

Controle de Caminhada Para Robôs Humanóides

KidSize

1st Dimitria Silveria

Escola de Engenharia Elétrica, Mecânica e de Computação

Universidade Federal de Goiás

Goiânia (GO), Brasil

dimitriasilveria.ds@gmail.com

Resumo—In this work, a walk control was developed for the center of mass of small humanoid robots, with height between 50 cm and 80 cm. A linear quadratic regulator (LQR) was used to control the position of the center of mass. Aiming at computational optimization, a control architecture is proposed in which the LQR controller is used to perform the stabilization of the center of mass and a proportional control with a feedforward term for the trajectory of the swinging feet. Dual quaternion algebra was used for the design of these controllers. The model used to describe the trajectory of the center of mass was the Linear Inverted Pendulum Model (LIPM). The results were validated through simulation with the open-source library Darwin OP3 offered by ROBOTIS. Results show that the developed control was able to stabilize the CoM and foot trajectories even in the presence of disturbances.

Index Terms—humanoid robots, linear quadratic regulator, dual quaternions, linear inverted pendulum model, Darwin OP3.

Resumo—Neste trabalho, foi desenvolvido um controle de caminhada para o centro de massa de robôs humanóides de pequeno porte, com altura entre 50 cm e 80 cm. Foi utilizado um regulador linear quadrático (LQR) para controlar a posição do centro de massa. Visando otimização computacional, é proposto uma arquitetura de controle no qual, utiliza-se o controlador LQR para realizar a estabilização do centro de massa e um controle proporcional com um termo de feedforward para a trajetória dos pés de balanço. A álgebra de quatérnios duais foi utilizada para o projeto destes controladores. O modelo utilizado para descrever a trajetória do centro de massa foi o Modelo do Pêndulo Invertido Linear (LIPM). Os resultados foram validados por meio de simulação com a biblioteca open-source Darwin OP3 oferecida pela ROBOTIS. Resultados mostram que o controle desenvolvido foi capaz de estabilizar as trajetórias do CoM e dos pés mesmo na presença de distúrbios.

Index Terms—robôs humanóides, regulador linear quadrático, duais quatérnios, modelo de pêndulo invertido linear, Darwin OP3.

I. INTRODUÇÃO

O Núcleo de Robótica Pequena Mecânica [1] é um grupo conduzido por alunos da Universidade Federal de Goiás, visando a participação em competições de robótica.

Um dos objetivos da equipe é a participação, no ano de 2023, na liga humanóide da RoboCup (competição mundial de robótica), na sub-liga *kidSize* (robôs com estatura entre 40cm – 80cm de altura) [2]. Para isso, o núcleo vêm desenvolvendo um robô inspirado no modelo Darwin OP3 [3], tornando necessário o desenvolvimento de um controle de caminhada para esse robô.

Muitos trabalhos aplicam o Modelo do Pêndulo Linear Invertido (LIPM) para gerar as trajetórias do centro de massa (CoM) de robôs humanóides, controlando o Ponto de Momento Zero (ZMP) [4]. Além disso, também é muito comum a utilização de métodos como o controle proporcional-integral-derivativo (PID) ou cinemática inversa aliada a matrizes de transformação homogênea [5]. O controle PID é muito eficiente quando o assunto é custo computacional, porém, não é muito robusto [6], principalmente quando comparado ao LQR. As matrizes de transformação homogênea trazem um elevado custo computacional e a cinemática inversa pode produzir singularidades [7], tais como infinitos valores possíveis de ângulos para uma determinada configuração do atuador final, entre outras.

O principal desafio enfrentado pelos humanóides dessa categoria é a limitação do hardware. Para se ter uma ideia, o robô REEM-C, da Pal Robotics com cerca de 1.65 m de altura, possui um computador com um processador Intel Core i7 2710QE x 2, de 4 núcleos e frequência de processamento de 2.1 GHz [8], enquanto que o OP3 possui um Intel Atom N2600 com 2 núcleos de 1.6 GHz [9].

Além das limitações de tamanho, outra regra da categoria determina que os robôs devem ter sensores o mais próximos aos sentidos humanos possíveis (o robô não pode, por exemplo, ter um sensor de ultrassom). Devido a isso, o robô do núcleo conta apenas com duas unidades de medição inercial (IMUs) e uma câmera monocular como sensores. Isso traz a necessidade de algoritmos de visão computacional robustos (que consomem boa parte do poder de processamento disponível), para que a tarefa de jogar futebol seja cumprida de forma eficaz. Aliado a isso, há uma limitação orçamentária por parte do núcleo de robótica, fazendo com que o robô do núcleo utilize uma Raspberry Pi 3B, de 1 GB de memória RAM e processador Cortex-A53 (ARMv8) 64-bit SoC com frequência de 1.4 GHz. Sendo assim, é necessário desenvolver um controle de caminhada com o menor custo computacional possível.

Esse trabalho propõe a utilização do LIPM para o robô Darwin OP3, juntamente com um controle Regulador Linear Quadrático (LQR) para o centro de massa (CoM), agregando robustez, utilizando-se álgebra de quatérnios duais, que gera uma redução do custo computacional [7]. A utilização do Darwin OP3 se deu porque o robô do próprio núcleo (que

ainda está em fase de desenvolvimento no momento do presente trabalho) será muito semelhante a ele, em massa e dimensões.

II. REVISÃO BIBLIOGRÁFICA

Dallali *et al.* [10] comparou o desempenho da estratégia, amplamente empregada, de implementar um controle PID para cada servo, com um sistema centralizado, que utilizava um LQR para controlar todos os servos. Sua conclusão foi a de que o sistema com o LQR era mais robusto para a tarefa de produzir a caminhada de um robô humanóide.

O trabalho apresentado em [11] propõe a utilização de um modelo dinâmico capaz de gerar trajetórias próximas ao caminhar humano, chamado de *3D Dual Spring-Load Inverted Pendulum (Dual-SLIP)*. Tal modelo considera o CoM como um pêndulo invertido com duas hastes. Cada haste possui uma mola. O robô humanóide utilizado foi o HUBO, que mede cerca de 1.20 m de altura e caminha a uma velocidade média de 1 km/h. Foi proposta, em [12], a utilização desse modelo, no mesmo robô, utilizando um controle LQR para o CoM, com álgebra de quatérnios duais, que promove uma redução no esforço computacional, em relação ao uso de matrizes de transformação homogênea [7].

Apesar de ser robusto, o Dual-SLIP possui uma limitação relacionada ao número Froude [13], definido como $Fr = v^2/gl$, em que v é a velocidade de caminhada, g é a aceleração gravitacional e l é o comprimento das pernas. O modelo propõe um processo de otimização, para calcular seus parâmetros, que é capaz de gerar caminhadas similares à humana, quando esse número está dentro do intervalo 0.02 – 0.22 [11], mas, para o Darwin OP3, cuja velocidade máxima é de 25 cm/s e $l \approx 0.24$, seu valor máximo é ≈ 0.0064 .

O tamanho dos robôs da categoria *kidSize* já é, por si só, um fator que traz limitações à capacidade de processamento de tais sistemas. Isso pode ser visto em [14], no qual é proposto um controle de caminhada para o robô TAO-PIE-PIE, da categoria *kidSize*. Nota-se que o principal fator levado em consideração é a redução do custo computacional. Utilizou-se um controle baseado na cinemática inversa. Trajetórias geradas por esse tipo de formalismo, podem apresentar singularidades (tais como infinitas soluções para uma única posição do atuador final, entre outras) [5].

Além disso, o objetivo do presente trabalho é sua futura aplicação no robô humanóide do Núcleo de Robótica Pequim Mecânico, que possuirá uma capacidade de processamento ainda mais limitada, conforme discutido na sessão I. Isso traz a necessidade de um controle de caminhada capaz de aliar robustez a um baixo custo computacional.

Para cumprir tal objetivo, optou-se pela utilização do modelo proposto por Kajita *et al.* [15], o *Linear Inverted Pendulum Model (LIPM)*. Esse modelo considera que a trajetória do CoM se mantém a uma altura constante, $z = z_c$ (vide seção IV). Essa redução de um grau de liberdade traz uma redução no custo computacional, afinal, apenas 2 equações diferenciais serão resolvidas, ao invés de 3. Nesse mesmo artigo, Kajita *et*

al. [15] utiliza o LIPM aliado a um controle para o ponto de momento zero (ZMP) [4].

A escolha realizada no presente trabalho foi a de associar o modelo dinâmico proposto por Kajita *et al.* [15] ao controle LQR, para o CoM utilizando álgebra de duais quatérnios, proposto em [12]

III. O ROBÔ HUMANÓIDE

O objetivo desse trabalho é desenvolver o controle de caminhada para o robô humanóide do Núcleo de Robótica Pequim Mecânico, que ainda está em processo de fabricação. Devido a isso, optou-se pela utilização do modelo Robotis Darwin OP3 [3]. Essa escolha deu-se ao fato de que esse robô cumpre os pré-requisitos da categoria *kidSize* (altura maior de 50 cm e inferior a 80 cm.). Além disso, o humanóide do núcleo está sendo projetado para ter dimensões e massa muito semelhantes ao OP3. Adicionalmente, a Robotis oferece uma biblioteca [16] *open source* que permite a utilização de seu robô em ambientes simulados, permitindo testes em simulação.

A figura 1 mostra as dimensões e massa desse robô, que foram utilizadas nesse trabalho.

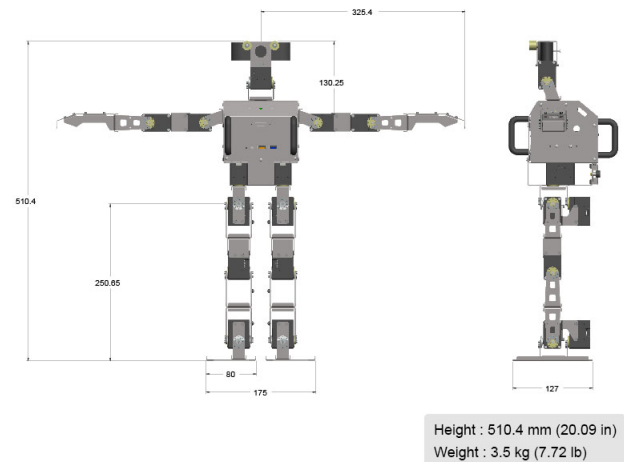


Figura 1. Darwin OP3: Dimensões (em mm) e Massa (em kg). [17]

A figura 2 mostra o robô que está sendo projetado pelos alunos do Pequim Mecânico. É possível notar a semelhança entre ele e o Darwin OP3 em dimensões. A massa do robô ainda não foi estimada, pois o núcleo ainda está decidindo alguns recursos de hardware que utilizará.

IV. MODELO DO PÊNDULO LINEAR INVERTIDO (LIPM)

A. Equações de Movimento

O modelo proposto por Kajita *et al.* [15] divide a caminhada em duas etapas. A primeira (chamada de etapa de suporte único) é o momento em que o robô humanóide está apoiado apenas sobre uma perna, enquanto a outra se desloca. A segunda etapa (suporte duplo) corresponde ao momento em que o robô se apoia sobre as duas pernas. Além disso, segundo ele, a dinâmica desse tipo de robô pode ser descrita de forma semelhante a de um pêndulo invertido que conecta a perna

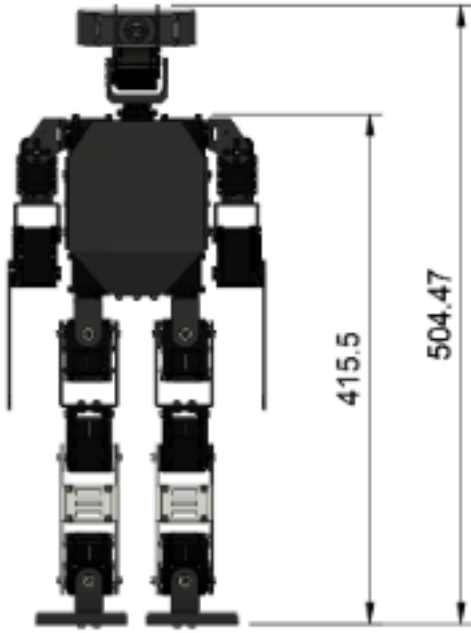


Figura 2. Robô Humanoide do Pequi Mecânico: Dimensões (em mm). [17]

apoiada ao chão ao centro de massa (CoM) do corpo do robô, conforme mostrado pela figura 3. A massa total é representada por m e r é a distância entre o pé de apoio e o CoM.

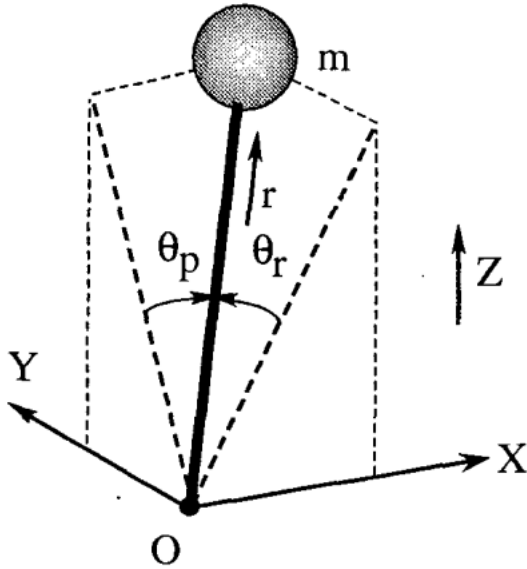


Figura 3. Pêndulo invertido 3D. [15]

As equações do pêndulo invertido, em coordenadas cartesianas, são dadas por:

$$m(-z\ddot{y} + y\ddot{z}) = \frac{D}{C_r} \tau_r - mgy, \quad (1)$$

$$m(z\ddot{x} + x\ddot{z}) = \frac{D}{C_p} \tau_p - mgx, \quad (2)$$

onde $C_r = \cos \theta_r$, $C_p = \cos \theta_p$, $D = \sqrt{C_r^2 + C_p^2 - 1}$ e g é a aceleração da gravidade. Além disso, (τ_r, τ_p, r) são as forças e torques associadas às variáveis de estado (θ_r, θ_p, r)

Para utilizar as equações do pêndulo na representação de uma caminhada bípede é necessário restringir o movimento no eixo z [4] a um plano, cuja equação é dada por:

$$z = k_x x + k_y y + z_c \quad (3)$$

Além disso, para a caminhada em um plano sem inclinação, que é o caso do presente trabalho, $k_x = k_y = 0$. Como nosso robô não está sujeito a nenhuma força externa, a não ser a gravidade, também temos que $(\tau_r = \tau_p = 0)$. Desta forma, as equações que descrevem a caminhada do nosso bípede são:

$$\ddot{y} = \frac{g}{z_c} y \quad (4)$$

$$\ddot{x} = \frac{g}{z_c} x \quad (5)$$

B. Trajetórias Geradas

As equações 12 e 11 geram duas trajetórias completamente independentes nos eixos x e y . Tais trajetórias e suas respectivas velocidades estão ilustradas na figura 4, para um tempo de suporte único $T_s = 0,1s$, tempo de suporte duplo $T_{dbl} = 0.04s$ de suporte duplo e $z_c = 0.208m$. A cada fase, o referencial dos eixos é colocado sobre o pé que está em contato no chão, ou seja, as trajetórias são chegadas localmente.

Os instantes de tempo em que as velocidades permanecem constantes em ambos os eixos, simultaneamente, correspondem às fases de suporte duplo [4]. Tais valores de velocidade são controladas pelo posicionamento do pé de balanço, em relação ao centro de massa, no início do suporte duplo, conforme será explicado com mais detalhes na seção IV-C.

A figura 5 mostra essa mesma trajetória, no plano xy , incluindo as posições dos pés a cada passo. Os pés estão enumerados de 0 a 10, sendo 0, as posições iniciais destes. O robô inicia a trajetória com a perna esquerda e finaliza com a perna direita. Os dois pés iniciam e terminam a caminhada alinhados, no eixo y . No último passo, a perna direita se desloca para trás, alinhando-se com a esquerda. Nota-se que essa curva não corresponde a uma caminhada em linha reta. Isso acontece porque, conforme proposto por [15], para determinar a direção da trajetória, é necessário implementar um controle para a direção do pé de balanço no final da fase de suporte único. Essa implementação não ocorreu porque não houve tempo o suficiente para isso, durante o desenvolvimento do presente trabalho.

As curvas mostradas nas figuras 4, 5 possuem formato semelhante àquelas que Kajita *et al.* [15] obteve para um robô de $1.50m$ de altura.

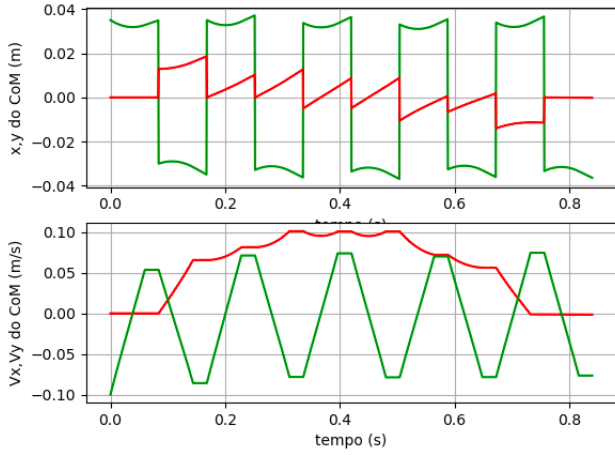


Figura 4. Posição x,y e velocidade do CoM, respectivamente, em função do tempo, para uma trajetória com 10 passos. As curvas vermelhas representam os valores para x e em verde, y. Os gráficos de posição "saltam" a distância do comprimento do passo em cada troca de pé de apoio, pois o pé que está em contato com o chão é utilizado como referência.

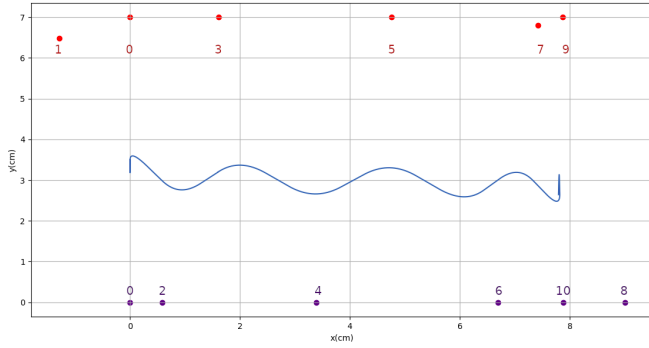


Figura 5. Trajetória 2D do centro de massa do robô, no plano xy, para 10 passos. A referência está situada na posição da perna direita no início da caminhada. Os círculos roxos representam as posições da perna direita a cada passo e os círculos vermelhos representam a perna direita.

C. Controle de Velocidade

Para modificar a velocidade do CoM em ambas as direções foi utilizado um controle proposto por [4] e [15], que determina a posição que o pé de balanço deve assumir, em relação ao CoM, no final de cada passo, conforme mostra a figura 6. A lei de controle é dada pela equação:

$$x_i^{(2)} = (aC_T(x_d - S_T T_c v_f^1) + b \frac{S_T}{T_c} (v_d - C_T v_f^1)) / D_T^1, \quad (6)$$

onde $C_T = \cosh \frac{T_s}{T_c}$, $S_T = \sinh \frac{T_s}{T_c}$, $T_c = \sqrt{\frac{z_c}{g}}$ e

$D_T = aC_T^2 + b(\frac{S_T}{T_c})^2$. Além disso, os ganhos do controle são $a, b > 0$. (x_d) e (v_d) são, respectivamente, a posição e

¹Essa equação está diferente da equação número (17) apresentada na referência [15]. Ao refazer os passos utilizado pelo autor, para deduzí-la, descobriu-se que era necessário fazer uma pequena correção, trocando o valor de v_d , no primeiro termo do lado direito da igualdade, por v_f^1

velocidade desejadas do CoM no final do i -ésimo passo, $v_f^{(1)}$ é a velocidade real e $x_i^{(2)}$ é a posição que o pé de balanço deve assumir, em relação ao CoM, ao final da fase de suporte único, para que a configuração desejada seja alcançada.

No presente trabalho, esse controle foi aplicado apenas na coordenada x . Os valores de a e b , para os 10 passos ilustrados nas figuras 5 e 4 estão ilustrados na tabela I. Pode-se observar que não há valores de a e b para o último passo, porque o controle ajusta a posição e velocidades desejadas no passo seguinte. Como não há passo seguinte, não há a necessidade de realizar os cálculos do controle.

Tabela I
GANHOS DO CONTROLE DE VELOCIDADE PARA OS 10 PASSOS

passo	1	2	3	4	5	6	7	8	9
a	30	36.5	20	2	0.001	2000	500	4200	200
b	1	5	50	3	5	0	0.01	0	0

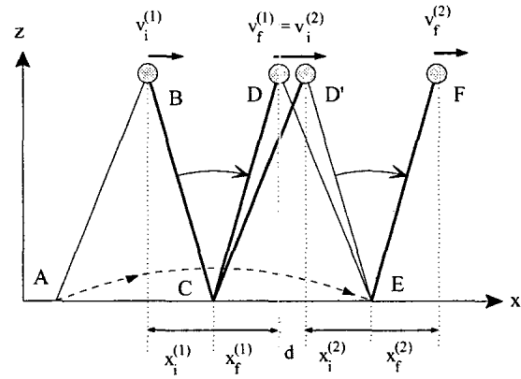


Figura 6. Dois passos consecutivos, no plano xz. O robô vai do ponto B a D na fase de suporte único, enquanto que o pé de balanço se move de A para E. Depois, vai de D até D' durante o suporte duplo, com velocidade $v_f^{(1)}$ (constante). A velocidade final do CoM ($v_f^{(2)}$), no passo seguinte, em F, pode ser controlada mudando a posição de E [15], [4]

V. CONTROLE DE CAMINHADA

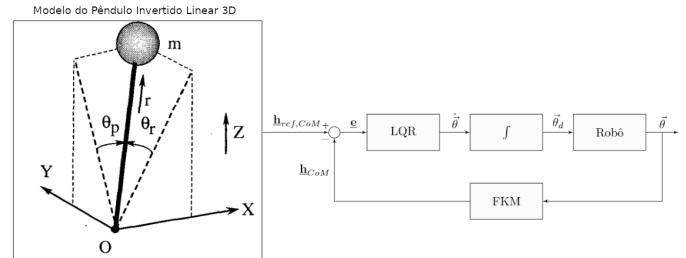


Figura 7. Estrutura do trabalho para o CoM

A figura 7 mostra um diagrama de blocos representando como os ângulos das juntas do robô são calculados. Nesse diagrama, $\underline{h}_{ref,CoM}$ e $\underline{h}_{ref,f}$ são, respectivamente, as posições e orientações de referência do CoM e dos pés e \underline{h}_{CoM} \underline{h}_f

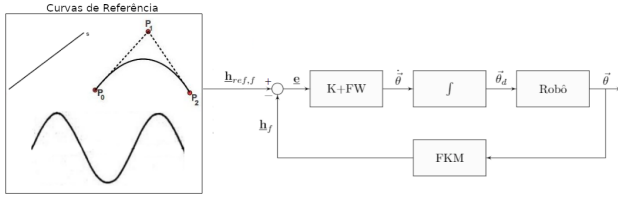


Figura 8. Estrutura do trabalho para os pés

são, respectivamente, as posições e orientações calculadas por cinemática direta (FKM), utilizando-se os ângulos dados pelos controles. O LQR recebe, como entrada, o erro entre as posições e orientações de referência (dadas pelo LIPM) e aquelas calculadas pela cinemática direta, do CoM. A partir disso, ele retorna as velocidades das juntas, da perna de suporte, que são integrados, para obter-se os ângulos que devem ser enviados ao robô. A figura 8 mostra que o controlador proporcional com *feedforward* (representado por K+FW) faz um processo semelhante, porém, sua entrada é o erro entre as posições do pé de balanço. Com isso, calcula-se as variáveis de junta para a perna que está se movimentando. As posições de referência são calculadas utilizando-se curvas de Bézier de segunda ordem, retas e senóides, conforme será comentado com mais detalhes na seção VI.

A. LQR para o CoM

Conforme mostrado por [12], é possível implementar um controle LQR (regulador linear quadrático), utilizando álgebra de quatérnios duais, para controlar a trajetória do CoM de um robô humanoíde, durante sua caminhada.

Esse controle foi escolhido no presente artigo pois oferece uma série de vantagens [18]:

- Por ser um controle de malha fechada que provê ganhos de realimentação que otimizam a trajetória do sistema [19], tem robustez em relação a ruídos e distúrbios [10].
- Baixo custo computacional ao utilizar álgebra de quatérnios, ao invés de matrizes de transformação homogênea, segundo [7]. Isso ocorre porque a representação matricial exige a utilização de 16 elementos para representar as poses de cada junta, enquanto que a representação por quatérnios duais exige apenas 8. Isso promove uma redução na quantidade de memória RAM e no tempo necessários para processar o código.
- O controlador é projetado no espaço de tarefas [20], ao invés das variáveis de juntas, simplificando sua implementação e evitando singularidades, se comparado ao uso da da cinemática inversa [5].
- A otimização da velocidade evita flutuações e sobressinais que podem ser causados por altas acelerações imprimidas aos motores pelos controles de baixo nível. [10]

1) *Formulação Matemática do LQR*: O problema do LQR proposto por [18] consiste em encontrar um controlador ideal

para o sistema²³:

$$\dot{\vec{e}} = \mathbf{A}\vec{e}(t) + \vec{u}(t) + \vec{c}, \quad (7)$$

em que $\vec{c} = -vec_8(\underline{h}_d^* \dot{\underline{h}}_d)$ é o produto do conjugado da posição desejada do CoM pela derivada da posição calculada por cinemática direta e vec_8 é um operador que transforma esse produto (originalmente representado em quatérnios duais) em um vetor de 8 posições. $\mathbf{A} = \bar{H}_8(\underline{h}_d^* \dot{\underline{h}}_d)$, onde \bar{H}_8 é o operador Hamiltoniano, que mapeia o dual quaternion em uma forma matricial e \underline{h}_d é a posição desejada do CoM e $\dot{\underline{h}}_d$ é sua derivada. Além disso, \vec{e} é o erro entre a trajetória calculada pela cinemática direta e a desejada, $u(t) = -\mathbf{N}\dot{\theta}$. $\mathbf{N} = \bar{H}_8(\underline{h}_d C_8 J)$, em que $C_8 = diag(1, -1, -1, -1, 1, -1, -1, -1)$, $J = \frac{\partial \underline{h}}{\partial \theta}$ é a matriz jacobiana do sistema e θ são os ângulos da perna de apoio [12].

Para isso, precisamos encontrar $u(t)$ que minimize a função custo:

$$\mathbf{F} = \frac{1}{2} \vec{e}(t_f)^T \mathbf{S} \vec{e}(t_f) + \frac{1}{2} \int_0^{t_f} (\vec{e}^T \mathbf{Q} \vec{e} + \vec{u}^T \mathbf{R} \vec{u}) dt, \quad (8)$$

onde as matrizes $\mathbf{S} \geq 0$, $\mathbf{Q} \geq 0$ e $\mathbf{R} > 0$ (que são determinadas pelo projetista) representam, respectivamente, o peso do erro ao final da trajetória, o custo do erro ao longo da trajetória e o ganho do controlador.

A minimização da equação 8 resulta em $\vec{u}(t) = -\mathbf{R}^{-1}(\mathbf{P}\vec{e} + \dot{\xi})$. Onde \mathbf{P} é um ganho proporcional, variável no tempo e $\dot{\xi}$ é um termo de avanço ponderado [12], [7]. Ambos são obtidos resolvendo as equações de Ricatti:

$$\dot{\mathbf{P}} = -\mathbf{P}\mathbf{A} - \mathbf{A}^T \mathbf{P} + \mathbf{P}\mathbf{R}^{-1} \mathbf{P} - \mathbf{Q} \quad (9)$$

$$\dot{\xi} = -\mathbf{A}^T \xi + \mathbf{P}\mathbf{R}^{-1} \xi - \mathbf{P}\vec{c} \quad (10)$$

Desta forma, as velocidades das juntas são dadas por:

$$\dot{\theta} = \mathbf{N}^\dagger \mathbf{R}^{-1} (\mathbf{P}\vec{e} + \dot{\xi}) \quad (11)$$

, em que \mathbf{N}^\dagger é a pseudo-inversa da matriz \mathbf{N}

A figura 9 mostra o algoritmo utilizado para obter os ângulos da perna de apoio por meio do controlador LQR.

B. Controle Proporcional com Feedforward para os Pés

Apesar da álgebra de quatérnios oferecer uma redução do custo computacional do LQR, esse controle ainda possui um custo elevado se comparado a um controle proporcional. Isso acontece porque os cálculos dos ganhos do LQR são feitos resolvendo equações de Ricatti. Desta forma, para reduzir a necessidade de poder de processamento, [12] propôs a utilização de um controle proporcional com *feedforward*, para as trajetórias dos pés. Essa escolha se deu porque esse controle, apesar de menos robusto que o LQR, ainda evita as singularidades da cinemática inversa. Conforme será mostrado

²As letras maiúsculas, em negrito, nas equações, são a notação escolhida para denotar matrizes

³o traço embaixo da letra é a notação que representa números quatérnios duais

- 1: Definir \mathbf{R}
- 2: Definir \mathbf{Q}
- 3: Definir \mathbf{S}
- 4: $\underline{\mathbf{h}} \leftarrow \underline{\mathbf{h}}(\theta)$
- 5: $\underline{\mathbf{h}}_d \leftarrow \underline{\mathbf{h}}_d(t)$
- 6: **while** $t < t_f$ **do**
- 7: Calcular \mathbf{A}
- 8: Calcular \mathbf{c}
- 9: Calcular \mathbf{J}
- 10: Calcular \mathbf{N}
- 11: Calcular \mathbf{N}^\dagger
- 12: Calcular \mathbf{e}
- 13: $\dot{\underline{\theta}} \leftarrow \mathbf{N}^\dagger \mathbf{R}^{-1} (\mathbf{P} \mathbf{e} + \xi)$
- 14: $\underline{\theta} \leftarrow \underline{\theta} \Delta t$
- 15: $\underline{\mathbf{h}} \leftarrow \underline{\mathbf{h}}(\underline{\theta})$
- 16: **end while**

Figura 9. Algoritmo do controle LQR [12]

na seção VI, ele mostrou eficiência suficiente em fazer com que os pés seguissem as trajetórias desejadas.

Desta forma, os ângulos do pé de balanço são calculados pela seguinte equação:

$$\dot{\underline{\theta}} = -\mathbf{N}^\dagger (\mathbf{K} \underline{\mathbf{e}} - \text{vec}_8(\underline{\mathbf{h}}^* \dot{\underline{\mathbf{h}}}_d)), \quad (12)$$

na qual $\mathbf{K} = k\mathbf{I}$ é o ganho do controlador (determinado pelo projetista do sistema), $\underline{\mathbf{h}}^*$ é o dual quatérnio conjugado da posição real do pé de balanço e $\dot{\underline{\mathbf{h}}}_d$ é a derivada da posição de referência do pé de balanço.

A figura 10 mostra o algoritmo utilizado para obter os ângulos da perna de balanço por meio do controlador proporcional com *feedforward*.

VI. RESULTADOS

Os resultados apresentados nessa seção são referentes ao algoritmo testado numa situação ideal, isto é, comparando-se apenas as trajetórias geradas pelo modelo dinâmico com aquelas calculadas pela cinemática direta utilizando-se os ângulos dos controles. A tabela II mostra os valores de ganhos utilizados em cada controle, em todos os passos da trajetória, para obter estes resultados

A. Centro de Massa

As figuras 11, 12, 13 e ?? mostram, respectivamente, a trajetória do CoM no eixo x , y , z e ângulo em quatérnio, do

- 1: Calcular $\underline{\mathbf{h}}_d(t)$
- 2: **while** $t < t_f$ **do**
- 3: Calcular \mathbf{J}
- 4: Calcular \mathbf{N}
- 5: Calcular \mathbf{N}^\dagger
- 6: Calcular \mathbf{e}
- 7: $\dot{\underline{\theta}} \leftarrow \mathbf{N}^\dagger (\mathbf{K} \mathbf{e})$
- 8: $\underline{\theta} \leftarrow \underline{\theta} \Delta t$
- 9: $\underline{\mathbf{h}} \leftarrow \underline{\mathbf{h}}(\underline{\theta})$
- 10: **end while**

Figura 10. Algoritmo do controle proporcional com *feedforward* [12]

Tabela II
GANHOS DOS CONTROLES PARA TODOS OS PASSOS

S	Q	R	K
0	500	0.001	900

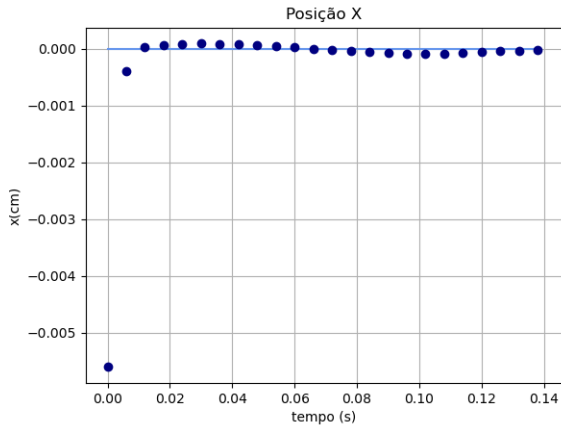
centro de massa, para os três primeiros passos da caminhada. No instante inicial é possível ver, nas figuras 11-a) e 13-a), há uma pequena diferença entre essas duas trajetórias, nos eixos x e z . Isso acontece porque os ângulos iniciais das juntas foram calculados manualmente, por cinemática inversa, considerando-se que os dois elos das pernas teriam o mesmo comprimento (situação não condizente com as verdadeiras medidas). Essa aproximação gerou um pequeno erro, da ordem de 10^{-4} em z e 10^{-5} em x . Tais erros não prejudicam o desempenho do sistema, pois são menores que a precisão dos servomotores (Dynamixel Mx-28 [21]). Ainda nas figuras 11 e 12, pode-se perceber que a trajetórias para os demais passos, gerados pelo LQR, coincidem com os valores desejados.

Nas figuras 12, 14, 15 e 16 é possível ver que o CoM segue a trajetória desejada para os três primeiros passos. Além disso, as figura 14, 15 e 16 mostram que o quadril não rotaciona durante a trajetória. Isso ocorre por causa da restrição de z constante, conforme será comentado com mais detalhes na seção VII

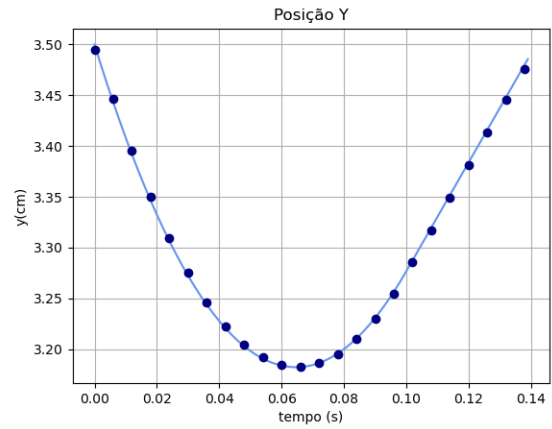
As figuras 17 mostra as trajetórias de referência e gerada pelo controle no plano xy e a figura 18 mostra as mesmas trajetórias no eixo z , em função do tempo. Percebe-se que ambas as trajetórias apresentaram valores muito próximos em toda a caminhada, mostrando que o LQR é eficaz o tempo todo.

B. Pé de Balanço

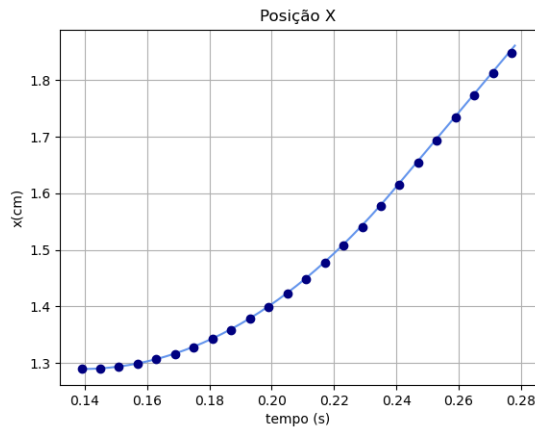
A caminhada inicia-se com um pequeno passo para trás, com o pé esquerdo. Já no segundo passo, o robô avança com o pé direito sendo o pé de balanço. Consequentemente, no



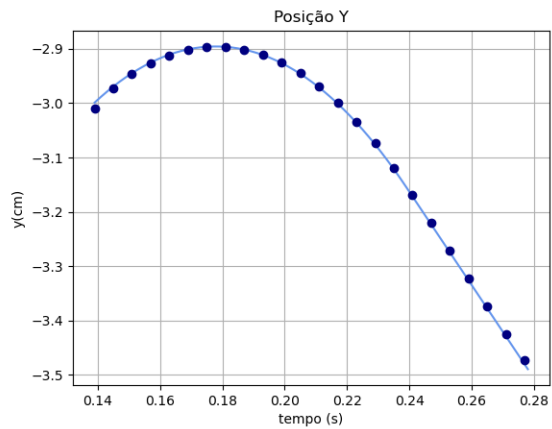
(a) X primeiro passo



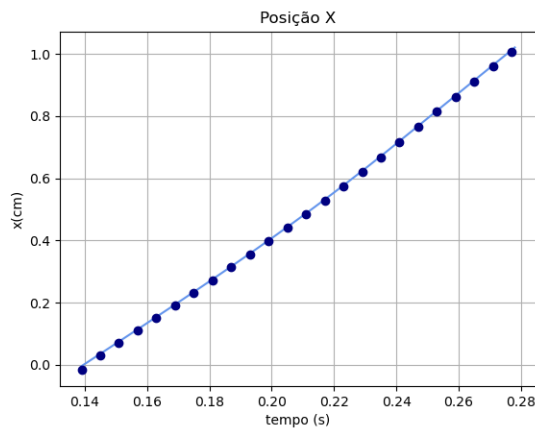
(a) Y primeiro passo



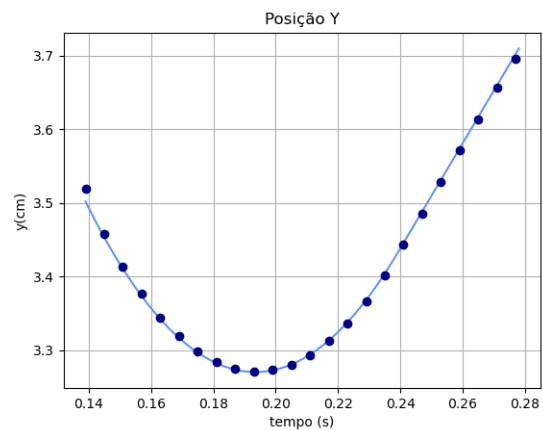
(b) X segundo passo



(b) Y segundo passo



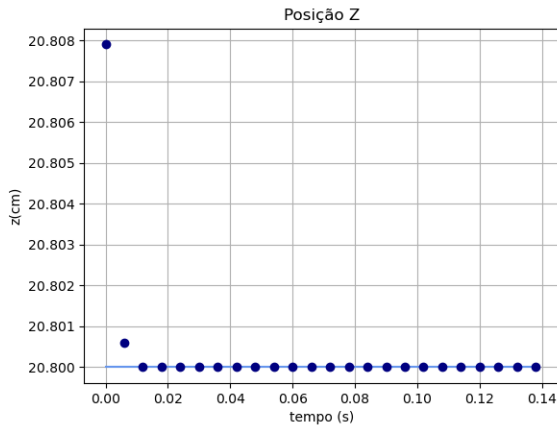
(c) X terceiro passo



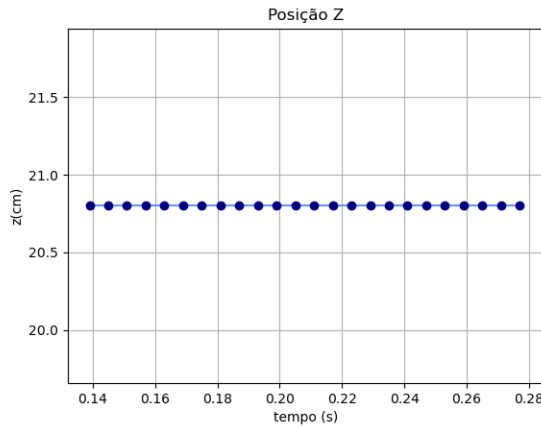
(c) Y terceiro passo

Figura 11. Trajetória do centro de massa no eixos x , em função do tempo, nos 3 primeiros passos da trajetória. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao passo inicial da caminhada

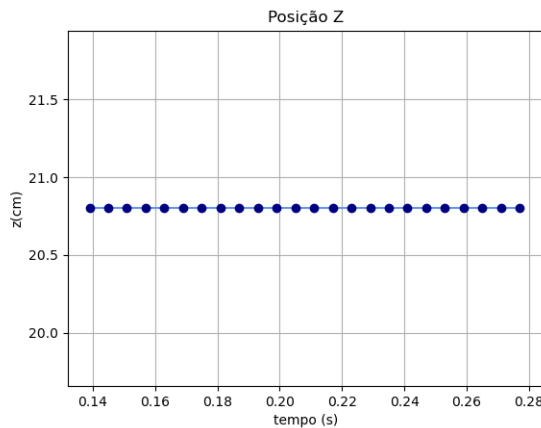
Figura 12. Trajetória do centro de massa no eixos y , em função do tempo, nos 3 primeiros passos da trajetória. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao passo inicial da caminhada



(a) Z primeiro passo



(b) Z segundo passo



(c) Z terceiro passo

Figura 13. Trajetória do centro de massa no eixo z , em função do tempo, nos 3 primeiros passos da trajetória. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos ângulos obtidos pelo controle, referentes ao passo inicial da caminhada

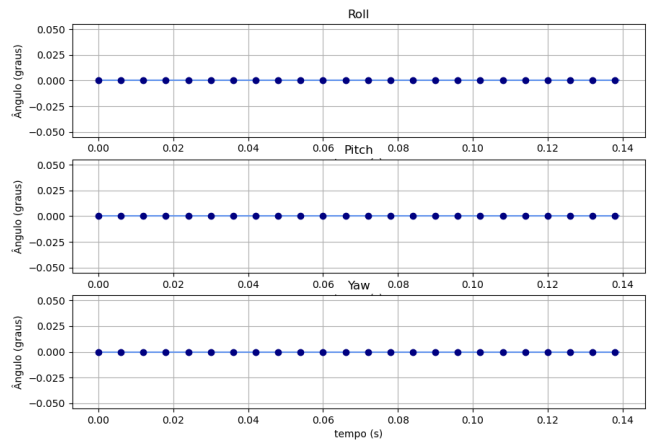


Figura 14. Trajetória dos ângulos de Euler do CoM em função do tempo, para o primeiro passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

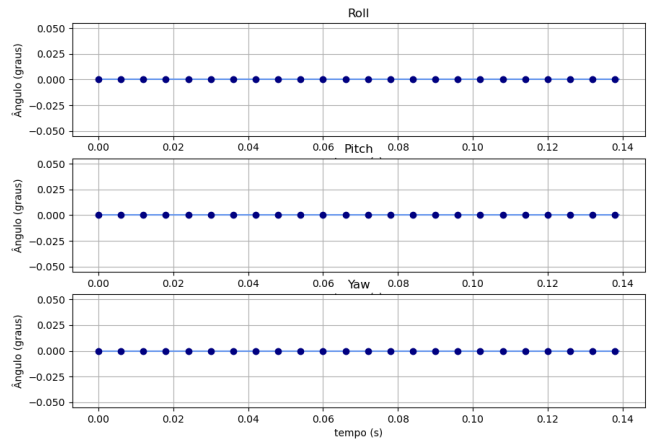


Figura 15. Trajetória dos ângulos de Euler do CoM em função do tempo, para o segundo passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

terceiro passo, o pé esquerdo avança. O padrão então se repete, até o final da caminhada.

Os valores de referência, do pé de balanço, foram geradas utilizando-se três curvas: curvas de Bézier de segunda ordem [22], no eixo z , retas no eixo x e nos dois primeiros passos do eixo y e uma senóide, no eixo y , nos demais passos. A necessidade de usar a senóide em alguns passos, ao invés de reta, surgiu porque o controle proporcional não convergia para uma trajetória constante com $y = 7$ cm (o controle só é capaz de gerar uma trajetória para retas cuja inclinação é diferente de 0). Sendo assim, optou-se por utilizar uma senóide de pequena amplitude de 0.7 cm, em tais casos. As trajetórias nos três eixos e o ângulo, do pé de balanço, calculadas por cinemática direta, com os ângulos do controlador proporcional com *feedforward* apresentaram valores muito próximos das curvas de referência. Isso pode ser visto nas figuras 19, 20,

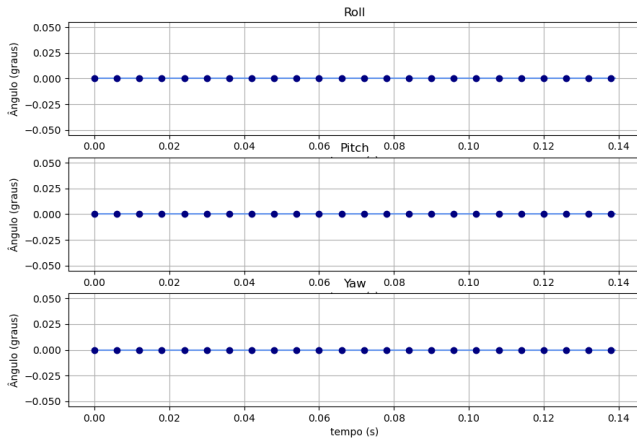


Figura 16. Trajetória dos ângulos de Euler do CoM em função do tempo, para o terceiro passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

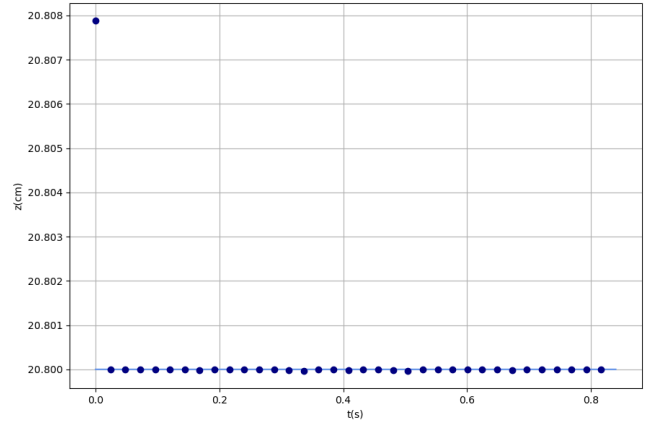


Figura 18. Trajetória de referência (curva contínua) e trajetória gerada pelo controle (círculos) no eixo z em função do tempo para 10 passos

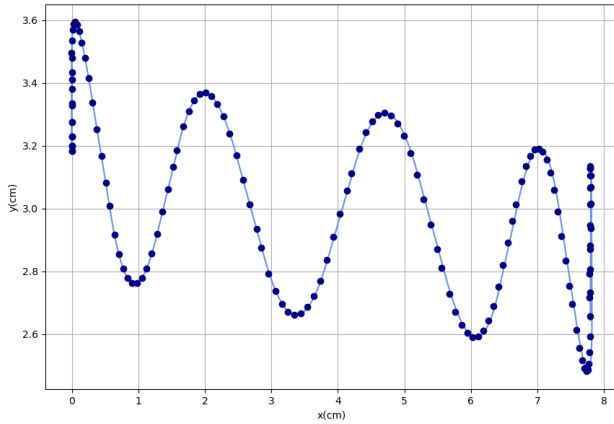


Figura 17. Trajetória de referência (curva contínua) e trajetória gerada pelo controle (círculos) no plano xy para 10 passos

21, 22, 23 e 24. Adicionalmente, as figura 22, 23 e 24 mostram ângulos de Euler nulos, ou seja, o pé de balanço não apresentou nenhuma rotação durante suas trajetórias. Isso ocorreu porque o controle da direção do pé de balanço não foi implementado.

As figuras 25 e 27 exibe, a trajetória dos pés direito e esquerdo, respectivamente durante os 10 passos, no plano xy . Percebe-se que o controle foi capaz de seguir a referência durante todos os passos. Além disso, nota-se que o último passo, do pé direito e o primeiro passo, do pé esquerdo, correspondem a recuos, conforme fora comentado na seção IV. Em 26 e 28 é possível ver a trajetória dos pés direito e esquerdo, respectivamente, no eixo z em função do tempo. Percebe-se que o controle também teve um bom desempenho nesse eixo, para ambos os pés.

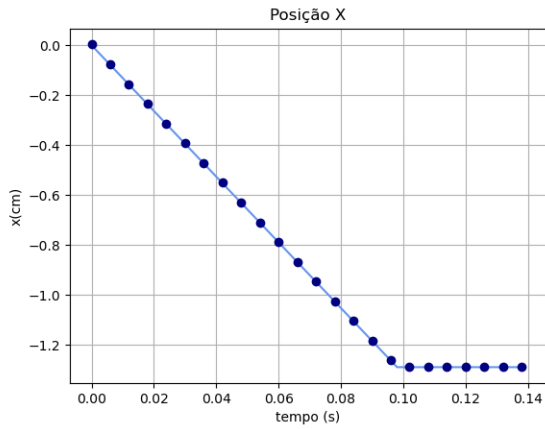
VII. SIMULAÇÃO

A figura 29 mostra o modelo disponibilizado pela ROBOTIS [16] para o simulador Gazebo [23], que foi utilizado nesse trabalho.

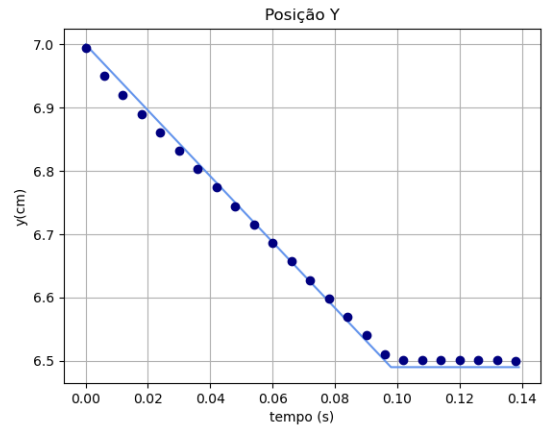
Algumas dificuldades foram encontradas no uso do simulação. A primeira delas era o fato de que os ganhos do controle dos servos (controles PID), não estava propriamente ajustados para o controle proposto nesse trabalho, fazendo com que a resposta dos motores ao comando para executar os ângulos adicionasse um pouco de ruído à trajetória. Isso pode ser visto na figura 30, na qual é possível observar vários sobressinais e pequenas oscilações nos valores dos ângulos do joelho da perna esquerda, durante a simulação. Além disso, o modelo da ROBOTIS permite apenas que os comandos dos ângulos sejam enviados apenas a um motor de cada vez. Desta maneira, enquanto o servo responsável pelo movimento de yaw to quadril, por exemplo, estava recebendo o novo valor de ângulo, os demais ainda estavam executando os valores antigos. Essa característica também trouxe ruídos para a trajetória. Esses ruídos podem ser vistos na figura 31, que mostra a trajetória do CoM do OP3, durante uma trajetória com 14 passos.

Para contornar esses problemas, a frequência de comandos do controle, que deveria ter sido de 100 Hz, conforme proposto por [15], foi reduzida para 40 Hz. Inicialmente, foi utilizado 100Hz, mas percebeu-se que a cada comando recebido, o robô balançava e, antes que o balanço parasse, outro comando era recebido, gerando cada vez mais balanço, provocando queda. Após isso, foram feitos vários testes, reduzindo-se a frequência de 10 em 10Hz, até chegar a 40Hz (com esse valor, o robô tinha tempo suficiente para se recuperar do balanço antes de executar o próximo comando). Desta forma, o equilíbrio foi mantido e a caminhada foi executada com êxito.

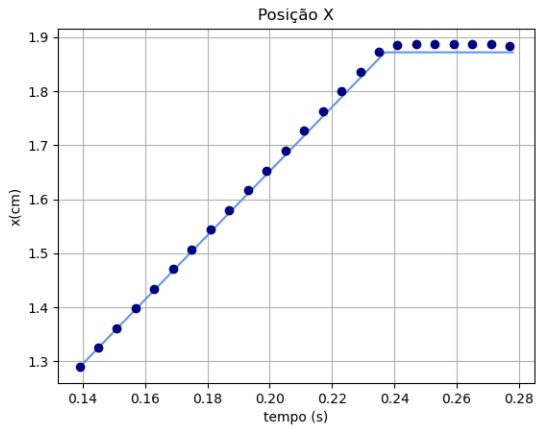
A figura 32 mostra as posições do *base_link*, que corresponde a um referencial cartesiano 3D, alinhado com o CoM no plano xy e cerca de 3 cm acima, no eixo z . É possível observar que a posição em z praticamente não varia ao longo



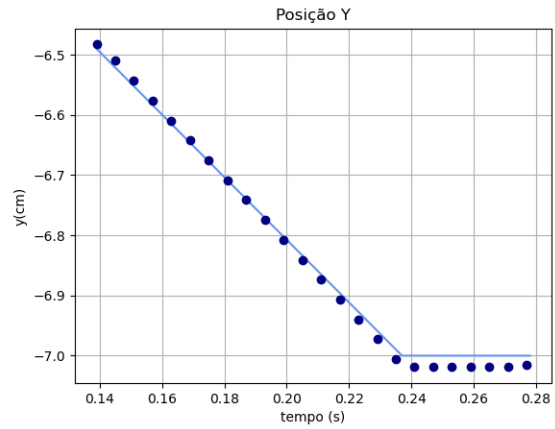
(a) X primeiro passo



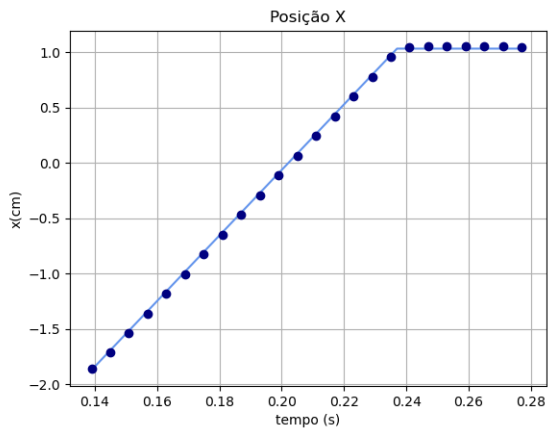
(a) Y primeiro passo



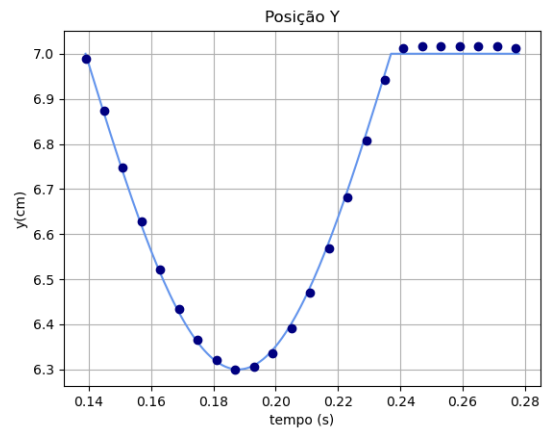
(b) X segundo passo



(b) Y segundo passo



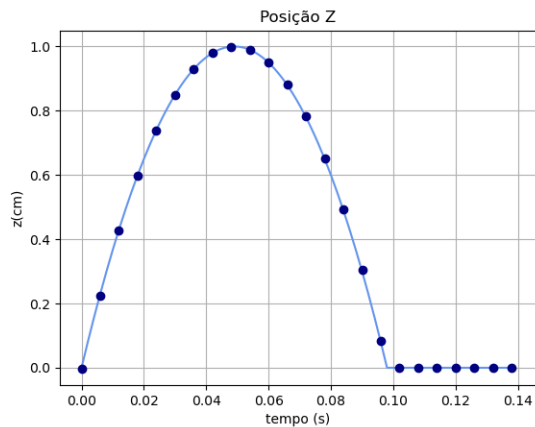
(c) X terceiro passo



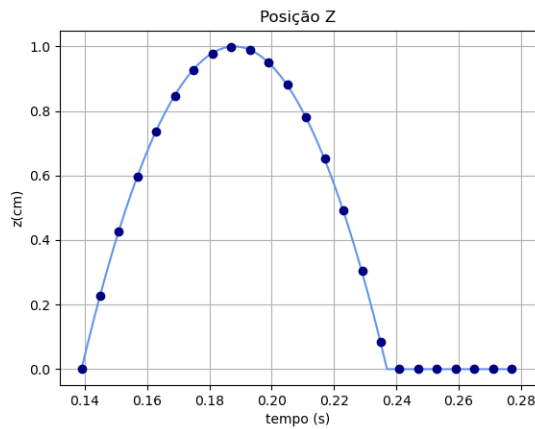
(c) Y terceiro passo

Figura 19. Trajetória do pé de balanço no eixo x em função do tempo, para os 3 primeiros passos. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

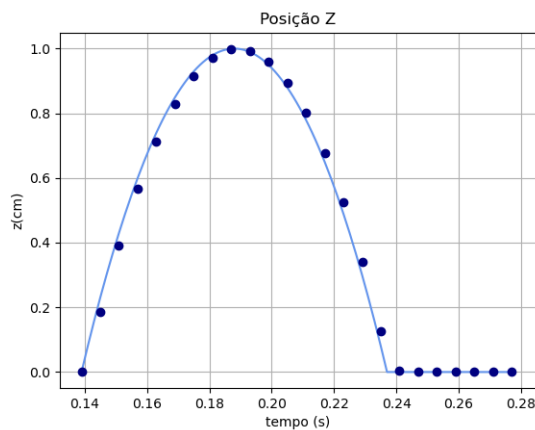
Figura 20. Trajetória do pé de balanço no eixo y em função do tempo, para os 3 primeiros passos. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada



(a) Z primeiro passo



(b) Z segundo passo



(c) Z terceiro passo

Figura 21. Trajetória do pé de balanço no eixo z em função do tempo, para os 3 primeiros passos. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

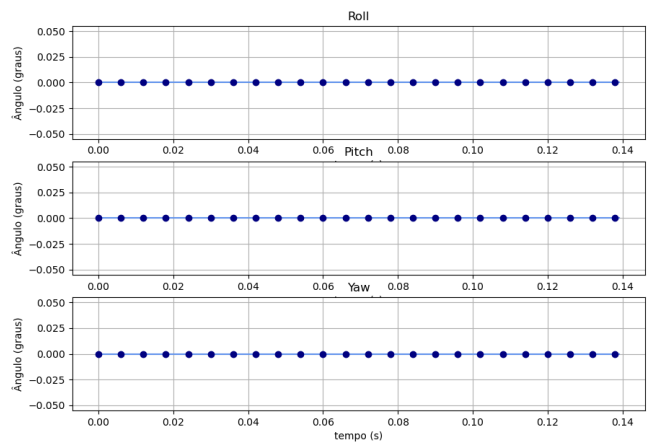


Figura 22. Trajetória dos ângulos de Euler do pé de balanço em função do tempo, para o primeiro passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

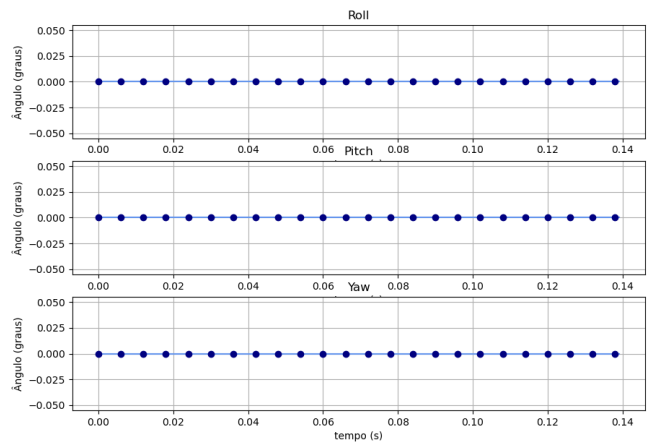


Figura 23. Trajetória dos ângulos de Euler do pé de balanço em função do tempo, para o segundo passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

do tempo, conforme previsto pelo modelo, discutido na seção IV. As pequenas flutuações ocorridas nesse eixo se devem aos ruídos já discutidos aqui.

A figura 33 mostra a trajetória executada pelo primeiro motor (de baixo para cima) dos tornozelos do robô (que corresponde ao movimento responsável pelo *roll*, em ângulos de Euler), no plano xy . Essas trajetórias são coincidentes com as trajetórias dos pés, por isso, serão usadas para analisá-las. Como é possível observar, as trajetórias são muito similares ao previsto pelas figuras 25 e 27. Além disso, também é possível notar que o pé esquerdo sai da posição nula e se move para trás, conforme também previsto pela figura 27

A figura 34 mostra a trajetória, no eixo z , em função do tempo, do mesmo motor. Observa-se que o valor inicial de z é de aproximadamente 2.9 cm e não 0. Isso ocorre porque esses motores estão um pouco acima dos pés do robô. Nota-

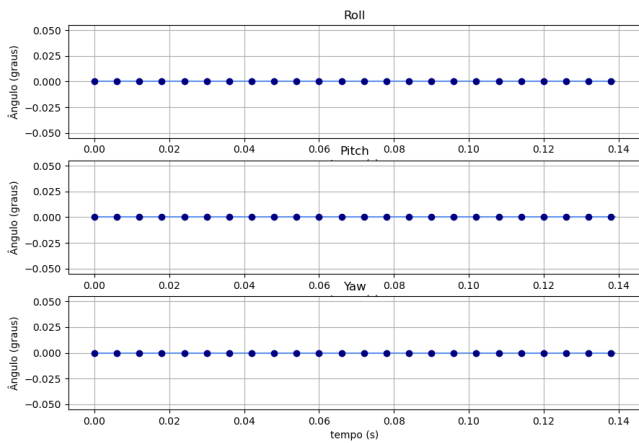


Figura 24. Trajetória dos ângulos de Euler do pé de balanço em função do tempo, para o terceiro passo. A curva contínua é a trajetória desejada e a curva com círculos é a trajetória calculada por cinemática direta, por meio dos pelos ângulos obtidos pelo controle, referentes ao primeiro passo da caminhada

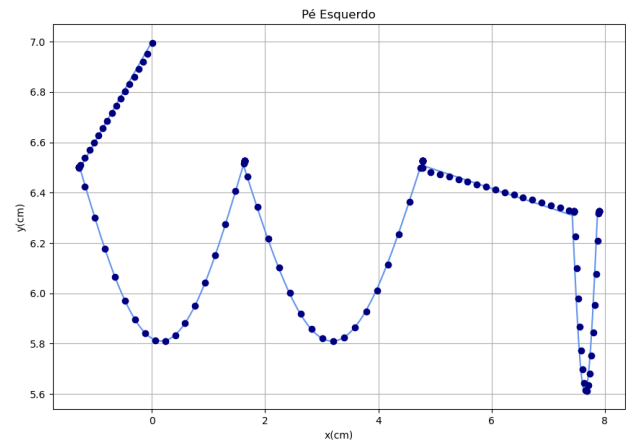


Figura 27. Trajetória do pé esquerdo, no plano xy durante todos os 10 passos. A curva contínua é a trajetória de referência e a curva com círculos é a trajetória executada pelo controle. A referência dos eixos é a posição inicial do pé direito

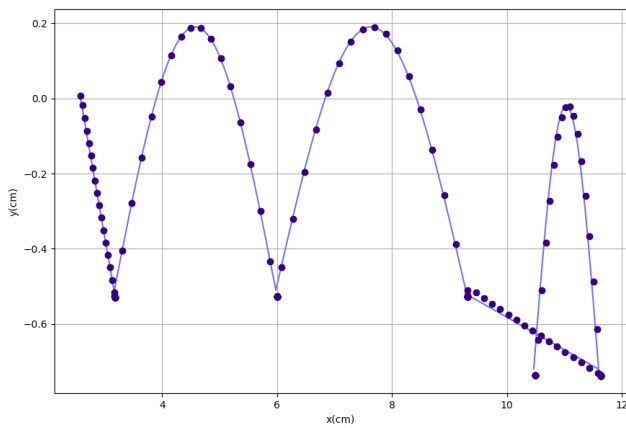


Figura 25. Trajetória do pé direito, no plano xy durante todos os 10 passos. A curva contínua é a trajetória de referência e a curva com círculos é a trajetória executada pelo controle. A referência dos eixos é a posição inicial do próprio pé

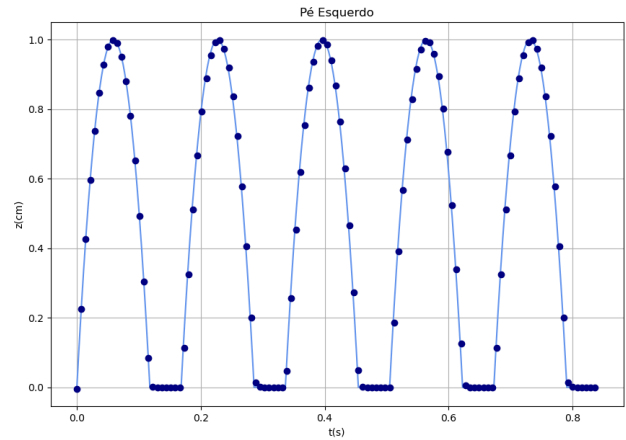


Figura 28. Trajetória do pé direito, no eixo z em função do tempo durante todos os 10 passos. A curva contínua é a trajetória de referência e a curva com círculos é a trajetória executada pelo controle. A referência dos eixos é a posição inicial do próprio pé

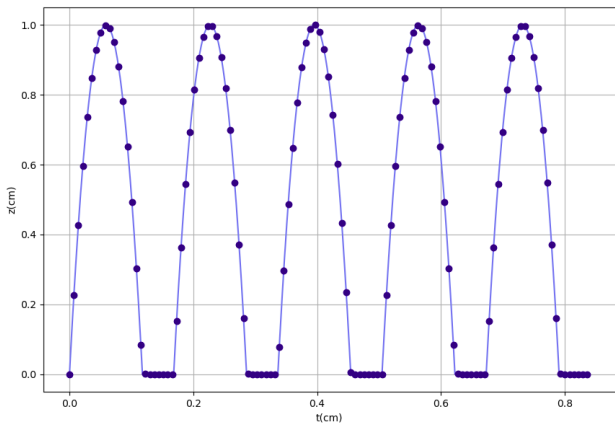


Figura 26. Trajetória do pé direito, no eixo z em função do tempo durante todos os 10 passos. A curva contínua é a trajetória de referência e a curva com círculos é a trajetória executada pelo controle. A referência dos eixos é a posição inicial do próprio pé

se que o valor máximo de amplitude das trajetórias foi de 3.9 cm, isso significa que a amplitude total foi de 1 cm, conforme o esperado (ver figuras 26 e 28).

Além disso, observa-se que, nos instantes de tempo em que o pé de balanço atinge seu valor máximo, o pé de apoio não encontra-se em seu valor mínimo. Isso porque o controle LQR gerou uma trajetória que imprimia valores diferentes de 0 no ângulo *roll* da perna de apoio. Normalmente, na caminhada humana, esse ângulo é mantido nulo e o ângulo *roll* do quadril é movimentado. Entretanto, como o Modelo do Pêndulo Invertido Linear prevê que a coordenada z é constante, a solução encontrada pelo controle para obedecer a essa condição foi a de manter ângulo *roll* do quadril nulo e movimentar o *roll* do tornozelo da perna de apoio. Isso pode ser confirmado pela figura 35, que mostra os valores assumidos pelo ângulo *roll* dos dois tornozelos. Nos instante inicial, por exemplo, a perna esquerda, que é a perna em

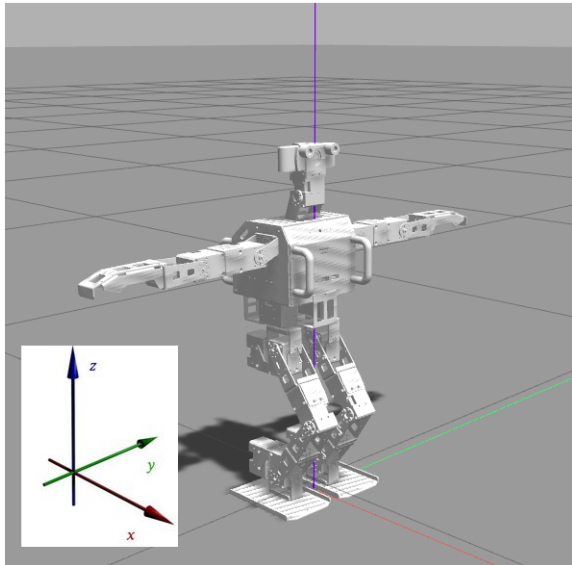


Figura 29. Darwin OP3 no ambiente de simulação do Gazebo

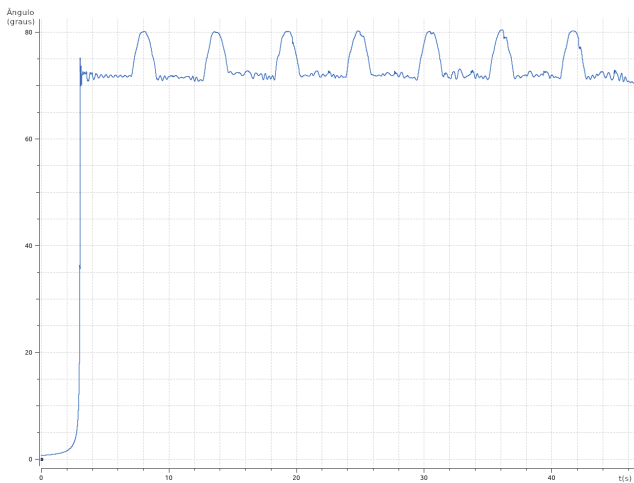


Figura 30. Valores de ângulo assumidos pela junta do joelho da perna esquerda

balanço, apresenta um valor de ângulo muito próximo a 0 (não é exatamente 0 devido às perturbações já mencionadas), enquanto que a perna de apoio (direita), apresenta valores mais distantes de 0. Esse padrão (perna de apoio mais distante de 0 e perna de balanço mais próxima de 0) se repete ao longo da caminhada.

Como é possível ver na figura 36, em determinados momentos, os valores assumidos pelos ângulos de Euler do centro de massa chegam, no máximo, a valores próximos 5° , mas esse valor é uma amplitude muito pequena, podendo-se considerar que eles apresentaram valores muito próximos de 0° . Isso confirma o que havia sido previsto pelas figura 14, 15 e 16. Essas pequenas oscilações de 5° ocorreram devido às perturbações já citadas aqui.

Como a biblioteca da ROBOTIS não inclui frames nos pés,

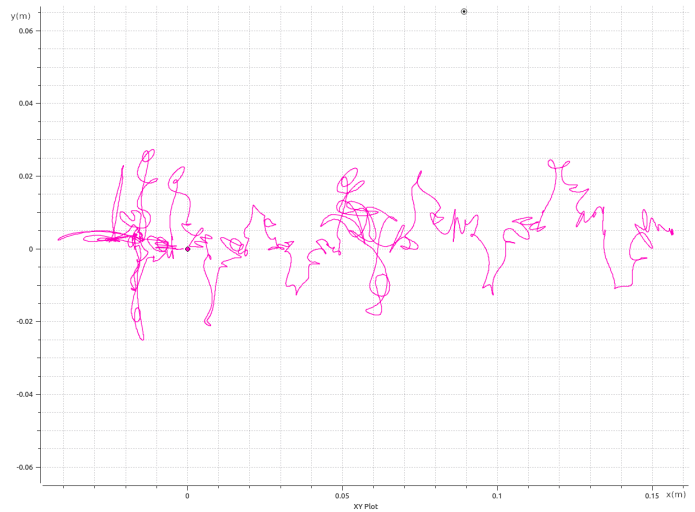


Figura 31. Trajetória, no plano xy , executada pelo CoM do robô durante uma simulação com 14 passos. A referência é a posição inicial do CoM no início da simulação

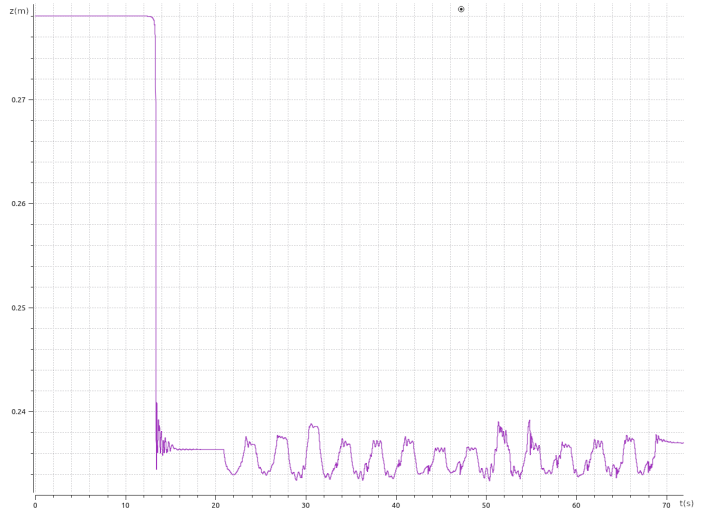


Figura 32. Trajetória no eixo z , em função do tempo executada pelo *base_link* do robô durante uma simulação com 14 passos. A referência é a posição inicial do mesmo frame no início da simulação

não é possível exibirmos aqui os ângulos de Euler destes, durante a trajetória, para comparar com o previsto pelas figura 22, 23 e 24

Para visualizar um vídeo da simulação, basta acessar o link <https://youtu.be/POFfcuMtqyg>

VIII. CONCLUSÃO E TRABALHOS FUTUROS

O modelo dinâmico do pêndulo invertido mostrou-se capaz de gerar uma trajetória compatível com a caminhada humana, para um robô de aproximadamente 51cm de altura, com um período de 0.14s entre cada passo.

Além disso, a ausência de sobressinais nas trajetórias calculadas por cinemática direta, utilizando-se os dois controles implementados neste trabalho, assim como suas semelhanças

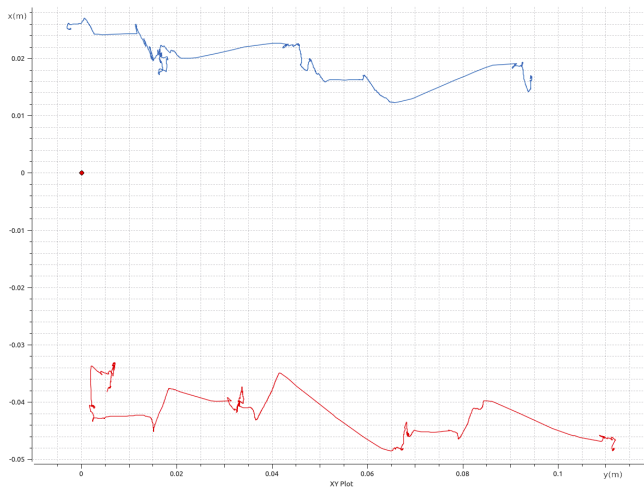


Figura 33. Trajetória, no plano xy , executada pelos primeiros motores (de baixo para cima) dos tornozelos do robô durante uma simulação com 14 passos. A referência dos eixos é a posição inicial do CoM no início da simulação. O tornozelo direito corresponde à curva em vermelho e o esquerdo corresponde à curva azul

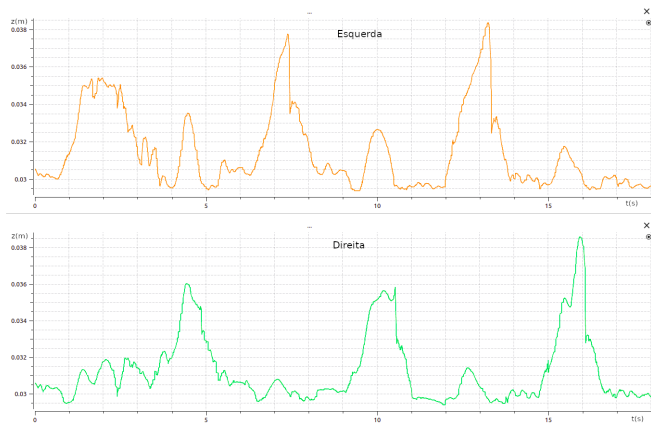


Figura 34. Trajetória no eixo z , em função do tempo executada pelos primeiros motores (de baixo para cima) dos tornozelos do robô durante uma simulação com 14 passos. A referência é o chão da simulação

com as respectivas referências mostram que ambos são eficientes para realizar o controle da caminhada de um robô humanoíde.

Apesar dos ruídos causados pela arquitetura da biblioteca de simulação da ROBOTIS, o robô conseguiu manter o equilíbrio e executar a trajetória, mostrando que os controles utilizados são robustos e resistente a ruídos.

A ausência de um robô real impossibilitou testes no mundo real.

Em trabalhos futuros, é necessário implementar o controle que determina a direção da caminhada. Também é importante adaptar os parâmetros do modelo (massa, comprimentos dos elos e ganhos dos controles) para os valores correspondentes ao humanoíde do Núcleo de Robótica Pequeni Mecânico e testá-lo. Após os testes no mundo real, também surgirá a

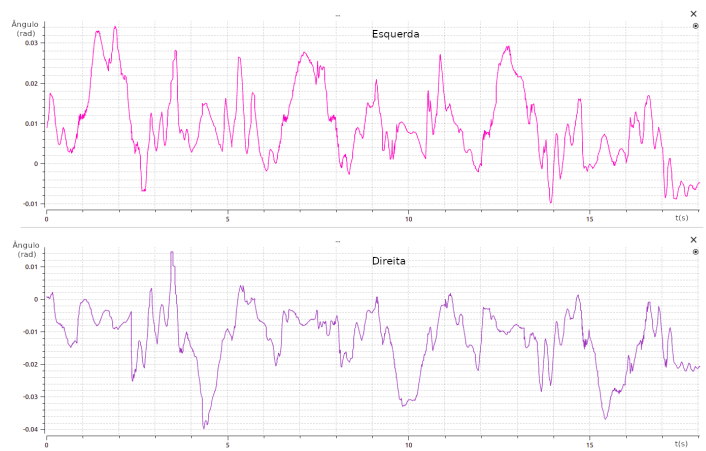


Figura 35. Trajetória dos ângulos $roll$, em função do tempo executada pelos primeiros motores (de baixo para cima) dos tornozelos do robô durante uma simulação com 14 passos.

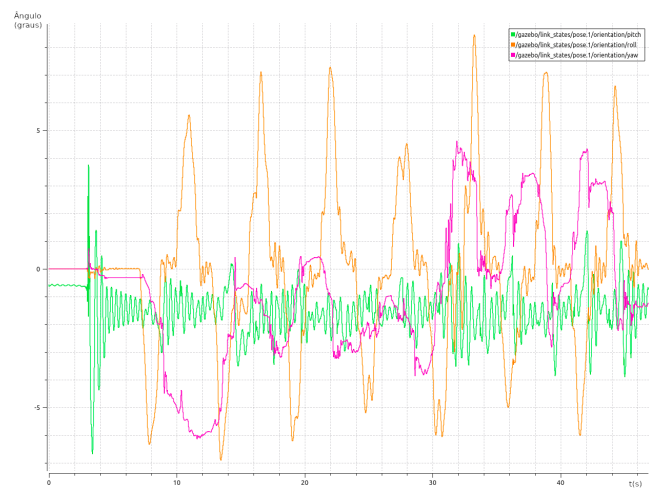


Figura 36. Trajetória dos ângulos $roll$ (laranja) $pitch$ (verde) e yaw (rosa), em função do tempo executada pelo centro de massa do robô durante uma simulação com 14 passos.

necessidade de melhorar o modelo para torná-lo capaz de lidar com as perturbações adicionais trazidas pelo mundo não ideal.

AGRADECIMENTOS

Agradeço à minha família por todo o suporte que me deram para que eu pudesse cursar a graduação.

Agradeço, também, ao Governo Federal e à Universidade Federal de Goiás (UFG), por promoverem uma educação pública e de qualidade.

Gostaria, também, de expressar minha gratidão ao Núcleo De Robótica Pequeni Mecânico, por acreditarem em mim e me oferecerem a oportunidade de fazer pesquisa na área de robótica.

Também agradeço a todos os amigos que a universidade me trouxe, pois, estes tiveram um papel muito importante na minha jornada acadêmica.

E, por último, deixo dois agradecimentos especiais. O primeiro aos professores da Escola de Engenharia Elétrica, Mecânica e de Computação, da UFG, por terem compartilhado seu conhecimento comigo, em especial aqueles que participaram da banca da minha defesa. O segundo, ao professor Dr. Marco Antônio Assfalk de Oliveira, por ter me orientado nessa jornada e ao Eng. Juan Ramón Belchior de França Silva, por também ter me orientado e ter permitido que eu desse continuidade ao trabalho dele.

[24] D. Silveria, “humanoidcontrol,” <https://github.com/dimitriasilveria/humanoidControl>, Agosto 2022.

REFERÊNCIAS

- [1] N. de Robótica Pequim Mecânico, “Pequim mecânico,” <https://www.pequimecanico.com.br/>, Agosto 2022.
- [2] RoboCup, “Robocupsoccer - humanoid,” <https://www.robocup.org/leagues/3>, Agosto 2022.
- [3] R. OP3, “Robotis e-manual,” <https://manual.robotis.com/docs/en/platform/op3/introduction/>, Janeiro 2021.
- [4] K. H. K. Y. Shuuji Kajita, Hirohisa Hirukawa, *Introduction to Humanoid Robotics*. Springer-Verlag Berlin Heidelberg, 2014.
- [5] B. S. Lorenzo Sciavicco, *Modelling and Control of Robot Manipulators*. Springer London, 2000.
- [6] S. Bagheri, T. Jafarov, L. Freidovich, and N. Sepehri, “Beneficially combining lqr and pid to control longitudinal dynamics of a smartfly uav,” in *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 2016, pp. 1–6.
- [7] B. V. Adorno, “Two-arm manipulation: From manipulators to enhanced human-robot collaboration,” Ph.D. dissertation, Université Montpellier II - Sciences et Techniques du Languedoc, 2011, unpublished thesis.
- [8] P. Robotics, “Reem-c technical specifications,” <https://web.archive.org/web/20130717154816/http://www.pal-robotics.com/press-kit/brochures/reemc-brochure.pdf>, Agosto 2022.
- [9] R. Advance, “Darwin op3 robotis,” <https://www.robot-advance.com/EN/art-darwin-op3-robotis-2358.htm>, Agosto 2022.
- [10] H. Dallali, G. A. Medrano-Cerda, and M. Brown, “A comparison of multivariable decentralized control strategies for robust humanoid walking,” in *UKACC International Conference on Control 2010*, 2010, pp. 1–6.
- [11] Y. Liu, “A dual-slip model for dynamic walking in a humanoid over uneven terrain,” Ph.D. dissertation, The Ohio State University, 2015, unpublished thesis.
- [12] J. R. Belchior, “Desenvolvimento de um sistema de controle para caminhada autônoma de um robô humanoide,” 2018.
- [13] M. J. O. Christopher L Vaughan, “Froude and the contribution of naval architecture to our understanding of bipedal locomotion,” *PubMed.gov*, vol. 1, no. 12, pp. 33–48, 2019.
- [14] J. Baltes and P. Lam, “Design of walking gaits for tao-pie-pie, a small humanoid robot,” *RoboCup 2003*, vol. 7, no. 3020, p. 355–365, 2004. [Online]. Available: <https://link.springer.com/content/pdf/bfm3A978-3-540-25940-42F1.pdf>
- [15] S. Kajita, F. Kanehiro, K. Kaneko, Kiyoslii Fujiwara, K. Yokoi, and H. Hirukawa, “A realtime pattern generator for biped walking,” *IEEE International Conference on Robotics Automation*, vol. 1, no. 4, pp. 31–37, 2002.
- [16] R. OP3, “Ros.org,” https://wiki.ros.org/robotis_op3, Janeiro 2021.
- [17] P. Robot, “Robotis op: Advanced ros based humanoid robot,” <https://www.pishrobot.com/en/product-en/robotis-op/>, Agosto 2022.
- [18] M. M. Marinho, L. F. C. Figueredo, and B. V. Adorno, “A dual quaternion linear-quadratic optimal controller for trajectory tracking,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 4047–4052.
- [19] K. Alexis, “Lqr brief overview,” <http://www.kostasalexis.com/lqr-control.html>, Agosto 2022.
- [20] M. R. Northwestern, “Task space and workspace,” <https://modernrobotics.northwestern.edu/nu-gm-book-resource/2-5-task-space-and-workspace/>, Agosto 2022.
- [21] R. e manual, “Mx-28ar, mx-28at,” <https://manual.robotis.com/docs/en/dxl/mx/mx-28/>, Agosto 2022.
- [22] F. C. Machado, “Curvas de bézier e desenho de fontes tipográficas,” https://www.ime.usp.br/~fabcm/files/projeto_bezier.pdf, Agosto 2022.
- [23] Gazebo, “Gazebo,” <https://gazebo.org/home>, Agosto 2022.