição final da extremidade do braço robótico seja calculada com base nos ângulos das juntas e nos comprimentos dos elos. Através do uso de equações trigonométricas, podemos determinar as coordenadas (x, y) da extremidade do braço robótico da seguinte forma:

$$x_e = l_1 * \cos(\theta_1) + l_2 * \cos(\theta_1 + \theta_2) + l_3 * \cos(\theta_1 + \theta_2 + \theta_3) \quad (3.1)$$

$$y_e = l_1 * \sin(\theta_1) + l_2 * \sin(\theta_1 + \theta_2) + l_3 * \sin(\theta_1 + \theta_2 + \theta_3) \quad (3.2)$$

$$\Phi_e = \theta_1 + \theta_2 + \theta_3 \quad (3.3)$$

Essas equações descrevem como as posições x e y mudam à medida que os ângulos das juntas são ajustados, permitindo-nos determinar a posição da extremidade do braço robótico para diferentes configurações.

Com as coordenadas da extremidade do robô, é possível calcular sua matriz Jacobiana, a qual desempenha um papel crucial na modelagem e controle de braços robóticos. Ela fornece informações importantes sobre a relação entre as velocidades das juntas (articulações) e as velocidades do efetuador final do robô, sendo uma ferramenta essencial para análise cinemática e dinâmica.

$$J = \begin{bmatrix} \frac{dx_e}{d\theta_1} & \frac{dy_e}{d\theta_1} & \frac{d\Phi_e}{d\theta_1} \\ \frac{dx_e}{d\theta_2} & \frac{dy_e}{d\theta_2} & \frac{d\Phi_e}{d\theta_2} \\ \frac{dx_e}{d\theta_3} & \frac{dy_e}{d\theta_3} & \frac{d\Phi_e}{d\theta_3} \end{bmatrix} \quad (3.4)$$

A importância da matriz Jacobiana na modelagem de um braço robótico pode ser destacada em diversos aspectos, sendo alguns deles:

- Relação entre Espaço de Juntas e Espaço Operacional:

A matriz Jacobiana permite transformar as velocidades das juntas, que estão no espaço de configuração do robô, nas velocidades lineares e angulares do efetuador final, que estão no espaço operacional. Isso é crucial para controlar o movimento do robô e garantir que ele execute tarefas de forma precisa.

- Análise Cinemática Direta e Inversa:

Na análise cinemática direta, a matriz Jacobiana é utilizada para calcular as velocidades do efetuador final a partir das velocidades das juntas. Na análise cinemática inversa, ela é empregada para determinar as velocidades das juntas necessárias para alcançar uma determinada velocidade no espaço operacional.

- Singularidades e Redundância:

A matriz Jacobiana ajuda a identificar regiões no espaço de configuração do robô onde as soluções são singulares, o que pode ser crítico para evitar situações indesejadas. Além disso, em sistemas redundantes, a matriz Jacobiana é útil para resolver o problema de como distribuir as velocidades nas juntas para otimizar o desempenho do robô.

- Controle de Movimento:

No controle de movimento, a matriz Jacobiana é usada para projetar leis de controle, estas relacionam as velocidades desejadas no espaço operacional com as velocidades necessárias nas juntas. Isso é vital para o controle preciso e eficiente do braço robótico.

- Otimização de Desempenho:

Ao se projetar um braço robótico para tarefas específicas, a matriz Jacobiana é fundamental para otimizar o desempenho do sistema, ajustando as características cinemáticas para atender aos requisitos da aplicação.

Em resumo, a matriz Jacobiana é uma ferramenta poderosa na modelagem e controle de braços robóticos, fornecendo uma ponte essencial entre as variáveis no espaço de juntas e as variáveis no espaço operacional, facilitando a análise e o projeto eficiente desses sistemas complexos.

Outro ponto de extrema importância na modelagem são os centros de massa. Estes apresentam um papel crucial no desempenho e controle eficiente de braços robóticos compostos por elos homogêneos. O centro de massa de um elo é o ponto onde toda a massa do elo pode ser considerada concentrada, resultando em um comportamento dinâmico mais simples para o sistema.

Em elos homogêneos, onde a densidade de massa é uniforme, o centro de massa está localizado no ponto médio do elo. Esse ponto é essencial para a análise dinâmica e controle do braço robótico, pois influencia diretamente nas propriedades de inércia do elo.

Para encontrar as coordenadas dos centros de massa de cada elo, é feita uma modelagem trigonométrica similar aos pontos da extremidade. Cabe frisar que somente neste caso é feita uma equação para cada elo. Com isto em consideração, as equações para cada centro de massa e suas respectivas angulações são:

- Coordenadas do centro de massa do elo 1:

$$x_{c1} = l_{c1} * \cos(\theta_1) \quad (3.5)$$

$$y_{c1} = l_{c1} * \sin(\theta_1) \quad (3.6)$$

$$\phi_{c1} = \theta_1 \quad (3.7)$$

- Coordenadas do centro de massa do elo 2:

$$x_{c2} = l_1 * \cos(\theta_1) + l_{c2} * \cos(\theta_1 + \theta_2) \quad (3.8)$$

$$y_{c2} = l_1 * \sin(\theta_1) + l_{c2} * \sin(\theta_1 + \theta_2) \quad (3.9)$$

$$\phi_{c2} = \theta_1 + \theta_2 \quad (3.10)$$

- Coordenadas do centro de massa do elo 3:

$$xc3 = l_1 * \cos(\theta_1) + l_2 * \cos(\theta_1 + \theta_2) + l_{c3} * \cos(\theta_1 + \theta_2 + \theta_3) \quad (3.5)$$

$$yc3 = l_1 * \sin(\theta_1) + l_2 * \sin(\theta_1 + \theta_2) + l_{c3} * \sin(\theta_1 + \theta_2 + \theta_3) \quad (3.12)$$

$$\Phi_{c3} = \theta_1 + \theta_2 + \theta_3 \quad (3.13)$$

O próximo passo da modelagem é encontrar a matriz de inércia, a qual desempenha um papel crucial na modelagem dinâmica de um robô, fornecendo informações fundamentais sobre a distribuição de massa e inércia do sistema. Essa matriz, frequentemente denotada por H , é uma representação matemática que caracteriza a resposta do robô a forças e torques aplicados nas juntas.

Na análise dinâmica, a matriz de inércia é essencial para formular as equações que descrevem o movimento do robô. Ela relaciona as forças e torques nas juntas às acelerações resultantes, permitindo uma compreensão aprofundada do comportamento dinâmico do sistema em diferentes situações.

Durante o desenvolvimento de algoritmos de controle de movimento, a matriz de inércia é uma peça-chave. Ela é usada para projetar estratégias de controle que levam em consideração as características dinâmicas do robô, garantindo um movimento preciso e eficiente. Além disso, compreender a matriz de inércia é crucial para reduzir os esforços e torques necessários nas juntas, otimizando a eficiência energética do sistema.

Em simulações e análises virtuais, a matriz de inércia é empregada para prever o comportamento dinâmico do robô em diferentes condições. Isso é utilizado durante a fase de *design*, permitindo aos engenheiros otimizar a estrutura do robô para atender aos requisitos de desempenho específicos. Esta matriz de inércia também desempenha um papel na identificação de configurações singulares, nas quais o robô pode ter dificuldades em realizar certos movimentos. Evitar essas situações é essencial para garantir um controle suave e preciso do robô em uma variedade de tarefas.

Além disso, em aplicações que envolvem interação física com o ambiente, como controle de força ou manipulação colaborativa, a matriz de inércia é essencial para projetar estratégias de controle de impedância. Isso permite que o robô responda

de maneira adequada a forças externas, garantindo segurança e precisão durante a interação com seu entorno.

Para a modelagem dessa matriz, é necessário fazer o Jacobiano linear e angular das posições dos centros de massa. Este cálculo é feito da mesma maneira que se faz para o Jacobiano da extremidade, porém, para este caso específico, a matriz linear e angular serão separadas.

$$J_L = \begin{bmatrix} \frac{dx_c}{d\theta_1} & \frac{dx_c}{d\theta_2} & \frac{dx_c}{d\theta_3} \\ \frac{dy_e}{d\theta_1} & \frac{dy_e}{d\theta_2} & \frac{dy_e}{d\theta_3} \end{bmatrix} \quad (3.14)$$

$$J_A = \begin{bmatrix} \frac{d\Phi_c}{d\theta_1} & \frac{d\Phi_c}{d\theta_2} & \frac{d\Phi_c}{d\theta_3} \end{bmatrix} \quad (3.15)$$

Com esses dados para cada centro de massa dos elos do robô, é possível calcular a inércia do sistema. A matriz de inércia é encontrada a partir da formulação abaixo:

$$H = \sum_{i=1}^n (m_i * J_L^{(i)T} * J_L^{(i)} + J_A^{(i)T} * I_i * J_A^{(i)}) \quad (3.16)$$

Após a determinação da parte inercial da modelagem, deve ser feita a parte das forças centrífugas e de Coriolis, dois elementos essenciais na modelagem dinâmica de sistemas mecânicos, desempenhando papéis importantes na representação precisa do comportamento de robôs em movimento.

A força centrífuga surge quando um robô se move ao longo de um caminho curvilíneo e experimenta uma aceleração centrípeta devido à sua aceleração angular. A determinação destas forças é fundamental para a formulação das equações dinâmicas do robô. Sua presença nessas equações é não linear, assim contribuindo para a representação fiel das interações entre as velocidades articulares e as

velocidades angulares, tornando possível prever com precisão o comportamento dinâmico do sistema.

Por outro lado, a força de Coriolis, surge devido à aceleração angular, quando um objeto se move em um sistema de referência rotativa, e desempenha um papel significativo na modelagem dinâmica de robôs. Essa força é particularmente importante quando as juntas do robô estão se acelerando ou desacelerando durante o movimento. Assim como a força centrífuga, a força de Coriolis introduz componentes não lineares nas equações dinâmicas do robô, afetando diretamente a precisão da modelagem. Elas têm impacto direto no controle de movimento do robô, influenciando a formulação de algoritmos de controle, que buscam otimizar a trajetória e minimizar esforços desnecessários nas juntas. A inclusão adequada da força centrífuga e da força de Coriolis é, portanto, de extrema importância para um controle eficiente e para assegurar que o robô possa executar movimentos suaves e precisos.

Além disso, o reconhecimento adequado dessas forças é vital para a eficiência energética do sistema robótico. Modelar incorretamente a força centrífuga e de Coriolis pode levar a projeções imprecisas dos esforços necessários nas juntas, resultando em um uso ineficiente de energia. Portanto, a consideração cuidadosa dessas forças na modelagem dinâmica não apenas melhora a precisão da representação matemática do sistema, mas também tem implicações práticas importantes no controle, na eficiência energética e no desempenho global do robô em uma variedade de aplicações.

Para determinar o valor numérico dessas duas forças, aplica-se a formulação abaixo, sendo que os valores de i , j e k representam, respectivamente, a equação principal de qual elo pertence e as variáveis referentes ao seu comportamento e as forças centrífugas quando j e k tem o mesmo valor e de Coriolis quando j e k apresentam valores distintos. Abaixo é demonstrado também o vetor de variáveis do sistema (\mathbf{q}), utilizado em notação para facilitar a generalização das forças mencionadas.

$$\mathbf{q} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \quad (3.17)$$

$$h_{ijk} = \frac{\partial H_{ij}}{\partial q_k} - \frac{\partial H_{jk}}{2 * \partial q_i} \quad (3.186)$$

A última componente física externa da modelagem do sistema seria a energia potencial. Na modelagem de um robô de 3 elos, a energia potencial gravitacional representa a energia que cada elo do robô, ao ser elevado em relação a um ponto de referência, adquire devido à força gravitacional que atua sobre ele. Essa energia potencial pode ser expressa, de forma geral, usando a fórmula:

$$U = \sum_{i=1}^n (m_i * g * y_{ci}) \quad (3.19)$$

Com a formulação geral da energia potencial, pode-se encontrar a equação para a força gravitacional de cada elo individualmente. Essa conta é de alta importância para o cálculo do torque, sendo este o objetivo final desta modelagem. Para encontrar a força gravitacional de cada elo, é necessário fazer a derivada da forma geral da energia potencial pela variável correspondente ao elo desejado, como demonstrado abaixo.

$$G_i = \frac{\partial U}{\partial q_i} \quad (3.20)$$

Após o equacionamento das forças e componentes inerciais, faz-se necessário equacionar a última parte da modelagem pertinente a este trabalho. Sendo esta parte a força realizada por cada motor ao movimentar as variáveis do sistema (θ_1 , θ_2 e θ_3). Como todas as variáveis do sistema são rotativas, todas as forças calculadas serão torques. A formulação para esta grandeza segue abaixo.

$$\tau_i = \sum_{j=1}^n H_{ij} * \ddot{q}_j + \sum_{j=1}^n \sum_{k=1}^n h_{ijk} * \dot{q}_j * \dot{q}_k + G_i \quad (3.21)$$

Com esta formulação finalizada, toda a dinâmica do robô é, assim, conhecida e pode ser calculada.

Este capítulo forneceu uma análise detalhada e abrangente da modelagem do braço robótico, abordando aspectos cruciais relacionados às equações dinâmicas, inércia, torque e considerações gravitacionais. Ao estabelecer as bases teóricas necessárias, buscamos compreender o comportamento dinâmico do sistema. No entanto, esta modelagem é apenas o ponto de partida para o tema abordado no trabalho. Nos próximos capítulos, a pesquisa se aprofundará em áreas específicas que são de interesse para o entendimento e contextualização do projeto. A modelagem terá aplicação na validação em simulação e otimização de parâmetros propostos posteriormente, solidificando a relação entre teoria e prática. Ao finalizar este capítulo, estabelecemos uma base para a contextualização e entendimento dos procedimentos que serão utilizados em capítulos posteriores.

3.2. Equacionamento de molas não lineares

Para o entendimento de molas não lineares, primeiramente será feita uma breve abordagem sobre molas lineares, seu equacionamento e formulação da energia potencial elástica.

Inicialmente o comportamento linear é adquirido pela lei de Hook. Esta dita que a força elástica no sistema será uma relação multiplicativa da deformação sofrida pela mola e sua constante elástica, a qual representa a força necessária para haver uma deformação unitária (usualmente medida em metro) na mola.

$$F = k * x \quad (3.22)$$

Tendo a equação da força, é possível encontrar a energia potencial armazenada na mola, fazendo uma integração em relação com a deformação sofrida no sistema, como mostrado na equação abaixo.

$$U_{elast} = \int F(x)dx \quad (3.23)$$

$$U_{elast} = F * \frac{x^2}{2} \quad (3.24)$$

Após esta rápida análise, pode-se iniciar a comparação de comportamento. O primeiro ponto a se comentar é que existem dois tipos de molas não lineares: as progressivas e as atenuantes.

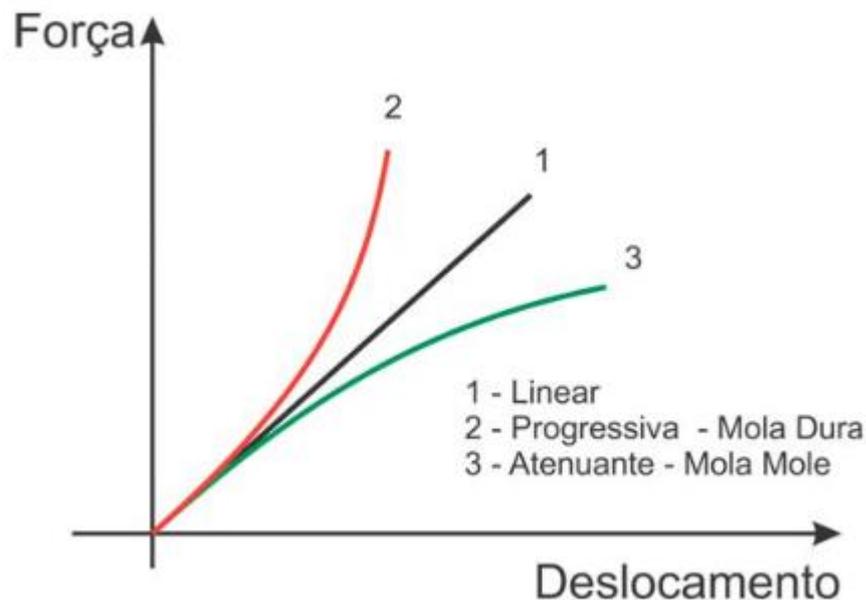


Figura 1 - Gráfico de comparação entre mola linear, progressiva e atenuante

Como demonstrado no gráfico acima, nota-se um comportamento específico para os 2 tipos de mola mencionados. A mola progressiva, também chamada de mola dura, apresenta um enrijecimento, fazendo com que a força necessária para que continue a deformação tenha um crescimento bem mais acelerado, comparado com uma mola linear.

O segundo tipo chamado de mola atenuante ou mola mole, apresenta um comportamento oposto à mola progressiva. A força necessária para fazer a deformação tem uma variação que a torna menos rígida conforme a deformação aumenta.

O estudo de Aranha (2016) fez uma comparação do comportamento não linear das molas com um oscilador de Duffing. Este é um tipo específico de sistema dinâmico não linear que exibe comportamento caótico e é frequentemente utilizado como modelo em problemas de mecânica, engenharia e física aplicada. Ele recebe esse nome em homenagem ao matemático holandês George Duffing, que contribuiu significativamente para o estudo de sistemas dinâmicos.

O sistema de oscilador de Duffing é descrito por uma equação diferencial não linear que representa a dinâmica de um objeto que oscila sob a influência de forças dissipativas e forças não lineares. A equação diferencial típica de um oscilador de Duffing é dada por:

$$F \cos(\omega t) = \ddot{x} + \delta \dot{x} + \beta x + \alpha x^3 \quad (3.25)$$

Os termos fundamentais na equação representam diferentes aspectos do sistema dinâmico. Utilizando-se da explicação dada por Aranha (2016), os termos têm o seguinte significado na equação:

- $\ddot{x} + \beta x$ – Esta parte representa o equacionamento de um oscilador harmônico;
- $\delta \dot{x}$ – Esta refere-se ao amortecimento no sistema;
- αx^3 – Parte correspondente a uma aproximação não linear do sistema, podendo haver outras, mas sempre mantendo um expoente ímpar;
- $F \cos(\omega t)$ – Corresponde a força externa oscilante no sistema.

Com as considerações acima, é possível representar a força de uma mola não linear utilizando-se da parte não linear e a parte não diferencial do oscilador harmônico, ficando da seguinte forma:

$$F = \beta x + \alpha x^3 \quad (3.26)$$

Com esta equação, que pode ser expandida com outros termos de não linearidade, foi generalizada uma equação que pode aproximar o comportamento das molas não lineares. Os termos β e α tem grande efeito na atuação da equação. Quando α apresenta valores positivos há um efeito endurecimento, já quando assume valores negativos apresenta um efeito de suavização. A repercussão da variação de β é um comportamento mais caótico quando apresenta valores abaixo de zero.

3.3. Biomecânica e atividade muscular

A biomecânica desempenha um papel importante no projeto e desenvolvimento de exoesqueletos de membros inferiores, influenciando sua eficácia e facilidade de uso. Este subcapítulo descreve a relação entre biomecânica e exoesqueletos e destaca a importância desses princípios no desenvolvimento de dispositivos eficazes e funcionais.

Um estudo realizado por Cenciarini (2011) destacou que embora os exoesqueletos tenham o potencial de melhorar o desempenho atlético e o desempenho físico, a literatura ainda carece de evidências fortes do sucesso dos exoesqueletos em pessoas saudáveis. Isso destaca a necessidade de uma compreensão mais profunda da biomecânica envolvida.

Um aspecto importante ao projetar um exoesqueleto é considerar parâmetros de projeto como cinemática, sua atuação e controle. As alterações nesses parâmetros dependem da aplicação específica do exoesqueleto. Os graus de liberdade (DOF) devem ser cuidadosamente selecionados para permitir movimentos naturais compatíveis com o corpo humano.

Além disso, a pesquisa destaca a importância de abordar as limitações de rotação do quadril e a OPD de joelhos e tornozelos. Outro elemento importante é a

análise dos torques articulares e seu papel no apoio ao movimento. A eficácia de um exoesqueleto está intimamente relacionada à sua capacidade de aplicar o torque correto às articulações para garantir um movimento suave e natural.

Afora a complacência cinemática e o comportamento dinâmico durante o exercício, a velocidade e a amplitude de movimento também são considerações importantes. A seleção do atuador e os requisitos de potência são fatores importantes na biomecânica do exoesqueleto. Um equilíbrio entre peso, propriedades inerciais e eficiência energética é essencial para garantir o conforto do usuário. Vale salientar que a forma como o exoesqueleto é fixado ao corpo também desempenha um papel importante na eficácia e facilidade de uso do dispositivo.

No entanto, é importante ressaltar que uma limitação significativa é a falta de avaliação quantitativa da eficácia da maioria das soluções de exoesqueleto dos membros inferiores. Mais estudos quantitativos são necessários para informar as decisões de design e validar empiricamente a eficácia desses dispositivos. A partir dessas considerações, pode-se inferir que a biomecânica desempenha um papel central no design de exoesqueletos dos membros inferiores.

Portanto, para superar as limitações atuais e alcançar melhorias significativas no desempenho humano, é necessária uma compreensão mais profunda dos aspectos biomecânicos envolvidos. A integração de dados quantitativos e a investigação contínua fazem-se imprescindíveis para melhorar a eficácia e a aplicação desses dispositivos, assim como para preparar o caminho rumo a um futuro promissor relativo à tecnologia de exoesqueletos passivos ou ativos.

4 Simulação no Matlab

Esta parte do trabalho foi focada em simular e testar o funcionamento do exoesqueleto como se fosse um braço robótico, sendo a primeira simulação feita com o movimento de agachamento e tendo como o primeiro elo a parte do pé e panturrilha, o segundo elo a coxa e o terceiro elo o torso. Esta divisão foi pensada para se ter uma distribuição de peso mais parecida com o real.

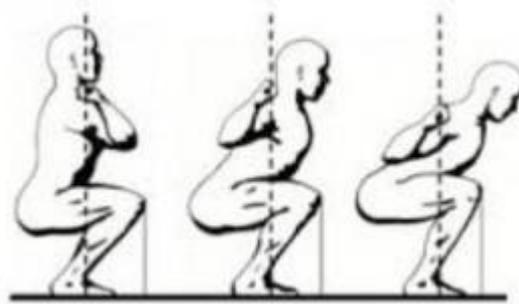


Figura 2 - Exemplo do movimento feito na primeira simulação

No intuito de simplificar os cálculos e, assim, diminuir as exigências computacionais, foi adotada a hipótese de que o elo 3 seria somente um ponto na extremidade do elo 2, contendo o restante da massa do corpo. Para testar a hipótese, foram refeitas os cálculos das equações até a matriz de inércia do sistema (eq 3.16)

Como o elo 3 é somente um ponto, as variáveis referentes a ele são todas nulas, inclusive sua inercia.

$$l_3 = 0 \quad (4.1)$$

$$l_{c3} = 0 \quad (4.2)$$

$$I_3 = 0 \quad (4.3)$$

Com esses valores em mente foram recalculadas as coordenadas dos centros de gravidades e suas Jacobianas respectivas.

Elo 1:

$$xc3 = l_{c1} * \cos(\theta_1) \quad (4.4)$$

$$yc3 = l_{c1} * \sin(\theta_1) \quad (4.5)$$

$$\phi_{c3} = \theta_1 \quad (4.6)$$

$$J_L = \begin{bmatrix} -l_{c1} * \sin(\theta_1) & 0 \\ l_{c1} * \cos(\theta_1) & 0 \end{bmatrix} \quad (4.7)$$

$$J_A = [1 \quad 0 \quad 0] \quad (4.8)$$

Elo 2:

$$xc3 = l_1 * \cos(\theta_1) + l_{c2} * \cos(\theta_1 + \theta_2) \quad (4.9)$$

$$yc3 = l_1 * \sin(\theta_1) + l_{c2} * \sin(\theta_1 + \theta_2) \quad (4.10)$$

$$\phi_{c3} = \theta_1 + \theta_2 \quad (4.11)$$

$$J_L = \begin{bmatrix} -l_1 * \sin(\theta_1) - l_{c2} * \sin(\theta_1 + \theta_2) & -l_{c2} * \sin(\theta_1 + \theta_2) & 0 \\ l_1 * \cos(\theta_1) + l_{c2} * \cos(\theta_1 + \theta_2) & l_{c2} * \cos(\theta_1 + \theta_2) & 0 \end{bmatrix} \quad (4.12)$$

$$J_A = [1 \quad 1 \quad 0] \quad (4.13)$$

Elo 3:

$$xc3 = l_1 * \cos(\theta_1) + l_2 * \cos(\theta_1 + \theta_2) \quad (4.14)$$

$$yc3 = l_1 * \sin(\theta_1) + l_2 * \sin(\theta_1 + \theta_2) \quad (4.15)$$

$$\Phi c3 = \theta_1 + \theta_2 + \theta_3 \quad (4.16)$$

$$J_L = \begin{bmatrix} -l_1 * \sin(\theta_1) - l_2 * \sin(\theta_1 + \theta_2) & -l_2 * \sin(\theta_1 + \theta_2) & 0 \\ l_1 * \cos(\theta_1) + l_2 * \cos(\theta_1 + \theta_2) & l_2 * \cos(\theta_1 + \theta_2) & 0 \end{bmatrix} \quad (4.17)$$

$$J_A = [1 \quad 1 \quad 1] \quad (4.18)$$

Tendo-se agora todas as equações necessárias para o cálculo da matriz de inercia, foram utilizados na equação [3.16] os resultados obtidos acima. Tendo como consequência a seguinte matriz de inércia:

$$H = \begin{bmatrix} \sigma_1 & \sigma_2 & 0 \\ \sigma_2 & I_2 + l_2^2 m_3 + l_{c2}^2 m_2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (4.19)$$

Sendo:

$$\sigma_1 = I_1 + I_2 + l_1^2(m_2 + m_3) + l_2^2 m_3 + l_{c1}^2 m_1 + l_{c2}^2 m_2 + 2l_1 l_2 m_3 \cos(\theta_2) + 2l_1 l_{c2} m_2 \cos(\theta_2)$$

$$\sigma_2 = l_2^2 m_3 + l_1 m_3 \cos(\theta_2) l_2 + l_{c2}^2 m_2 + l_1 l_{c2} m_2 \cos(\theta_2) + I_2$$

Como na matriz de inércia, não há nenhum componente relacionado ao elo 3 não nulo e a massa encontrada do elo 3 está presente nas equações, a hipótese adotada foi confirmada. As variáveis utilizadas na simulação foram as seguintes:

m1(kg)	5
m2(kg)	8
m3(kg)	67
l1(m)	0.53
l2(m)	0.44
lc1(m)	0.265
lc2(m)	0.22
I1(m ⁴)	0.1170
I2(m ⁴)	0.1291
g(m/s ²)	9.81

As posições iniciais das juntas foram pensadas para se evitar uma situação de singularidade, que seria quando $\theta_1 = 90^\circ$ e $\theta_2 = 0^\circ$. Com isto em consideração, a posição inicial do sistema foi projetada para ser o mais próximo da situação de singularidade sem comprometer a simulação. Assim, os valores iniciais para as variáveis angulares foi de $\theta_1 = 88^\circ$ e $\theta_2 = 4^\circ$. E sua posição final foi $\theta_1 = 50.1^\circ$ e $\theta_2 = 90.5^\circ$

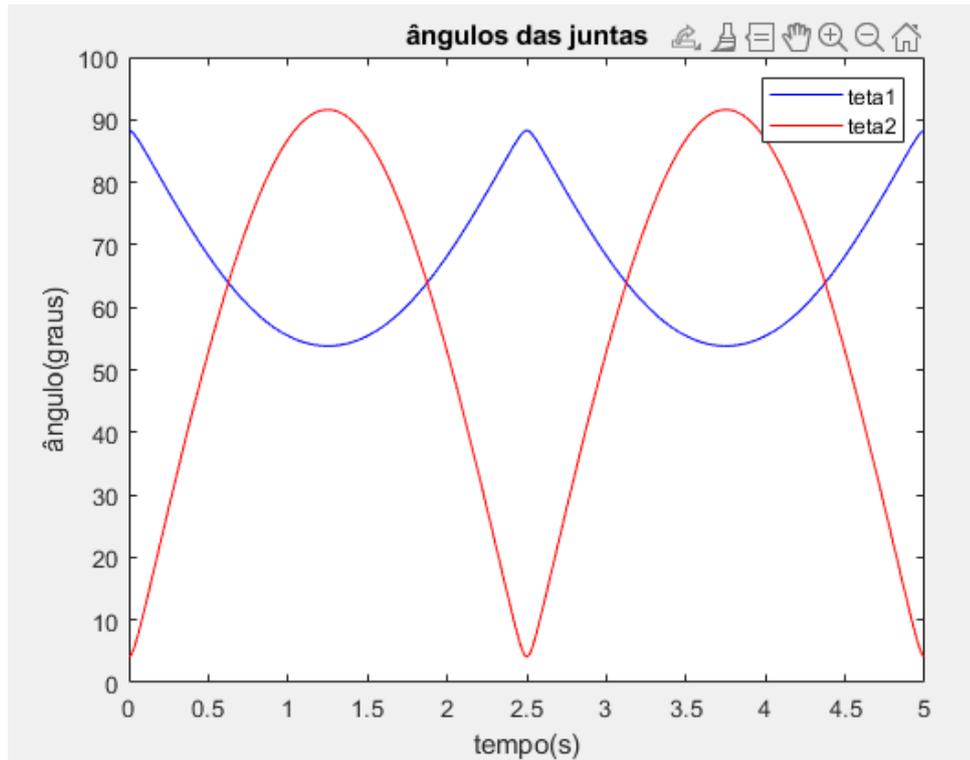


Figura 3 - Variação da posição angular de θ_1 e θ_2 durante a simulação.

Utilizando as equações (3.18). (3.20) e (3.21), foram calculados o torque e a força gravitacional instantâneos durante a simulação.

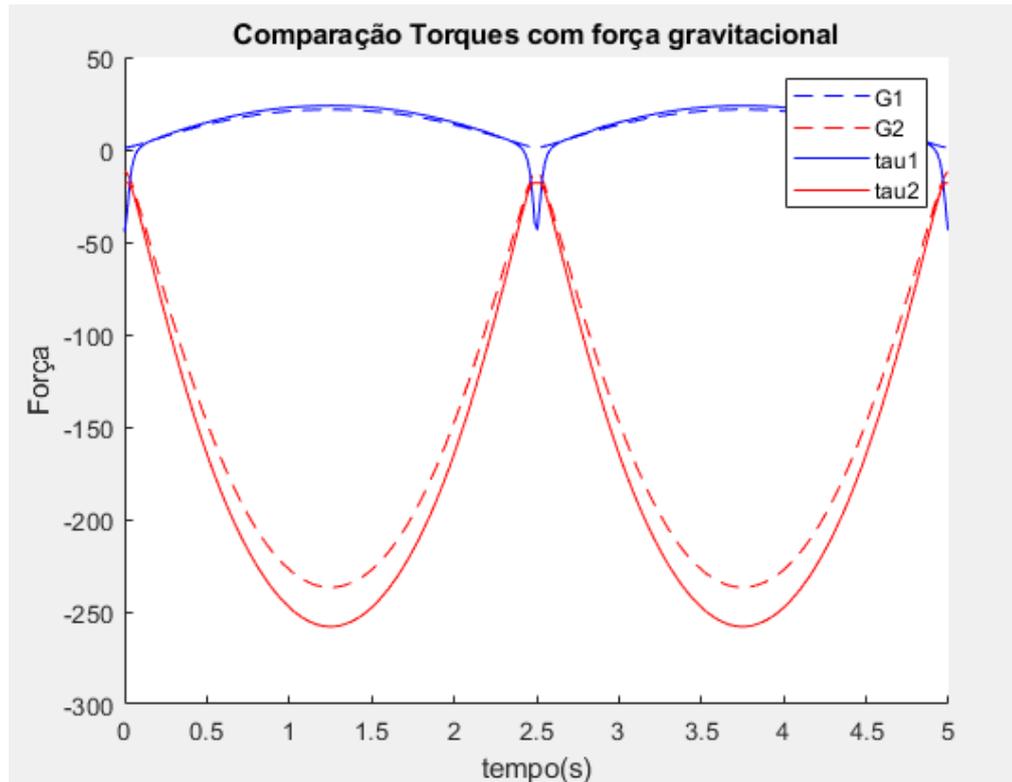


Figura 4 - Gráfico de comparação entre Torques e força gravitacional

Após a análise dos resultados obtidos, nota-se que há uma coerência com o comportamento esperado na situação simulada. Sendo que, quanto mais alto for o valor de θ_2 , maior será a componente do peso sendo sustentada pelos músculos da coxa.

Com esses resultados, pode ser iniciada a modelagem das molas. Primeiramente, deve ser selecionado um posicionamento ideal para a fixação do exoesqueleto na perna do usuário, sendo este posicionamento demonstrado na figura abaixo. A escolha da posição definitiva será dependente da intensidade em que a mola irá amenizar a carga imposta na coxa. Sendo d a distância entre o calcanhar e o ponto de fixação e x a distância desde o início do segundo elo até o ponto de fixação.

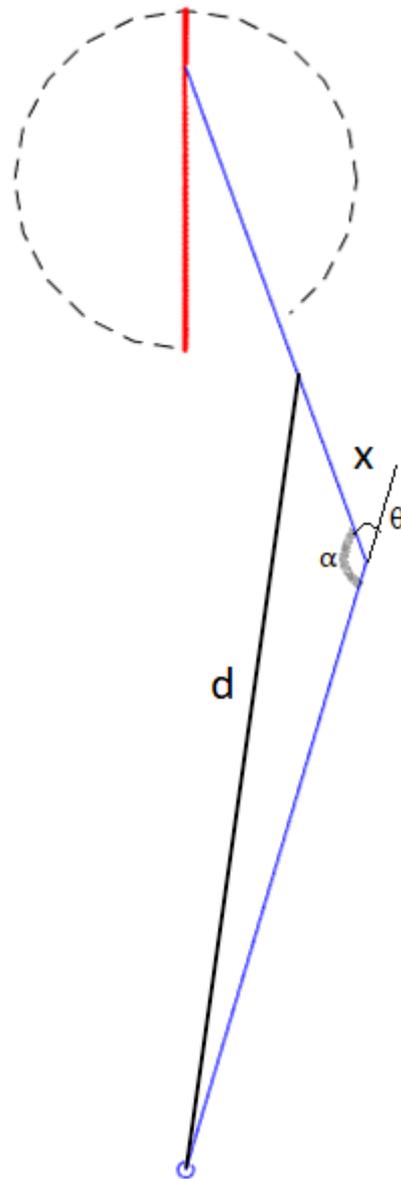


Figura 5 - Demonstração do posicionamento de d e x

Nota-se que na figura é formado um triângulo, o qual é formado pelos lados d , x e l_1 . Como l_1 é um valor conhecido e, segundo a notação de Denavit-Hatzenberg demonstrada na figura 9, é possível perceber que o ângulo formado entre os lados do triângulo x e l_1 , que será chamado de α , tem o seguinte valor:

(4.20)

$$\alpha = 180^\circ - \theta_2$$

Com isso, aplicando-se a lei dos cossenos, é possível se obter um valor de **d** dependente de θ_2 .

$$d = \sqrt{l_1^2 + 2 \cos(\theta_2) l_1 x + x^2} \quad (4.21)$$

Assim, foi estipulado um primeiro valor para a força máxima da mola. Este valor estipulado foi calculado para que, quando a mola estivesse comprimida, esta possuísse uma força equivalente a 35kg. Este valor para a mola é somente um valor inicial. Após a escolha da posição de fixação final, serão comparadas quatro molas lineares com diferentes forças elásticas para se determinar a mola linear definitiva do sistema. Abaixo estão os cálculos realizados para se encontrar a equação da mola:

$$F = m * g \quad (4.22)$$

$$F = k * (d_0 - d_{min}) \quad (4.23)$$

Considerou-se **d₀** o valor máximo de **d** para aquela configuração e **d_{min}**, o menor valor. Igualando estas equações será possível obter o valor de **k** para cada **x** diferente.

Para a seleção de fixação, foi colocado como limite superior o valor de **x** igual a metade de **I₂** e o menor quando **x** tiver o valor de um sexto de **I₂**. Esta seleção foi pensada para que haja uma distância entre o ponto mínimo e o joelho para que a força elástica possa gerar um torque e também para que o ponto de fixação não tenha uma elevação exagerada, pois isto iria restringir a liberdade de movimentação do usuário em algumas situações, como, por exemplo, no ato de se sentar.

Após essas considerações, foi feita a simulação com 4 possibilidades de valores para **x**:

Opção 1	Opção 2	Opção 3	Opção 4
$l_2/2$	$l_2/3$	$l_2/4$	$l_2/6$

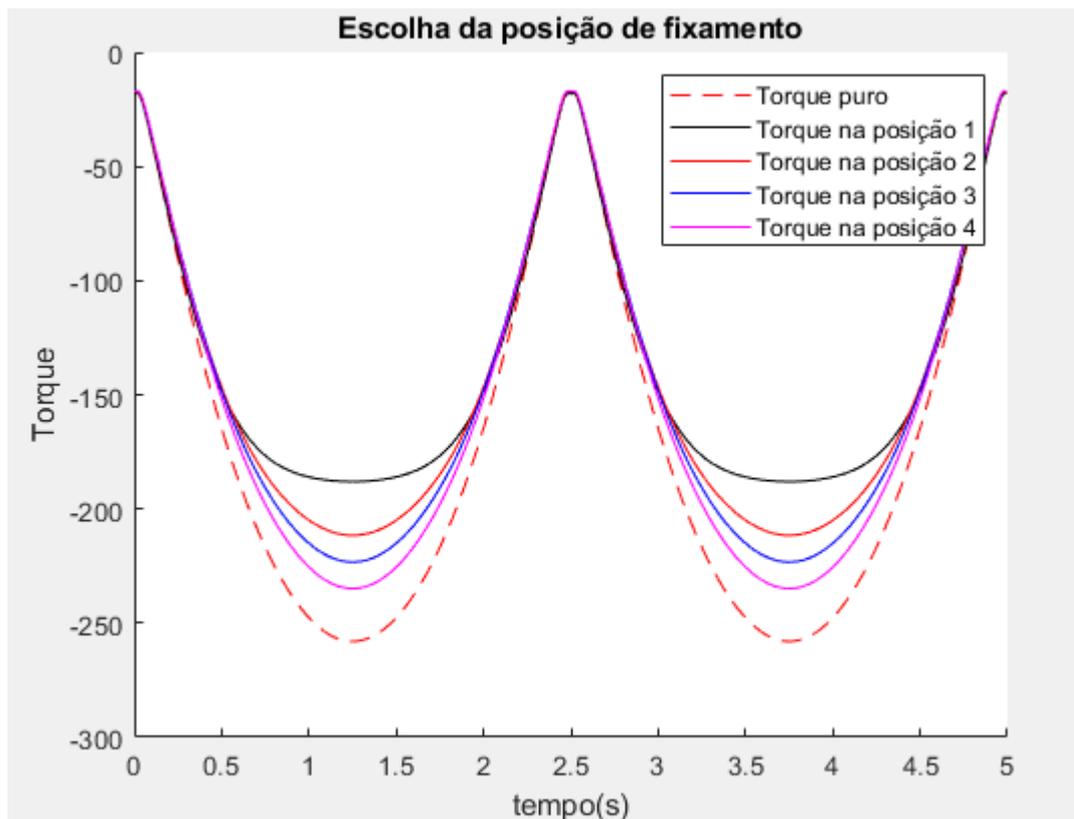


Figura 6 – Comparação do torque sem mola com o torque de diferentes valores de x .

Após a análise do gráfico, foi escolhido a opção 1. Já que nela houve a maior redução de torque avaliada. Assim fixando x para o valor de $0.22m$ ($\frac{l_2}{2}$) da extremidade do elo 1 ao longo do elo 2.

Com o posicionamento de fixação escolhido, foram calculadas mais três molas para serem comparadas com a inicial: uma com a força equivalente a 25kg (mola 2), outra a 45kg (mola 3) e a última a 55kg (mola 4). O gráfico abaixo mostra a comparação das quatro molas e o torque sem mola. Nota-se que todas as molas melhoram o sistema, porém constatou-se que a mola mais apropriada para o projeto seria a mola 3. Tendo em vista que a pesquisa tem como foco pessoas em reabilitação

muscular, isso faz com que o comportamento de um trecho de torque constante seja mais interessante para o usuário. Outra opção seria a mola 4, porém o comportamento de inflexão não iria oferecer o mesmo exercício disponível pela mola 3, podendo auxiliar ao usuário que volte à normalidade mais rapidamente.

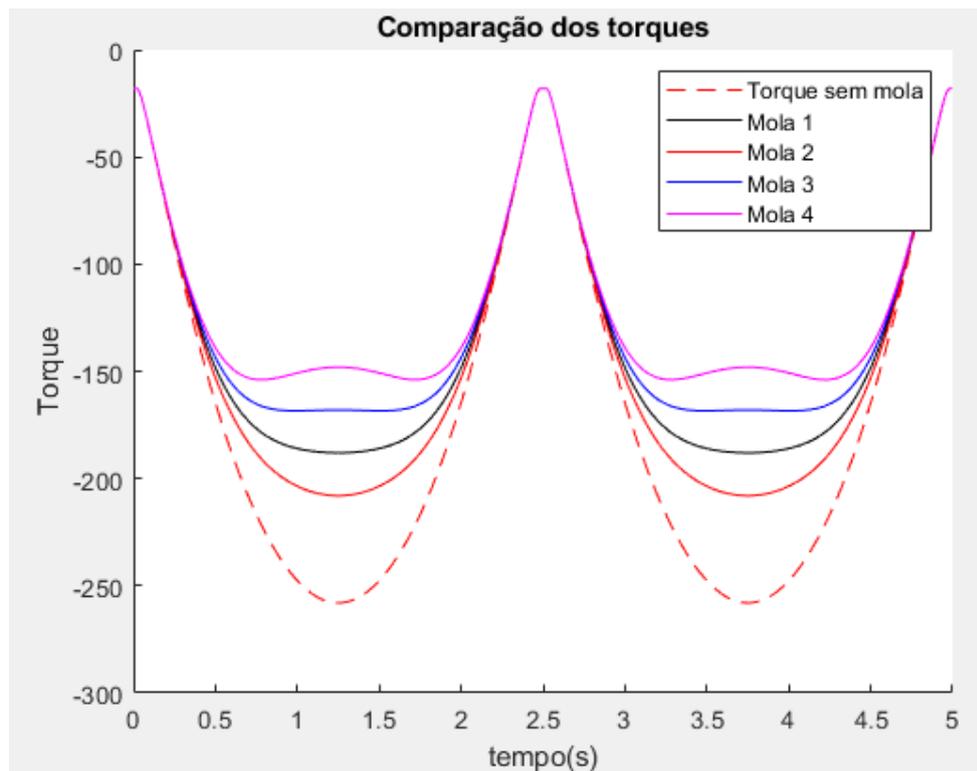


Figura 7 – Comparação dos torques para escolha da mola linear.

Após a escolha da mola linear foram estipuladas quatro equações de comportamento não linear, de modo a facilitar o cálculo. Porém, para se ater aos parâmetros estabelecidos pela literatura consultada, foram modeladas em correlação a cada mola uma equação em conformidade à encontrada na literatura, seguindo o estudo de Aranha (2016) sobre a equação de Duffing.

Todas as molas foram estipuladas utilizando como base a máxima deformação possível no sistema, essa informação foi encontrada comparando o valor de d_0 , por ser o valor máximo que d pode assumir (0.7496m) com o valor mínimo que d pode assumir (0.5721m)

A primeira mola foi estipulada com uma equação cúbica, então esta já estaria de acordo com a equação de Duffing, tendo o valor numérico estimado o valor de α e β sendo zero.

$$F = \left(\frac{441.45}{0.1775^3} \right) * (x)^3 \quad (4.24)$$

Já em relação às outras, foi necessário que se resolvessem igualdades para se encontrar os valores de α e β . Em sequência a isso, as equações foram comparadas à equação equivalente presente na literatura e se avaliou sua disparidade com o comportamento.

Para equacionar as molas da literatura precisaram ser resolvidas igualdades para encontrar os valores de α e β , sendo esse cálculo apresentado no anexo “Equações”. Com isto, as equações foram comparadas à equação equivalente constante na literatura consultada e foi avaliada sua similaridade com o comportamento.

A equação não linear 2 e sua equação equivalente da literatura é a seguinte:

$$Fnl2 = \left(\frac{441.45}{0.1775^2} \right) * (x)^2 \quad (4.25)$$

$$Eeq2 = 708.416 * x + 56453 * x^3 \quad (4.26)$$

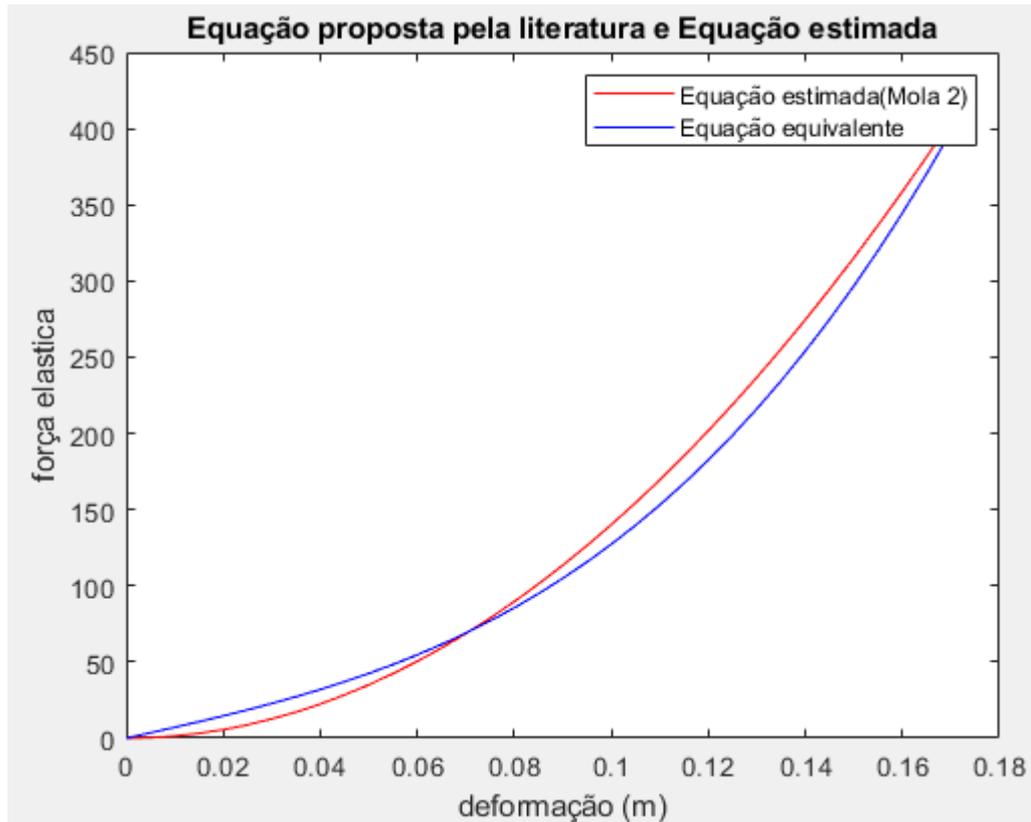


Figura 8 - Comparação entre a equação estimada e a equivalente para a mola 2

Nota-se que as duas equações têm um comportamento próximo, podendo ser considerada válida a equação estipulada para facilitação de cálculos, já que apresentam pouca variação, sendo a mola estipulada um pouco mais rígida.

Equação não linear 3 e sua equação equivalente da literatura:

$$Fnl3 = \left(\frac{441.45}{0.1775^4} \right) * (x)^4 \quad (4.27)$$

$$Eq3 = -177.40 * x + 84569 * x^3 \quad (4.28)$$

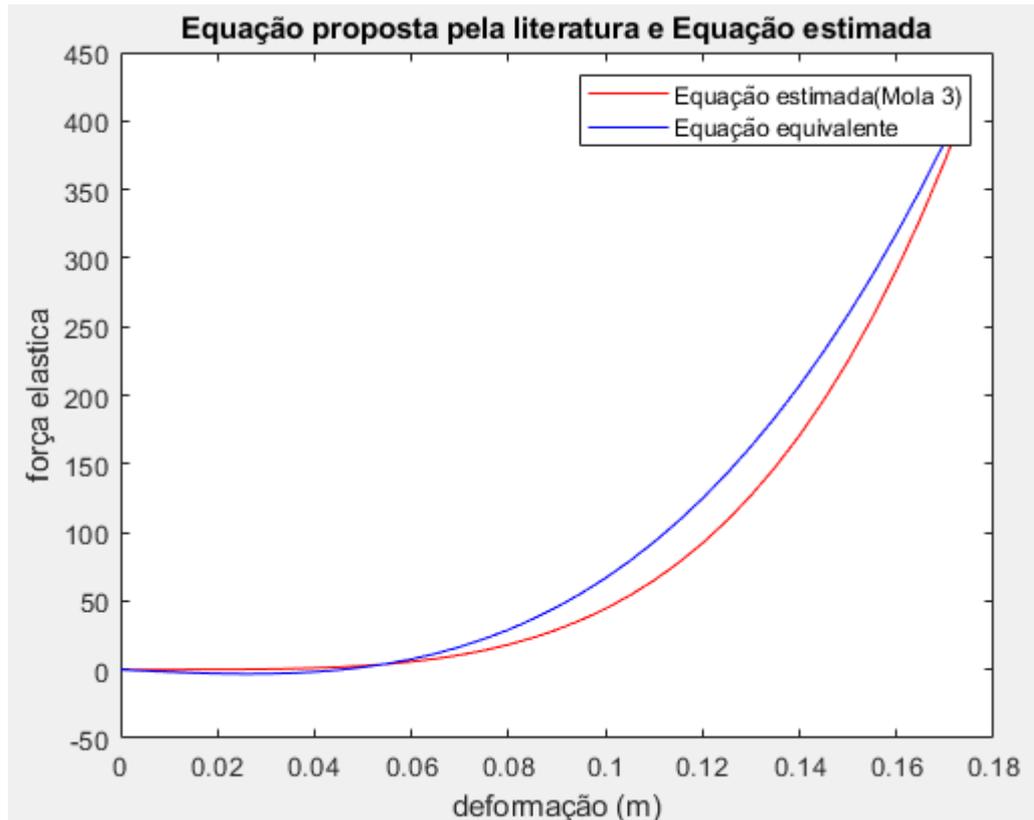


Figura 9 - Comparação entre a equação estimada e a equivalente para a mola 3

Assim como a mola 2, houve um comportamento próximo da equação equivalente da literatura. Porém neste caso, o comportamento da mola estipulada foi mais atenuado que a sua equação equivalente.

Equação não linear 4 e sua equação equivalente da literatura:

$$Fnl3 = \left(\frac{441.45}{0.1775^5} \right) * (x)^5 \quad (4.29)$$

$$Eeq4 = -621.06 * x + 98650 * x^3 \quad (4.30)$$

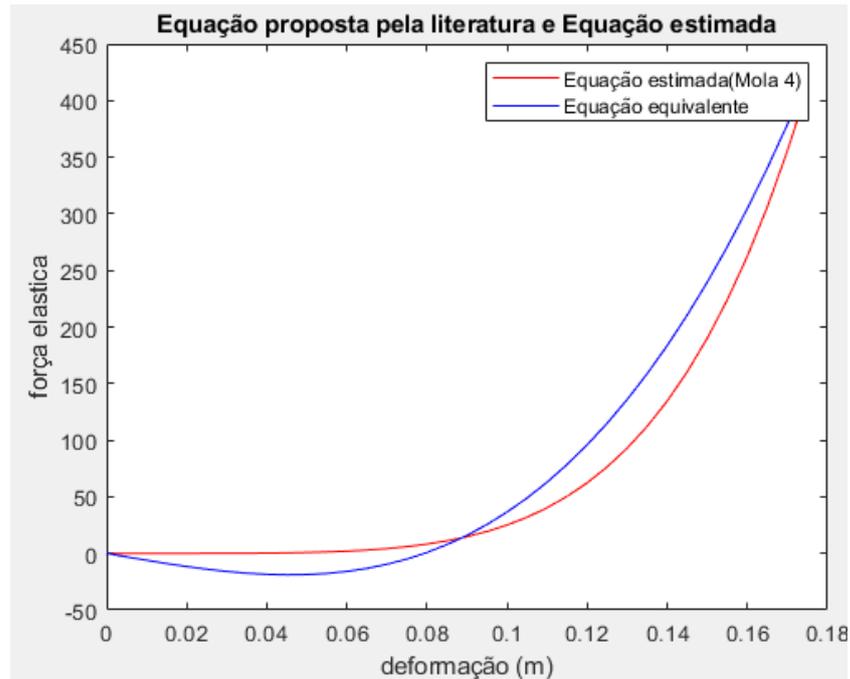


Figura 10 - Comparação entre a equação estimada e a equivalente para a mola 4

A última mola teve um comportamento um pouco mais destoante da sua equação equivalente, porém essa variação não aparenta ser de grandeza suficiente para afetar de forma alarmante o comportamento da mola.

Tendo finalizado o equacionamento de todas as molas não lineares, será calculado o torque da mesma forma que foi calculado utilizando as molas lineares, porém, desta vez, o resultado será comparado com o torque da mola linear.

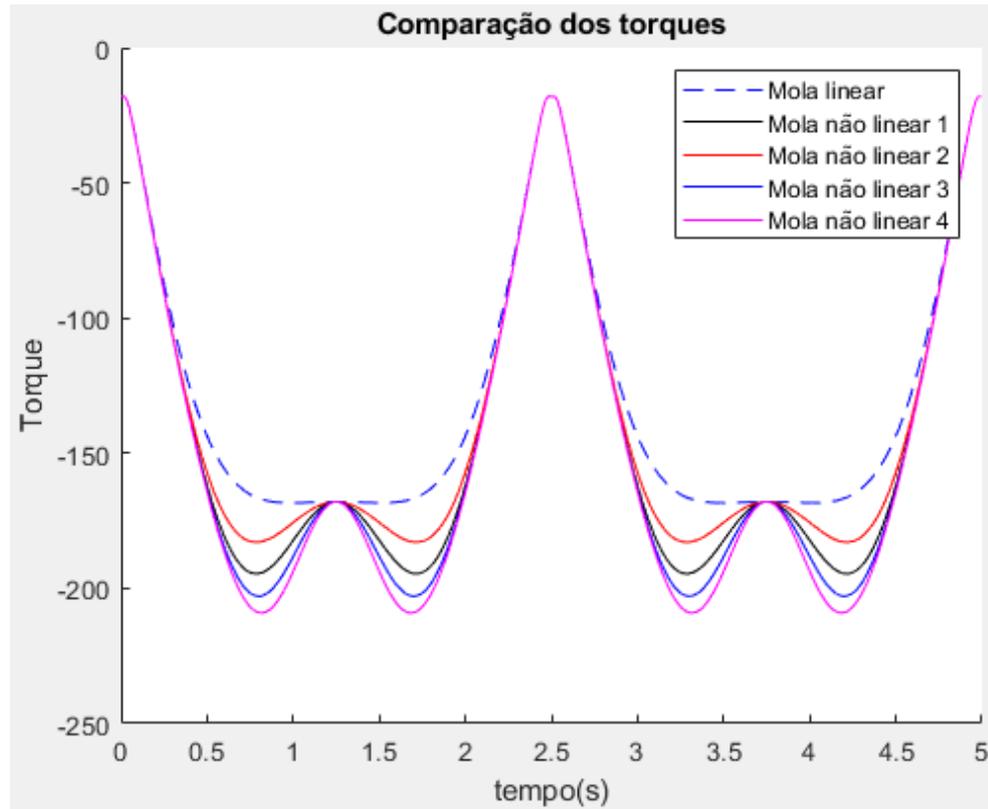


Figura 11 - Comparação do torque das 4 molas não lineares com o da mola linear

No gráfico acima, é possível notar que o comportamento da mola 2 apresenta apenas uma leve defasagem do comportamento da mola linear. Essa comparação se torna ainda mais evidente no gráfico abaixo, quando são incluídos somente o torque sem mola, da mola linear e a segunda mola não linear.

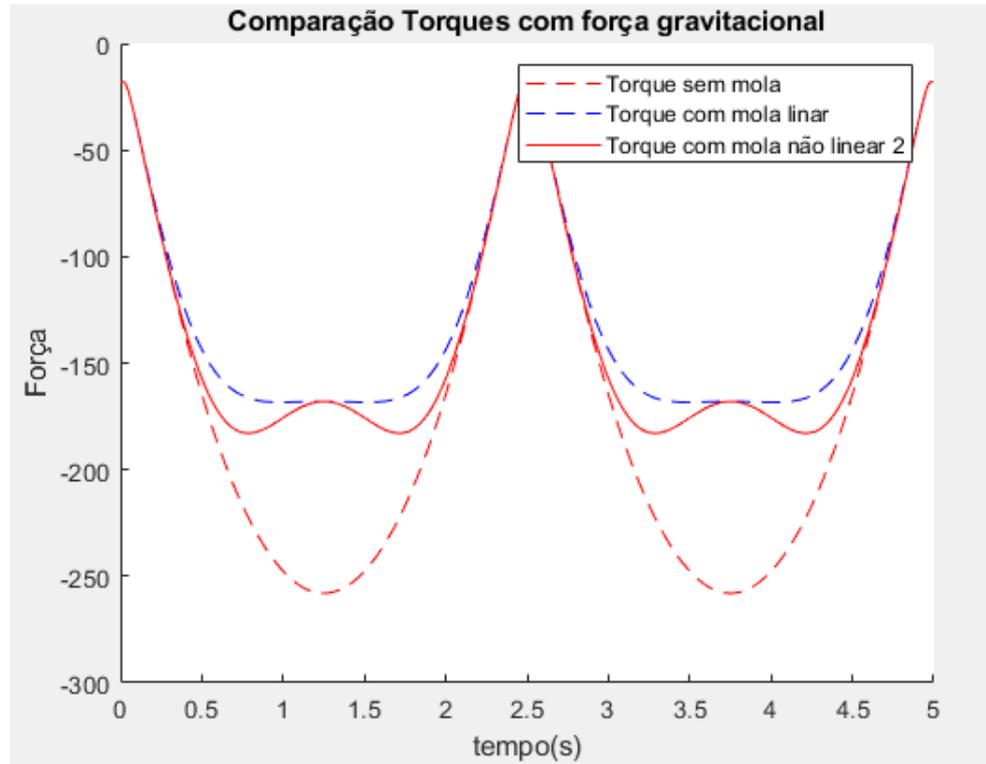


Figura 12 - Gráfico de torque sem mola, da mola linear e da segunda mola não linear

Esta última simulação apresentou uma possível escolha para a mola não linear que melhor se adequaria ao sistema, porém ainda há uma questão de alta importância a se considerar. A primeira simulação foi realizada em uma situação em que as molas compensariam a carga do usuário ao se agachar ou subir escadas, mas durante o processo de subida de escadas há um movimento de levantar a perna. Durante esse movimento, a mola iria fazer uma força prejudicial ao movimento, o que demandaria mais energia gasta para fazer o movimento. O propósito da realização de testes de molas não lineares seria justamente minimizar este comportamento prejudicial.

Para testar essa nova situação, foi feita uma nova modelagem do sistema. Nesta ocasião, não foi mais necessário levar em consideração o peso do torço, já que, ao se levantar uma perna durante a subida de escadas, necessariamente haverá alguma forma de apoio, seja a da outra perna ou de um corrimão, que sustentará o peso. Assim, na nova modelagem a origem do sistema foi transladada para a cintura do usuário.

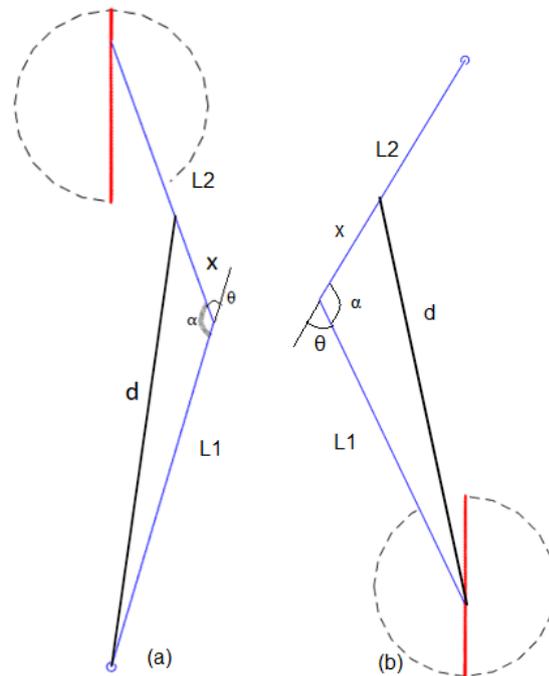


Figura 13 - (a) Modelagem da primeira equação (b) modelagem da segunda equação.

Na nova simulação, o antigo elo 1 será chamado de elo 2 e o elo 2 será chamado de elo 1. Como mencionado anteriormente, não haverá mais o elo 3 nem a sua massa concentrada em um ponto. O novo movimento levará em conta somente o peso dos elos 1 e 2, necessitando, assim, de menos torque para realizar o movimento. A presença de molas neste caso será sempre prejudicial, pois ao haver deformação nas molas a força potencial elástica geral irá sempre agir contra o movimento, demandando mais energia para fazê-lo



Figura 14 - Exemplo do movimento simulado nesta parte

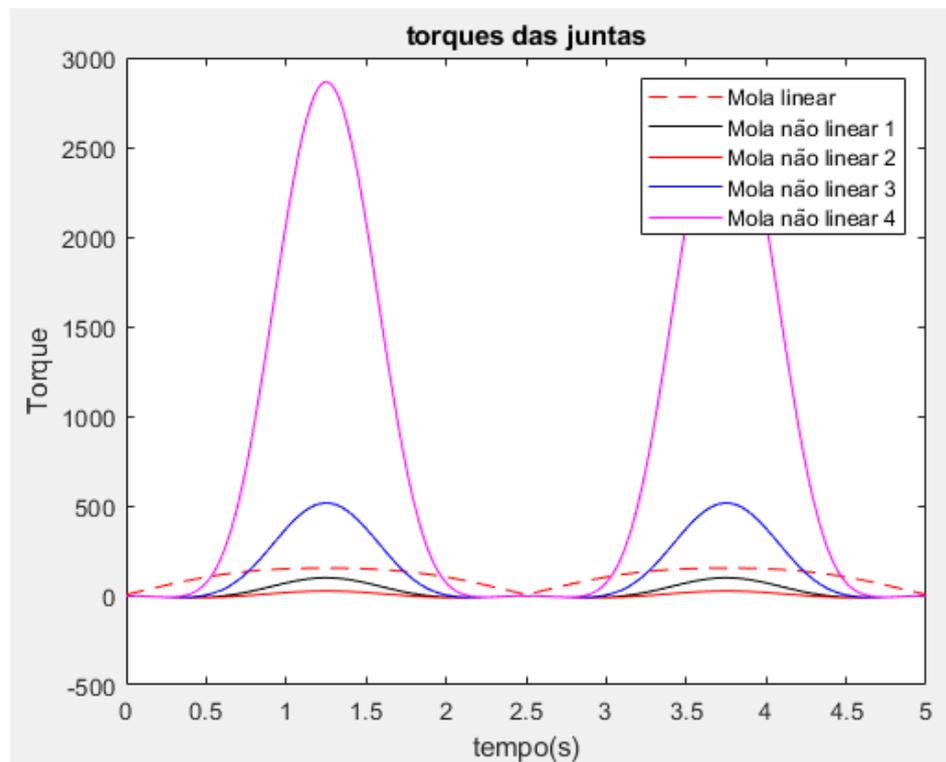


Figura 15 - Comparação dos torques sem mola, com mola linear e não lineares no movimento de levantar a perna

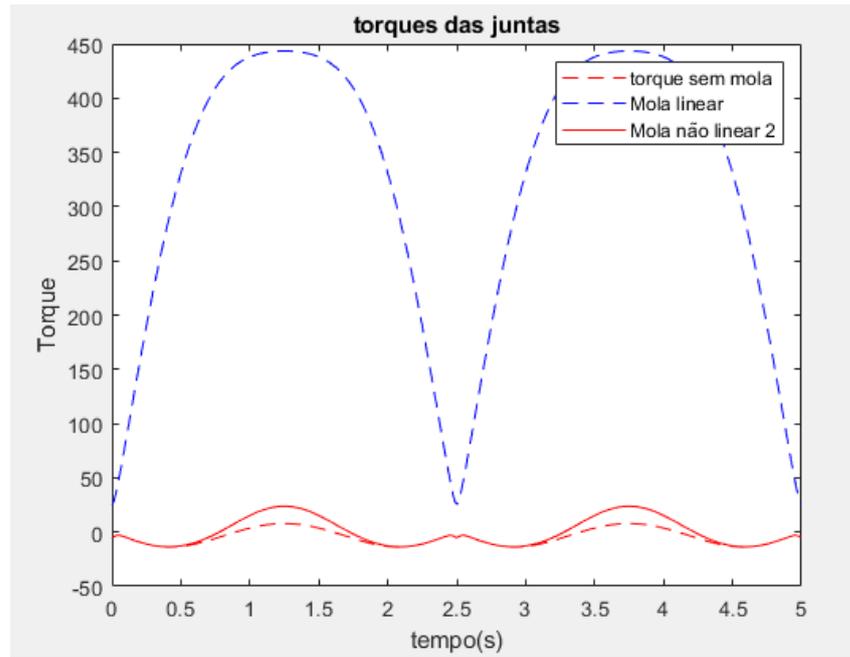


Figura 16 - Comparação do torque sem mola, com mola linear e com a mola não linear 2.

Com este gráfico, nota-se que o torque da mola 2 apresenta um comportamento extremamente próximo do torque sem mola. Já a mola linear está fazendo um torque com módulo aproximadamente 80% maior que o apresentado na última simulação. Este resultado reforça a versatilidade das molas não lineares e sua capacidade de manter o usuário com uma liberdade de movimentos apropriada e um suporte considerável em situações de cargas pesadas na coxa.

5 Testes no robô humanoide

Um robô humanoide é um dispositivo mecânico e eletrônico que imita características físicas e funcionais dos seres humanos. Geralmente, eles têm uma estrutura física semelhante à forma humana, incluindo cabeça, tronco, braços e pernas, bem como mobilidade autônoma. Alguns desses robôs são equipados com sensores avançados para perceber o ambiente, conseguem se comunicar por meio de fala, gestos ou expressões faciais, podem possuir capacidades de inteligência artificial e aprendizado de máquina. As aplicações dos robôs humanoides são diversas, abrangendo campos como educação, assistência médica, entretenimento, entre outras. No contexto específico para os testes deste projeto, será utilizado o robô ROBOTIS BIOLOID GP equipado com servo motores AX-18A da DYNAMIXEL.



Figura 17 - ROBOTIS BIOLOID GP

As propriedades da bateria utilizada no robô foram avaliadas antes dos testes, para se poder ter uma ideia da sua carga e corrente máximas.

Tabela 1 - Bateria do humanoide

Volts	%	Corrente (mAh)	(Wh)	(Ws)
12,6	100%	1000	11,1	39960

No primeiro teste, o objetivo foi medir a variação de carga na bateria após realizar uma série específica de movimentos no robô humanoide. Essa medição foi realizada com base na variação da carga da bateria do robô antes e depois da execução dos movimentos.

Nesta primeira etapa, o humanoide fez os movimentos sem nenhuma mola ou outro tipo de sistema que poderia ajudar em seus movimentos. O movimento dele foi cronometrado, pois assim seria possível fazer um cálculo de energia posteriormente para testar a hipótese de que a instalação de uma mola em somente uma das pernas seria suficiente para reduzir a energia gasta durante o movimento.

A rotina de movimentos consistiu em realizar 6 agachamentos consecutivos. Essa sequência de movimentos foi repetida 4 vezes. Antes e depois de cada repetição, a carga da bateria do robô foi medida. A diferença na carga da bateria antes e depois de cada repetição dos movimentos é uma indicação direta da energia consumida durante a execução desses movimentos, como exemplificada pela figura abaixo:

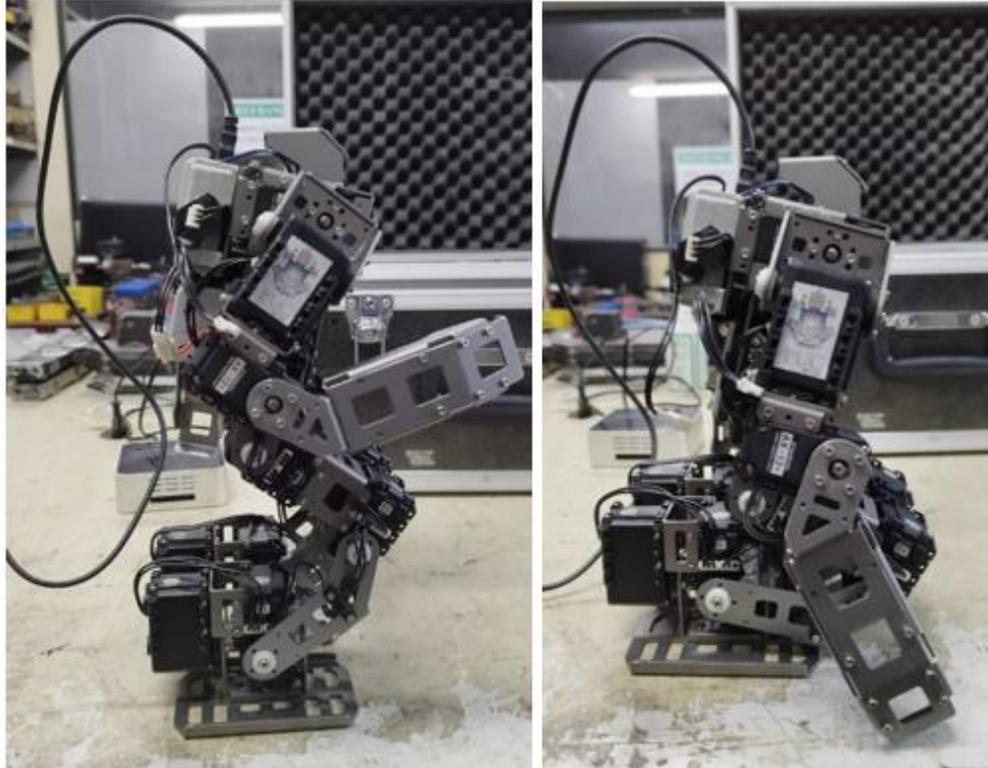


Figura 18 - (a) início do agachamento (b) meio e final do agachamento

Após a primeira testagem, foram coletados os seguintes dados:

Tabela 2 - Dados do teste sem mola

Sem mola			
Tensão	%	Corrente(mAh)	Energia (Wh)
12,3	89,0%	890	9,879
12,19	85,0%	850	9,435
12,14	82,0%	820	9,102
12,07	79,0%	790	8,769
12,02	76,0%	760	8,436
Repetições	Variação de Energia(J)		Variação em Wh
R1	1598,4	em 12s	0,444
R2	1198,8	em 12s	0,333
R3	1198,8	em 12s	0,333
R4	1198,8	em 12s	0,333

Após esse primeiro teste foi acoplada uma mola na perna do robô como demonstrado na figura abaixo.

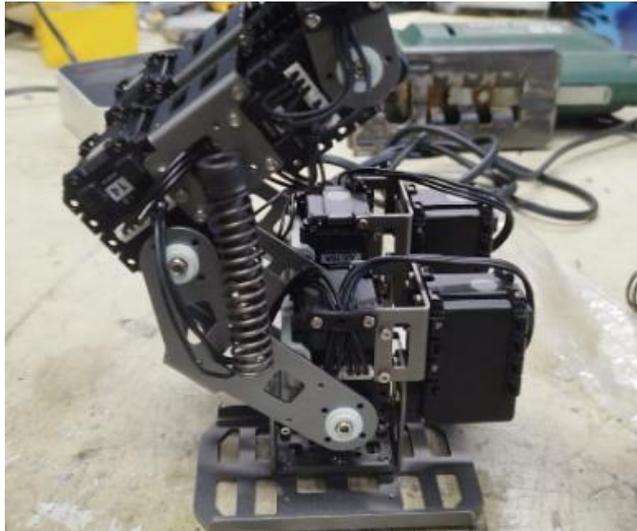


Figura 19 - Mola presa no robô humanoide

A mola utilizada foi uma mola de compressão para aplicação industrial 4004. Esta mola veio com alguns detalhes especificados no próprio anúncio de venda.

Tabela 3 - Dados da mola linear utilizada no humanoide

Comprimento	55 mm
Diâmetro interno	9,5 mm
Diâmetro externo	12,5 mm
Diâmetro do fio	1,47 mm

Dentro da mola, foi inserido um parafuso m8 para que a flambagem fosse reduzida e que funcionasse de maneira tal que, quando a mola tentasse flambar, ela iria exercer uma força no parafuso, mas este não iria deformar, obrigando, assim, a mola a manter um comportamento em sua maioria linear, pois ainda havia uma leve deformação lateral de flambagem.

Após os testes com a mola, foram obtidos os seguintes valores referentes à energia do sistema:

Tabela 4 - Dados do teste com molas

Com mola			
Tensão	%	Corrente(mAh)	Energia (Wh)
12,29	89,0%	890	9,879
12,22	86,0%	860	9,546
12,18	84,0%	840	9,324
12,13	82,0%	820	9,102
12,1	80,0%	800	8,88
Repetição	Variação de Energia(J)		Variação em Wh
R1	1198,8	em 12s	0,333
R2	799,2	em 12s	0,222
R3	799,2	em 12s	0,222
R4	799,2	em 12s	0,222

Nota-se que nos dois testes houve uma perda maior de bateria na primeira repetição, essa perda pode estar relacionada ao robô assumindo a posição e o tempo de espera até que o primeiro movimento fosse realizado.

Ao se analisar as tabelas, fica evidente que a inclusão da mola teve efeitos benéficos para o sistema, reduzindo a energia consumida pelo robô. Considerando diferenças de cargas constantes, foi feito um cálculo para transformar a energia de Wh para kJ ($1\text{Wh}=3600\text{J}$). Avaliando a variação de carga encontrada, a energia consumida pode ser encontrada.

Ao se comparar os valores de energia gastos tanto em Wh, quanto Jaule, é descoberta uma economia de 33.33% na energia gasta durante o movimento. Essa economia comprova os gráficos e resultados obtidos nas simulações anteriormente feitas.

6 Trabalhos futuros

6.1. Projeto de sistema não linear de molas

Como a não-linearidade de molas pode ser gerada de diversas formas, o sistema de molas em série é uma solução prática para projetar sistemas de suporte de carga que podem se adaptar de forma eficaz às condições variáveis de deformação. Essa configuração permite a conexão de várias molas em um único sistema, onde cada mola contribui para a rigidez global do sistema. À medida que uma força é aplicada ao sistema, todas as molas são comprimidas inicialmente. No entanto, à medida que a deformação aumenta, uma característica notável deste sistema se manifesta: o número de molas em operação começa a diminuir gradualmente.

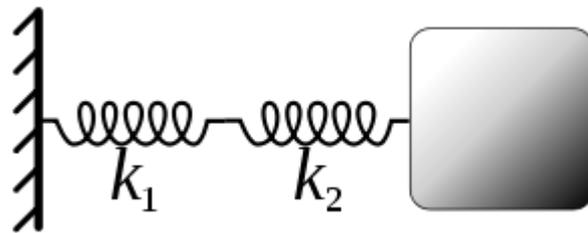


Figura 20 - Exemplo de molas em série

$$k_{eqv} = \sum_{i=1}^n \frac{1}{k_i}$$

Esse comportamento é intrínseco à configuração em série. As molas mais fracas ou mais curtas atingem primeiro o limite de compressão, enquanto as demais continuam a ser comprimidas. Isso resulta em um comportamento não linear, no qual a rigidez efetiva do sistema varia em relação à deformação. Essa característica não linear é fundamental para alcançar o objetivo do projeto, que é criar uma força potencial elástica equivalente a uma certa percentagem do peso do usuário quando todas as molas estiverem comprimidas.

O projeto desse sistema complexo envolve uma seleção criteriosa das molas individuais, levando em consideração suas características de rigidez, comprimento, diâmetro interno, diâmetro de fio e material. As molas mais fracas ou mais curtas devem ser posicionadas mais próximas de uma das extremidades para garantir que sejam as primeiras a comprimir. Isso assegura que o sistema se adapte de maneira eficaz à carga aplicada, fornecendo uma resposta adequada ao peso do usuário e às variações na deformação.

A aplicação deste conceito de molas em série com comportamento não linear é diversificada, abrange áreas mais cotidianas, tendo como exemplo equipamentos esportivos, tais como tênis de corrida e bicicletas, além de móveis ergonômicos, como cadeiras de escritório. Além destes, sistemas de suspensão em veículos e equipamentos de segurança também podem se beneficiar de tal tipo de projeto. Em todos esses contextos, a capacidade do sistema de molas em série de fornecer flexibilidade e desempenho adequados é altamente valorizada, apresentando uma grande possibilidade de aprofundamento e estudo.

6.2. Modelagem considerando o terceiro elo íntegro

A modelagem de um exoesqueleto passivo, inicialmente concebido como um braço robótico de dois elos e, posteriormente, estendido com um terceiro elo, oferece perspectivas valiosas para a simulação e compreensão da dinâmica do sistema. Ao incorporar um terceiro elo ao exoesqueleto, é introduzido um maior grau de liberdade e flexibilidade, o que pode ser crucial para otimizar a interação com o usuário.

A adição desse terceiro elo permite uma adaptação mais eficaz aos movimentos naturais do corpo humano, especialmente em tarefas que envolvem ambientes tridimensionais e complexos. A flexibilidade adicional também pode melhorar a capacidade de o exoesqueleto seguir os padrões de movimento do usuário de maneira mais natural e eficiente.

Ao se modelar o exoesqueleto com três elos, ampliam-se potencialmente seu alcance e capacidade de ajuste para diferentes anatomias e movimentos. Essa expansão na configuração do sistema pode ser instrumental na simulação de cenários

mais realistas e na compreensão aprofundada das interações entre o exoesqueleto e o usuário.

Apesar dos desafios adicionais na modelagem matemática de sistemas com mais elos, as vantagens oferecidas por um exoesqueleto de três elos na simulação e análise dinâmica superam essas complexidades. A capacidade de se replicar de maneira mais fiel os movimentos naturais do corpo humano pode levar a avanços significativos na otimização e eficácia dos exoesqueletos, seja para aplicações de reabilitação, assistência em atividades diárias ou aprimoramento de desempenho em diversas áreas.

Em resumo, a expansão de um exoesqueleto passivo de dois para três elos não apenas aprimora a modelagem dinâmica, mas também oferece uma representação mais precisa e versátil das capacidades do dispositivo. Essa abordagem pode ser determinante para impulsionar o desenvolvimento de exoesqueletos mais eficientes e adaptáveis, alinhados com as necessidades específicas dos usuários e aplicações.

6.3. Construção de um protótipo

A construção de um protótipo de exoesqueleto passivo, incluindo a utilização de molas em série, demanda uma grande quantidade de conhecimentos distintos. Inicialmente, é essencial realizar uma validação minuciosa do projeto. Isso envolve a revisão detalhada do design conceitual, simulações de movimentos, e análise de viabilidade técnica do exoesqueleto.

A escolha dos materiais é um passo crítico na construção do protótipo. Considerando a utilização de molas em série, projetos como estudos de comportamentos não lineares podem ser abordados. A análise de propriedades mecânicas é imperativa para garantir que os materiais selecionados ofereçam durabilidade, leveza e resistência necessárias para o sucesso do dispositivo.

O dimensionamento dos componentes é outro aspecto fundamental. Molas, estruturas e articulações devem ser dimensionadas com precisão para garantir a funcionalidade e segurança do exoesqueleto. Testes práticos de conforto com usuários reais são muito importantes para avaliar a eficácia do exoesqueleto na

prática. Durante esses testes, aspectos como ergonomia, liberdade de movimento e adaptabilidade do dispositivo às atividades cotidianas podem ser avaliados.

A integração de molas em série no exoesqueleto é estratégica para proporcionar flexibilidade e suavidade nos movimentos como foi demonstrado nas simulações realizadas neste trabalho. Essa abordagem visa otimizar a resposta dinâmica do sistema, garantindo que o exoesqueleto seja capaz de se adaptar de forma eficiente às diversas atividades realizadas pelos usuários.

Em conclusão, a construção bem-sucedida de um protótipo de exoesqueleto passivo demanda uma combinação cuidadosa de conhecimento de diversas áreas. Essa abordagem integrada é essencial para impulsionar o desenvolvimento de exoesqueletos mais eficientes e adaptáveis, alinhados com as necessidades específicas dos usuários e aplicações.

7 Conclusão

Neste estudo, alcançamos uma simulação bem-sucedida ao modelar o comportamento de molas não lineares, proporcionando uma compreensão mais profunda de sua aplicação em exoesqueletos passivos. Essa simulação não apenas validou a precisão teórica, como também abriu perspectivas promissoras para a implementação prática dessas molas, especialmente em contextos de reabilitação muscular. Ao antecipar com sucesso o comportamento não linear, nosso trabalho contribui significativamente para o avanço da tecnologia de exoesqueletos, fornecendo uma ferramenta eficaz para otimizar o suporte à reabilitação.

Os testes conduzidos no humanoide foram essenciais para validar a credibilidade das simulações. Os resultados práticos obtidos durante os agachamentos demonstraram que as melhorias no consumo de bateria eram consistentes com as previsões teóricas. Essa congruência entre simulação e experimentação prática destaca a coerência de nossa abordagem, sugerindo que as simulações podem ser confiáveis para prever o comportamento de molas não lineares em situações simuladas. Esse alinhamento entre teoria e prática é fundamental para estabelecer uma base sólida para futuras implementações práticas em ambientes de reabilitação muscular.

A potencial aplicação dessas descobertas em processos de reabilitação muscular é promissora. A eficiência energética aprimorada oferecida pelas molas não lineares pode ser um fator crucial na redução da fadiga muscular durante exercícios de reabilitação, proporcionando maior conforto e melhorando a adesão dos pacientes aos programas de tratamento. Essa perspectiva positiva sugere que nossa pesquisa tem implicações práticas que interferem diretamente na melhoria da qualidade de vida de indivíduos em processo de reabilitação.

Em conclusão, nossa simulação foi bem sucedida e os testes práticos no humanoide não apenas validaram a viabilidade das molas não lineares em exoesqueletos passivos, como também destacaram seu potencial significativo em melhorar a eficiência energética e, conseqüentemente, a experiência de reabilitação muscular. Estamos otimistas em relação a implicações futuras desta pesquisa,

antecipando que servirá como um ponto de partida sólido para avanços inovadores na interseção entre tecnologia de exoesqueletos e reabilitação muscular.

8 REFERÊNCIAS BIBLIOGRÁFICAS

- [1]. ZHOU, L. *et al.* Design of a passive lower limb exoskeleton for walking assistance with gravity compensation. **Mechanism and Machine Theory**, [s. l.], v. 150, 2020. DOI 10.1016/j.mechmachtheory.2020.103840.
- [2]. ZHANG, Q.; TIAN, K.; GUO, H. Development of an Instrumented and Passive Exoskeleton for the Lower Limb Rehabilitation. **2009 International Association of Computer Science and Information Technology - Spring Conference**, [s. l.], 2009. DOI 10.1109/iacsit-sc.2009.78
- [3]. BARTENBACH, V.; GORT, M.; RIENER, R. Concept and design of a modular lower limb exoskeleton. **2016 6th IEEE International Conference on Biomedical Robotics and Biomechatronics (BioRob)**, [s. l.], 2016. DOI 10.1109/biorob.2016.7523699.
- [4]. PIRJADE, Y. M. *et al.* Design and Fabrication of a Low-cost Human Body Lower Limb Exoskeleton. **2020 6th International Conference on Mechatronics and Robotics Engineering (ICMRE)**, [s. l.], 2020. DOI 10.1109/icmre49073.2020.9065128.
- [5]. UNLUHISARCIKLI, O. *et al.* Design and control of a robotic lower extremity exoskeleton for gait rehabilitation. **2011 IEEE/RSJ International Conference on Intelligent Robots and Systems**, [s. l.], 2011. DOI 10.1109/iros.2011.6094973.
- [6]. ARANHA, N. *et al.* A lei de Hooke e as molas não-lineares, um estudo de caso. (Portuguese). **Caderno Brasileiro de Ensino de Física**, [s. l.], v. 38, n. 4, p. 1–10, 2016. DOI 10.1590/1806-9126-RBEF-2016-0102.
- [7]. ROEBROECK, M. E. *et al.* Biomechanics and muscular activity during sit-to-stand transfer. **Clinical Biomechanics ; volume 9, issue 4, page 235-244 ; ISSN 0268-0033**, [s. l.], 1994. DOI 10.1016/0268-0033(94)90004-3.
- [8]. CENCIARINI, M.; DOLLAR, A. M. Biomechanical considerations in the design of lower limb exoskeletons. **IEEE ... International Conference on**

Rehabilitation Robotics : [proceedings], [s. l.], v. 2011, p. 5975366, 2011.
DOI 10.1109/ICORR.2011.5975366.

9 ANEXO A – Código de Matlab para primeira simulação

```

clear
close all
global l1 l2 l3 lc1 lc2 lc3 m1 m2 m3 I1 I2 I3 g;
global xc yc r f ;
global caso ;

caso = 1; %0 sem controle, 1 CTC(PD+ff) 2Hz

l1 = 0.53; l2 = 0.44; l3=0; %metro
lc1 = l1/2; lc2 = l2/2; lc3=0; %elo homogêneo
m1 = 5; m2 = 8; m3=67; %kg
I1 = m1*l1^2/12; I2 = m2*l2^2/12; I3 = 0; %assumindo uma barra homogênea
g = 9.81; %m/s^2, gravidade
xc =0;
ycmin = (l1+l2)*sin(pi/4);
ycmax = (l1+l2)*sin(pi/180*88);
yc = (ycmin+ycmax)/2; r = (ycmax-ycmin)/2; %centro e raio da trajetória
f = 0.4; %hz
w = 2.51327412; %frequencia em radianos por segundo

x0 = 0;
y0 = ycmax;
%cinemática inversa:
teta2 = acos((x0^2+y0^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2 = -teta2; %caso queira usar a solução com cotovelo para cima
teta1 = atan2(y0,x0) - acos((x0^2+y0^2+l1^2-l2^2)/(2*l1*sqrt(x0^2+y0^2)));
%teta3 = (pi/180)*90 - teta1 - teta2

q0 = [teta1 teta2 0 0 ]'; %estado inicial em radianos: [teta1 teta2 teta3
teta1ponto teta2ponto teta3ponto]

tf = 5; %tempo final de simulação
t = [0:0.015:tf];
[t,q] = ode45('robosimtcc1',t,q0);
qponto =zeros(size(q));
H11 = zeros(size(q(:,1)));
H12 = zeros(size(q(:,1)));
H21 = zeros(size(q(:,1)));
h = zeros(size(q(:,1)));
G1 = zeros(size(q(:,1)));
G2 = zeros(size(q(:,1)));
U = zeros(size(q(:,1)));
Um = zeros(size(q(:,1)));
T = zeros(size(q(:,1)));
L = zeros(size(q(:,1)));
Lm = zeros(size(q(:,1)));
Udiff = zeros(size(q(:,1)));
H22 = m3*l2^2 + m2*lc2^2 + I2;
x1 = l2/2;

```

```

x2= l2/3;
x3= l2/4;
x4= l2/6;
k1 = 1934;
k2 = 1.3817e+03;
k3 = 2.4870e+03;
k4 = 3.0397e+03;
teta2min = min(q(:,2));
d0 = sqrt(l1^2+x1^2-2*l1*x1*cos(pi-teta2min));
d = sqrt(l1^2+x1^2-2*l1*x1*cos(pi-q(:,2)));
delta_dmax = d0 - min(d);
%d01 = sqrt(l1^2+x1^2-2*l1*x1*cos(pi-teta2min));
%d02 = sqrt(l1^2+x2^2-2*l1*x2*cos(pi-teta2min));
%d03 = sqrt(l1^2+x2^2-2*l1*x3*cos(pi-teta2min));
%d03 = sqrt(l1^2+x2^2-2*l1*x4*cos(pi-teta2min));
% Um = m1*g*yc1+m2*g*yc2+m3*g*yc3 + 0.5*k*(d-d0)^2;
G2m1 = G2;
G2m2 = G2;
G2m3 = G2;
G2m4 = G2;
G2n11 = G2;
G2n12 = G2;
G2n13 = G2;
G2n14 = G2;

for index= 1:size(t)

    qponto(index,:)=robosimtcc1(t(index),q(index,:));

    %valores calculados

    H11(index,1) = I1 + I2 + l1^2*m2 + l1^2*m3 + l2^2*m3 + lc1^2*m1 + lc2^2*m2 +
    2*l1*l2*m3*cos(q(index,2)) + 2*l1*lc2*m2*cos(q(index,2));

    H12(index,1) = m3*l2^2 + l1*m3*cos(q(index,2))*l2 + m2*lc2^2 +
    l1*m2*cos(q(index,2))*lc2 + I2;

    H21(index,1)=H12(index,1);

    %só ha um componente principal

    h(index,1)=(l1*sin(q(index,2))*(l2*m3 + lc2*m2));

    G1(index,1) = g*m3*(l2*cos(q(index,1) + q(index,2)) + l1*cos(q(index,1))) +
    g*m2*(lc2*cos(q(index,1) + q(index,2)) + l1*cos(q(index,1))) +
    g*lc1*m1*cos(q(index,1));

    G2(index,1) = g*cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2);

    T(index,1) = 0.5*(H11(index,1)*q(index,3)^2 +
    2*H12(index,1)*q(index,3)*q(index,4) + H22*q(index,4)^2)*1;

    U(index,1) = g*m3*(l2*sin(q(index,1) + q(index,2)) + l1*sin(q(index,1))) +
    g*m2*(lc2*sin(q(index,1) + q(index,2)) + l1*sin(q(index,1))) +
    g*lc1*m1*sin(q(index,1));

    Um(index,1) = (k1*(d0 - sqrt(l1^2 + 2*cos(q(index,2))*l1*x1 + x1^2))^2)/2 +
    g*m3*(l2*sin(q(index,1) + q(index,2)) + l1*sin(q(index,1))) +

```

$$g*m2*(lc2*\sin(q(index,1) + q(index,2)) + l1*\sin(q(index,1))) + g*lc1*m1*\sin(q(index,1));$$

$$Udiff(index,1) = U(index,1) - Um(index,1);$$

$$L(index,1) = T(index,1) - U(index,1);$$

$$Lm(index,1) = T(index,1) - Um(index,1);$$

$$G2x1(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k1*l1*x1*\sin(q(index,2)) + (d0*k1*l1*x1*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2x2(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k1*l1*x2*\sin(q(index,2)) + (d0*k1*l1*x2*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x2 + x1^2};$$

$$G2x3(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k1*l1*x3*\sin(q(index,2)) + (d0*k1*l1*x3*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x3 + x1^2};$$

$$G2x4(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k1*l1*x4*\sin(q(index,2)) + (d0*k1*l1*x4*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x4 + x1^2};$$

$$G2m1(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k1*l1*x1*\sin(q(index,2)) + (d0*k1*l1*x1*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2m2(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k2*l1*x1*\sin(q(index,2)) + (d0*k2*l1*x1*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2m3(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k3*l1*x1*\sin(q(index,2)) + (d0*k3*l1*x1*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2m4(index,1) = g*\cos(q(index,1) + q(index,2))*(l2*m3 + lc2*m2) - k4*l1*x1*\sin(q(index,2)) + (d0*k4*l1*x1*\sin(q(index,2)))/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2n11(index,1) = g*l2*m3*\cos(q(index,1) + q(index,2)) + g*lc2*m2*\cos(q(index,1) + q(index,2)) + (78938.06*l1*x1*\sin(q(index,2))*(d0 - \sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2}))^3/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2n12(index,1) = g*l2*m3*\cos(q(index,1) + q(index,2)) + g*lc2*m2*\cos(q(index,1) + q(index,2)) + (14011.51*l1*x1*\sin(q(index,2))*(d0 - \sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2}))^2/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2n13(index,1) = g*l2*m3*\cos(q(index,1) + q(index,2)) + g*lc2*m2*\cos(q(index,1) + q(index,2)) + (444721.47*l1*x1*\sin(q(index,2))*(d0 - \sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2}))^4/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

$$G2n14(index,1) = g*l2*m3*\cos(q(index,1) + q(index,2)) + g*lc2*m2*\cos(q(index,1) + q(index,2)) + (2505473.04*l1*x1*\sin(q(index,2))*(d0 - \sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2}))^5/\sqrt{l1^2 + 2*\cos(q(index,2))*l1*x1 + x1^2};$$

```

G2nl4test(index,1) = -(g*l2*m3*cos(q(index,1) + q(index,2)) +
g*lc2*m2*cos(q(index,1) + q(index,2)) + (2505473.04*l1*x1*sin(q(index,2))*(d0 -
sqrt(l1^2 + 2*cos(q(index,2))*l1*x1 + x1^2))^5)/sqrt(l1^2 +
2*cos(q(index,2))*l1*x1 + x1^2));
end

```

```

%cinemática direta:

```

```

x = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
y = l1*sin(q(:,1)) + l2*sin(q(:,1)+q(:,2));

```

```

% Jacobiano

```

```

J11 = -l1*sin(q(:,1)) - l2*sin(q(:,1)+q(:,2));
J12 = -l2*sin(q(:,1)+q(:,2));
%J13=1;
J21 = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
J22 = l2*cos(q(:,1)+q(:,2));
% J23=1;
% J31=0;
% J32=0;
% J33=1;

```

```

% Jacobiano dos centros

```

```

% J11c1 = -lc1*sin(q(:,1));
% J12c1 = 0;
% J21c1 = lc1*cos(q(:,1));
% J22c1 = 0;
% J13c1 = 0;
% J23c1 = 0;
%
% J11c2 = -l1*sin(q(:,1)) - lc2*sin(q(:,1)+q(:,2));
% J12c2 = -lc2*sin(q(:,1)+q(:,2));
% J21c2 = l1*cos(q(:,1)) + lc2*cos(q(:,1)+q(:,2));
% J22c2 = lc2*cos(q(:,1)+q(:,2));
% J13c2 = 0;
% J23c2 = 0;
%
% J11c3 = -l1*sin(q(:,1)) - l2*sin(q(:,1)+q(:,2));
% J12c3 = -l2*sin(q(:,1)+q(:,2));
% J21c3 = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
% J22c3 = l2*cos(q(:,1)+q(:,2));
% J13c3 = 0;
% J23c3 = 0;

```

```

% toques sem molas

```

```

for i = 1:size(t)
tau1(i,1) = qponto(i,3).*H11(i)+ qponto(i,4).*H12(i)-2*h(i)*q(i,3)*q(i,4)-
h(i)*q(i,4)*q(i,4) + G1(i);
tau2(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) + G2(i);
end;

```

```

%torques com mola linear
for i = 1:size(t)
    tau2x1(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2x1(i);
    tau2x2(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2x2(i);
    tau2x3(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2x3(i);
    tau2x4(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2x4(i);

    tau2m1(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2m1(i);
    tau2m2(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2m2(i);
    tau2m3(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2m3(i);
    tau2m4(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2m4(i);
end

%torques com mola nao linear

for i = 1:size(t)
    tau2n11(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n11(i);
    tau2n12(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n12(i);
    tau2n13(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n13(i);
    tau2n14(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n14(i);
    taudiff(i,1)=tau2m3(i)-tau2n12(i,1);
end

taudiffmed=median(taudiff);
Potmed=taudiffmed*w; %potencia media
EnergiaGasta1=Potmed*5; %diferente de energia entre a mola linear e a não

% fh = figure(1);
% set(fh, 'doublebuffer', 'on'); % for smoothness in the animation
% fh.WindowState = 'maximized';
% drawnow;
%
% %posição da origem do sistema de coordenadas 1:
% x1 = l1*cos(q(:,1));
% y1 = l1*sin(q(:,1));
% plot(0,0,'bo')
% hold on
% axis([-l1/2,xc+1.5*r,-l1/2,yc+1.5*r])
% axis equal
% plot(xc+r*cos([0:pi/10:2*pi]), yc+r*sin([0:pi/10:2*pi]), 'k--');
% for i = 1:size(t,1)
%     plot([0 x1(i) x(i)], [0 y1(i) y(i)], 'b'); %desenha o manipulador
%     drawnow;
%     plot([0 x1(i) x(i)], [0 y1(i) y(i)], 'w'); %apaga o manipulador
%     plot(x(i),y(i), 'r. '); %desenha o rastro da extremidade

```

```

% end

% figure(2)
% plot(t,q(:,1)*180/pi,'b')
% hold on
% plot(t,q(:,2)*180/pi,'r')
% title('ângulos das juntas')
% xlabel('tempo(s)')
% ylabel('ângulo(graus)')
% legend('teta1','teta2')

% figure(3)
% plot(x,y)
% hold on
% plot(xc+r*cos([0:pi/10:2*pi]), yc+r*sin([0:pi/10:2*pi]), 'k--');
% axis equal
% switch caso
%     case 0
%         title('sem controle (motores desligados)')
%     case 1
%         title('CTC com PD + feedforward, 2Hz')
% end
% xlabel('extremidade do manipulador x (m)')
% ylabel('extremidade do manipulador y (m)')

figure(4)
hold on
% plot(t,G1,'b--')
% plot(t,G2,'r--')
% plot(t,G2m1,'k')
% plot(t,G2m2,'r')
% plot(t,G2m3,'b')
% plot(t,G2m4,'m')

% plot(t,tau1,'b')
% plot(t,tau2,'r')
% plot(t,tau2x1,'k')
% plot(t,tau2x2,'r')
% plot(t,tau2x3,'b')
% plot(t,tau2x4,'m')

plot(t,tau2,'r--')
% plot(t,tau2m1,'k')
% plot(t,tau2m2,'r')
% plot(t,tau2m3,'b')
% plot(t,tau2m4,'m')

plot(t,tau2m3,'b--')
% plot(t,tau2n11,'k')
plot(t,tau2n12,'r')
% plot(t,tau2n13,'b')
% plot(t,tau2n14,'m')

%plot(t,tau2m2,'k')
title('Comparação Torques com força gravitacional')
%title('Escolha da posição de fixamento')
%title('Comparação dos torques com molas lineares')
%title('Comparação dos torques com molas não lineares')

```

```
xlabel('tempo(s)')
ylabel('Força')
legend('Torque sem mola', 'Torque com mola linear', 'Torque com mola não linear 2')
%legend('G2', 'G2m1', 'G2m2', 'G2m3', 'G2m4' )
%legend('Torque puro', 'Torque na posição 1', 'Torque na posição 2', 'Torque na
posição 3', 'Torque na posição 4' )
%legend('Torque sem mola', 'Mola 1', 'Mola 2', 'Mola 3', 'Mola 4')
%legend('Mola linear', 'Mola não linear 1', 'Mola não linear 2', 'Mola não linear
3', 'Mola não linear 4')

% figure(5)
% plot(t,T,'b')
% hold on
% plot(t,U,'r')
% plot(t,Um,'r--')
% plot(t,L,'k')
% plot(t,Lm,'k--')
% title('Lagrange')
% xlabel('tempo(s)')
% ylabel('energia')
% legend('T', 'U', 'Um', 'L', 'Lm')
```

10 Anexo B – Função para simulação 1

```

function qponto = robosimtcc1(t, q)
%equações para simular um robô de 2 graus de liberdade
global l1 l2 l3 lc1 lc2 lc3 m1 m2 m3 I1 I2 I3 g ;
global xc yc r f ;
global caso gamainv;

teta1 = q(1);
teta2 = q(2);
%teta3 = q(x);
teta1p = q(3); %derivada de teta1
teta2p = q(4); %derivada de teta2
%teta3p = q(y); %derivada de teta3
t

%valores calculados
H11 = I1 + I2 + l1^2*m2 + l1^2*m3 + l2^2*m3 + lc1^2*m1 + lc2^2*m2 +
2*l1*l2*m3*cos(teta2) + 2*l1*lc2*m2*cos(teta2);
H12 = m3*l2^2 + l1*m3*cos(teta2)*l2 + m2*lc2^2 + l1*m2*cos(teta2)*lc2 + I2;
H13 = 0;
H21=H12;
H22 = m3*l2^2 + m2*lc2^2 + I2;
H23 = 0;
%só ha um componente principal
h=l1*sin(teta2)*(l2*m3 + lc2*m2);
G1 = g*m3*(l2*cos(teta1 + teta2) + l1*cos(teta1)) + g*m2*(lc2*cos(teta1 + teta2) +
l1*cos(teta1)) + g*lc1*m1*cos(teta1);
G2 = g*cos(teta1 + teta2)*(l2*m3 + lc2*m2);

%trajetória desejada:
xd = 0 ;
yd = yc + r*cos(2*pi*f*t);
xdp = 0;
ydp = -r*sin(2*pi*f*t)*(2*pi*f);
xdpp = 0;
ydp = -r*cos(2*pi*f*t)*(2*pi*f)^2;
%cinemática inversa:
teta2d = acos((xd^2+yd^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2d = -teta2d; %caso queira usar a solução com cotovelo para cima
teta1d = atan2(yd,xd) - acos((xd^2+yd^2+l1^2-l2^2)/(2*l1*sqrt(xd^2+yd^2)));
%teta3d = (pi/2) - teta2d - teta1d
%Jacobiano
c1d = cos(teta1d); c12d = cos(teta1d+teta2d); s1d = sin(teta1d); s12d =
sin(teta1d+teta2d);
Jd = [[-l1*s1d-l2*s12d -l2*s12d];[l1*c1d+l2*c12d l2*c12d]];
tetadp = inv(Jd)*[xdp ydp]';
teta1dp = tetadp(1); teta2dp = tetadp(2);%teta3p = tetadp(3);
tetadpp = inv(Jd)*[xdpp+l1*c1d*teta1dp^2+l2*c12d*(teta1dp+teta2dp)^2;
ydpp+l1*s1d*teta1dp^2+l2*s12d*(teta1dp+teta2dp)^2];
teta1dpp = tetadpp(1); teta2dpp = tetadpp(2); %teta3dpp = tetadpp(3);

```

```

wn = 15; Kp = wn^2; %frequência natural desejada (para ambas as juntas)
zeta = 0.76; Kd = 2*zeta*wn; %overshoot de 2%, settling time ts = 4/(zeta*wn)
%0 sem controle, 1 CTC(PD+ff) 0.5Hz
if (caso==0) %sem controle
    tau1 = 0;
    tau2 = 0;
elseif (caso==1)%controle de torque computado com PD + feedforward
    u1 = teta1dpp + Kp*(teta1d-teta1) + Kd*(teta1dp-teta1p);
    u2 = teta2dpp + Kp*(teta2d-teta2) + Kd*(teta2dp-teta2p);
    tau1 = H11*u1 + H12*u2 + G1 - 2*h*teta1p*teta2p - h*teta2p^2;
    tau2 = H22*u2 + H12*u1 + G2 + h*teta1p^2;

end

% Equações da dinâmica: (teta1pp é a segunda derivada de teta1, idem para teta2pp
e teta2)
% tau1 = H11*teta1pp + H12*teta2pp - h*teta2p^2 - 2*h*teta1p*teta2p + G1;
% tau2 = H22*teta2pp + H12*teta1pp + h*teta1p^2 + G2;
% resolvendo, temos (ou pode escrever as equações abaixo em forma vetorial):
teta1pp = (H22*tau1-H12*tau2+H22*h*teta2p^2+H22*2*h*teta1p*teta2p-
H22*G1+H12*h*teta1p^2+H12*G2)/(H11*H22-H12^2);
teta2pp = (tau2 - H12*teta1pp - h*teta1p^2 - G2)/H22;

    qponto = [teta1p teta2p teta1pp teta2pp]'; %precisa ser vetor coluna

end

```

11 Anexo C – Modelagem para primeira simulação

```

function qponto = robosimtcc1(t, q)
%equações para simular um robô de 2 graus de liberdade
global l1 l2 l3 lc1 lc2 lc3 m1 m2 m3 I1 I2 I3 g ;
global xc yc r f ;
global caso gamainv;

teta1 = q(1);
teta2 = q(2);
%teta3 = q(x);
teta1p = q(3); %derivada de teta1
teta2p = q(4); %derivada de teta2
%teta3p = q(y); %derivada de teta3
t

%valores calculados
H11 = I1 + I2 + l1^2*m2 + l1^2*m3 + l2^2*m3 + lc1^2*m1 + lc2^2*m2 +
2*l1*l2*m3*cos(teta2) + 2*l1*lc2*m2*cos(teta2);
H12 = m3*l2^2 + l1*m3*cos(teta2)*l2 + m2*lc2^2 + l1*m2*cos(teta2)*lc2 + I2;
H13 = 0;
H21=H12;
H22 = m3*l2^2 + m2*lc2^2 + I2;
H23 = 0;
%só ha um componente principal
h=l1*sin(teta2)*(l2*m3 + lc2*m2);
G1 = g*m3*(l2*cos(teta1 + teta2) + l1*cos(teta1)) + g*m2*(lc2*cos(teta1 + teta2) +
l1*cos(teta1)) + g*lc1*m1*cos(teta1);
G2 = g*cos(teta1 + teta2)*(l2*m3 + lc2*m2);

%trajetória desejada:
xd = 0 ;
yd = yc + r*cos(2*pi*f*t);
xdp = 0;
ydp = -r*sin(2*pi*f*t)*(2*pi*f);
xdpp = 0;
ydp = -r*cos(2*pi*f*t)*(2*pi*f)^2;
%cinemática inversa:
teta2d = acos((xd^2+yd^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2d = -teta2d; %caso queira usar a solução com cotovelo para cima
teta1d = atan2(yd,xd) - acos((xd^2+yd^2+l1^2-l2^2)/(2*l1*sqrt(xd^2+yd^2)));
%teta3d = (pi/2) - teta2d - teta1d
%Jacobiano
c1d = cos(teta1d); c12d = cos(teta1d+teta2d); s1d = sin(teta1d); s12d =
sin(teta1d+teta2d);
Jd = [[-l1*s1d-l2*s12d -l2*s12d];[l1*c1d+l2*c12d l2*c12d]];
tetadp = inv(Jd)*[xdp ydp]';
teta1dp = tetadp(1); teta2dp = tetadp(2);%teta3p = tetadp(3);
tetadpp = inv(Jd)*[xdpp+l1*c1d*teta1dp^2+l2*c12d*(teta1dp+teta2dp)^2;
ydpp+l1*s1d*teta1dp^2+l2*s12d*(teta1dp+teta2dp)^2];
teta1dpp = tetadpp(1); teta2dpp = tetadpp(2); %teta3dpp = tetadpp(3);

```

```

wn = 15; Kp = wn^2; %frequência natural desejada (para ambas as juntas)
zeta = 0.76; Kd = 2*zeta*wn; %overshoot de 2%, settling time ts = 4/(zeta*wn)
%0 sem controle, 1 CTC(PD+ff) 0.5Hz
if (caso==0) %sem controle
    tau1 = 0;
    tau2 = 0;
elseif (caso==1)%controle de torque computado com PD + feedforward
    u1 = teta1dpp + Kp*(teta1d-teta1) + Kd*(teta1dp-teta1p);
    u2 = teta2dpp + Kp*(teta2d-teta2) + Kd*(teta2dp-teta2p);
    tau1 = H11*u1 + H12*u2 + G1 - 2*h*teta1p*teta2p - h*teta2p^2;
    tau2 = H22*u2 + H12*u1 + G2 + h*teta1p^2;

end

% Equações da dinâmica: (teta1pp é a segunda derivada de teta1, idem para teta2pp
e teta2)
% tau1 = H11*teta1pp + H12*teta2pp - h*teta2p^2 - 2*h*teta1p*teta2p + G1;
% tau2 = H22*teta2pp + H12*teta1pp + h*teta1p^2 + G2;
% resolvendo, temos (ou pode escrever as equações abaixo em forma vetorial):
teta1pp = (H22*tau1-H12*tau2+H22*h*teta2p^2+H22*2*h*teta1p*teta2p-
H22*G1+H12*h*teta1p^2+H12*G2)/(H11*H22-H12^2);
teta2pp = (tau2 - H12*teta1pp - h*teta1p^2 - G2)/H22;

    qponto = [teta1p teta2p teta1pp teta2pp]'; %precisa ser vetor coluna

end

```

12 – Anexo D – Código de Matlab para segunda simulação

```

clear
close all
global l1 l2 lc1 lc2 m1 m2 m3 I1 I2 g;
global xc yc r f ;
global caso ;

caso = 1; %0 sem controle, 1 CTC(PD+ff) 2Hz

l2 = 0.53; l1 = 0.44; %metro
lc1 = l1/2; lc2 = l2/2; %elo homogêneo
m2 = 5; m1 = 8; %kg
I1 = m1*l1^2/12; I2 = m2*l2^2/12; %assumindo uma barra homogênea
g = -9.81; %m/s^2, gravidade
xc = 0;
ycmin = (l1+l2)*sin(pi/4);
ycmax = (l1+l2)*sin(pi/180*88);
yc = (ycmin+ycmax)/2; r = (ycmax-ycmin)/2; %centro e raio da trajetória
f = 0.4; %hz
w = 2.51327412; %frequencia em radianos por segundo

x0 = 0;
y0 = ycmax;
%cinemática inversa:
teta2 = acos((x0^2+y0^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2 = -teta2; %caso queira usar a solução com cotovelo para cima
teta1 = atan2(y0,x0) - acos((x0^2+y0^2+l1^2-l2^2)/(2*l1*sqrt(x0^2+y0^2)));
%teta3 = (pi/180)*90 - teta1 - teta2

q0 = [teta1 teta2 0 0]'; %estado inicial em radianos: [teta1 teta2 teta1ponto
teta2ponto]

tf = 5; %tempo final de simulação
t = 0:0.015:tf;
[t,q] = ode45('robosimtcc2',t,q0);
qponto =zeros(size(q));
H11 = zeros(size(q(:,1)));
H12 = zeros(size(q(:,1)));
H21 = zeros(size(q(:,1)));
h = zeros(size(q(:,1)));
G1 = zeros(size(q(:,1)));
G2 = zeros(size(q(:,1)));
H22 = m2*lc2^2 + I2;
x = l1/2;
k = 2.4870e+03;
d = zeros(size(q(:,1)));
% d = sqrt(l1^2+x^2-2*l1*x*cos(pi-q(:,2)));
% teta2min = min(q(:,2));
d0 = sqrt(l2^2+x^2-2*l2*x*cos(pi-0.0708));
% %Um = m1*g*yc1+m2*g*yc2+m3*g*yc3 + 0.5*k*(d-d0)^2;
G2m = G2;

```

```

G2n11 = G2;
G2n12 = G2;
G2n13 = G2;
G2n14 = G2;

for index= 1:size(t)

    qponto(index,:)=robosimtcc2(t(index),q(index,:));

    %valores calculados

    H11(index,1) = I1 + I2 + l1^2*m2 + lc1^2*m1 + lc2^2*m2 +
    2*l1*lc2*m2*cos(q(index,2));

    H12(index,1) = m2*lc2^2 + l1*m2*cos(q(index,2))*lc2 + I2;

    H21(index,1)=H12(index,1);

    %só ha um componente principal

    h(index,1)=(l1*sin(q(index,2))*(l2*m3 + lc2*m2));

    G1(index,1) = (l1*cos(q(index,1)) + g*m2*(lc2*cos(q(index,1) + q(index,2)) +
    l1*cos(q(index,1)))) + g*lc1*m1*cos(q(index,1));

    G2(index,1) = (g*cos(q(index,1) + q(index,2))*(lc2*m2));

    d(index,1) = sqrt(l2^2+x^2-2*l2*x*cos(pi-q(index,2)));

    G2m(index,1) = (g*lc2*m2*cos(q(index,1) + q(index,2)) + k*l2*x*sin(q(index,2))
    + (d0*k*l2*x*sin(q(index,2)))/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));

    G2n11(index,1)= (g*lc2*m2*cos(q(index,1) + q(index,2)) +
    (78938.06*l2*x*sin(q(index,2)))*(d0 - sqrt(l2^2 + 2*cos(q(index,2))*l2*x +
    x^2))^3)/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));

    G2n12(index,1)= (g*lc2*m2*cos(q(index,1) + q(index,2)) +
    (14011.506*l2*x*sin(q(index,2)))*(d0 - sqrt(l2^2 + 2*cos(q(index,2))*l2*x +
    x^2))^3)/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));

    G2n13(index,1)= (g*lc2*m2*cos(q(index,1) + q(index,2)) +
    (444721.465*l2*x*sin(q(index,2)))*(d0 - sqrt(l2^2 + 2*cos(q(index,2))*l2*x +
    x^2))^3)/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));

    G2n14(index,1)= (g*lc2*m2*cos(q(index,1) + q(index,2)) +
    (2505473.042*l2*x*sin(q(index,2)))*(d0 - sqrt(l2^2 + 2*cos(q(index,2))*l2*x +
    x^2))^3)/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));

    G2n14test(index,1)= (g*lc2*m2*cos(q(index,1) + q(index,2)) +
    (2505473.042*l2*x*sin(q(index,2)))*(d0 - sqrt(l2^2 + 2*cos(q(index,2))*l2*x +
    x^2))^3)/sqrt(l2^2 + 2*cos(q(index,2))*l2*x + x^2));
end

%cinemática direta:

```

```

x = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
y = l1*sin(q(:,1)) + l2*sin(q(:,1)+q(:,2));

% Jacobiano
J11 = -l1*sin(q(:,1)) - l2*sin(q(:,1)+q(:,2));
J12 = -l2*sin(q(:,1)+q(:,2));
%J13=1;
J21 = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
J22 = l2*cos(q(:,1)+q(:,2));
% J23=1;
% J31=0;
% J32=0;
% J33=1;

% Jacobiano dos centros
% J11c1 = -lc1*sin(q(:,1));
% J12c1 = 0;
% J21c1 = lc1*cos(q(:,1));
% J22c1 = 0;
% J13c1 = 0;
% J23c1 = 0;
%
% J11c2 = -l1*sin(q(:,1)) - lc2*sin(q(:,1)+q(:,2));
% J12c2 = -lc2*sin(q(:,1)+q(:,2));
% J21c2 = l1*cos(q(:,1)) + lc2*cos(q(:,1)+q(:,2));
% J22c2 = lc2*cos(q(:,1)+q(:,2));
% J13c2 = 0;
% J23c2 = 0;
%
% J11c3 = -l1*sin(q(:,1)) - l2*sin(q(:,1)+q(:,2));
% J12c3 = -l2*sin(q(:,1)+q(:,2));
% J21c3 = l1*cos(q(:,1)) + l2*cos(q(:,1)+q(:,2));
% J22c3 = l2*cos(q(:,1)+q(:,2));
% J13c3 = 0;
% J23c3 = 0;

% toques sem molas
for i = 1:size(t)
    tau1(i,1) = qponto(i,3).*H11(i)+ qponto(i,4).*H12(i)-2*h(i)*q(i,3)*q(i,4)-
h(i)*q(i,4)*q(i,4) + G1(i);
    tau2(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) + G2(i);
end

%torques com mola linear
for i = 1:size(t)
    tau2m(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2m(i);
end

%torques com mola nao linear

for i = 1:size(t)

```

```

    tau2n11(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n11(i);
    tau2n12(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n12(i);
    tau2n13(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n13(i);
    tau2n14(i,1)=qponto(i,4).*H22+ qponto(i,3).*H12(i) + h(i)*q(i,4)*q(i,3) +
G2n14(i);
    taudiff(i,1)=tau2m(i)-tau2n12(i,1);
end
taudiffmed=median(taudiff);
Potmed=taudiffmed*w; %potencia media
EnergiaGasta2=Potmed*5; %diferente de energia entre a mola linear e a não

% i=(0:00025:01775);
% ym1=zeros(size(i));
% ym2=zeros(size(i));

fh = figure(1);
set(fh, 'doublebuffer', 'on'); % for smoothness in the animation
fh.WindowState = 'maximized';
drawnow;

%posição da origem do sistema de coordenadas 1:
x1 = l1*cos(q(:,1));
y1 = l1*sin(q(:,1));
plot(0,0,'bo')
hold on
axis([-l1/2,xc+1.5*r,-l1/2,yc+1.5*r])
axis equal
plot(xc+r*cos([0:pi/10:2*pi]), yc+r*sin([0:pi/10:2*pi]), 'k--');
for i = 1:size(t,1)
    plot([0 x1(i) x(i)], [0 y1(i) y(i)], 'b'); %desenha o manipulador
    drawnow;
    plot([0 x1(i) x(i)], [0 y1(i) y(i)], 'w'); %apaga o manipulador
    plot(x(i),y(i), 'r. '); %desenha o rastro da extremidade
end

figure(2)
plot(t,q(:,1)*180/pi, 'b')
hold on
plot(t,q(:,2)*180/pi, 'r')
title('ângulos das juntas')
xlabel('tempo(s)')
ylabel('ângulo(graus)')
legend('teta1', 'teta2')

figure(3)
plot(x,y)
hold on
plot(xc+r*cos([0:pi/10:2*pi]), yc+r*sin([0:pi/10:2*pi]), 'k--');
axis equal
switch caso
    case 0
        title('sem controle (motores desligados)')
    case 1
        title('CTC com PD + feedforward, 2Hz')
end

```

```

xlabel('extremidade do manipulador x (m)')
ylabel('extremidade do manipulador y (m)')

figure(4)

plot(t,tau2,'r--')
hold on

plot(t,tau2m,'b--')
%plot(t,tau2n11,'k')
plot(t,tau2n12,'r')
%plot(t,tau2n13,'b')
%plot(t,tau2n14,'m')

title('torques das juntas')
xlabel('tempo(s)')
ylabel('Torque')
legend("torque sem mola", 'Mola linear', 'Mola não linear 2')
%legend("torque puro", 'Mola linear', 'Mola não linear 1', 'Mola não linear 2', 'Mola
não linear 3', 'Mola não linear 4')

% figure(5)
% hold on
% plot(t,G2,'k')
% plot(t,G2m,'r')
% plot(t,G2n11,'b')
% plot(t,G2n12,'m')
% plot(t,G2n13,'g')
% plot(t,G2n14,'c')
% title('Forças gravitacionais')
% xlabel('tempo(s)')
% ylabel('Força')
% legend('Gravitacional sem mola', 'Com mola linear', 'Mola não linear 1', 'Mola não
linear 2', 'Mola não linear 3', 'Mola não linear 4')

```

13 Anexo E – Função para segunda simulação

```

function qponto = robosimtcc2(t, q)
%equações para simular um robô de 2 graus de liberdade
global l1 l2 lc1 lc2 m1 m2 m3 I1 I2 g ;
global xc yc r f ;
global caso gamainv;

teta1 = q(1);
teta2 = q(2);
teta1p = q(3); %derivada de teta1
teta2p = q(4); %derivada de teta2
t

%valores calculados
H11 = m2*l1^2 + 2*m2*cos(teta2)*l1*lc2 + m1*lc1^2 + m2*lc2^2 + I1 + I2;
H12 = m2*lc2^2 + l1*m2*cos(teta2)*lc2 + I2;
H21=H12;
H22 = m2*lc2^2 + I2;
%só ha um componente principal
h=l1*sin(teta2)*(l2*m3 + lc2*m2);
G1 = g*m2*(lc2*cos(teta1 + teta2) + l1*cos(teta1)) + g*lc1*m1*cos(teta1);
G2 = g*lc2*m2*cos(teta1 + teta2);

%trajetória desejada:
xd = 0 ;
yd = yc + r*cos(2*pi*f*t);
xdp = 0;
ydp = -r*sin(2*pi*f*t)*(2*pi*f);
xdpp = 0;
ydp = -r*cos(2*pi*f*t)*(2*pi*f)^2;
%cinemática inversa:
teta2d = acos((xd^2+yd^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2d = -teta2d; %caso queira usar a solução com cotovelo para cima
teta1d = atan2(yd,xd) - acos((xd^2+yd^2+l1^2-l2^2)/(2*l1*sqrt(xd^2+yd^2)));
%teta3d = (pi/2) - teta2d - teta1d
%Jacobiano
c1d = cos(teta1d); c12d = cos(teta1d+teta2d); s1d = sin(teta1d); s12d =
sin(teta1d+teta2d);
Jd = [-l1*s1d-l2*s12d -l2*s12d];[l1*c1d+l2*c12d l2*c12d]];
tetadp = inv(Jd)*[xdp ydp]';
teta1dp = tetadp(1); teta2dp = tetadp(2);%teta3p = tetadp(3);
tetadpp = inv(Jd)*[xdpp+l1*c1d*teta1dp^2+l2*c12d*(teta1dp+teta2dp)^2;
ydpp+l1*s1d*teta1dp^2+l2*s12d*(teta1dp+teta2dp)^2];
teta1dpp = tetadpp(1); teta2dpp = tetadpp(2); %teta3dpp = tetadpp(3);

wn = 15; Kp = wn^2; %frequência natural desejada (para ambas as juntas)
zeta = 0.76; Kd = 2*zeta*wn; %overshoot de 2%, settling time ts = 4/(zeta*wn)
%0 sem controle, 1 CTC(PD+ff) 0.5Hz
if (caso==0) %sem controle
    tau1 = 0;

```

```

    tau2 = 0;
elseif (caso==1)%controle de torque computado com PD + feedforward
    u1 = teta1dpp + Kp*(teta1d-teta1) + Kd*(teta1dp-teta1p);
    u2 = teta2dpp + Kp*(teta2d-teta2) + Kd*(teta2dp-teta2p);
    tau1 = H11*u1 + H12*u2 + G1 - 2*h*teta1p*teta2p - h*teta2p^2;
    tau2 = H22*u2 + H12*u1 + G2 + h*teta1p^2;

end

% Equações da dinâmica: (teta1pp é a segunda derivada de teta1, idem para teta2pp
e teta2)
% tau1 = H11*teta1pp + H12*teta2pp - h*teta2p^2 - 2*h*teta1p*teta2p + G1;
% tau2 = H22*teta2pp + H12*teta1pp + h*teta1p^2 + G2;
% resolvendo, temos (ou pode escrever as equações abaixo em forma vetorial):
teta1pp = (H22*tau1-H12*tau2+H22*h*teta2p^2+H22*2*h*teta1p*teta2p-
H22*G1+H12*h*teta1p^2+H12*G2)/(H11*H22-H12^2);
teta2pp = (tau2 - H12*teta1pp - h*teta1p^2 - G2)/H22;

    qponto = [teta1p teta2p teta1pp teta2pp]'; %precisa ser vetor coluna

end

```

14 Anexo F – Modelagem para segunda simulação

```

function qponto = robosimtcc2(t, q)
%equações para simular um robô de 2 graus de liberdade
global l1 l2 lc1 lc2 m1 m2 m3 I1 I2 g ;
global xc yc r f ;
global caso gamainv;

teta1 = q(1);
teta2 = q(2);
teta1p = q(3); %derivada de teta1
teta2p = q(4); %derivada de teta2
t

%valores calculados
H11 = m2*l1^2 + 2*m2*cos(teta2)*l1*lc2 + m1*lc1^2 + m2*lc2^2 + I1 + I2;
H12 = m2*lc2^2 + l1*m2*cos(teta2)*lc2 + I2;
H21=H12;
H22 = m2*lc2^2 + I2;
%só ha um componente principal
h=l1*sin(teta2)*(l2*m3 + lc2*m2);
G1 = g*m2*(lc2*cos(teta1 + teta2) + l1*cos(teta1)) + g*lc1*m1*cos(teta1);
G2 = g*lc2*m2*cos(teta1 + teta2);

%trajetória desejada:
xd = 0 ;
yd = yc + r*cos(2*pi*f*t);
xdp = 0;
ydp = -r*sin(2*pi*f*t)*(2*pi*f);
xdpp = 0;
ydp = -r*cos(2*pi*f*t)*(2*pi*f)^2;
%cinemática inversa:
teta2d = acos((xd^2+yd^2-l1^2-l2^2)/(2*l1*l2)); %solução com cotovelo para baixo
%teta2d = -teta2d; %caso queira usar a solução com cotovelo para cima
teta1d = atan2(yd,xd) - acos((xd^2+yd^2+l1^2-l2^2)/(2*l1*sqrt(xd^2+yd^2)));
%teta3d = (pi/2) - teta2d - teta1d
%Jacobiano
c1d = cos(teta1d); c12d = cos(teta1d+teta2d); s1d = sin(teta1d); s12d =
sin(teta1d+teta2d);
Jd = [-l1*s1d-l2*s12d -l2*s12d];[l1*c1d+l2*c12d l2*c12d]];
tetadp = inv(Jd)*[xdp ydp]';
teta1dp = tetadp(1); teta2dp = tetadp(2);%teta3p = tetadp(3);
tetadpp = inv(Jd)*[xdpp+l1*c1d*teta1dp^2+l2*c12d*(teta1dp+teta2dp)^2;
ydpp+l1*s1d*teta1dp^2+l2*s12d*(teta1dp+teta2dp)^2];
teta1dpp = tetadpp(1); teta2dpp = tetadpp(2); %teta3dpp = tetadpp(3);

wn = 15; Kp = wn^2; %frequência natural desejada (para ambas as juntas)
zeta = 0.76; Kd = 2*zeta*wn; %overshoot de 2%, settling time ts = 4/(zeta*wn)
%0 sem controle, 1 CTC(PD+ff) 0.5Hz
if (caso==0) %sem controle
    tau1 = 0;

```

```

    tau2 = 0;
elseif (caso==1)%controle de torque computado com PD + feedforward
    u1 = teta1dpp + Kp*(teta1d-teta1) + Kd*(teta1dp-teta1p);
    u2 = teta2dpp + Kp*(teta2d-teta2) + Kd*(teta2dp-teta2p);
    tau1 = H11*u1 + H12*u2 + G1 - 2*h*teta1p*teta2p - h*teta2p^2;
    tau2 = H22*u2 + H12*u1 + G2 + h*teta1p^2;

end

% Equações da dinâmica: (teta1pp é a segunda derivada de teta1, idem para teta2pp
e teta2)
% tau1 = H11*teta1pp + H12*teta2pp - h*teta2p^2 - 2*h*teta1p*teta2p + G1;
% tau2 = H22*teta2pp + H12*teta1pp + h*teta1p^2 + G2;
% resolvendo, temos (ou pode escrever as equações abaixo em forma vetorial):
teta1pp = (H22*tau1-H12*tau2+H22*h*teta2p^2+H22*2*h*teta1p*teta2p-
H22*G1+H12*h*teta1p^2+H12*G2)/(H11*H22-H12^2);
teta2pp = (tau2 - H12*teta1pp - h*teta1p^2 - G2)/H22;

    qponto = [teta1p teta2p teta1pp teta2pp]'; %precisa ser vetor coluna

end

```

15 Anexo G – Equações

```

close all
clear
syms alpha1 beta1 alpha2 beta2 alpha3 beta3
x=0:0.005:0.1775;
k3 = 2.4870e+03;
eqn1 = 14011.5057*0.1775^2==beta1*0.1775+alpha1*0.1775^3;
eqn2 = 14011.5057*0.0887^2==beta1*0.1+alpha1*0.0887^3;
s1= (solve([eqn1,eqn2],[alpha1,beta1])); %s1 devolve valores para alpha1 e beta1
beta1=708.4157; %valor numerico de alpha1
alpha1= 5.6453e+04; %valor numerico de beta1
eqn3 = (441.45/0.1775^4)*0.1775^4== beta2*0.1775+alpha2*0.1775^3;
eqn4 = (441.45/0.1775^4)*0.0887^4== beta2*0.1775+alpha2*0.0887^3;
s2= (solve([eqn3,eqn4],[alpha2,beta2])); %s2 devolve valores para alpha2 e beta2
beta2=-177.4029; %valor numerico de alpha2
alpha2= 8.4569e+04; %valor numerico de beta2
eqn5 = (441.45/0.1775^5)*0.1775^5== beta3*0.1775+alpha3*0.1775^3;
eqn6 = (441.45/0.1775^5)*0.0887^5== beta3*0.0887+alpha3*0.0887^3;
s3= (solve([eqn5,eqn6],[alpha3,beta3])); %s3 devolve valores para alpha3 e beta3
beta3= -621.0602; %valor numerico de alpha3
alpha3= 9.8650e+04; %valor numerico de beta3
N12= zeros(size(x));
N13= zeros(size(x));
N14= zeros(size(x));
Lit= zeros(size(x));
Lit2= zeros(size(x));
Lit3= zeros(size(x));
for i=1:36
    N12(i)= 14011.5057*x(i)^2;
    N13(i)= 444721.465*x(i)^4;
    N14(i)= 2505473.04*x(i)^5;
    Lit(i)= alpha1*x(i)^3+beta1*x(i);
    Lit2(i)= alpha2*x(i)^3+beta2*x(i);
    Lit3(i)= alpha3*x(i)^3+beta3*x(i);
end
figure(1)
plot(x,N12,'r')
hold on
plot(x,Lit,'b')
title('Equação proposta pela literatura e Equação estimada')
xlabel('deformação (m)')
ylabel('força elastica')
legend('Equação estimada(Mola 2)', 'Equação equivalente')

figure(2)
plot(x,N13,'r')
hold on
plot(x,Lit2,'b')
title('Equação proposta pela literatura e Equação estimada')
xlabel('deformação (m)')
ylabel('força elastica')
legend('Equação estimada(Mola 3)', 'Equação equivalente')

```

```
figure(3)
plot(x,Nl4,'r')
hold on
plot(x,Lit3,'b')
title('Equação proposta pela literatura e Equação estimada')
xlabel('deformação (m)')
ylabel('força elastica')
legend('Equação estimada(Mola 4)','Equação equivalente')
```