



Fundação Universidade Federal do ABC

Pró reitoria de pesquisa

Av. dos Estados, 5001, Santa Terezinha, Santo André/SP, CEP 09210-580

Bloco L, 3ºAndar, Fone (11) 3356-7617

iniciacao@ufabc.edu.br

Relatório de iniciação científica referente ao edital: 03/2023

Nome do aluno: Matheus Lourenço Aires de Pontes Siqueira

Assinatura do aluno:

Nome do orientador: Magno Enrique Mendoza Meza

Assinatura do orientador:

Título do projeto: Projeto e construção de robô quadrúpede de baixo custo para pesquisas de locomoção, controle e fins didáticos de robótica.

Palavras-chave do projeto: Robô quadrúpede, legged robot, manufatura aditiva, Arduino, open source, biomimética

Área do conhecimento do projeto: robótica, controle e automação

Bolsista: CNPQ

Santo Andre

26/09/2024

Sumário

1	Resumo	3
2	Introdução	3
3	Fundamentação Teórica	5
4	Metodologia	10
4.1	Materiais e métodos	10
4.2	Etapas de pesquisa	13
5	Resultados e discussão dos resultados	14
6	Conclusão e perspectiva de trabalhos futuros	23
7	Referências	24

1 Resumo

Este trabalho consiste no projeto e construção de um robô quadrúpede, cuja locomoção por pernas mecânicas biomiméticas confere mais acessibilidade a terrenos irregulares, se comparado à robôs com rodas. O projeto visa baixo custo, composto por peças impressas em 3D, parafusos, rolamentos e componentes eletrônicos acessíveis nos mercados, além da programação de código aberto visando uma tecnologia facilmente replicável para futuras pesquisas em laboratórios de universidades ou escolas técnicas. Esta ferramenta será útil como *framework* nos estudos de controle, locomoção, inteligência artificial e diversas áreas relacionadas a ampla área de robôs móveis. Ao final do projeto estará disponível um protótipo de robô quadrúpede com todos seus componentes operando adequadamente, e é desejável realizar testes de locomoção e leis de controle para validar o seu funcionamento.

2 Introdução

Nos últimos anos, o uso de robôs tem se expandido para além das áreas industriais, como aeroespacial, a medicina e etc [5], e o tipo do robô aqui proposto, robô com pernas ou *legged robots*, está cada vez mais popular devido às boas performances em terrenos desafiadores e que são inacessíveis aos robôs tradicionais com rodas, devido à sua natureza quadrúpede e mais dinâmica de locomoção [9]. A exemplo, tem-se os robôs *Spot* da *Boston Dynamics*, *ANYmal C* da *Anybotics* e o *Mini Cheetah* do *MIT* que são robôs biomiméticos e de uso geral para atividades em superfícies irregulares como escadas, minas subterrâneas e na neve.

Apesar do crescente uso desta tecnologia, estes robôs ainda apresentam um custo elevado. Em vista disso, a *Open Dynamic Robot Initiative* [4], compostas por entidades científicas internacionais de renome, propuseram a compilação de um repositório online que auxilia na construção destes robôs, cobrindo diversas áreas adjacentes, com peças impressas em 3D e eletrônicos disponíveis nos mercados.

Entretanto, para a realidade brasileira, há peças e atuadores que necessitam ser importados, tornando a implementação de um robô deste consideravelmente dispendioso. Sendo assim, a proposta deste projeto é construir, baseado nos materiais e métodos alternativos disponíveis online, uma forma mais simples e funcional deste tipo de robô, com custos e materiais acessíveis aos laboratórios de universidades e escolas brasileiras.

Nas figuras 1 e 2 podem ser observadas respectivamente a vista em perfil do robô e o seu sistema coordenadas, sendo (X_0, Y_0, Z_0) as coordenadas centrais.

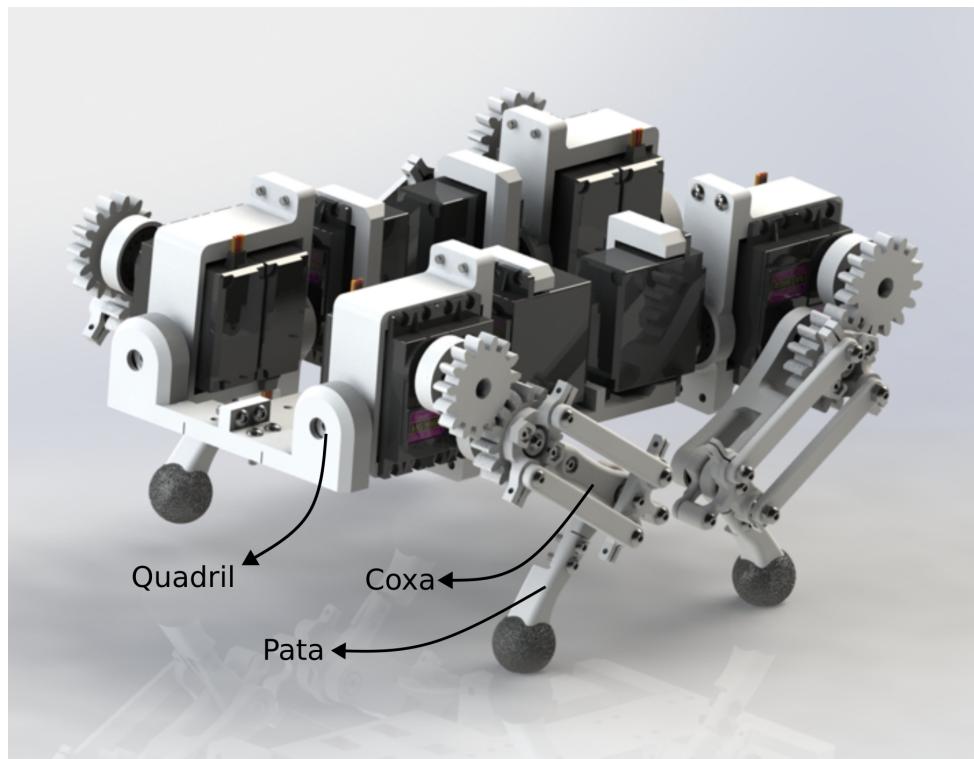


Figura 1: Anatomia do protótipo final.

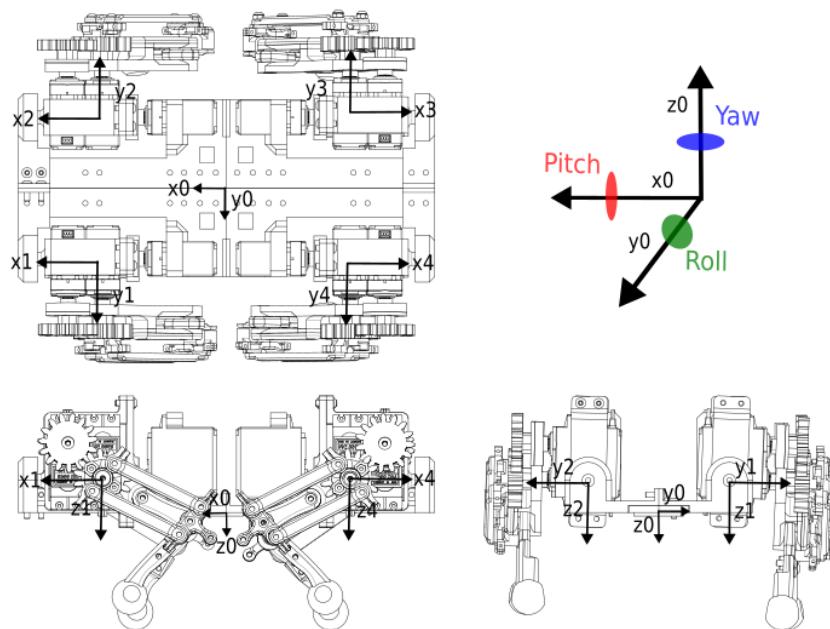


Figura 2: Sistema de coordenadas do robô

3 Fundamentação Teórica

Para fazer um robô quadrúpede, o componente mais importante são as pernas mecânicas, que são elementos compostos por barras, elos e motores entre as juntas. Estas componentes devem conter comprimentos e geometrias que garantam uma vantagem mecânica. Para realizar o movimento adequado de uma perna, tem-se os servo motores, que transformam os ângulos das juntas conforme desejado, além destes possuírem um elevado torque em um pequeno espaço, o que os possibilita servir como músculos nas juntas. Além da parte mecânica, tem se as relações matemáticas e computacionais necessárias para a correta ordenação esperada de suas juntas conforme coordenadas de entrada, são essas as relações determinadas de cinemática inversa e direta.

A começar com a anatomia do robô, segundo a figura 1, tem-se que cada perna deste possui 3 graus de liberdade, um motor para cada, sendo assim é possível representar um diagrama simplificado de elos e juntas para estudar a dinâmica de seu movimento, como mostrado na figura 3, em que $(\theta_1, \theta_2, \beta)$ são os ângulos variáveis das juntas rotacionadas pelos motores servo.

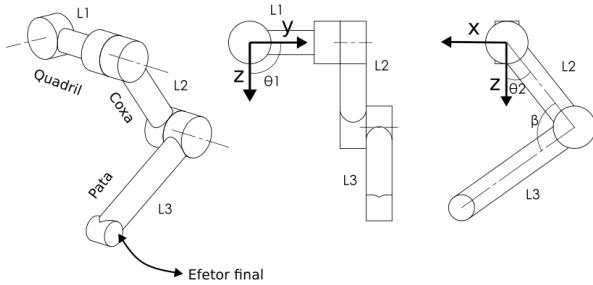


Figura 3: Ilustração simplificada de uma perna

Sobre o design dos elementos da perna, tem-se que as barras paralelas de transmissão de movimento localizadas entre a coxa nas figuras 1 e 2 seguem a lei de Grashof para 4 elos conectados em um paralelogramo [7], que configuram em uma baixa dissipação de energia, dinâmica simplificada e alta rigidez [5], o que permite que os atuadores se concentrem próximos ao corpo do robô e não dispersos nas juntas adicionando peso e interferindo mecanicamente, o que aumentaria o momento de inércia e dissipação de torque útil entre as juntas conectadas. A vantagem deste mecanismo é melhor ilustrada quando o robô levanta uma perna e isso não o desequilibra, pois cada perna é leve se comparada com as que usam motores diretamente nas juntas [1].

Projetar um mecanismo é desenvolver um que transforme movimento em vantagem mecânica [7], para isso, os valores ideais de (L_1, L_2, L_3) dispostos na figura 3 consideram

restrições físicas do mundo real: peso, comprimento dos elos da perna e torque do motor. Com isso, tem-se o estudo matemático do caso extremo de uso dos atuadores indicado pelas equações **1**, **2** e **3**, em que o robô utiliza de no mínimo 2 motores para sustentação, como estudado em [1]. Sendo F_1 e F_2 as forças geradas por cada motor e τ o torque de cada motor, obtém-se em **3** o comprimento máximo de cada elo, em relação a uma massa e torque do motor já conhecidos.

$$F_1 + F_2 \geq mg \quad (1)$$

$$\frac{\tau}{L_i} + \frac{\tau}{L_i} \geq mg \quad (2)$$

$$\frac{2\tau}{mg} \geq L_i, i = 1, 2, 3 \quad (3)$$

Para a correta ordenação das juntas e elos, é utilizado a relação matemáticas denominada de cinemática direta, a função que recebe como entrada os ângulos desejados ($\theta_1, \theta_2, \beta$) e retorna as coordenadas do efetor final, que assim como desenvolvido em [9] pode ser determinado usando o algoritmo de *Denavit-Hartenberg*, que para este robô cada perna possui os parâmetros indicados na tabela **1**.

θ	d	a	α
θ_1	0	0	$\frac{\pi}{2}$
0	L_1	0	0
θ_2	0	L_2	0
β	0	L_3	0

Tabela 1: Parâmetros DH

Este método matemático é útil para descobrir as matrizes homogêneas de transformação e translação da base (motor mais próximo do quadril ou origem 0) até o efetor final (ponta da pata ou ponto número 3), obtendo-se assim o vetor coluna responsável pelas coordenadas finais do efetor $[X_3^0, Y_3^0, Z_3^0]^T$ dado por **4**. Este vetor auxilia na determinação do espaço de trabalho da perna robótica para guiar simulações e trajetórias das juntas dentro dos limites, ao plotar-se a saída dessas transformações tendo como entrada os limites angulares de cada junta.

$$\begin{bmatrix} L_1 \sin(\theta_1) + L_2 \cos(\theta_1) \cos(\theta_2) - L_3 \sin(\theta_2) \sin(\beta) \cos(\theta_1) + L_3 \cos(\theta_1) \cos(\theta_2) \cos(\beta) \\ -L_1 \cos(\theta_1) + L_2 \sin(\theta_1) \cos(\theta_2) - L_3 \sin(\theta_1) \sin(\theta_2) \sin(\beta) + L_3 \sin(\theta_1) \cos(\theta_2) \cos(\beta) \\ L_2 \sin(\theta_2) + L_3 \sin(\theta_2) \cos(\beta) + L_3 \sin(\beta) \cos(\theta_2) \end{bmatrix} \quad (4)$$

Outra relação para a correta ordenação das juntas é a da cinemática inversa, que tem

como entrada as coordenadas x,y,z do efetor final e como saída os valores dos ângulos dos motores servos calculados ($\theta_1, \theta_2, \beta$). Dentre as diversas formas de resolver este problema, esta relação pode ser desenvolvida por meio do método geométrico que utiliza da lei dos cossenos, lei dos senos e de matrizes de transformação para assim obter-se uma solução analítica para esse problema, assim como desenvolvido em [3]. A solução para esta perna é dada pelas equações **6**, **10** e **12**, em que (x, y, z) são as variáveis de entrada para cada motor efetuar a saída $(\theta_1, \theta_2, \theta_3)$ desejada para a correta trajetória do efetor final. Como consequência do design de barras de transmissão paralelas, é usado a relação $\beta = \frac{3\pi}{2} + \theta_3$, em que θ_3 refere-se à entrada do motor servo localizado nas juntas responsáveis pelo movimento da pata.

$$c = \sqrt{z^2 + y^2} \quad (5)$$

$$\theta_1 = \pi - \text{atan2}(z, y) - \sin^{-1} \left(\frac{L_1}{c} \right) \quad (6)$$

$$R_x = -\frac{\pi}{2} + \theta_1 \quad (7)$$

$$z_2 = y \sin(R_x) + z \cos(R_x) \quad (8)$$

$$h = \sqrt{z_2^2 + x^2} \quad (9)$$

$$\theta_2 = \cos^{-1} \left(\frac{h^2 + L_2^2 - L_3^2}{2 \cdot L_2 \cdot h} \right) - \text{atan2}(x, z) \quad (10)$$

$$\beta = \frac{-h^2 + L_2^2 + L_3^2}{2 \cdot L_2 \cdot L_3} \quad (11)$$

$$\theta_3 = -\theta_2 + \frac{3\pi}{2} - \cos^{-1}(\beta) \quad (12)$$

Com a correta ordenação do efetor final do plano cartesiano (x,y,z), pode-se aplicar equações de movimento para cada perna de um robô quadrúpede para assim ordenar o movimento total de todo o corpo. Para este movimento, é necessário que cada perna possua uma equação que planeja a trajetória do efetor final a fim da locomoção correta, a exemplo deseja-se que o efetor faça uma meia curva cicloide para o algoritmo de deslocamento, esta trajetória será discutida mais adiante.

A equação auxiliar na trajetória usada neste projeto é a curva paramétrica de *Bezier* [10] explicitada pelas equações **13** e **14**, que é uma curva que se ajusta por pontos de controle (**figura 4**).

$$B(t) = \sum_{i=0}^n \binom{n}{i} (1-t)^{n-i} t^i P_i \quad (13)$$

No caso deste projeto, são necessários apenas 4 pontos de controle:

$$B(t) = (1-t)^3 P_0 + 3(1-t)^2 t P_1 + 3(1-t)t^2 P_2 + t^3 P_3 \quad (14)$$

Onde P_0 , P_1 , P_2 e P_3 são as coordenadas de controle e t varia de 0 a 1.

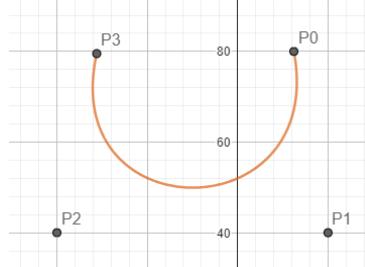


Figura 4: Curva paramétrica de Bezier

A trajetória do efetor final está dividida em duas fases de deslocamento, são elas a de *swing* e *stance*, que são respectivamente a fase em que o efetor final está em uma trajetória curva determinada pelos pontos de controle da curva de Bezier (figura 4) e a outra uma retilínea no solo para impulsionar o robô. Devido à adaptabilidade desta curva por pontos de controle, tem-se a possibilidade de mudança de parâmetros para ambas as fases, sendo o caso para a de *swing* mostrada na figura 5 em que é plotado varias trajetórias de alturas diferentes mas mantendo o comprimento do *stance* constante

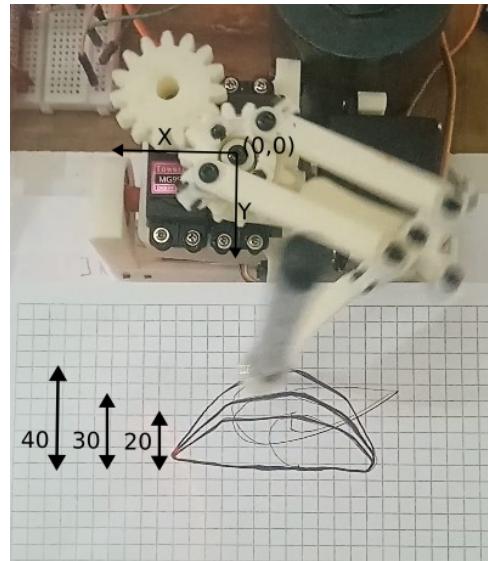


Figura 5: Trajetória da perna robótica com *stance* constante e *swing* variável

Há também a possibilidade de mudança dos parâmetros da a fase de *stance*, como

mostrado na figura 6, em que o efetor final percorre diferentes distâncias retilíneas logo após às trajetórias curvas.

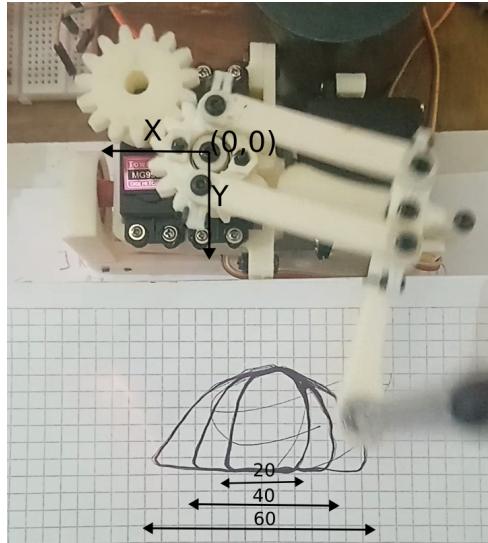


Figura 6: Trajetória da perna robótica com *stence* variável

Por fim, para o software embarcado, além das equações relacionadas às ordenações de juntas e trajetórias de efetor final, tem-se os algoritmos de locomoção, ou *gaits*, apresentados em [8]. Para este projeto, cada perna usa do *crawl gait*, que é composto pelo movimento de *swing* de uma pata por vez enquanto as outras 3 estão apoiadas no chão formando um triângulo contendo o centro de massa de todo o robô, resultando em uma forma relativamente lenta porém estável de locomoção. Além deste algoritmo de locomoção quadrúpede, há também o *trot gait* que é o mais observado em situação reais em animais mamíferos, além de mais rápido, o que se caracteriza por duas patas na fase de *swing* simultaneamente enquanto duas na fase de *stance* a cada iteração. Este ultimo método, por mais que natural e rápido, demanda um controle de malha fechada para o equilíbrio do robô, pois este movimento pode ser instável. Para isso, considera-se o uso de controle *PID* com um sensor *IMU* para alimentar a malha fechada com os valores de ângulos em torno das coordenadas centrais (X_0, Y_0, Z_0), para assim aplicar as devidas matrizes de transformação e translação no efetor final para obter-se o equilíbrio do robô. Como último *gait* será testado brevemente o *bound gait*, a forma de andar que mais se assemelha à de coelhos, entretanto este algoritmo é bastante instável e demanda mais energia se comparado aos algoritmos anteriores.

4 Metodologia

4.1 Materiais e métodos

Para a construção deste robô, são aplicados métodos que cobrem as áreas de eletrônica, como os atuadores e controladores, da mecânica como o design do robô até a fabricação das peças, e por fim da computação com o software embarcado no microcontrolador, responsável pelo controle ordenado das juntas, além da simulação da área em que este processo pode ocorrer para uma locomoção mais eficiente.

A começar pela eletrônica, para o controle dos atuadores e todos os componentes eletrônicos, será usado o microcontrolador *ESP32*, o responsável pelo núcleo e a inteligência do robô, com poder de processamento superior ao *Arduino*, ambos em faixas de preços semelhantes. Esta placa possui conexão sem fio, como o *BLE* e *Wi-Fi*, que possibilita a comunicação à distância com o robô, como envio de imagens, dados e diversos outros comandos além de possuir 34 *GPIOs*, os pinos de entrada e saída de uso geral, que garantem mais conexões de periféricos para os futuros usos gerais deste robô. Além disso, será usado o módulo multiplexador *PCA9685 I₂C* para processamento paralelo de sinal *PWM*, disponibilizando o controlador central para mais periféricos visando futuras pesquisas.

Para a locomoção são empregados servo motores de alto torque redutores de metal do modelo *MG996R*, que possibilitam o controle das juntas das pernas ao especificar o ângulo desejado para o motor efetuar. É necessário um teste de torque em cada motor para assim checar a viabilidade do funcionamento correto das manipulações das juntas. Isso pode ser obtido conforme o experimento em que o braço de alavanca com comprimento conhecido é acoplado ao motor que levanta uma massa crítica conhecida, como demonstrado na figura 7.



Figura 7: Teste de torque para servo motores

Após a medição de torque máximo do motor, assim como o seu consumo de potência elétrica, estes atuadores também precisam ser calibrados para seguir a correta saída angular desejada. Para isso basta relacionar por meio de regressão linear a saída *PWM* que o controlador fornece com a leitura real de ângulo. Para a realização deste experimento, foi construído uma base de ajustes angulares descrito na figura 8, que após cada deslocamento ajustável do ponteiro rosa nos números em graus, obtém-se o equivalente valor em *PWM*, parra assim plotar uma relação linear *PWM* x Ângulo em graus, como demonstrado na figura 9.

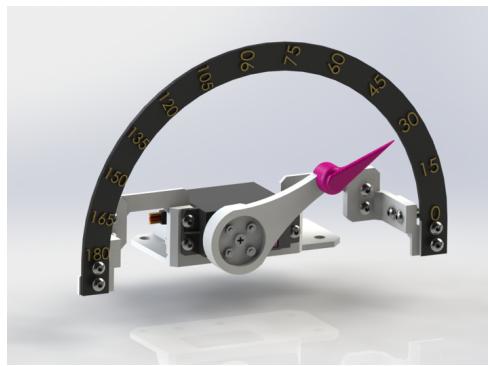


Figura 8: Calibração angular

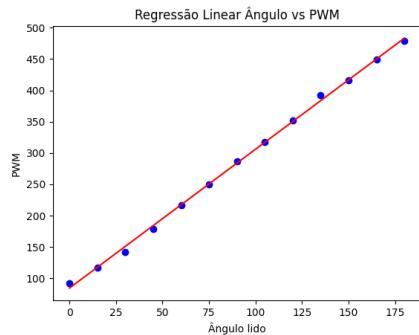


Figura 9: Regressão linear entre PWM e ângulos em graus

Visando a implementação de um sistema de alimentação própria do robô, considera-se a potência consumida durante os testes de torque máximo de cada motor, para assim obter-se a necessidade energética deste robô. Com os devidos dados de tensão e corrente consumidas, tem-se a criação customizada de um banco de baterias de lítio 18650 2600 mAh adequado para o consumo simultâneo energético de cada perna, conectadas a um sensor de balanceamento e proteção *BMS*, para impedir riscos de incêndios e proporcionar recargas seguras para o banco de baterias.

Para a parte mecânica do robô e das peças, usa-se a manufatura aditiva ou impressão 3D, que está cada vez mais utilizada em prototipagem de peças. Além das peças impressas em 3D, usa-se em conjunto parafusos M2 e M3 para a correta fixação, e rolamentos de 11x7x3 , 7x4x3 mm e 20x8x4 mm para uma rotação de baixo atrito entre as juntas. O projeto de peças dá-se pelo software de *CAD Solidworks*, que possibilita não só o projeto, mas também testes e simulações dinâmicas e estáticas, que por fim pouparam um tempo considerável para a implementação das peças e mecanismos do robô antes de sua impressão 3D.

A fim de se obter o design correto das pernas e do robô como um todo, tem-se que além das restrições físicas de comprimento determinadas pela equação 3, há também a escolha de formas e a criação de mecanismos eficientes para realizar a tarefa desejada do robô, e isto necessita de diversas tentativas e erros além de estudos de casos para compreender o resultado obtido em comparação ao esperado, pois no mundo real há diversos fatores complexos que se somam em um sistema. Para isso, usa-se da metodologia *VDI 2221* para a identificação de problemas visando uma solução no design final, como apresentados no livro de *Nigel Cross 2014 [2]* e exemplificados na figura 10.

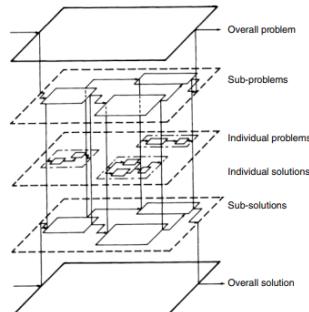


Figura 10: Método VDI 2221

Tratando-se sobre a criação de *softwares* para o robô, no caso do *software* embarcado será usado o ambiente de desenvolvimento *Arduino IDE*, uma plataforma *open source* compatível com o controlador *ESP32*, com diversas bibliotecas úteis para implementação de leitura e controle de sensores. Para outro software que visa determinar o espaço de trabalho do robô, para conhecer-se os limites das juntas e evitar singularidades (casos em que há divisão por zero ou é impossível apenas uma única solução para a ordenação das juntas), é realizado um que plota tridimensionalmente as equações apresentadas em 4, em que as coordenadas do efetor final recebem como entrada os limites dos ângulos de cada motor.

A fim de validar e mensurar as equações de movimento, utiliza-se uma caneta acoplada

ao efetor final, para assim conferir o movimento desejado e mensurar a exatidão do resultado esperado. Estas técnicas estão apresentadas nas figuras **4** e **5**, em que realizam-se diversos testes para a curva de *Bezier*.

Após validar as partes elétricas, mecânicas e computacionais de apenas uma perna, pode-se unir 4 destas em um corpo em comum, conferindo assim um robô de 12 graus de liberdade, capaz de executar movimentos independentes e coordenados por equações de cinemática inversa, podendo-se aplicar os algoritmos de *crawl gait* e *trot gait* como explicados anteriormente.

Por fim, com o robô funcionando em malha aberta, é desenvolvido o controle em malha fechada, para conferir equilíbrio estático e dinâmico durante os deslocamentos do robô. Cada pata deve realizar uma transformação conforme o exigido pela entrada do controle, sendo esta fornecida pelo sensor *IMU MPU-6050*, amplamente usado em projetos com o *ESP32* e de baixo custo. O sistema de controle usará do algoritmo de *PID*, em que o erro será a diferença entre o estado desejado, 0 graus em torno dos eixos X_0 e Y_0 (figura **2**), e o estado observado, os ângulos em torno dos eixos X_0 e Y_0 fornecidos pelo sensor *IMU*.

4.2 Etapas de pesquisa

- Etapa 1
 - Etapa 1.a. Pesquisa de componentes eletrônicos e mecânicos adequados
 - Etapa 1.b. Modelagem matemática, mecânica (CAD) e computacional (software) de uma perna do robô.
 - Etapa 1.c. Impressão das peças, construção e testes de uma perna robótica.
- Etapa 2
 - Etapa 2.a. Remodelagem da perna conforme resultados empíricos
 - Etapa 2.b. Modelagem, impressão de peças e construção do corpo total do robô.
 - Etapa 2.c. Testes do robô e redação do relatório parcial
- Etapa 3
 - Etapa 3.a. Conclusão do relatório parcial
 - Etapa 3.b. Ajustes na construção final do robô conforme resultados empíricos.
 - Etapa 3.c. Redação e conclusão do relatório final

Etapa	Mês											
	01	02	03	04	05	06	07	08	09	10	11	12
1.a	x	x	x									
1.b		x	x	x								
1.c			x	x	x							
2.a				x	x	x						
2.b					x	x	x					
2.c						x	x	x				
3.a							x	x	x			
3.b								x	x	x		
3.c									x	x	x	x

5 Resultados e discussão dos resultados

A começar com os resultados obtidos das constantes, obteve-se os dados da **tabela 2**:

Constante	Valor	Unidade	Descrição
L_1	34	mm	comprimento quadril
L_2	55	mm	comprimento coxa
L_3	55	mm	comprimento pata
C	150, 76	mm	comprimento do robô
L	81, 72	mm	largura do robô
τ	4	kg.cm	torque obtido do motor servo
P_0	(20, 90)	mm	ponto de controle curva de Bezier
P_1	(20, 65)	mm	ponto de controle curva de Bezier
P_2	(-20, 65)	mm	ponto de controle curva de Bezier
P_3	(-20, 90)	mm	ponto de controle curva de Bezier
X_0	0	mm	coordenada central do robô
Y_0	0	mm	coordenada central do robô
Z_0	0	mm	coordenada central do robô

Tabela 2: Tabela de constantes obtidas

Para o protótipo final obtido, obteve-se a ilustração pelo software de *CAD* mostrados nas figuras **1** e **2**, mais adiante serão vistas fotos do robô em funcionamento. Para a concepção da perna, figuras **5** e **6**, houveram diversas modificações de design para as peças e mecanismos do robô, sendo estas guiadas pela metodologia *VDI 2221* de resolução de problemas na criação de um design eficiente, e as demais restrições físicas de torque apresentadas pela equação **3**. O fluxograma apresentado na figura **11** demonstra a busca

pelo design final de acordo com a metodologia mencionada, sendo que cada design teve seus problemas relacionados a torque, geometria e volume disponível para cargas e módulos.

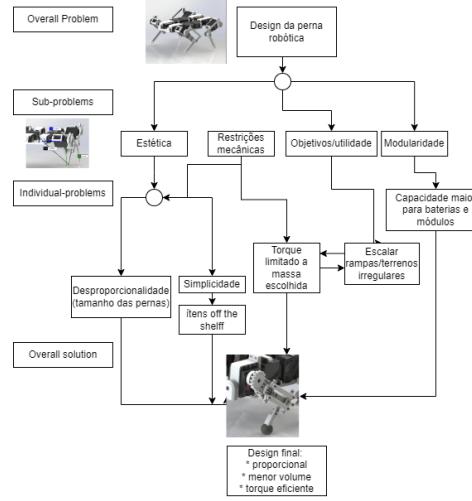


Figura 11: Decisão para solução de design usando o método VDI 2221

O torque medido dos motores escolhidos *MG996R* (procedimento da figura 7) foi de aproximadamente 4kg.cm , o que para uma massa máxima desejada de 1200 g para o robô, valor estimado devido ao peso das baterias com atuadores e demais elementos, fornece um resultado máximo de comprimento de elos de $L = 65\text{mm}$, o que gerou bastante conflito na primeira versão da perna robótica demonstrada na figura 12 em que o comprimento L_1 do robô excede o valor máximo de 65 mm, sendo assim necessária toda uma reinvenção do mecanismo por barras paralelas de transmissão e engrenagens, que retirou o motor servo que acompanhava o movimento da coxa para uma versão em que apenas o quadril move os dois servos responsáveis pela coxa e pata, como exemplificados na figura 13.

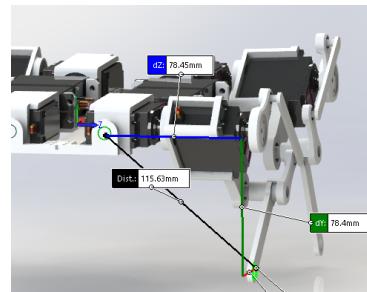


Figura 12: Erro de design perna V1: comprimento L_1 ultrapassa braço de torque máximo permitido

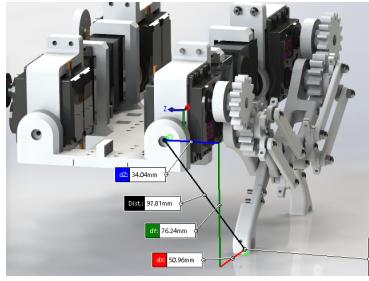


Figura 13: Correcção de design com transmissão por barras paralelas, com menor valor do comprimento L_1

Anterior às barras duplas, usou-se apenas uma única barra para a transmissão do movimento do motor à pata, entretendo após testes empíricos (figura 14), as pernas se dobravam com o próprio peso (próximo de 200g) devido à flexibilidade do material usado na confecção das peças: o termoplástico ABS. A solução obtida ao adicionar outra barra paralela para a transmissão de movimento não só conferiu menos flexibilidade mas também agilidade ao mecanismo.



Figura 14: Teste de resistência de mecanismo de transmissão por uma barra

Com todos os demais testes, os valores finais para os comprimentos dos elos foram os obtidos na **tabela 2**. Após resolver a questão mecânica, foi desenvolvido o controle das juntas. Com os parâmetros de Denavit-Hartenberg que levam à matriz de coordenadas apresentadas em 4 do efetor final $[X_3^0, Y_3^0, Z_3^0]^T$, obteve-se os seguintes gráficos de simulação do espaço de trabalho de cada perna robótica, dada pela figura 15.

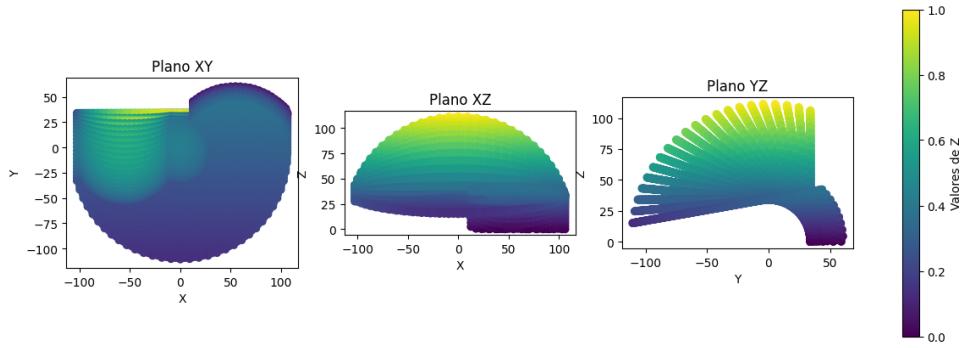


Figura 15: Simulação de espaço de trabalho de uma perna robótica

Após a implementação do software embarcado que utilizou-se da cinemática inversa apresentada pelas equações **6**, **10** e **12**, guiando-se pelas simulações da figura **15**, determinou-se uma área útil para a atuação da perna robótica, em que há uma razoável resposta do efetor final à coordenadas cartesianas, como demonstrado nas figuras **16** e **17**, assim como mostrado anteriormente nas figuras **4** e **5** em que há a realização da curva de *Bézier* com pontos de controle variáveis.

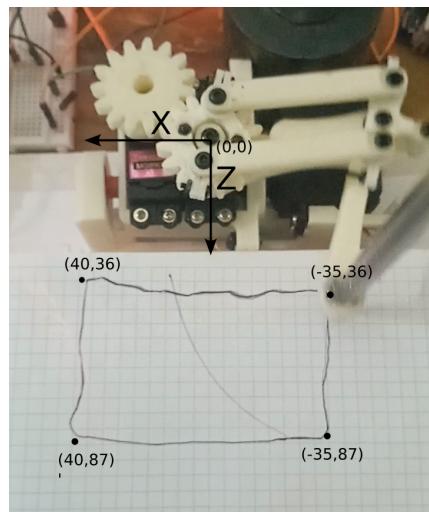


Figura 16: Determinação empírica de espaço de trabalho de uma perna robótica

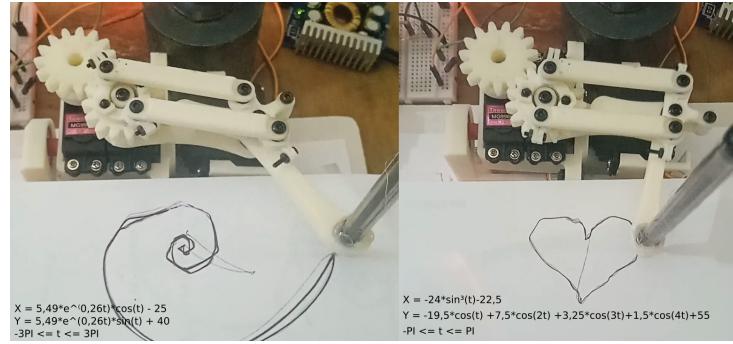


Figura 17: Teste de realização de coordenadas cartesianas pelo efetor final

Juntando todas as pernas em um só corpo, obteve-se o robô completo (**figura 1**). Com todo este conjunto montando, testou-se o funcionamento adequado de robô como um todo. O primeiro teste foi feito com as rotações do centro geométrico X_0, Y_0, Z_0 (**figura 2**) em torno dos três eixos cartesianos, sendo essas os movimentos de *roll*, *yaw* e *pitch*, descritas nas **figuras 18, 19 e 20**.



Figura 18: Movimento de *roll*

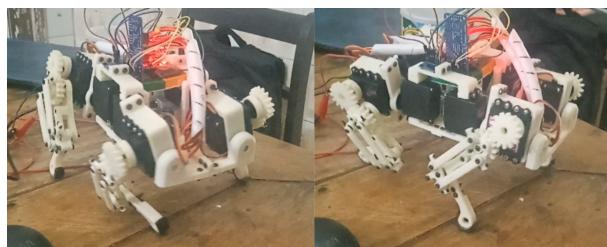


Figura 19: Movimento de *yaw*

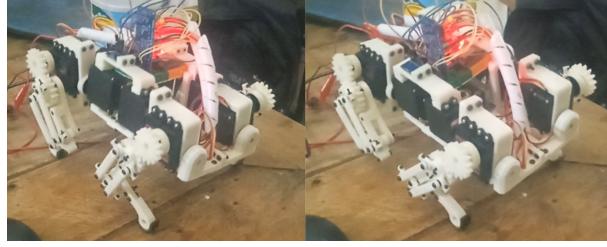


Figura 20: Movimento de *pitch*

Assim como desenvolvido em [9], as matrizes homogêneas responsáveis para a rotação e translação de cada perna para ajustar o centro X_0, Y_0, Z_0 aqui é dado pela **equação 15**, sendo o vetor $[x, y, z]^T$ as coordenadas atuais do efetor final e o vetor $[x', y', z']^T$ as coordenadas transformadas que são a entrada para o cálculo de cinemática inversa.

$$R \cdot T_{centro} \cdot T \cdot [x, y, z]^T = [x', y', z']^T \quad (15)$$

Sendo R o produto das matrizes homogêneas de transformação de *roll*, *yaw* e *pitch*:

$$\mathbf{R}_{roll} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) & 0 \\ 0 & \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{yaw} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{R}_{pitch} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

E T as matrizes de translação e rotação de referencial de cada perna individual em relação ao centro geométrico (referência dada na **figura 2**):

Para a perna na região frontal-direita

$$\mathbf{T}_{FD} = \begin{bmatrix} 1 & 0 & 0 & \frac{C}{2} \\ 0 & 1 & 0 & \frac{L}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para a perna na região frontal-esquerda

$$\mathbf{T}_{FE} = \begin{bmatrix} 1 & 0 & 0 & \frac{C}{2} \\ 0 & -1 & 0 & -\frac{L}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para a perna na região traseira-direita

$$\mathbf{T}_{TD} = \begin{bmatrix} -1 & 0 & 0 & -\frac{C}{2} \\ 0 & 1 & 0 & \frac{L}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para a perna na região traseira-esquerda

$$\mathbf{T}_{TE} = \begin{bmatrix} -1 & 0 & 0 & -\frac{C}{2} \\ 0 & -1 & 0 & -\frac{L}{2} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

E por último T_{centro} a matriz de translação do centro geométrico, igual para todas as pernas:

$$\mathbf{T}_{centro} = \begin{bmatrix} 1 & 0 & 0 & X_0 \\ 0 & 1 & 0 & Y_0 \\ 0 & 0 & 1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Uma demonstração de todas as três rotações pode ser conferida com o código *Rotations-and-translations-DEMO* disponível em [6].

Após os testes de rotações do robô como um todo, foi testado com sucesso os algoritmos de deslocamento: o *trot gait* (**figura 21**) e *bound gait* (**figura 22**).



Figura 21: Algoritmo de *trot gait*

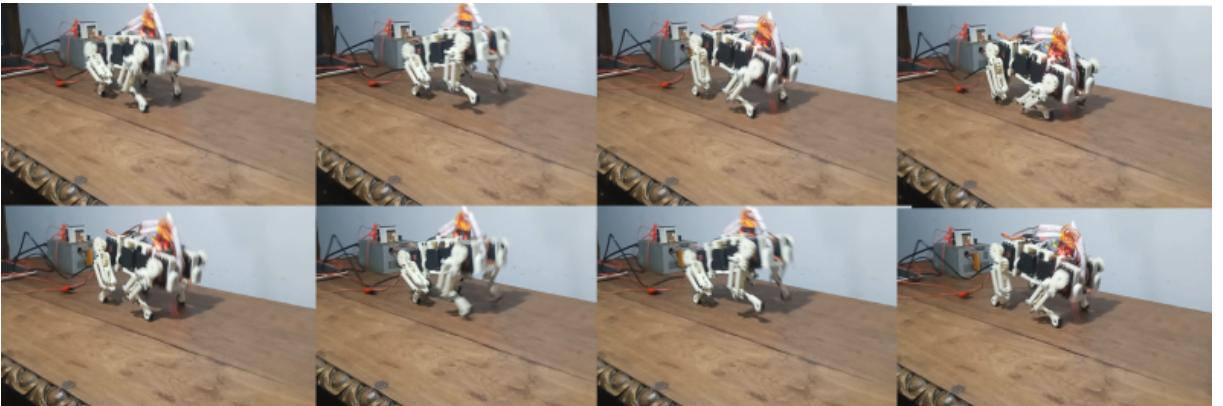


Figura 22: Algoritmo de *bound gait*

Pelo escopo e tempo do projeto, o algoritmo de *crawl gait* não foi implementado com sucesso, pois foi necessário estimar o centro de massa manualmente e encontrar valores de rotação para a sequência do andar, o que é consideravelmente dispendioso com os recursos de tempo disponíveis, se caso deseja-se a implementação bem sucedida deste método. Uma demonstração dos algoritmos de deslocamento pode ser conferida com o código *Gait-testing* disponível em [6].

O foco por fim foi na instabilidade que acompanhou os métodos de deslocamento implementados nas imagens acima. Com o baixo desempenho esperado do torque dos atuadores, a performance do deslocamento do robô foi um pouco lenta se comparado à movimentos naturais de animais quadrúpedes e à robôs quadrúpedes já em uso nas indústrias, além disso esses movimentos sofreram bastante instabilidades, o que motivou a busca da criação de um controle em malha fechada a fim de estabilizar o andar do robô. Primeiramente considerou-se buscar a estabilização estática do robô, sendo o objetivo do controle garantir que o corpo do robô esteja sempre paralelo ao chão, independente da

inclinação. A **figura 23** demonstra o resultado obtido

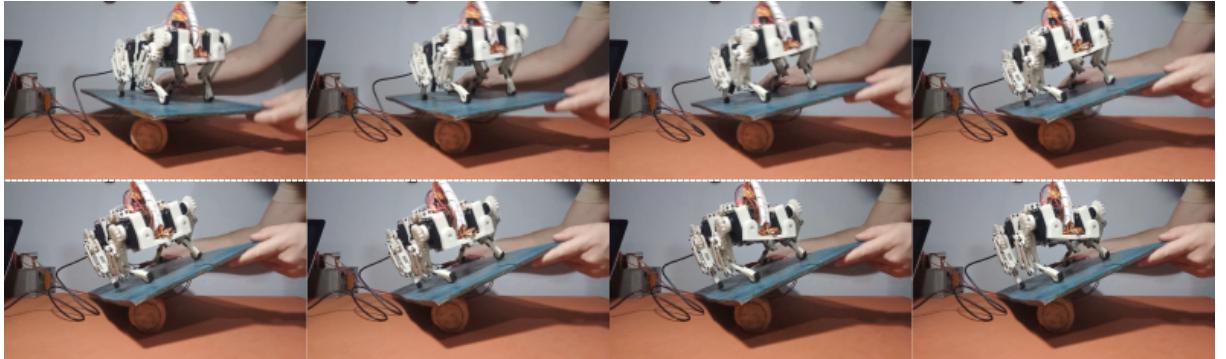


Figura 23: Controle estático de equilíbrio rejeitando mudança na inclinação

Para esse experimento foi desenvolvido um controlador PID de dois graus de liberdade (**figura 24**) em que os valores lidos do acelerômetro *MPU6050* de *roll* e *pitch* (filtrados por um filtro passa baixa) são a entrada do sistema, sendo a saída desta a correção destes ângulos. O controlador apresentou valores aproximados de tempo de subida e acomodação respectivamente de 1 e 2 segundos.

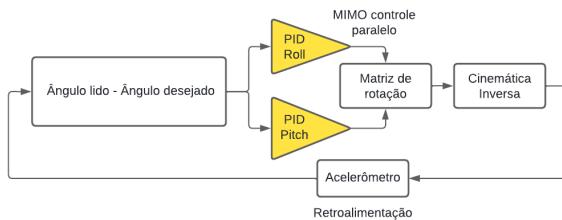


Figura 24: Diagrama de PID paralelo com dois graus de liberdade

Após resolver os diversos detalhes mecânicos e computacionais, para a parte elétrica, foram necessárias uma fonte de 6v com 5A de capacidade durante todo o desenvolvimento e teste do robô. A versão final foi alimentada com um *pack* caseiro de baterias de lítio 4S com 8,3 v de tensão de saída na placa reguladora BMS conectadas a um *stepdown* de até 5A para limitar a tensão para os atuadores. Entretanto, para fins de melhor performance, considerou-se estudar os movimentos com a fonte externa de energia pois o conjunto de baterias adicionou um peso significante para o robô, a ponto de afetar a sua performance. Caso deseja-se o diagrama de conexões elétricas, confira [6].

Por fim, o projeto apresentou um custo total apresentado pela tabela 3 abaixo:

Item	Quantidade	Uso	Valor
Servo MG996R	12x	Atuador	R\$ 550,80
Plástico ABS (1 kg)	2x	Confecção das peças	R\$ 160,00
Roladores	8x	Movimento das juntas robóticas	R\$ 3,00
Módulo PCA9685 I2C	1x	Controle dos atuadores	R\$ 40,76
ESP32	1x	Controlador	R\$ 45,00
Pack de bateria	1x	Alimentação balanceada	R\$ 135,00
Diversos	-	Cabos, parafusos e etc.	R\$ 76,90
Total	-	-	R\$ 1011,46

Tabela 3: Custo Total

6 Conclusão e perspectiva de trabalhos futuros

Como conclusão deste trabalho, foi observado que um robô com pernas ou quadrúpede é uma tecnologia promissora para situações em que rodas não são acessíveis, e diversas iniciativas internacionais tem criado versões amplamente já em uso e é desejável a implementação desta tecnologia em solo nacional.

Pôde-se compreender que para a criação de peças e mecanismos do robô, restrições físicas como comprimentos dos elos são guiadas por equações de torque que lidam com vantagem mecânica para um mecanismo eficiente. Além do mais que para a identificação de problemas ocorrentes no desenvolvimento de mecanismos e peças, usou-se da metodologia *VDI 2221* que auxilia em uma abordagem focada na solução geral de problemas de design.

Observou-se também que a ordenação de um robô quadrúpede segue equações para mapear o movimento das juntas com a ponta da pata, ou o efetor final, sendo essas relações denominadas de cinemáticas direta e inversa, que usam respectivamente do algoritmo de *Denavit-Hartenberg* e do método geométrico para encontrar a solução a estes problemas. Ademais, após esta habilidade do robô em realizar deslocamentos nas coordenadas cartesianas, é possível plotar equações, sendo uma delas fundamentais para a trajetória de cada perna do robô, como a curva de *Bezier* que neste projeto contém 4 pontos de controle para o ajuste desejável do deslocamento da pata robótica.

Após a simulação do espaço de trabalho do robô, pôde-se guiar a determinação empírica da área de trabalho de uma perna robótica com o controle via software embarcado. Portanto, pôde-se obter uma perna robótica para um robô quadrúpede com 3 graus de

liberdade, capaz de receber como entrada coordenadas cartesianas e ajustar automaticamente o ângulo das juntas, possibilitando a plotagem de equações de movimento e futuras aplicações. Todo o material para a reprodução deste robô se encontra em [6].

Sobre a perspectiva de trabalhos futuros, sugere-se motores *brushless* de alto torque e sem limitações angulares [4], diferentes dos motores aqui usados (*MG996R*) que possui uma amplitude máxima de apenas 180 graus, para assim obter uma perna robótica mais robusta e flexível com um maior espaço de trabalho. Além disso, pelo tempo de projeto disponível, não foi possível desenvolver um controle de malha fechada que estabilizasse dinamicamente o andar do robô, esta questão é um estudo a parte sugerido para futuros projetos que possam aproveitar deste aqui proposto. Podem ser analisadas técnicas modernas de controle como equação no estado de espaços ou diversas técnicas abordadas em [4], entretanto é de boa prática lembrar a simplicidade do projeto para alcançar o seu objetivo de ser replicado em escolas e universidades para fins didáticos e de pesquisa.

7 Referências

Referências

- [1] Androtics. How to design a diy quadruped robot. Disponível em: <https://www.youtube.com/watch?v=SU3fmMUz9Zg>. Acesso em: 16 de maio de 2023, 2021.
- [2] Nigel Cross. *Engineering Design Methods: Strategies for Product Design*. John Wiley & Sons, Chichester, 5 edition, 2014.
- [3] EngineerM. Part 1 - how to solve inverse kinematics of a 4 leg robot. Disponível em: <https://www.youtube.com/watch?v=qFE-zuD6jok>. Acesso em: 21 de março de 2024, 2023.
- [4] F. Grimminger, A. Meduri, M. Khadiv, J. Viereck, M. Wüthrich, M. Naveau, V. Berenz, S. Heim, F. Widmaier, T. Flayols, J. Fiene, A. Badri-Spröwitz, and L. Righetti. An open torque-controlled modular robot architecture for legged locomotion research. *IEEE Robotics and Automation Letters*, 5(2):3650–3657, 2020.
- [5] J.Y. Kim. Task based kinematic design of a two dof manipulator with a parallelogram five-bar link mechanism. *science direct*, 2006.
- [6] M.L.A.P.Siqueira. Lui-abc. <https://github.com/MathLou/Lui-ABC>, 2022. Acesso em: 21 mar. 2024.
- [7] Arun K. Natesan. Kinematic analysis and synthesis of four-bar mechanisms for straight line coupler curves, 1994. Tese, Rochester Institute of Technology.
- [8] Robot Mania. Quadruped robot basics. Disponível em: https://www.youtube.com/watch?v=0_2swSMecB4&t=311s. Acesso em: 08 de julho de 2023, 2021.
- [9] Muhammed Arif Sen, Veli Bakircioglu, and Mete Kalyoncu. Inverse kinematic analysis of a quadruped robot. *International Journal of Scientific & Technology Research*, 6(09):285, 2017.
- [10] Xuanqi Zeng, Songyuan Zhang, Hongji Zhang, Xu Li, Haitao Zhou, and Yili Fu. Leg trajectory planning for quadruped robots with high-speed trot gait. *Applied Sciences*, 2019.