

VINÍCIUS ABRÃO DA SILVA MARQUES

**MÉTODOS ALGÉBRICOS APLICADOS AO
ESTUDO DO MOVIMENTO DE ROBÔS
HUMANOIDES**



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA**

2016

VINÍCIUS ABRÃO DA SILVA MARQUES

**MÉTODOS ALGÉBRICOS APLICADOS AO ESTUDO DO
MOVIMENTO DE ROBÔS HUMANOIDES**

Tese APROVADA pelo
Programa de Pós-Graduação em
Engenharia Mecânica da
Universidade Federal de Uberlândia,
como parte dos requisitos para
obtenção do título de DOUTOR EM
ENGENHARIA MECÂNICA.

Área de concentração: Mecânica dos
Sólidos e Vibrações.

Orientador: Prof. Dr. João Carlos
Mendes Carvalho.

Co-orientador: Univ. Prof. Mag. Dr.
Dr. h.c. Manfred Husty (Universidade
de Innsbruck – Áustria).

UBERLÂNDIA - MG

2016

Dados Internacionais de Catalogação na Publicação (CIP)
Sistema de Bibliotecas da UFU, MG, Brasil.

M357m Marques, Vinícius Abrão da Silva, 1989-
2016 Métodos algébricos aplicados ao estudo do movimento de robôs
humanoides / Vinícius Abrão da Silva Marques. - 2016.
148 f. : il.

Orientador: João Carlos Mendes Carvalho.

Coorientador: Manfred L. Husty.

Tese (doutorado) - Universidade Federal de Uberlândia, Programa
de Pós-Graduação em Engenharia Mecânica.

Inclui bibliografia.

1. Engenharia mecânica - Teses. 2. Robôs - Teses. 3. Geometria
algébrica - Teses. 4. Dinâmica - Teses. I. Carvalho, João Carlos Mendes,
1954- II. Husty, Manfred L. III. Universidade Federal de Uberlândia.
Programa de Pós-Graduação em Engenharia Mecânica. IV. Título.

CDU: 621

VINÍCIUS ABRÃO DA SILVA MARQUES

MÉTODOS ALGÉBRICOS APLICADOS AO ESTUDO DO MOVIMENTO DE ROBÔS
HUMANOIDES

Tese **APROVADA** pelo Programa
de Pós-Graduação em Engenharia
Mecânica da Universidade Federal
de Uberlândia.

Banca Examinadora:

Prof. Dr. João Carlos Mendes Carvalho – Orientador - UFU

Prof. Dr. Tarcisio Antonio Hess Coelho - USP

Prof. Dr. Daniel Martins - UFSC

Prof. Dr. Leonardo Sanches - UFU

Prof. Dr. Francisco Paulo Lépure Neto - UFU

Uberlândia, 16 de Dezembro de 2016

AGRADECIMENTOS

Eu agradeço inicialmente a Deus por minha vida e pelas oportunidades de concluir este trabalho.

Agradeço a toda minha família, em especial minha mãe Catarina, meu pai Bolívar e meu irmão Gustavo.

Um agradecimento especial à minha esposa Fernanda, pela sua companhia e por sempre me apoiar em minhas decisões.

Agradeço ao meu orientador Professor João Carlos Mendes Carvalho e ao Professor Manfred Husty que em grande parte contribuíram com o conhecimento necessário para desenvolver esta tese. Também agradeço ao Professor Rogério Sales Gonçalves e ao Martin Pfurner pelas suas grandes contribuições durante cada etapa deste trabalho.

Eu gostaria de agradecer aos membros do departamento de Geometria e CAD da Universidade de Innsbruck, Áustria. À todo o suporte técnico, receptividade e companheirismo durante o inesquecível ano de pesquisa e grande aprendizado de 2015.

Agradeço à Equipe De Desenvolvimento Em Robótica Móvel, EDROM, da Universidade Federal de Uberlândia, pelo suporte durante os testes experimentais.

Todo o meu agradecimento a cada um dos Professores da Faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia, pelo aprendizado em Engenharia Mecatrônica e Mecânica desde 2006.

Agradeço à Universidade Federal do Triângulo Mineiro por me dar a oportunidade de dedicar o meu tempo ao desenvolvimento deste trabalho desde 2013.

Agradeço ao CNPq por financiar este trabalho no programa Ciência sem Fronteiras e o meu período sanduíche na Universidade de Innsbruck.

MARQUES, V. A. S. **Métodos Algébricos Aplicados ao Estudo do Movimento de Robôs Humanoides**. 2016. 148 f., Tese de Doutorado, Universidade Federal de Uberlândia, Brasil.

Resumo

Robôs humanoides são mecanismos multicorpos com aparência humana, com duas pernas, dois braços e uma cabeça. Existem diferentes aplicações nas quais robôs humanoides podem ser utilizados, como em atividades domésticas, resgate de vítimas em desastres naturais e no auxílio à reabilitação de humanos. O modelo cinemático de robôs humanoides tem sido extensivamente estudado e atualmente podem ser encontradas diferentes soluções para este problema. O interesse é obter modelos que representem o movimento dos robôs de maneira mais similar possível ao dos humanos. Porém, apesar das soluções apresentadas na literatura, os robôs humanoides continuam caminhando de maneira pouco similar à humana, mesmo em terrenos planos e sem a ação de perturbações externas. Possíveis razões para estas diferenças são as limitações do modelo cinemático inverso utilizados em seus projetos. Em geral, os movimentos são analisados em dois planos independentes, mantendo o tronco e os pés com a mesma orientação, impossibilitando a obtenção de uma solução geral para descrever a trajetória espacial do robô humanoide. São aplicados métodos convencionais de análise geométrica baseados em matrizes de transformação e expressões trigonométricas, nas quais as orientações dos corpos são em geral descritas por ângulos de Euler. Esta abordagem apresenta singularidade na parametrização do movimento dada pelo problema de *Gimbal Lock*. A complexidade de se interpolar os movimentos descritos pelos ângulos de Euler é outro problema. Nesse sentido, o objetivo deste trabalho é apresentar um novo método para resolver a cinemática inversa de robôs humanoides mantendo seu equilíbrio dinâmico. O método se fundamenta na geometria algébrica e dual quaternions para obter o modelo cinemático inverso, o qual é capaz de descrever o movimento espacial de robôs humanoides sem restrições à orientação e aos deslocamentos desejados. A metodologia proposta permite interpolar os ângulos, velocidades e acelerações angulares das articulações, resultando em funções polinomiais, permitindo também obter a aceleração do centro de massa. A análise da estabilidade dinâmica é feita para o movimento teórico proposto, por simulações computacionais e por testes experimentais, com base no método do Ponto de Momento Nulo (ZMP – *Zero Moment Point*). A influência dos efeitos dinâmicos também é avaliada pela variação da aceleração do centro de massa do robô. O método foi aplicado no projeto do movimento de dois robôs de diferentes tamanhos. Os testes realizados confirmaram a estabilidade dinâmica da trajetória proposta, validando a metodologia para projetar e simular os movimentos de robôs humanoides.

Palavras chave: Robôs humanoides, geometria algébrica, estabilidade dinâmica, dual quaternions.

MARQUES, V. A. S. **Algebraic Methods Applied to Study the Movement of Humanoid Robots** 2016. 148 p., Ph.D. Dissertation, Federal University of Uberlândia, Brazil.

Abstract

Humanoid robots are multibody systems with appearance of humans, with two legs, two arms and a head. There are different applications where humanoid robots can be used, as in domestic's activities, rescue of victims in natural disasters and to support the human rehabilitation. The kinematic model of humanoid robots has been extensively studied and there are different approaches to solve it. The interest is to obtain such models that represent the humanoid gait as similar as possible to the human gait. Despite the solutions presented in the literature, the robots still walking with behavior different of human's, even on flat surfaces and without external disturbances. Possible reasons for these differences are the limitations of the inverse kinematics model used for the robot design. In general, the motions are split on two independent planes, keeping constant the torso and feet orientation, that does not give a general solution to describe a spatial trajectory of the humanoid robot. A common approach to obtain these models applies the conventional geometric methods to model the inverse kinematics, with matrix transformation and trigonometric expressions, where the multibody orientation is, in general, given by Euler angles. This approach presents parametrization singularities given by Gimbal Lock problem. The complexity to interpolate the motion described by Euler angles is another problem. Then, the objective of this work is to present a new method to solve the humanoid robot's inverse kinematics maintaining its dynamic equilibrium. The method is based on algebraic geometry and dual quaternions approach to obtain the inverse kinematics model, which is able to describe the spatial motion of humanoid robots without restrictions of orientation and required displacement. The proposed methodology enable to interpolate the joint angles, velocities and accelerations, resulting in polynomial functions, allowing to obtain the acceleration of the center of mass. The dynamic stability analysis is done to the proposed motion, by computational simulations and experimental tests, based on the method of Zero Moment Point (ZMP). The influence of dynamic effects is also evaluated by the variation of the robot center of mass acceleration. The method was applied to design the gait of two robots of different sizes. The tests have shown the dynamic stability of the proposed trajectory and validated the methodology to design and simulate the humanoid robot gait.

Keywords: Humanoid robots, algebraic geometry, dynamic stability, dual quaternions.

LISTA DE SÍMBOLOS

${}^j\mathbf{a}_P$: Vetor que define a aceleração linear do ponto P escrito em relação ao referencial j .

$a_{m,n}$: Elemento da linha m e coluna n da matriz de passagem homogênea jA_i .

$D_{i/j}$: Dual quaternion correspondente à matriz de passagem homogênea jA_i .

dt : Incremento temporal.

\mathbf{G} : Vetor campo gravitacional.

$h_i(v_2)$: Coordenadas dos hiperplanos que definem a estrutura serial 3R canônica.

h_j : Coordenadas dos hiperplanos que definem a estrutura esférica não canônica.

0I_i : Matriz de Inércia do corpo i em relação ao referencial inercial.

I_3 : Matriz identidade 3x3.

\mathbf{M}_P : Momento resultante em relação ao ponto P .

m_i : Massa do corpo i .

\mathbf{p} : Parte vetorial do quaternion P .

p_0 : Parte escalar do quaternion P .

p_i : Elemento do quaternion P .

jR_i : Matriz que define a orientação do referencial i em relação ao referencial j .

${}^j\mathbf{r}_P$: Vetor que define a posição do ponto P escrito em relação ao referencial j .

${}^j\mathbf{r}_{A/B}$: Vetor da posição relativa do ponto A em relação ao ponto B escrito no referencial j .

S: Dual quaternion correspondente à postura em que as estruturas seriais 3R canônica e esférica não canônica se interceptam.

T : Dual quaternion correspondente à matriz ${}^1T_{6*}$.

jT_i : Matriz de passagem homogênea do referencial j ao referencial i .

u_i : Coordenada angular associada à articulação i .

\dot{u}_i : Velocidade articular da articulação i .

\ddot{u}_i : Aceleração articular da articulação i .

v_i : Tangente do arco metade da coordenada articular u_i .

X_i, Y_i, Z_i : Eixos coordenados do referencial Σ_i .

x_i : Elemento do quaternion X.

$\alpha_i, \theta_i, a_i, d_i$: Parâmetros de Denavit-Hartenberg.

Δt : Intervalo de tempo entre duas posturas que se deseja interpolar.

η_i : Tangente do arco metade do parâmetro de Denavit-Hartenberg α_i .

(θ, Φ, ψ) : Ângulos de Euler (arfagem, rolagem, guinada).

Σ_0 : Referencial inercial.

Σ_B : Referencial de base para cálculo do movimento das pernas do robô humanoide.

Σ_i : Referencial i .

${}^j\omega_i$: Vetor velocidade angular do corpo i escrito no referencial j .

${}^j\dot{\omega}_i$: Vetor aceleração angular do corpo i escrito no referencial j .

SUMÁRIO

1. INTRODUÇÃO	1
2. ROBÔS HUMANOIDES	6
2.1. Histórico dos robôs humanoides.....	6
2.2. Estudo do movimento de robôs humanoides.....	9
2.2.1. Principais aspectos da abordagem proposta	9
2.2.2. Características da marcha humana	12
3. MODELO CINEMÁTICO DIRETO	18
3.1. Parâmetros de Denavit-Hartenberg.....	18
3.2. Descrição do modelo.....	20
4. MODELO CINEMÁTICO INVERSO.....	27
4.1. Singularidade na parametrização do movimento	27
4.2. Geometria Algébrica	30
4.3. Quaternions.....	33
4.4. Dual quaternions	36
4.5. Mapeamento Cinemático.....	40
4.6. Equações de uma estrutura serial canônica	42
4.6.1. Estrutura serial 2R canônica.....	42
4.6.2. Estrutura serial 3R canônica.....	46
4.7. Equações de uma estrutura serial 6R.....	51
4.8. Algoritmo para solução da cinemática inversa	53
5. SOLUÇÃO DO PROBLEMA CINEMÁTICO INVERSO	59
5.1. Trajetória Proposta	59
5.1.1. Deslocamento do centro de massa e dos pés.....	60
5.1.2. Orientação do centro de massa e dos pés	62
5.3. Solução do modelo cinemático inverso para a trajetória proposta.....	63
6. INTERPOLAÇÃO DO MOVIMENTO.....	71
6.1. Interpolação no espaço Cartesiano.....	72
6.2. Interpolação no espaço das juntas.....	73
6.2.1. Interpolação linear	73

6.2.2. Interpolação polinomial	74
6.3. Interpolação do movimento do robô humanoide	76
7. ESTABILIDADE DA TRAJETÓRIA PROPOSTA	85
7.1. Avaliação teórica	85
7.1.1. Método do Ponto de Momento Nulo	85
7.1.2. Análise da estabilidade.....	89
7.1.3. Influência dos parâmetros da trajetória	90
7.2. Simulação computacional	92
7.3. Testes experimentais	94
8. INFLUÊNCIA DE FATORES DINÂMICOS	98
8.1. Sensores Inerciais	98
8.2. Comparação teórico-experimental da aceleração do centro de massa	100
8.3. Método de avaliação teórica da estabilidade	102
9. CONCLUSÕES.....	105
10. TRABALHOS FUTUROS.....	109
REFERÊNCIAS BIBLIOGRÁFICAS	111
APÊNDICE A	121
APÊNDICE B	136
APÊNDICE C	142

CAPÍTULO I

INTRODUÇÃO

Durante o desenvolvimento de robôs móveis um dos maiores desafios está relacionado à obtenção de modelos teóricos que melhor representam o comportamento físico do robô. Tais modelos são utilizados principalmente para calcular o movimento dos atuadores de maneira que a trajetória do robô seja estável. De acordo com Angeles (2003) e Raibert *et al.* (1986), a estabilidade do movimento pode ser dividida em estabilidade estática e dinâmica. A estabilidade estática é a habilidade de manter a configuração mediante esforços reativos, enquanto a estabilidade dinâmica é a habilidade de manter a configuração mediante os esforços reativos e de inércia. São as estabilidades estática e dinâmica que garantem que o robô não se desequilibre durante o movimento.

A estabilidade de um robô móvel está diretamente relacionada ao número de pernas sobre as quais ele se apoia. Para robôs humanoides, a trajetória a ser percorrida durante sua caminhada deve ser previamente definida e cinematicamente avaliada, já que o robô se mantém em pé sobre duas pernas e, em alguns momentos, em somente uma perna.

Existe na literatura atual uma vasta quantidade de trabalhos relacionados à avaliação cinemática de robôs humanoides. No entanto, várias considerações são feitas de maneira a simplificar sua modelagem teórica. De maneira geral, a abordagem utilizada consiste em simplificar a representação do movimento em 2 planos, sendo estes frontal e lateral (sagital) ao robô. Em seguida, são propostas as trajetórias dos pés e do centro de massa nestes dois planos.

Dentro desta abordagem, Han (2012) apresenta o Método de Caminhada Intuitiva. Neste método, o movimento dos pés é definido em ambos os planos seguindo trajetórias senoidais. Apesar da possibilidade do deslocamento espacial dos pés do humanoide se assemelhar às características dos movimentos humanos, o método proposto por Han (2012) não permite que seja definida uma trajetória semelhante também para o movimento do centro de massa, a qual é definida sempre

paralela ao solo. Além da simplificação no deslocamento, são verificadas ainda considerações referentes às orientações dos pés e do centro de massa. Durante todo o movimento, o tronco é mantido sempre com uma mesma orientação, e os pés se movem sempre paralelos ao solo, de modo que o movimento do robô é realizado com várias restrições.

Metodologias semelhantes são encontradas em Kofinas (2012), Kofinas *et al.* (2013), Zorjan e Hugel (2013), Mistry *et al.* (2008), Zadeh *et al.* (2011), Ali *et al.* (2010), Meredith e Maddock (2015), Williams II (2012), Rokbani e Alimi (2012) e Nunez *et al.* (2012).

Após definidas as trajetórias e orientações dos pés e do centro de massa, é necessário calcular o movimento das articulações que movem o robô. Este cálculo é realizado utilizando os métodos da cinemática inversa.

Os trabalhos citados na literatura de robôs humanoides se baseiam em equações trigonométricas e orientações dadas de acordo com os ângulos de Euler. De acordo com Kenwrigth (2012), o uso de ângulos de Euler pode resultar em alguns problemas, como o de singularidade na parametrização do modelo, além de não ser possível suas interpolações diretamente ao longo da trajetória proposta.

São encontradas ainda na literatura relacionada à modelagem cinemática de robôs humanoides, abordagens baseadas em inteligência artificial. Neste tipo de abordagem são utilizados sensores e controladores em malha fechada, os quais definem parâmetros para as tomadas de decisão do robô durante o movimento. Em Wollherr (2005), a definição dos movimentos articulares é feita através de algoritmos de otimização. O uso de algoritmo genético e inteligência artificial é encontrado também em Arakawa e Fukuda (1997), Capi *et al.* (2001) e Katic e Vukobratovic' (2003).

Os métodos numéricos destes algoritmos apresentam a desvantagem de requerer um elevado custo computacional. Isso se torna ainda mais crítico quando os cálculos do modelo inverso são feitos pelo robô durante a caminhada, ou seja, não há uma trajetória previamente definida e analisada.

Neste trabalho é apresentada uma metodologia de projeto do movimento de robôs humanoides baseada em dual-quaternions, os quais eliminam os problemas existentes pela utilização dos ângulos de Euler. Além disso, o método proposto permite calcular o movimento das articulações para trajetórias desejadas, sem restrições para os movimentos dos pés e do centro de massa do robô humanoide, tanto em termos de posição como de orientação.

A modelagem analítica baseada em geometria algébrica e dual quaternions permite obter todas as possíveis soluções do modelo cinemático inverso. A solução do

modelo se fundamenta na resolução de um sistema de equações algébricas obtidas pelo mapeamento cinemático das posturas propostas para o movimento do robô.

É proposta também uma metodologia capaz de simular a trajetória desejada levando em conta sua estabilidade dinâmica. Esta análise leva em consideração os ângulos, as velocidades e as acelerações angulares das articulações, avaliando suas influências na aceleração do centro de massa do robô.

De maneira a definir completamente o movimento do robô, são avaliadas diferentes maneiras de interpolar a trajetória, sendo proposto um método que permite obter funções polinomiais que descrevam os ângulos, as velocidades e as acelerações angulares das articulações.

Não foram encontradas referências bibliográficas para o uso de dual quaternions aplicados ao problema de cinemática inversa de robôs humanoides. Neste trabalho é utilizado a álgebra de quaternions para a solução teórica da modelagem cinemática de robôs humanoides, acompanhada de validação numérica e experimental da estabilidade dinâmica dos movimentos propostos.

Os testes experimentais realizados mostraram que não é necessário o uso de controle em malha fechada para garantir a caminhada do robô humanoide nas condições de terreno plano e sem perturbação, validando o algoritmo proposto para simular a estabilidade do movimento.

Assim o objetivo deste trabalho consiste em planejar o movimento de robôs humanoides de maneira que ele seja estável e resolvendo a movimentação das articulações (cinemática inversa) a partir da solução de um sistema de equações polinomiais, evitando singularidades na parametrização do movimento.

Para atingir este objetivo será adotada a seguinte metodologia de trabalho.

Inicialmente, são definidas as trajetórias do centro de massa e dos pés do robô humanoide tendo como referência o estudo da marcha humana. As trajetórias incluem os perfis de posição e de orientação do centro de massa e dos pés a cada instante. Para o estudo cinemático, a trajetória foi discretizada em dez posturas.

A cinemática direta do robô humanoide é resolvida utilizando os tradicionais parâmetros de Denavit-Hartenberg.

Em seguida é feito o mapeamento cinemático direto que consiste em obter o dual quaternion correspondente às matrizes de transformação homogêneas obtidas na cinemática direta.

A partir dos dual quaternions obtidos é apresentado um algoritmo que permite obter um sistema de equações polinomiais cuja solução corresponde à solução da cinemática inversa. As coordenadas articulares são então interpoladas nas dez posturas.

A estabilidade do robô durante sua movimentação é feita de maneira teórica, usando o Método do Ponto de Momento Nulo (ZMP – “Zero Moment Point”) e experimentalmente.

Por fim é proposto um método para avaliar a estabilidade do robô quando sujeito a perturbações (esforços) aleatórios externos.

Tais objetivos são alcançados de acordo com a seguinte divisão de capítulos:

- No Capítulo II são descritos o estado da arte de robôs humanoides e uma breve avaliação do movimento humano durante sua caminhada. Desta maneira busca-se projetar o movimento do robô com características as mais humanas possíveis.

- No Capítulo III é apresentado o modelo cinemático direto utilizado para simular o movimento do robô.

- No Capítulo IV é descrita a fundamentação teórica para a modelagem inversa baseada em dual quaternions. Um algoritmo de solução da cinemática inversa de robôs humanoides é proposto para calcular a variação das coordenadas articulares das pernas do robô para posturas definidas ao longo de sua trajetória.

- O Capítulo V trata dos diferentes parâmetros necessários para descrever a trajetória e orientação do robô humanoide, baseados em posturas discretas. Um movimento desejado para o robô é proposto para a realização de simulações numéricas e testes experimentais. O modelo cinemático inverso é aplicado para calcular os valores das coordenadas articulares relativas à cada postura da trajetória discreta proposta.

- No Capítulo VI são discutidas diferentes possibilidades de se interpolar os movimentos do robô, tanto no espaço Cartesiano como no espaço das juntas. Uma interpolação polinomial no espaço das juntas é proposta. Diferentes ordens das funções polinomiais são avaliadas de acordo com sua influência na aceleração do centro de massa do robô humanoide.

- No Capítulo VII é avaliada a estabilidade dinâmica do movimento proposto pelo Método do Ponto de Momento Nulo (ZMP – “Zero Moment Point”). Os parâmetros da trajetória são verificados de acordo com sua influência na estabilidade dinâmica do movimento do humanoide. Além dos resultados teóricos, são apresentados também os

resultados das simulações numéricas utilizando-se o software Adams e os resultados dos testes experimentais.

- No Capítulo VIII são comparadas as acelerações teóricas e experimentais do centro de massa do robô humanoide. Durante a realização dos testes experimentais é aplicado o Filtro de Madgwich para o processamento dos dados dos sensores inerciais, o qual também se baseia em quaternions. Os efeitos de esforços dinâmicos são avaliados de acordo com sua influência na aceleração do centro de massa. Por fim, é proposto um método de simulação para avaliar a estabilidade de trajetórias de robôs humanoides, o qual leva em consideração a influência de fatores dinâmicos. É então apresentada uma proposta para definir um limite de estabilidade dinâmica baseado na aceleração do centro de massa, e com isso conseguir prever teoricamente qual deve ser a trajetória que o robô deve ter para manter a estabilidade.

- No Capítulo IX são apresentados as principais conclusões deste trabalho.

- Por fim, no Capítulo X são apresentadas algumas sugestões para trabalhos futuros .

CAPÍTULO II

ROBÔS HUMANOIDES

2.1. Histórico dos robôs humanoides

De acordo com Siciliano e Khatib (2008), desde o século XIV, artistas e até mesmo mágicos já se interessavam por criar mecanismos artificiais com aspectos humanos. Leonardo da Vinci se destacou não apenas por suas pinturas, como também por ser o primeiro a documentar um humanoide autômato em 1495.

Já no século 20, entre a Primeira Guerra Mundial e a década de 70, houve um período de interrupção no progresso e criação destes mecanismos, sendo retomado em seguida por pesquisadores e cientistas, e não mais por artistas. Kato *et al.* (1973) apresentaram no Japão o primeiro robô antropomórfico, denominado WABOT 1, capaz de realizar passos lentos usando simples controladores.

Em 1972, Vukobratovic´ na Iugoslávia apresentou um dos trabalhos mais importantes em robótica, usado até hoje como referência em pesquisas de estabilidade na locomoção de robôs móveis, denominado Método do Ponto de Momento Nulo (VUKOBRATOVIC´; BOROVARAC, 2004). Estes foram alguns dos primeiros trabalhos que despertaram o interesse no estudo da estabilidade dinâmica de robôs com pernas.

Durante a década seguinte, Raibert *et al.* (1984) iniciaram no MIT (*Massachusetts Institute of Technology*) estudos sobre a estabilidade de robôs com uma, duas e quatro pernas, desenvolvendo experimentos com controle dinâmico ativo e obtendo resultados expressivos.

Simultaneamente, Waldron *et al.* (1984) desenvolveram um hexápode denominado veículo de suspensão adaptativa (*Adaptive Suspension Vehicle*), um sistema capaz de andar sobre terrenos irregulares e dirigido por ação humana.

McGeer (1990) propôs o Método de Comportamento Naturalmente Cíclico, no qual duas pernas são capazes de se locomover sem ação de atuadores ativos em um plano inclinado. Tal método, denominado “dinâmica passiva” (*passive-dynamic*), tem

sido utilizado tanto para movimentação em planos inclinados ou não, tais como em Wisse *et al.* (2004), Chevallereau *et al.* (2003) e Collins *et al.* (2005).

Contribuindo amplamente com o desenvolvimento tecnológico de robôs humanoides, surgem nos anos 2000 as primeiras pesquisas realizadas por empresas como a Honda e seu robô ASIMO (Disponível em: < <http://asimo.honda.com/> >; Acesso em: 15/10/2016), além dos robôs QRIO da Sony (Disponível em: < http://www.sony.net/SonyInfo/News/Press_Archive/200312/03-060E/ >; Acesso em: 15/10/2016) e HRP da Kawada (Disponível em: < <http://global.kawada.jp/mechatronics/hrp4.html> >; Acesso em: 15/10/2016).

Pequenos robôs voltados ao entretenimento também surgem nesta época, como por exemplo o robô NAO da Aldebaran-Robotics (Disponível em: < www.aldebaran-robotics.com >; Acesso em: 15/10/2016). Além de popularizar as aplicações com humanoides, estes robôs são usados no auxílio educacional de crianças em escolas e para o entretenimento de pacientes em hospitais. Alguns kits, como os da Bioloid (Disponível em: < www.robotis.com >; Acesso em: 15/10/2016), também surgem no mercado reduzindo os custos e facilitando o desenvolvimento de pesquisas em Universidades.

Na última década, diversos grupos de pesquisa vem se destacando pelo mundo. Dentre estes pode-se citar o Laboratório de Robótica da Universidade Técnica de Munique, com o robô LOLA (BUSCHMANN *et al.*, 2009); Na Coréia do Sul, a equipe do Laboratório HUBO (LUO *et al.*, 2014 e KIM *et al.*, 2007), que em 2015 venceu o desafio DARPA (Disponível em: < <http://www.darpa.mil/program/darpa-robotics-challenge> >; Acesso em: 15/10/2016); e ainda os trabalhos desenvolvidos na Universidade de Tóquio pelo Professor Masayuki Inaba (MUROOKA *et al.*, 2015).

O Grupo de Robótica do MIT (Disponível em: < <https://robotics.mit.edu/> >; Acesso em: 15/10/2016) em parceria com a DARPA e a Boston Dynamics (Disponível em: < <http://www.bostondynamics.com/> >; Acesso em: 15/10/2016) podem ainda ser citados como referência em pesquisas atuais em robôs humanoides e robótica móvel em geral. Apesar de não terem trabalhos científicos publicados, frequentemente publicam vídeos com os resultados obtidos.

Na Universidade Federal de Uberlândia surgiu em 2009 a Equipe de Desenvolvimento em Robótica Móvel, EDROM (Disponível em: < <https://ufuedrom.wordpress.com/> >; Acesso em: 15/10/2016), que é formada principalmente por estudantes do curso de Engenharia Mecatrônica e desenvolve robôs humanoides para participar da ROBOCUP (Disponível em: < <http://www.robocup.org.br/> >; Acesso em: 15/10/2016), campeonato de futebol de robôs.

Na Figura 2.1 são apresentadas imagens de alguns robôs humanoides desenvolvidos nos últimos anos. A Figura 2.2 apresenta um dos robôs projetado pela EDROM.



Figura 2.1. Robôs humanoides: (a) HUBO (www.drc-hubo.com); (b) Johnnie (<https://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/johnnie/>); (c) Atlas (http://www.bostondynamics.com/robot_Atlas.html); (d) LOLA (<http://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/lola/>); (e) NAO (www.ald.softbankrobotics.com); (f) Darwin (www.robotis.com); (g) ASIMO (asimo.honda.com); (h) Bioloid (www.robotis.com), (Acessos em: 30/09/2016).

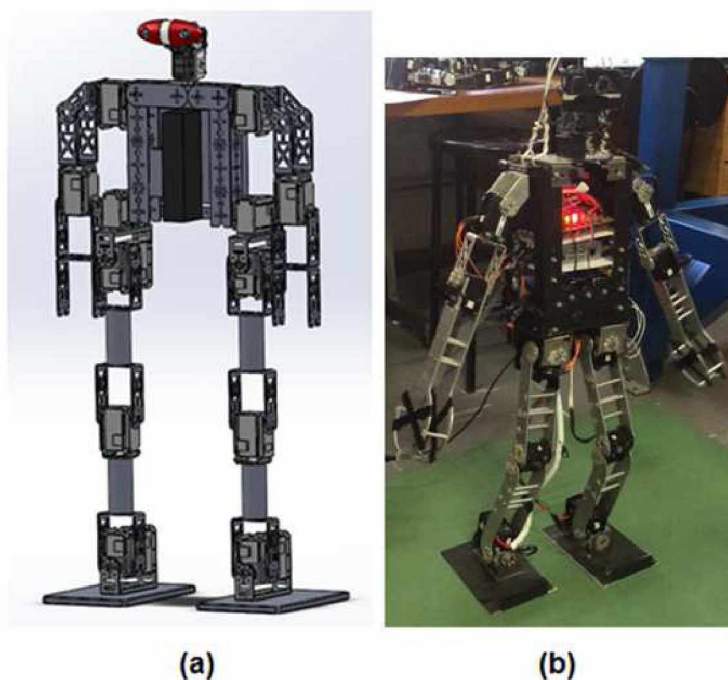


Figura 2.2. Robô humanoide projetado pela EDROM/UFU: (a) Modelo CAD; (b) Protótipo.

2.2. Estudo do movimento de robôs humanoides

Embora seja observado um grande avanço no estudo dos robôs bípedes, pode-se verificar ainda que existe a necessidade de uma melhor solução para problemas tais como: sua estabilidade durante seu caminhar; sua movimentação corporal mais semelhante à humana; e possuir dimensões dos pés proporcionais aos pés humanos.

Após o breve histórico apresentado sobre alguns dos robôs humanoides desenvolvidos ao longo da história da robótica móvel, esta seção visa apresentar os principais conceitos relacionados ao movimento de robôs humanoides. Inicialmente são descritas as motivações para se propor um novo método de estudo dos movimentos de robôs humanoides, procurando contribuir para a solução dos problemas citados no parágrafo anterior. Como referência para tal estudo, é necessário também apresentar os aspectos da marcha humana, afim de desenvolver robôs humanoides com características ainda mais semelhantes aos humanos.

2.2.1. Principais aspectos da abordagem proposta

A modelagem teórica dos movimentos de robôs humanoides é o principal foco deste trabalho. Na literatura, a geração de trajetórias e avaliação de sua estabilidade é

descrita em inúmeros trabalhos tais como em Li *et al.* (2012), Park *et al.* (2006), Schmitz *et al.* (2012), Kagami (2002), Zhe *et al.* (2007), Stasse *et al.* (2008), Kajita *et al.* (2010), Sugihara (2009) e Mitobe *et al.* (2000).

Porém, apesar da vasta literatura sobre o tema, pode-se destacar duas diferenças principais da metodologia proposta neste trabalho. A primeira está relacionada com o método de solução da cinemática inversa, que utiliza dual quaternions, eliminando o problema de singularidade de parametrização dos dados e permitindo que a trajetória do centro de massa e dos pés do robô sejam definidas de forma similar ao comportamento humano. A segunda está relacionada com a análise da estabilidade do robô durante sua movimentação a partir do estudo da aceleração do centro de massa, por sua vez obtida a partir da interpolação dos movimentos articulares.

No caso em que os esforços atuantes sobre o robô humanoide são previamente conhecidos, torna-se possível modelar o seu comportamento dinâmico e ainda calcular os valores de torques dos atuadores ao longo do movimento. Porém, na maioria dos casos reais, não é possível calcular o torque resultante nos atuadores. Isto ocorre porque não é possível prever todos os esforços que atuam sobre o robô, principalmente sobre condições de terrenos e ambientes desconhecidos. Neste trabalho é proposto que qualquer esforço seja considerado no modelo através de sua contribuição na aceleração do centro de massa do robô, conforme descrito no Capítulo VIII. Além disso, são definidos limites de estabilidade para o movimento que podem ser previamente simulados. Visto que perturbações afetam diretamente a aceleração do centro de massa, diferentes trajetórias podem ser simuladas afim de definir os valores limites de aceleração que garantam a estabilidade do movimento. Estas análises de estabilidade são baseadas no Método do Ponto de Momento Nulo e comparadas com simulações computacionais e testes experimentais.

A vantagem do algoritmo proposto neste trabalho é a possibilidade de simular previamente diferentes trajetórias, considerando diferentes variações de aceleração do centro de massa devido à esforços externos, e avaliando assim qual deve ser o movimento dos atuadores que resultam em uma trajetória desejada.

A ideia de avaliar o equilíbrio baseado no estudo de movimento do centro de massa é encontrada na literatura em trabalhos de biomecânica. Nataraj (2011) estuda o problema de controle do equilíbrio humano avaliando estatisticamente a influência de perturbações externas na aceleração do centro de massa. Ele observou uma relação direta entre perturbações e os dados de aceleração mensurados por sensores inerciais posicionados no centro de massa. Nataraj descreve que a aceleração do centro de massa representa o comportamento global do sistema dinâmico, e o

relaciona ao controle de postura e do equilíbrio. Além disso, ele destaca que são necessários apenas sensores inerciais posicionados na altura do abdômen, onde está localizado 75% da massa corporal, como descrito também por Horak e Nashner (1986), Krishnamoorthy *et al.* (2003) e Winter (1990).

De maneira geral, o problema de geração de trajetória de mecanismos é definido por Weber (2000) como um problema no qual se deve computar os movimentos espaciais de cada grau de liberdade do sistema, isto é, os sinais temporais das posições, velocidades e acelerações. Desta forma é definida a maneira que um mecanismo se move de uma posição inicial à uma final, sujeito a restrições que podem ser geométricas, cinemáticas e/ou dinâmicas. Pode-se ainda ser definidas posições intermediárias nas quais deseja-se que a trajetória passe. É importante que a trajetória seja suave, isto é, que a primeira e segunda derivadas da posição espacial ao longo do tempo sejam funções contínuas, caso contrário podem haver vibrações no movimento do mecanismo.

Murray (2010) define a geração de trajetória como um problema cinemático com condições de contorno definidas, sendo estas as posições espaciais iniciais e finais. Este problema pode ser resolvido tanto no espaço Cartesiano quanto no espaço das juntas. O primeiro consiste em determinar os deslocamentos, velocidades e acelerações globais do mecanismo, enquanto que o segundo consiste em obter os movimentos articulares, ou seja, os deslocamentos, velocidades e acelerações de cada articulação.

No caso do movimento de robôs humanoides, o problema de geração de trajetória depende dos seguintes parâmetros no espaço Cartesiano e no espaço das juntas:

- Espaço Cartesiano: Orientações, deslocamentos, velocidades e acelerações lineares e angulares do centro de massa e dos pés.
- Espaço das juntas: Visto que são utilizados somente atuadores rotativos, se baseia nos ângulos, velocidades e acelerações angulares das articulações.

Além da avaliação do equilíbrio, o presente trabalho também discute diferentes possibilidades para propor o movimento de robôs humanoides e calcular seus movimentos articulares. A análise cinemática do movimento pode ser feita para uma trajetória espacial sem restrições para os pés e o centro de massa, buscando um movimento com características mais semelhantes possível à humana.

2.2.2. Características da marcha humana

Para propor uma trajetória para robôs humanoides que seja semelhante à humana, é necessário conhecer o comportamento espacial dos movimentos realizados ao longo da caminhada humana.

Kharb *et al.* (2011) descreve a caminhada humana passando pelas seguintes fases: enquanto uma perna se move para frente, a outra atua como apoio ao movimento, até o momento em que ambos os pés estão em contato com o solo. Se apenas um pé está em contato com o solo, trata-se do período denominado de simples apoio. Pelo breve momento em que ambos os pés ficam em contato com o solo, tem-se o período de duplo apoio. Durante o passo, as pernas revezam o seu comportamento. A perna que inicialmente estava suportando o corpo pelo pé em contato com o solo passa a se movimentar para frente, enquanto a outra perna que se movimentava, passa a servir de apoio. Cada perna se move durante o período denominado de balanço, e um ciclo de passo é convencionado como sendo o período em que esta sequência de movimentos se repete. As etapas do passo humano, segundo Whittle (2007), são ilustradas na Fig. 2.3.

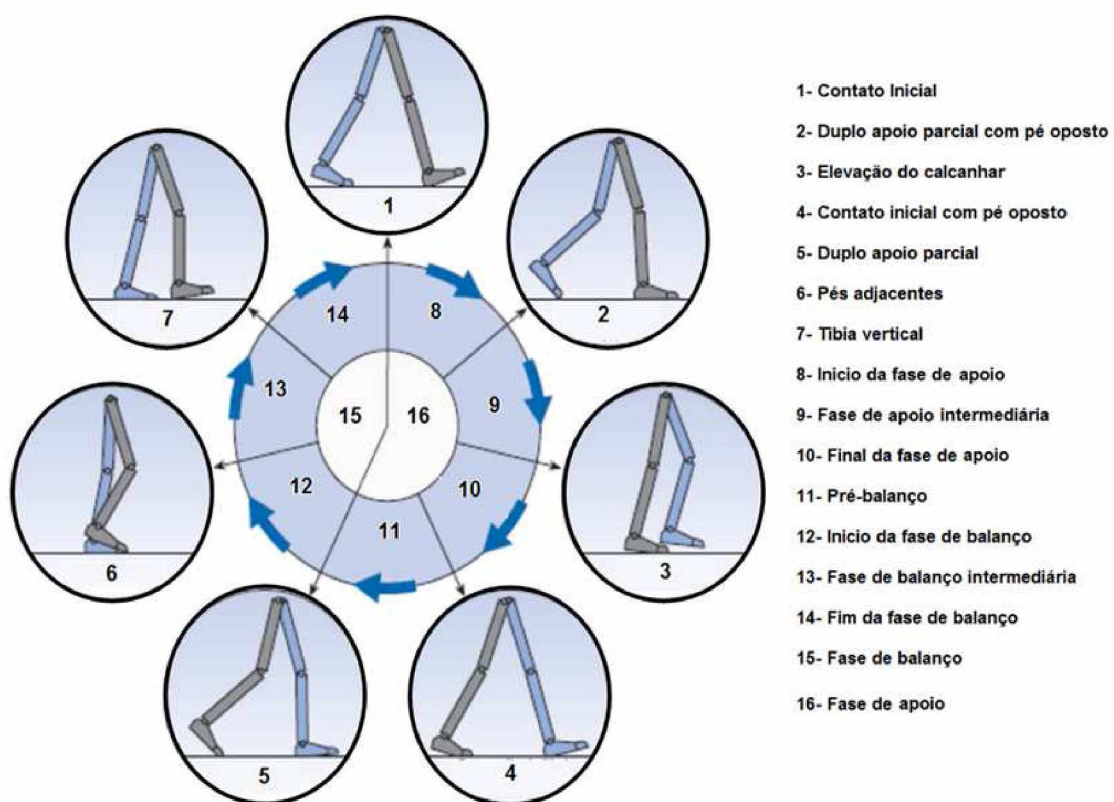


Figura 2.3. Eventos durante o ciclo do passo (Adaptado de WHITTLE, 2007).

Observa-se na Fig. 2.3 as etapas dos movimentos de contato do calcanhar e dos dedos com o solo, e da configuração das pernas enquanto suporte e nos períodos de movimento. Um ciclo de passo pode ainda ser definido como o intervalo de tempo entre a repetição de um destes eventos.

Apesar das diferentes orientações dos pés ao longo do ciclo do passo humano apresentado, na modelagem do movimento de robôs humanoides é definido um movimento para descrever a trajetória dos pés mantendo-o sempre paralelo ao solo, como apresentado por Han (2012) e ilustrado na Fig. 2.4.

As restrições impostas ao movimento visam simplificar o modelo cinemático utilizado por Han (2012). No entanto, o movimento resultante se torna pouco similar ao humano.

Além da trajetória dos pés, o movimento humano também apresenta maior complexidade quando se leva em conta a trajetória do centro de massa. Chou *et al.* (2000) apresenta uma análise biomecânica do movimento humano descrevendo a trajetória do centro de massa e dos pés em suas componentes espaciais: frontal (ou antero-posterior), vertical e mediolateral, sendo referidas nesta tese como frontal, vertical e lateral, respectivamente, conforme Figs. 2.5 e 2.6.

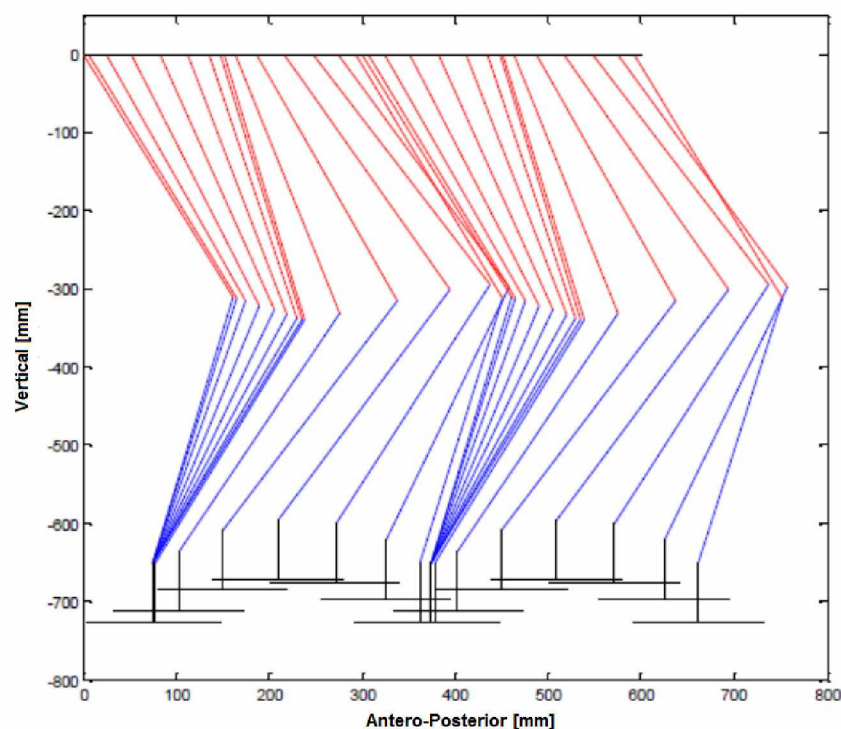


Figura 2.4-a. Vista lateral da trajetória dos pés durante o movimento (Adaptado de HAN, 2012).

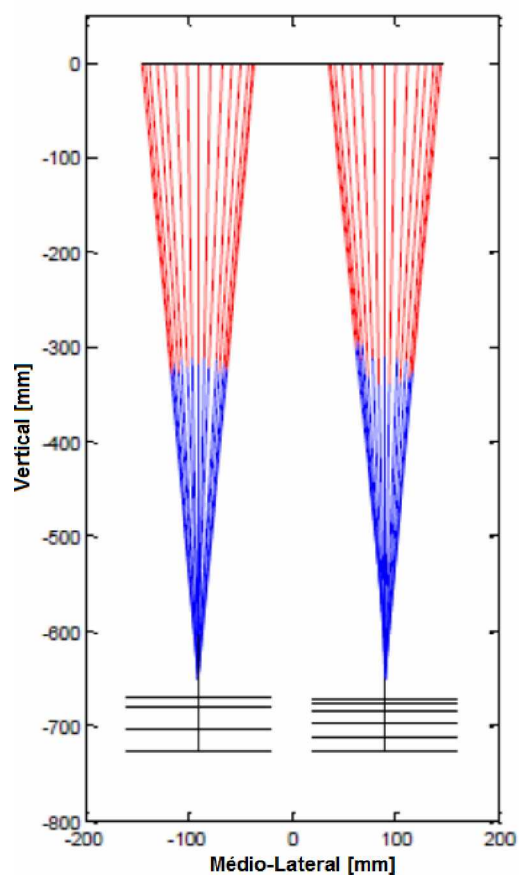


Figura 2.4-b. Vista frontal da trajetória dos pés durante o movimento (Adaptado de HAN, 2012).

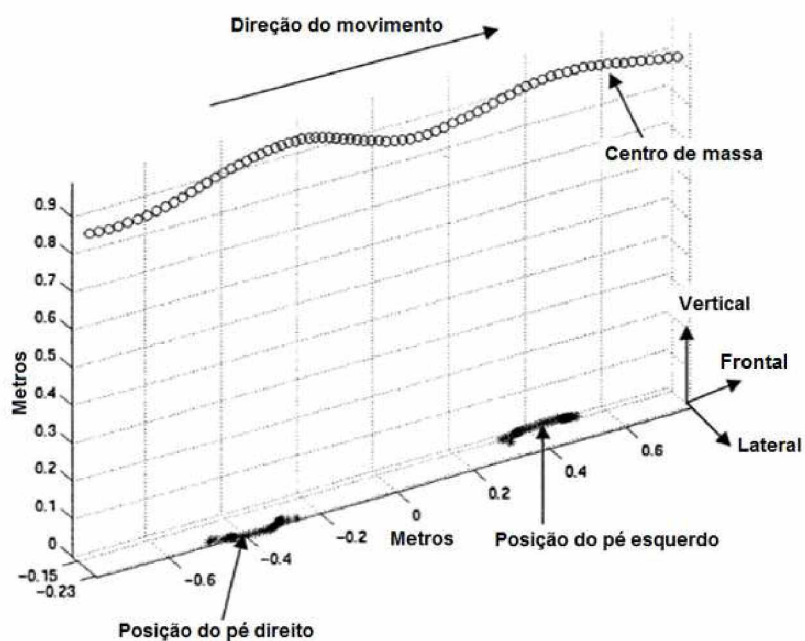


Figura 2.5. Trajetória espacial do centro de massa e dos pés ao longo da caminhada humana (Adaptado de CHOU *et al.*, 2000).

Na Figura 2.6 observa-se que o centro de massa segue uma trajetória aproximadamente senoidal nas direções frontal e lateral, além de se deslocar verticalmente, sendo estas direções definidas como apresentado na Fig. 2.5.

A fim de monitorar o comportamento do movimento humano, sensores inerciais tem sido aplicados para mensurar a aceleração dos pés ao longo do ciclo do passo. Zijlstra e Hof (2003) apresentam as componentes da aceleração dos pés e a força de reação do contato dos pés com o solo durante a caminhada humana, conforme Fig. 2.7.

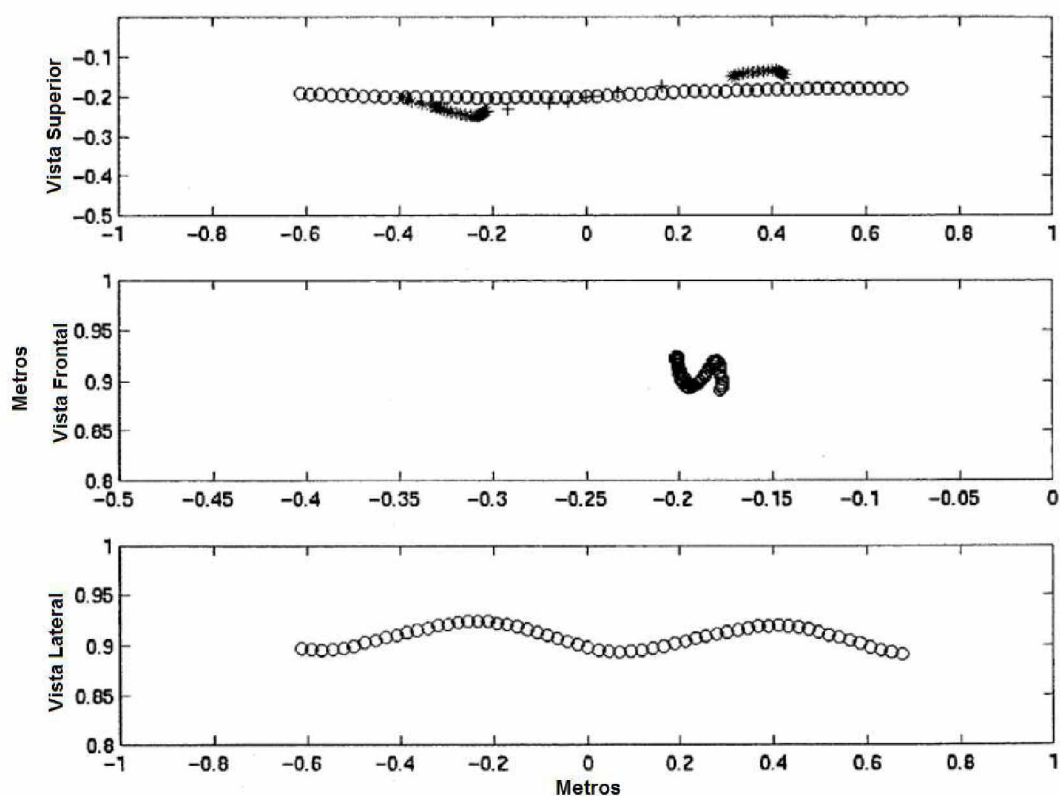


Figura 2.6. Vistas superior, frontal e lateral da trajetória espacial do centro de massa (o) e dos pés (+) durante a caminhada humana (Adaptado de CHOU *et al.*, 2000).

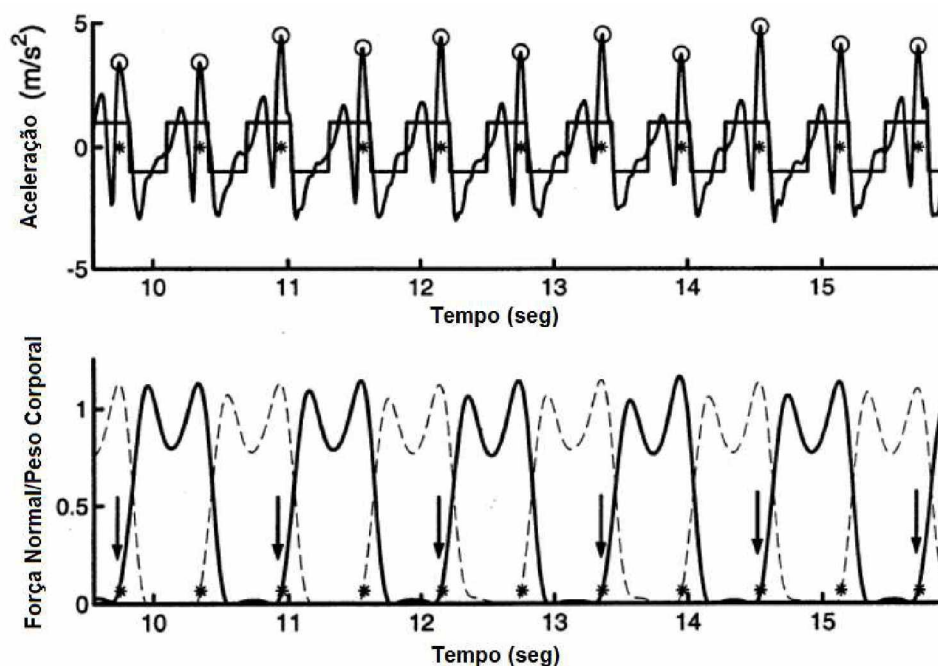


Figura 2.7. Aceleração dos pés e força de reação do contato dos pés com o solo dividida pelo peso corporal (Adaptado de ZIJLSTRA; HOF, 2003).

Na Figura 2.7 os círculos indicam o início e fim de um ciclo de passo, também representado pelo sinal quadrado que altera entre -1 e 1 indicando as fases de contato e movimento do pé direito, respectivamente. Os asteriscos indicam o momento em que o pé entra em contato com o solo. As linhas sólidas são referentes ao movimento do pé direito e as linhas tracejadas, ao movimento do pé esquerdo. A aceleração mensurada se refere à componente na direção frontal do movimento.

Na Figura 2.8 são apresentadas as componentes frontal, vertical e lateral da aceleração dos pés, sendo a vertical subtraída da aceleração gravitacional. No eixo horizontal está indicada a etapa do ciclo, sendo 0 o instante em que se inicia o contato do pé com o solo e 50 o instante em que o pé se levanta e avança para frente. Os resultados são de uma pessoa do sexo feminino andando com diferentes velocidades, indicadas à direita.

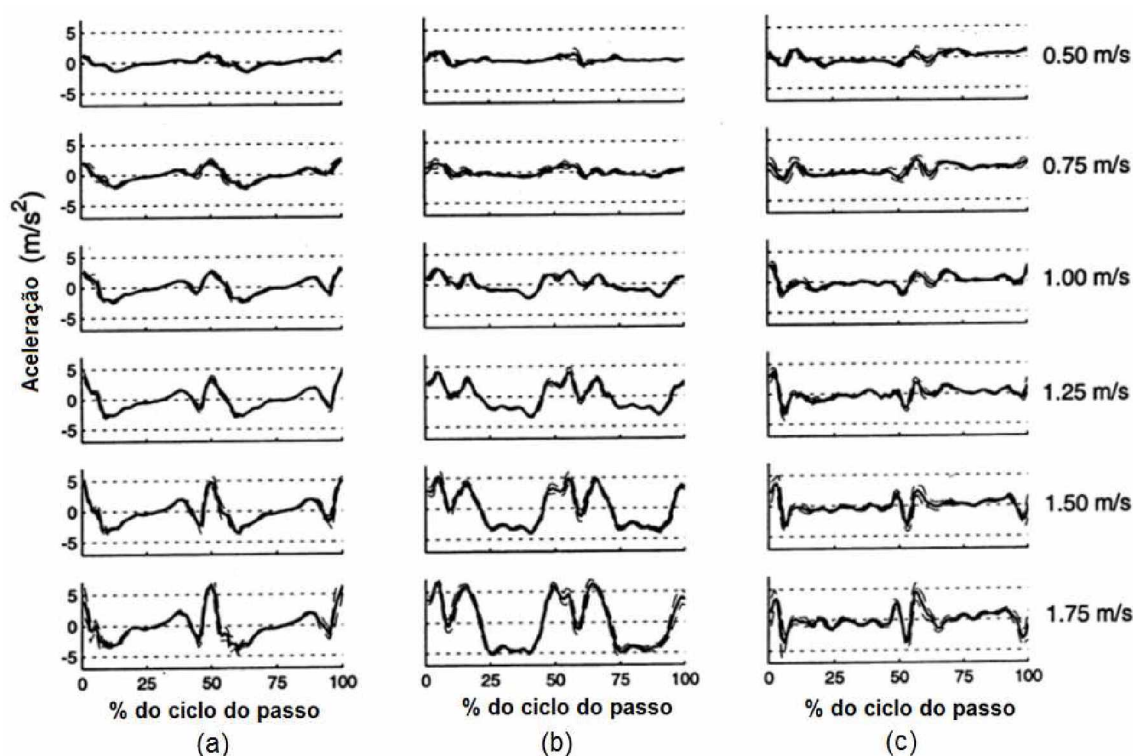


Figura 2.8. Componentes da aceleração dos pés: (a) frontal; (b) vertical; (c) lateral
(Adaptado de ZIJLSTRA; HOF, 2003).

Os dados experimentais indicam que o movimento dos pés e do centro de massa possuem componentes espaciais ao longo das três direções de movimento. Visto que se deseja que o movimento de robôs humanoides se assemelhem ao movimento humano, deve-se considerar estas três componentes espaciais também ao definir a trajetória desejada para o robô, e não apenas uma simplificação como verificado nos trabalhos já publicados.

Neste trabalho são propostas trajetórias espaciais tanto para os pés quanto para o centro de massa, além de permitir que suas orientações possam ser definidas de maneira espacial. Com isso, busca-se eliminar as simplificações encontradas em diversas publicações, nas quais a orientação dos pés e do centro de massa são mantidas sempre paralelas ao solo e a componente vertical do deslocamento do centro de massa é definida como sendo nula.

CAPÍTULO III

MODELO CINEMÁTICO DIRETO

A modelagem cinemática direta tem como objetivo relacionar o movimento das articulações, consideradas como variáveis de entrada, com a trajetória de um ponto de interesse do mecanismo. Na análise de movimento de um robô humanoide é possível, através do modelo cinemático direto, calcular a posição do seu centro de massa para cada conjunto de valores instantâneos dos ângulos das coordenadas articulares das articulações de suas pernas. Com o modelo cinemático direto é possível também simular o movimento do robô.

Neste trabalho são utilizados os parâmetros de Denavit-Hartenberg (1955). Neste capítulo, uma análise cinemática é realizada para obter as matrizes de passagem homogênea entre os referenciais posicionados nas articulações das pernas e no centro de massa do robô humanoide.

Diferentes notações podem ser encontradas na literatura para definir os referenciais e assim obter os parâmetros de Denavit-Hartenberg, como pode ser visto em Angeles (2003), Craig (2005) e Pfurner (2006). Neste trabalho é utilizada a nomenclatura conforme apresentado na Seção 3.1.

Para isso é definida a estrutura serial utilizada para representar as pernas do robô humanoide; a posição de cada referencial e as dimensões geométricas da estrutura.

3.1. Parâmetros de Denavit-Hartenberg

Os parâmetros de Denavit-Hartenberg têm sido largamente utilizados na modelagem de estruturas robóticas. Isso justifica seu uso nesta tese. No entanto, qualquer método que permita descrever a posição e a orientação de um corpo rígido no espaço pode ser utilizado.

Uma estrutura serial nR é definida como um mecanismo composto pela sequência de $n+1$ corpos, incluindo a base, conectados por n articulações rotativas. Os referenciais são numerados a partir do zero, da base da estrutura serial, até o

referencial $n+1$, localizado no elemento terminal. A articulação i conecta os corpos $i-1$ e i , e o referencial Σ_i é preso no corpo i .

Os parâmetros de Denavit-Hartenberg utilizados para a modelagem cinemática direta de uma estrutura serial são ilustrados na Fig. 3.1.

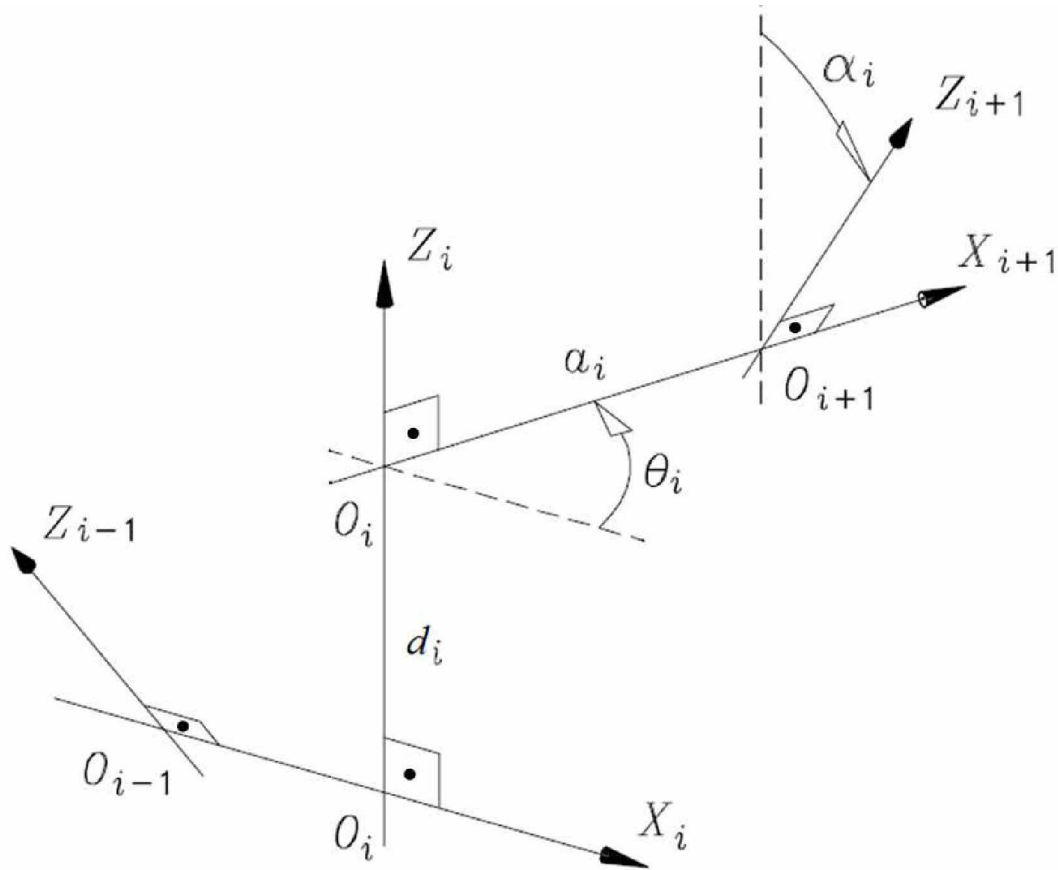


Figura 3.1. Parâmetros de Denavit-Hartenberg (Adaptado de ANGELES, 2003).

A definição dos referenciais Σ_i ($i=1, \dots, n+1$), segue a seguinte regra: O eixo Z_i é o eixo da articulação i , com o sentido arbitrário; a origem O_i do referencial Σ_i é posicionada na interseção do eixo Z_i com a perpendicular comum entre Z_{i-1} e Z_i . Nos casos em que os eixos Z_{i-1} e Z_i forem paralelos, uma escolha adequada da posição de O_i , pode simplificar as operações. O eixo X_i possui a direção da perpendicular comum entre Z_{i-1} e Z_i , com sentido de Z_{i-1} para Z_i . O eixo Y_i é definido pela regra da mão direita.

Os parâmetros de Denavit-Hartenberg (α_i , θ_i , a_i , d_i) da matriz de passagem homogênea ${}^i T_{i+1}$, do referencial Σ_i para o referencial Σ_{i+1} , são definidos por: α_i representa o ângulo entre os eixos Z_i e Z_{i+1} medido em torno de X_{i+1} ; a_i é o comprimento da perpendicular comum entre Z_i e Z_{i+1} na direção de X_{i+1} ; d_i é a distância entre X_i e X_{i+1} na direção de Z_i ; e θ_i , sendo o ângulo entre os eixos X_i e X_{i+1} medido em torno de Z_i . O parâmetro θ_i inclui a coordenada articular u_i , que define o movimento da articulação i em torno do eixo Z_i .

Nesta tese, quando todas as rotações u_i são nulas, o mecanismo está em sua condição inicial.

O referencial de base Σ_0 pode ser definido arbitrariamente. No entanto, uma definição adequada pode simplificar as operações, sempre respeitando as regras apresentadas anteriormente, conforme Fig. 3.1. Isto também é válido para a localização do referencial Σ_{n+1} , localizado no elemento terminal.

A matriz de passagem homogênea do referencial i ao referencial $i+1$ contendo os parâmetros de Denavit-Hartenberg, ${}^iT_{i+1}$, pode ser obtida por:

$${}^iT_{i+1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a_i \cos(\theta_i) & \cos(\theta_i) & -\cos(\alpha_i) \sin(\theta_i) & \sin(\alpha_i) \sin(\theta_i) \\ a_i \sin(\theta_i) & \sin(\theta_i) & \cos(\alpha_i) \cos(\theta_i) & -\sin(\alpha_i) \cos(\theta_i) \\ d_i & 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix}. \quad (3.1.1)$$

Então, a matriz de transformação homogênea do referencial Σ_0 ao referencial Σ_{n+1} , posicionado no elemento terminal, é obtida pela multiplicação das matrizes,

$${}^0T_{n+1} = {}^0T_1 \cdot {}^1T_2 \cdot \dots \cdot {}^nT_{n+1}. \quad (3.1.2)$$

3.2. Descrição do modelo

O modelo cinemático definido para o robô humanoide é composto por uma cintura, onde se encontra o seu centro de massa (CM), e que serve como elemento de fixação para as duas pernas, sendo que cada perna consiste em uma estrutura serial 6R, conforme esquematizado nas Figs. 3.2 e 3.3.

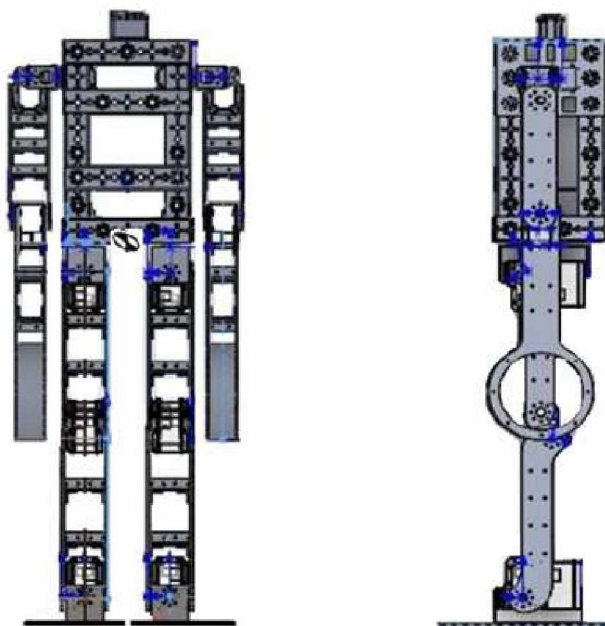


Figura 3.2. Robô humanoide utilizado na modelagem.

Para a análise cinemática do robô humanoide os movimentos dos membros superiores são desprezados durante a trajetória proposta. Apesar dos braços não se moverem, suas inércias, assim como as inércias do tronco e da cabeça, são consideradas para obter a posição do centro de massa do corpo do robô humanoide. Cabe salientar que é possível modelar o comportamento do robô considerando o movimento dos membros superiores. Porém, esse não é o objeto de estudo nesta fase do trabalho.

Para o cálculo da posição do centro de massa do robô foi feita a sua montagem no *software* SolidWorks, sendo então consideradas as inércias de todos os seus membros. A posição do seu centro de massa obtida pelo *software* foi validada experimentalmente posicionando-se o robô deitado sobre um cilindro, que por sua vez estava apoiado sobre uma superfície plana. A posição do robô sobre o cilindro foi ajustada até que o sistema assumisse a configuração de equilíbrio, obtendo-se desta forma a confirmação da posição do seu centro de massa.

Os referenciais das articulações das pernas foram posicionados e orientados conforme indicados na Fig. 3.3, onde também são apresentados os comprimentos considerados para a sua modelagem cinemática, seguindo a definição dos parâmetros de Denavit-Hartenberg, descritos anteriormente.

Para o estudo das trajetórias descritas pelos pés e pelo centro de massa, foi considerado um referencial de base, Σ_B , na ponta do pé de apoio. Para simplificar a modelagem, foi desprezada a espessura do pé e considerada a origem de Σ_B , O_B , sobre o eixo da primeira articulação, conforme esquematizado na Fig. 3.3-a. Um

referencial Σ_{CM} foi definido no centro de massa, com seu eixo Z_{CM} na vertical. Os eixos Z_i ($i=B, 1, 2, \dots, 6$), da perna esquerda, foram definidos como equipolentes com os eixos Z_j ($j=8, 9, \dots, 14$), da perna direita, conforme Fig. 3.3-b. Além disso, a posição do centro de massa foi considerada fixa em relação às articulações da cintura, desprezando as variações devido ao movimento do robô.

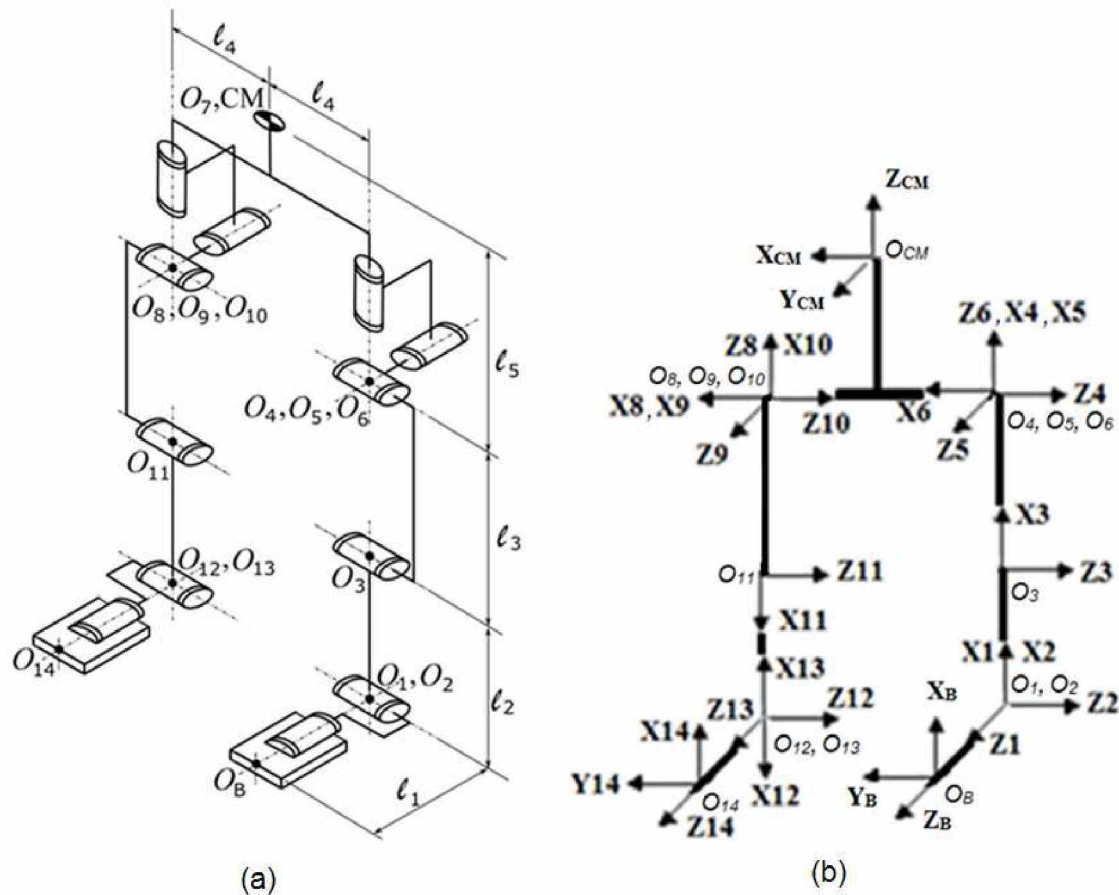


Figura 3.3. Robô humanoide: (a) Configuração das articulações; (b) Referenciais.

As seis articulações de cada perna podem ser descritas como:

- uma articulação esférica na região da cintura, permitindo sua rotação em relação ao tronco em 3 direções. Essa articulação pode ser representada por três articulações de rotação com eixos ortogonais entre si e concorrentes em um mesmo ponto;
- uma articulação rotativa na região dos joelhos, permitindo flexioná-la (dobrar a perna) e,
- duas articulações rotativas na região dos tornozelos, permitindo a rotação dos pés em torno das direções lateral e frontal ao robô.

Para descrever a trajetória do robô foi definido um referencial inercial Σ_o , inicialmente coincidente com o referencial de base Σ_B , cujas direções definidas como vertical, lateral e frontal são apresentadas na Fig. 3.4.

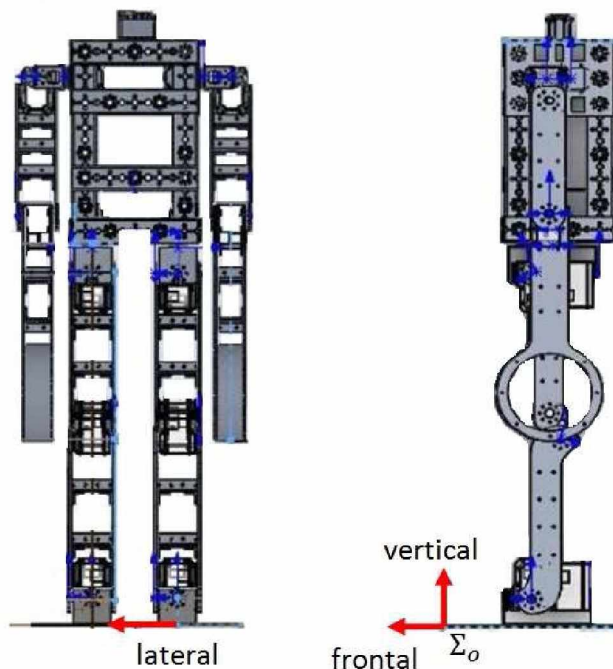


Figura 3.4. Posição do referencial inercial Σ_o e as direções definidas como vertical, lateral e frontal.

O referencial de base da estrutura serial utilizada para representar o movimento de cada perna, denominado por Σ_B , é posicionado na extremidade frontal do pé de apoio e, na condição inicial do movimento, é posicionado no pé esquerdo. A trajetória do centro de massa em relação ao referencial de base, Σ_B , é definida pelo movimento da estrutura 6R representativa da perna esquerda. Já o movimento do pé direito em relação ao referencial de base, Σ_B , é obtido pela composição das duas estruturas 6R relativas a cada uma das pernas, resultando em uma estrutura serial 12R.

Durante a simulação do movimento, o referencial de base é posicionado inicialmente no pé esquerdo, o qual é mantido parado em contato com o solo durante a primeira metade do ciclo do passo. Para o movimento subsequente, independente da quantidade de passos, a posição e orientação do referencial inercial Σ_o se mantém inalterada. Já Σ_B alterna para estar sempre na extremidade do pé de apoio, ou seja, quando o pé de apoio passa a ser o direito, o referencial de base passa a ser posicionado na extremidade do pé direito. O referencial inercial é mantido na posição

inicialmente definida, de maneira que as componentes dos deslocamentos dos corpos sejam sempre incrementadas ao longo do movimento.

Os parâmetros de Denavit-Hartenberg da estrutura serial 12R, conforme Fig. 3.3, são dados na Tab. 3.1.

Tabela 3.1. Parâmetros de Denavit-Hartenberg do robô humanoide, conforme Fig. 3.3.

i	a_i	d_i	α_i	θ_i
B	0	$d_B = -l_1$	0	0
1	0	0	$\pi/2$	u_1
2	$a_2 = l_2$	0	0	u_2
3	$a_3 = l_3$	0	0	u_3
4	0	0	$-\pi/2$	u_4
5	0	0	$\pi/2$	$u_5 + \pi/2$
6	$a_6 = l_4$	$d_6 = l_5$	0	u_6
7=CM	$a_7 = l_4$	$d_7 = -l_5$	0	0
8	0	0	$-\pi/2$	u_7
9	0	0	$\pi/2$	$u_8 - \pi/2$
10	$a_{10} = l_3$	0	0	$u_9 - \pi$
11	$a_{11} = l_2$	0	0	u_{10}
12	0	0	$-\pi/2$	$u_{11} + \pi$
13	0	$d_{13} = l_1$	0	u_{12}

A partir dos parâmetros de Denavit-Hartenberg, apresentados na Tab. 3.1, é possível calcular a matriz de passagem homogênea, ${}^B T_{CM}$, do centro de massa em relação ao referencial de base, posicionado no pé de apoio,

$${}^B T_{CM} = {}^B T_1 {}^1 T_2 {}^2 T_3 {}^3 T_4 {}^4 T_5 {}^5 T_6 {}^6 T_{CM}. \quad (3.1.3)$$

De maneira semelhante, a partir dos 12 graus de liberdade do modelo, considerando as duas pernas, é possível obter a matriz de passagem homogênea, ${}^B T_{14}$, do pé direito em relação ao referencial de base,

$${}^B T_{14} = {}^B T_{CM} {}^{CM} T_8 {}^8 T_9 {}^9 T_{10} {}^{10} T_{11} {}^{11} T_{12} {}^{12} T_{13} {}^{13} T_{14}. \quad (3.1.4)$$

Durante a segunda metade do ciclo de cada passo, quando o pé direito está em contato com o solo, os referenciais são enumerados a partir do pé direito até a extremidade do pé esquerdo, passando pelo centro de massa, conforme

esquemático na Fig. 3.5. Neste caso, os parâmetros de Denavit-Hartenberg apresentados na Tab. 3.1 são os mesmos, exceto os parâmetros $\theta_5 = u_5 - \pi/2$ e $\theta_9 = u_8 + \pi/2$. Esse procedimento permite obter o mesmo equacionamento para todo o ciclo do movimento.

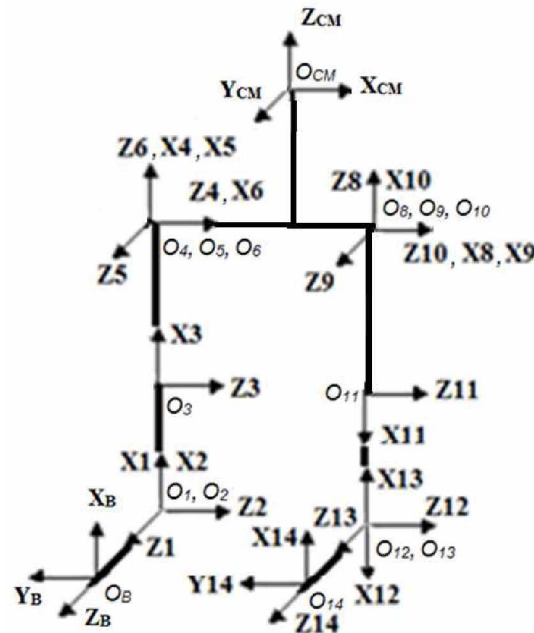


Figura 3.5. Referenciais para a situação em que o pé direito é o pé de apoio.

Para realizar as simulações do movimento do robô humanoide, foi criado um algoritmo em Matlab o qual recebe os parâmetros de Denavit-Hartenberg e os valores angulares das articulações como dados de entrada, obtendo-se como saída os deslocamentos de cada membro do corpo. O algoritmo é apresentado no Apêndice A.

Já a solução do modelo cinemático inverso é feita dividindo o robô humanoide em duas partes, cada uma com 6 graus de liberdade, facilitando a modelagem do movimento de cada perna separadamente. Isto é possível já que ambas apresentam a mesma estrutura, apenas alternando o movimento durante a caminhada. Feita a divisão do mecanismo em duas estruturas seriais 6R, o modelo inverso depende dos parâmetros de entrada, ou seja, das orientações e deslocamentos do centro de massa e de ambos os pés. A partir destes parâmetros, a solução do modelo inverso é obtida conforme apresentado no Capítulo IV.

Para visualizar os resultados das simulações do movimento do robô humanoide, um ciclo de passo foi discretizado inicialmente em 10 posturas, conforme apresentado na Fig. 3.6. Este mesmo movimento será posteriormente interpolado, conforme apresentado no Capítulo VI. Para computar as posições e orientações do

centro de massa e do pé em movimento, foram utilizadas as Eqs. (3.1.3) e (3.1.4), sendo possível também analisar suas trajetórias espaciais conforme apresentado no Capítulo V.

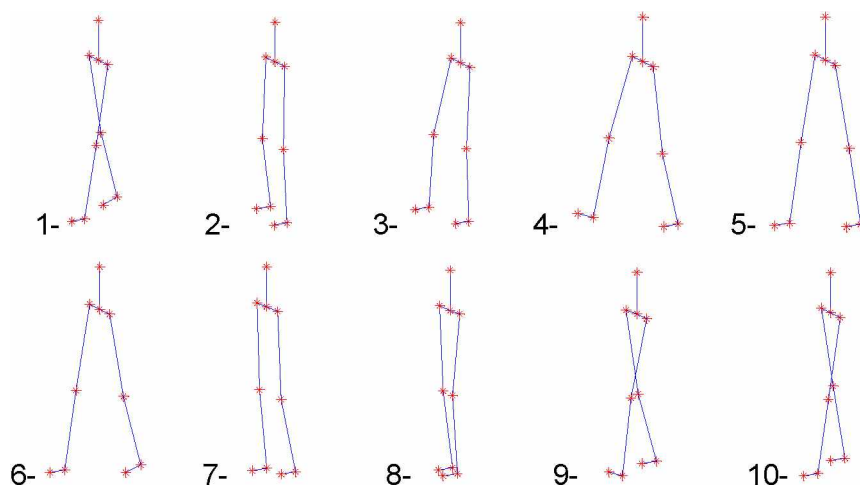


Figura 3.6. Simulação de dez posturas do robô humanoide ao longo de um ciclo de passo.

Após definido o modelo cinemático direto para o robô humanoide, o próximo passo é desenvolver o modelo cinemático inverso, assunto do próximo capítulo.

CAPÍTULO IV

MODELO CINEMÁTICO INVERSO

A cinemática inversa visa obter o movimento das articulações para que o elemento de interesse descreva uma trajetória previamente definida, tanto em posição quanto em orientação. Isso equivale a inverter as equações da cinemática direta. Como, em geral, essas equações são não lineares, nem sempre é uma tarefa fácil, além da necessidade de resolver o problema das singularidades. Em alguns casos, o problema de singularidade na parametrização do movimento pode ser resolvido pelo uso de outro tipo de coordenadas para descrever a trajetória desejada (posição e orientação).

O método proposto neste trabalho para a solução da cinemática inversa se baseia na geometria algébrica e dual quaternions, o que resulta em um sistema de equações polinomiais, ao invés de equações trigonométricas, geralmente obtidas, tais como em Han (2012). Este método tem sido utilizado na modelagem de vários tipos de estruturas seriais e paralelas tais como em Husty (1996), Husty *et al.* (2007, 2008), Walter e Husty (2009, 2010), Pfurner (2006), Caro *et al.* (2015) e Nurahmi *et al.* (2015).

Neste capítulo, inicialmente são apresentados alguns conceitos preliminares de geometria algébrica. Os problemas de singularidade de parametrização do movimento que surgem ao considerar os ângulos de Euler para definir a orientação são discutidos, assim como as vantagens em se aplicar dual quaternions.

Em seguida é descrito o Método de Mapeamento Cinemático, o qual representa a postura do mecanismo na forma de dual quaternions. A aplicação deste método ao mecanismo resulta em um sistema de equações algébricas, cuja solução leva à solução do problema inverso.

4.1. Singularidade na parametrização do movimento

De acordo com Kenwright (2012), existem diferentes maneiras de representar a orientação e posição de corpos rígidos no espaço. Em seu trabalho são comparadas

as vantagens e desvantagens das matrizes de passagem homogênea e as representações através de Eixo - Ângulo, ângulos de Euler e dual quaternions.

Os principais aspectos que diferenciam estes métodos são a possibilidade de interpolação, o custo computacional e a robustez matemática, sendo esta última referente à representação unívoca do movimento e à análise de continuidade.

Ângulos de Euler é a maneira mais utilizada nos campos de física e computação gráfica, pois são bem intuitivos, não necessitam normalização e representam a orientação do corpo rígido através de apenas 3 parâmetros. No entanto, existem 12 diferentes possibilidades de se representar a orientação, por consistir em 3 rotações sucessivas em torno dos 3 eixos coordenados. Ao utilizar os ângulos de Euler, ocorre o problema de singularidade de parametrização do movimento, denominado de “Gimbal Lock”.

Gimbal Lock ocorre quando a orientação de dois eixos ficam em uma mesma direção. Um exemplo que ilustra este problema é apresentado na Fig. 4.1.

Na Figura 4.1-a são indicadas as direções para representação da orientação através dos ângulos de Rolagem-Arfagem-Guinada, na qual a ordem de rotação é dada inicialmente em torno da direção Z do referencial inercial, e na sequência em torno dos eixos Y e depois X do referencial fixo ao corpo.

Na Figura 4.1-b os ângulos dos giros são nulos. Já nas Figuras 4.1-c e 4.1-d o sistema se encontra com ângulo de arfagem igual a 90° . Observa-se nesta condição que a direção dos ângulos de rolagem e guinada ficam coincidentes, isto significa que a representação perde um grau de liberdade. Na prática, por exemplo, qualquer sensor utilizado para mensurar o movimento teria problemas para indicar a orientação do sistema todas as vezes que o ângulo de arfagem se aproximar de 90° .

Este problema ocorre independente de qual das 12 sequências de rotações é escolhida para representar a orientação do sistema com ângulos de Euler, o que só pode ser resolvido utilizando um método diferente.

De acordo com Kenwright (2012) outros problemas apresentados por ângulos de Euler aparecem ao interpolar o movimento em condições próximas às de “Gimbal Lock”, o que também é discutido em Alan e Mart (1992).

O método Eixo - Ângulo (\vec{e}, ε) , representa a orientação de um corpo rígido através de um ângulo de giro, ε , em torno de uma direção indicada por um vetor unitário $\vec{e} = (e_x, e_y, e_z)$, necessitando assim de 4 números para sua representação. A desvantagem deste método é que a interpolação do movimento não pode ser feita simplesmente pela interpolação linear destes 4 elementos. A normalização dos elementos também pode gerar imprecisões numéricas no valor do ângulo de rotação. Além disso, ângulos múltiplos de 2π representam a mesma orientação.

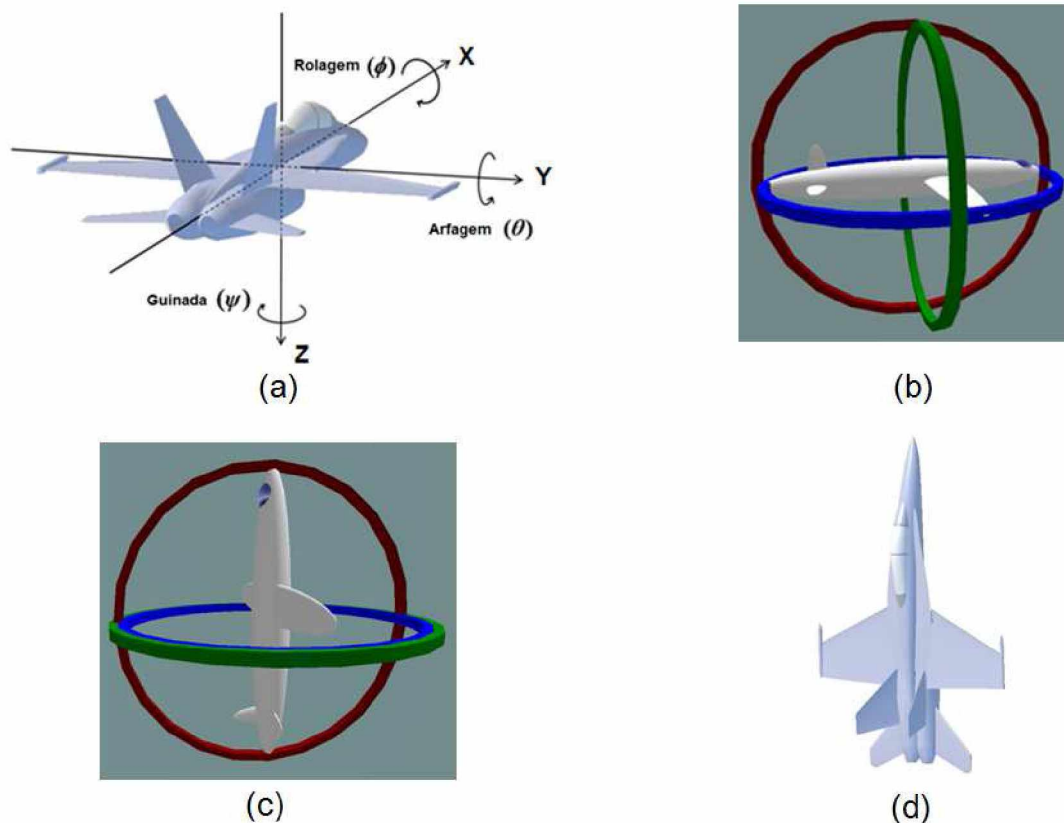


Figura 4.1. a) Ângulos de rolagem, arfagem e guinada; b) Condição em que os ângulos são nulos c-d) Condição de Gimbal Lock com arfagem de 90° . (CHRobotics, Disponível em: <<http://www.chrobotics.com/library/understanding-euler-angles>>, Acesso em: 30/09/2016).

Ainda segundo Kenwright (2012), as matrizes de passagem homogêneas são de simples implementação porém também apresentam algumas desvantagens. São necessários 12 elementos para indicar a orientação e translação de um corpo rígido. A parte rotacional, dada por uma sub-matriz ortogonal 3×3 , pode ser re-normalizada, mas a operação tem elevado custo computacional (GILBERT, 1986). As matrizes de passagem homogêneas também não podem ser diretamente interpoladas.

Em Schilling (2011), Varshney *et al.* (1998) e Kenwright (2012) são descritas as vantagens de utilizar dual quaternions para representar a posição e orientação de um corpo rígido. Dentre estas se destacam o baixo custo computacional e a simplicidade de implementação das operações. São necessários apenas 8 elementos para indicar tanto a orientação como a translação de um corpo rígido no espaço. Dual quaternions são livres de singularidade na parametrização do movimento de corpos rígidos e podem ser diretamente interpolados, como será mostrado no Capítulo VI. Em resumo,

segundo os autores, é a maneira mais eficiente de representar a mudança de referenciais para descrever o movimento de corpos rígidos.

Pode ser citado como desvantagem de dual quaternions a dificuldade de interpreta-los, isto é, é necessário alguma experiência em implementações e simulações para que se possa interpretar diretamente os resultados.

4.2. Geometria Algébrica

Nesta seção, os fundamentos de geometria algébrica aplicados à análise cinemática de corpos rígidos são apresentados através de um exemplo, adaptado a partir de Wampler e Sommese (2011). O problema consiste em modelar o movimento da estrutura serial 2R plana apresentada na Fig. 4.2.

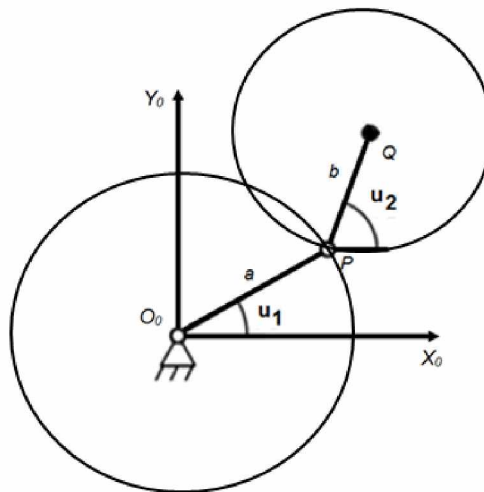


Figura 4.2. Estrutura serial 2R plana.

A posição do elemento terminal é dada pelas coordenadas do ponto Q em relação ao referencial $O_0X_0Y_0$,

$$Q = (Q_x, Q_y), \quad (4.2.1)$$

sendo,

$$\begin{bmatrix} Q_x \\ Q_y \end{bmatrix} = \begin{bmatrix} a \cos(u_1) + b \cos(u_2) \\ a \sin(u_1) + b \sin(u_2) \end{bmatrix}. \quad (4.2.2)$$

Já a posição do ponto P é dada por

$$\begin{bmatrix} P_x \\ P_y \end{bmatrix} = \begin{bmatrix} a \cos(u_1) \\ a \sin(u_1) \end{bmatrix}. \quad (4.2.3)$$

Para este problema, a solução do modelo cinemático direto é única, ou seja, para cada conjunto de ângulos u_1 e u_2 , a posição do elemento terminal Q é univocamente determinada pela Eq. (4.2.2).

O problema cinemático inverso consiste em calcular os valores de u_1 e u_2 quando a posição do elemento terminal, (Q_x, Q_y) , é conhecida.

Um primeiro método possível para se obter os valores de u_1 e u_2 é dado pela utilização da Lei dos Cossenos para o triângulo OPQ. Desta forma, os valores de u_1 e u_2 são obtidos através de relações trigonométricas.

Uma outra maneira é converter a Eq. (4.2.2) em um sistema de equações polinomiais, onde as equações trigonométricas podem ser convertidas em equações polinomiais a partir da introdução das variáveis auxiliares, por exemplo, fazendo as seguintes substituições:

$$\begin{aligned} x_1 &= \cos(u_1), & y_1 &= \sin(u_1), \\ x_2 &= \cos(u_2), & y_2 &= \sin(u_2), \end{aligned} \quad (4.2.4)$$

e considerando as identidades trigonométricas,

$$\begin{aligned} \cos^2(u_1) + \sin^2(u_1) &= 1, \\ \cos^2(u_2) + \sin^2(u_2) &= 1. \end{aligned} \quad (4.2.5)$$

Após substituir a Eq. (4.2.4) na Eq. (4.2.2) e juntamente com a Eq. (4.2.5), pode-se escrever:

$$\begin{bmatrix} ax_1 + bx_2 - Q_x \\ ay_1 + by_2 - Q_y \\ x_1^2 + y_1^2 - 1 \\ x_2^2 + y_2^2 - 1 \end{bmatrix} = \mathbf{0}. \quad (4.2.6)$$

onde $\mathbf{0}$ representa o vetor nulo $(0,0,0,0)^T$.

A solução do problema cinemático inverso é obtida pela solução do sistema da Eq. (4.2.6), que possui 4 incógnitas (x_1, x_2, y_1, y_2) e 4 equações, sendo 2 lineares e 2 quadráticas. Em seguida, a partir das Eq. (4.2.4), obtém-se os valores de u_1 e u_2 . Note

que, nesta abordagem, o número de incógnitas duplica, passando de 2 (u_1, u_2) para 4 (x_1, x_2, y_1, y_2).

Uma outra abordagem para estudar o movimento da estrutura serial 2R se refere à interpretação geométrica de seu espaço de trabalho, sendo possível utilizar geometria algébrica para obter as equações polinomiais que representam o mecanismo. Isto é feito a partir da localização do ponto P por meio das distâncias conhecidas $a = \overline{OP}$ e $b = \overline{PQ}$, além das coordenadas dos pontos O e Q também conhecidas. O ponto P pertence à intersecção de dois círculos, sendo o primeiro de raio a e centro em O , e o segundo de raio b e centro em Q .

Sendo a posição do ponto P dada pelas coordenadas P_x e P_y , as equações polinomiais resultantes podem ser obtidas a partir das equações dos círculos, ou seja:

$$\left[\begin{array}{c} P_x^2 + P_y^2 - a^2 \\ (P_x - Q_x)^2 + (P_y - Q_y)^2 - b^2 \end{array} \right] = 0. \quad (4.2.7)$$

Pela interpretação geométrica do sistema de equações, Eq. (4.2.7), são obtidas duas soluções, o que é esperado da intersecção de dois círculos.

Esta última abordagem ilustra como representar mecanismos utilizando um sistema de equações polinomiais. Walter e Husty (2010) descrevem as vantagens da utilização das abordagens geométrica e algébrica. De acordo com os autores, a representação polinomial nem sempre é utilizada devido à complexidade envolvida em obtê-las. Esta complexidade é devido à grande quantidade de parâmetros que são necessários para descrever curvas e superfícies no espaço.

Apesar de mais complexos, os métodos algébricos são amplamente utilizados em computação gráfica, como descrito em Cox *et al.* (2007), onde são apresentados alguns algoritmos para a solução de problemas de CAD (*Computer Aided Design*).

Husty (1996) apresenta um algoritmo baseado em geometria algébrica para resolver polinomialmente o problema de cinemática direta de plataformas de Stewart-Gough. Métodos algébricos também são aplicados por Husty e Karger (2000, 2002) para resolver o problema de *Self-Motion* de plataformas de Stewart-Gough. Também é apresentado por Husty *et al.* (2007) uma metodologia baseada em geometria algébrica para síntese de mecanismos planares de quatro barras, esféricos e espaciais. A estrutura do tipo 3-UPU é avaliada por geometria algébrica em Walter *et al.* (2009).

De acordo com Walter e Husty (2010), métodos fundamentados em Bases de Gröbner tem suas limitações para solução de problemas que envolvem equações mais

complexas, quando os polinômios são de ordem elevada ou com um grande número de variáveis. A abordagem por eles proposta resulta em um algoritmo que pode ser aplicado a diferentes tipos de mecanismos resultando em soluções globais, ao invés de soluções locais como obtidas pela Teoria de Helicoides.

Com o algoritmo proposto em Walter e Husty (2010), é possível obter equações lineares, quadráticas, cúbicas ou quárticas para descrever a cinemática dos mecanismos, generalizando a abordagem de problemas cinemáticos através de geometria algébrica.

Com base nos trabalhos de Husty, Pfurner (2006) solucionou o problema da cinemática inversa para estruturas seriais 6R, sendo o algoritmo por ele proposto utilizado nesta tese para solucionar a cinemática inversa de robôs humanoides.

4.3. Quaternions

Para a solução do problema de cinemática inversa de robôs humanoides através de geometria algébrica, é necessário a aplicação de dual quaternions para representar o movimento. Antes de definir os conceitos de dual quaternions, é descrito na sequência a álgebra de quaternions.

De acordo com Pfurner (2006), W. R. Hamilton foi o primeiro a introduzir os quaternions em 1840, porém, os elementos básicos da álgebra de quaternions podem ser encontradas em Euler (1776). Hamilton demonstrou que a posição relativa de dois vetores no espaço tridimensional pode ser descrita por 4 parâmetros. De maneira similar à representação planar por números complexos, o caso espacial é feito através de quaternions.

Seja $P = (p_0, p_1, p_2, p_3)$ um quaternion no qual cada elemento pertence ao conjunto dos números reais, p_0 é a parte escalar e

$$\mathbf{p} = (p_1, p_2, p_3), \quad (4.3.1)$$

a parte vetorial do quaternion P , respectivamente.

Seja $Q = (q_0, q_1, q_2, q_3)$ um segundo quaternion. A multiplicação entre os quaternions P e Q é definida como

$$P * Q = (p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3, p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2, p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1, p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0). \quad (4.3.2)$$

A regra de multiplicação utilizada na Eq. (4.3.2) pode ser entendida escrevendo os quaternions P e Q como

$$\begin{aligned} P &= p_0 \mathbf{1} + p_1 \mathbf{i} + p_2 \mathbf{j} + p_3 \mathbf{k} = p_0 + \mathbf{p}, \\ Q &= q_0 \mathbf{1} + q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k} = q_0 + \mathbf{q}, \end{aligned} \quad (4.3.3)$$

e aplicando-se as relações:

$$\begin{aligned} \mathbf{1} * \mathbf{1} &= \mathbf{1}, \quad \mathbf{1} * \mathbf{i} = \mathbf{i}, \quad \mathbf{1} * \mathbf{j} = \mathbf{j}, \quad \mathbf{1} * \mathbf{k} = \mathbf{k}, \\ \mathbf{i} * \mathbf{1} &= \mathbf{i}, \quad \mathbf{i} * \mathbf{i} = -\mathbf{1}, \quad \mathbf{i} * \mathbf{j} = \mathbf{k}, \quad \mathbf{i} * \mathbf{k} = -\mathbf{j}, \\ \mathbf{j} * \mathbf{1} &= \mathbf{j}, \quad \mathbf{j} * \mathbf{i} = -\mathbf{k}, \quad \mathbf{j} * \mathbf{j} = -\mathbf{1}, \quad \mathbf{j} * \mathbf{k} = \mathbf{i}, \\ \mathbf{k} * \mathbf{1} &= \mathbf{k}, \quad \mathbf{k} * \mathbf{i} = \mathbf{j}, \quad \mathbf{k} * \mathbf{j} = -\mathbf{i}, \quad \mathbf{k} * \mathbf{k} = -\mathbf{1}. \end{aligned}$$

A multiplicação de um escalar real X por um quaternion P , pode ser feita da mesma maneira como apresentado pela Eq. (4.3.2), considerando X como um quaternion cuja parte vetorial é nula, ou seja, $X = (X, 0, 0, 0)$. A multiplicação de quaternions é uma operação associativa, porém não é comutativa.

O quaternion conjugado de P é definido por

$$\tilde{P} = p_0 \mathbf{1} - p_1 \mathbf{i} - p_2 \mathbf{j} - p_3 \mathbf{k} = p_0 - \mathbf{p}. \quad (4.3.4)$$

A norma do quaternion P é definida por

$$\|P\| = \sqrt{P * \tilde{P}} = \sqrt{p_0^2 + p_1^2 + p_2^2 + p_3^2}, \quad (4.3.5)$$

e um quaternion é dito normalizado se $\|P\| = 1$. A inversa de P é definida por

$$P^{-1} = \frac{\tilde{P}}{\|P\|}, \quad (4.3.6)$$

de forma que a inversa de um quaternion normalizado é igual ao seu conjugado.

Um quaternion é dito vetorial se sua parte escalar é nula. O produto de dois quaternions vetoriais $P = p_1 \mathbf{i} + p_2 \mathbf{j} + p_3 \mathbf{k}$ e $Q = q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}$ é dado por

$$\begin{aligned}
P * Q &= (p_1 \mathbf{i} + p_2 \mathbf{j} + p_3 \mathbf{k}) * (q_1 \mathbf{i} + q_2 \mathbf{j} + q_3 \mathbf{k}) \\
&= - (p_1 q_1 + p_2 q_2 + p_3 q_3) \mathbf{1} + (p_2 q_3 - p_3 q_2) \mathbf{i} - (p_1 q_3 - p_3 q_1) \mathbf{j} + (p_1 q_2 - p_2 q_1) \mathbf{k} \\
&= - \mathbf{pq} + \mathbf{p} \times \mathbf{q},
\end{aligned} \tag{4.3.7}$$

o qual não é um quaternion vetorial, a menos que \mathbf{p} seja ortogonal à \mathbf{q} . Para o caso em que \mathbf{p} é ortogonal à \mathbf{q} , o produto entre P e Q representa geometricamente uma rotação de \mathbf{q} em torno de \mathbf{p} , com uma mudança de comprimento de \mathbf{q} . Caso P seja um quaternion vetorial normalizado e \mathbf{p} é ortogonal à \mathbf{q} , então o produto $P * Q$ descreve uma rotação de \mathbf{q} em torno de \mathbf{p} , sem alteração de comprimento. Todo quaternion normalizado pode ser escrito como

$$P = \cos(\theta) + p_1 \sin(\theta) \mathbf{i} + p_2 \sin(\theta) \mathbf{j} + p_3 \sin(\theta) \mathbf{k}, \tag{4.3.8}$$

no qual $\mathbf{p}=(p_1, p_2, p_3)$ é um vetor unitário e θ define um ângulo de giro em torno de \mathbf{p} . O produto deste quaternion normalizado por um quaternion vetorial Q , cuja parte vetorial \mathbf{q} é ortogonal a \mathbf{p} , é dado por

$$P * \mathbf{q} = \sin(\theta) (\mathbf{p} \times \mathbf{q}) + \cos(\theta) \mathbf{q}. \tag{4.3.9}$$

A Equação (4.3.9) descreve uma rotação do vetor \mathbf{q} em torno de \mathbf{p} com um ângulo θ . No entanto as rotações descritas são limitadas pela condição de ortogonalidade do vetor com o eixo de rotação.

A rotação de um vetor \mathbf{q} em torno de um eixo arbitrário dado pela direção de \mathbf{p} , não limitados pela condição de ortogonalidade, é definida por

$$\text{Ad } P(\mathbf{q}) = P * \mathbf{q} * \tilde{P} = P * \mathbf{q} * P^{-1}. \tag{4.3.10}$$

A operação $\text{Ad } P(\mathbf{q})$ indica uma rotação de um quaternion vetorial \mathbf{q} em torno da direção de \mathbf{p} com um ângulo θ , sendo $P = \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right) \mathbf{p}$ um quaternion normalizado, com \mathbf{p} não necessariamente ortogonal a \mathbf{q} .

É possível, a partir dessas operações, generalizar as rotações representando-as por uma matriz de transformação ortogonal. Isto é feito aplicando o operador $\text{Ad } X$ aos vetores unitários \mathbf{i} , \mathbf{j} e \mathbf{k} , que representam os eixos coordenados de um referencial j , com o quaternion $X = x_0 \mathbf{1} + x_1 \mathbf{i} + x_2 \mathbf{j} + x_3 \mathbf{k}$, o que resulta em

$$\begin{aligned}
X * i * \tilde{X} &= (x_0 \mathbf{1} + x_1 \mathbf{i} + x_2 \mathbf{j} + x_3 \mathbf{k}) \mathbf{i} (x_0 \mathbf{1} - x_1 \mathbf{i} - x_2 \mathbf{j} - x_3 \mathbf{k}) \\
&= (x_0 \mathbf{i} - x_1 \mathbf{1} - x_2 \mathbf{k} + x_3 \mathbf{j}) (x_0 \mathbf{1} - x_1 \mathbf{i} - x_2 \mathbf{j} - x_3 \mathbf{k}) \\
&= (x_0^2 + x_1^2 - x_2^2 - x_3^2) \mathbf{i} + 2(x_1 x_2 + x_0 x_3) \mathbf{j} + 2(x_1 x_3 - x_0 x_2) \mathbf{k}, \\
X * j * \tilde{X} &= 2(x_1 x_2 - x_0 x_3) \mathbf{i} + (x_0^2 - x_1^2 + x_2^2 - x_3^2) \mathbf{j} + 2(x_2 x_3 + x_0 x_1) \mathbf{k}, \\
X * k * \tilde{X} &= 2(x_1 x_3 + x_0 x_2) \mathbf{i} + 2(x_2 x_3 - x_0 x_1) \mathbf{j} + (x_0^2 - x_1^2 - x_2^2 + x_3^2) \mathbf{k}.
\end{aligned} \tag{4.3.11}$$

$$Ad X = \begin{pmatrix} x_0^2 + x_1^2 - x_2^2 - x_3^2 & 2(x_1 x_2 - x_0 x_3) & 2(x_1 x_3 + x_0 x_2) \\ 2(x_1 x_2 + x_0 x_3) & x_0^2 - x_1^2 + x_2^2 - x_3^2 & 2(x_2 x_3 - x_0 x_1) \\ 2(x_1 x_3 - x_0 x_2) & 2(x_2 x_3 + x_0 x_1) & x_0^2 - x_1^2 - x_2^2 + x_3^2 \end{pmatrix}. \tag{4.3.12}$$

A matriz apresentada na Eq. (4.3.12) corresponde à matriz ${}^i R_{i+1}$, que representa todas as rotações do referencial $i + 1$ em relação ao referencial i , ou seja, a matriz de passagem de um referencial i para o referencial $i + 1$, e seus elementos também são conhecidos como parâmetros de Euler. Desta forma, é possível obter uma relação direta entre as representações de rotações por quaternions e pelos parâmetros de Euler. Com o objetivo de representar as translações junto com a orientação do corpo rígido, torna-se necessário o conceito de dual quaternions.

4.4. Dual quaternions

De acordo com Pfurner (2006), a álgebra dos dual quaternions é a mais importante álgebra para análises cinemáticas por fornecer uma representação do movimento de corpos rígidos sem singularidades de parametrização. Nesta seção são apresentadas as principais definições relativas aos dual quaternions.

Um dual quaternion pode ser escrito como

$$\begin{aligned}
P &= X + \varepsilon Y \\
&= (x_0 + \varepsilon y_0) \mathbf{1} + (x_1 + \varepsilon y_1) \mathbf{i} + (x_2 + \varepsilon y_2) \mathbf{j} + (x_3 + \varepsilon y_3) \mathbf{k} \\
&= \mathbf{a} + \mathbf{p},
\end{aligned} \tag{4.4.1}$$

no qual os coeficientes $X = (x_0, x_1, x_2, x_3)$ e $Y = (y_0, y_1, y_2, y_3)$ são quaternions e ε é um valor infinitesimal ($\varepsilon^2=0$), o que resulta em um vetor representativo para os dual quaternions com 8 elementos.

Na Eq. (4.4.1) tem-se ainda que $\mathbf{a} = [(x_0 + \varepsilon y_0) \mathbf{1}]$ e $\mathbf{p} = \mathbf{p}_1 + \varepsilon \mathbf{p}_2$, sendo $\mathbf{p}_1 = (x_1, x_2, x_3)$, $\mathbf{p}_2 = (y_1, y_2, y_3)$.

O produto dos dual quaternions $P = \mathbf{a} + \mathbf{p}$ e $Q = \mathbf{b} + \mathbf{q}$ é definido como,

$$P * Q = \mathbf{ab} - \mathbf{pq} + \mathbf{aq} + \mathbf{bp} + \mathbf{p} \times \mathbf{q}. \quad (4.4.2)$$

Existem dois tipos de relações de conjugação para dual quaternions. O conjugado biquaternion é definido como

$$\tilde{P} = [(x_0 + \varepsilon y_0) \mathbf{1}] - \mathbf{p}_1 - \varepsilon \mathbf{p}_2, \quad (4.4.3)$$

e o conjugado biquaternion dual é definido como

$$P_\varepsilon = [(x_0 - \varepsilon y_0) \mathbf{1}] + \mathbf{p}_1 - \varepsilon \mathbf{p}_2. \quad (4.4.4)$$

A norma de um dual quaternion é definida como

$$\|P\| = \sqrt{P \tilde{P}}. \quad (4.4.5)$$

O produto de um dual quaternion pelo seu conjugado biquaternion é dado por

$$\begin{aligned} P * \tilde{P} &= \mathbf{a}^2 + \mathbf{pp} \\ P * \tilde{P} &= x_0^2 + 2\varepsilon x_0 y_0 + \mathbf{p}_1 \mathbf{p}_1 + 2\varepsilon \mathbf{p}_1 \mathbf{p}_2 \\ P * \tilde{P} &= x_0^2 + \mathbf{p}_1 \mathbf{p}_1 + 2\varepsilon (x_0 y_0 + \mathbf{p}_1 \mathbf{p}_2) \\ P * \tilde{P} &= x_0^2 + x_1^2 + x_2^2 + x_3^2 + 2\varepsilon (x_0 y_0 + x_1 y_1 + x_2 y_2 + x_3 y_3). \end{aligned} \quad (4.4.6)$$

Um dual quaternion é dito normalizado ou unitário se $\|P\| = 1$. Para que isso ocorra, a partir da Eq. (4.4.6), obtém-se as relações,

$$\begin{aligned} x_0^2 + x_1^2 + x_2^2 + x_3^2 &= 1, \\ x_0 y_0 + x_1 y_1 + x_2 y_2 + x_3 y_3 &= 0. \end{aligned} \quad (4.4.7)$$

De acordo com Kenwright (2013), a transformação homogênea de um corpo rígido no espaço pode ser representada por dual quaternions unitários de acordo com a seguinte notação:

$$P = q_r + \varepsilon \frac{1}{2} q_r q_t \quad (\text{Rotação seguida de translação}) \quad (4.4.8)$$

ou

$$P = q_r + \varepsilon \frac{1}{2} q_t q_r \quad (\text{Translação seguida de rotação}). \quad (4.4.9)$$

Sendo os termos

$$q_r = \left(\cos\left(\frac{\theta}{2}\right), n_x \cdot \sin\left(\frac{\theta}{2}\right), n_y \cdot \sin\left(\frac{\theta}{2}\right), n_z \cdot \sin\left(\frac{\theta}{2}\right) \right), \quad (4.4.10)$$

$$q_t = (0, t_x, t_y, t_z), \quad (4.4.11)$$

os quaternions de rotação e translação, respectivamente.

O termo $q_r q_t$ é obtido pela multiplicação de ambos os quaternions. Tem-se ainda que (n_x, n_y, n_z) é um vetor unitário que representa a direção do eixo de rotação e θ o ângulo de rotação.

Os dual quaternions podem representar uma rotação pura da mesma maneira que os quaternions. Neste caso,

$$q_t = (0, 0, 0, 0). \quad (4.4.12)$$

Para representar uma translação pura, tem-se que

$$q_r = (1, 0, 0, 0). \quad (4.4.13)$$

De acordo com Pfurner (2006), é possível obter a matriz de passagem homogênea a partir dos termos de um dual quaternion. Para isso, é definido um operador $\text{Ad}_e P$, aplicado a um dual quaternion R de forma a representar tanto a rotação quanto a translação de um corpo rígido.

Considerando $R = 1 + \varepsilon \mathbf{r}$ um dual quaternion, o operador $\text{Ad}_e P$ é definido por,

$$\text{Ad}_e P(R) = P_\varepsilon * R * \tilde{P}, \quad (4.4.14)$$

sendo P_ε dado pela Eq. (4.4.4) e \tilde{P} , pela Eq. (4.4.3), resultando em:

$$\begin{aligned}
\text{Ad}_e P(R) &= (X - \varepsilon Y) * (1 + \varepsilon \mathbf{r}) * (\tilde{X} + \varepsilon \tilde{Y}) \\
&= 1 + \varepsilon (X * \mathbf{r} * \tilde{X} + X * \tilde{Y} - Y * \tilde{X}),
\end{aligned} \tag{4.4.15}$$

salientando que $X * \tilde{X} = 1$ pois X é normalizado.

Pela análise da Eq. (4.4.15) pode-se verificar que o termo $X * \tilde{Y} - Y * \tilde{X}$ representa a parte de translação do operador e o termo $X * \mathbf{r} * \tilde{X}$ representa a parte rotacional, como visto nas Eqs. (4.3.11) e (4.3.12).

A parte de translação pode ser desenvolvida como,

$$\begin{aligned}
X * \tilde{Y} - Y * \tilde{X} &= 2(-x_0 y_1 + x_1 y_0 - x_2 y_3 + x_3 y_2) \mathbf{i} + 2(-x_0 y_2 + x_1 y_3 + x_2 y_0 - x_3 y_1) \mathbf{j} + \\
&+ 2(-x_0 y_3 - x_1 y_2 + x_2 y_1 + x_3 y_0) \mathbf{k}.
\end{aligned} \tag{4.4.16}$$

resultando assim nas 3 componentes em \mathbf{i} , \mathbf{j} e \mathbf{k} dos termos de translação da matriz de passagem homogênea do referencial i ao referencial $i + 1$ que, combinado com a Eq. (4.3.12), resulta em,

$${}^i A_{i+1} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ l & x_0^2 + x_1^2 - x_2^2 - x_3^2 & 2(x_1 x_2 - x_0 x_3) & 2(x_1 x_3 + x_0 x_2) \\ m & 2(x_1 x_2 + x_0 x_3) & x_0^2 - x_1^2 + x_2^2 - x_3^2 & 2(x_2 x_3 - x_0 x_1) \\ n & 2(x_1 x_3 - x_0 x_2) & 2(x_2 x_3 + x_0 x_1) & x_0^2 - x_1^2 - x_2^2 + x_3^2 \end{pmatrix}, \tag{4.4.17}$$

na qual

$$\begin{aligned}
l &= 2(-x_0 y_1 + x_1 y_0 - x_2 y_3 + x_3 y_2), \\
m &= 2(-x_0 y_2 + x_1 y_3 + x_2 y_0 - x_3 y_1), \\
n &= 2(-x_0 y_3 - x_1 y_2 + x_2 y_1 + x_3 y_0).
\end{aligned} \tag{4.4.18}$$

e pela condição de normalização tem-se que,

$$x_0^2 + x_1^2 + x_2^2 + x_3^2 = 1.$$

A Equação (4.4.17) representa um mapeamento entre os parâmetros de um dual quaternion para a matriz de passagem homogênea correspondente.

4.5. Mapeamento Cinemático

Como foi visto na seção anterior, a orientação e translação de um corpo rígido no espaço podem ser representadas através de dual quaternions. Nesta seção é apresentado o mapeamento de Study, utilizado para obter os elementos do dual quaternion, também chamados de parâmetros de Study.

Para obter os parâmetros de Study é necessário o mapeamento inverso ao apresentado nas Eq. (4.4.17) e (4.4.18). Isto significa que é necessário calcular o dual quaternion a partir dos elementos da matriz de passagem homogênea. Em Golub e Van Loan (1996) é apresentado o Método de Cayley para este mapeamento, porém trata-se de um método que não é livre de singularidades. Diferente do Método de Cayley, é apresentado aqui o método proposto por Study (1891, 1903), o qual é livre de singularidades. Esses trabalhos de Study podem ser encontrados apenas em Alemão, os quais segundo Husty e Schröcker (2010), são trabalhos baseados em uma literatura complexa e de difícil entendimento. A mesma metodologia é então traduzida e discutida em Husty *et al.* (1997) e Husty e Schröcker (2010), nos quais é apresentada a descrição do mapeamento de Study utilizada neste trabalho. Mais referências podem ser encontradas em Selig (2005), Bottema e Roth (1979) e McCarthy (2000).

De acordo com Husty e Schröcker (2010), o método de Study permite mapear um ponto no espaço projetivo de dimensão igual a sete a partir dos 12 elementos da matriz de passagem homogênea. Isto quer dizer que são calculados os 8 elementos do dual quaternion satisfazendo as condições de normalização

$$x_0^2 + x_1^2 + x_2^2 + x_3^2 = 1, \quad (4.5.1)$$

$$x_0 y_0 + x_1 y_1 + x_2 y_2 + x_3 y_3 = 0. \quad (4.5.2)$$

que foram introduzidos na Eq. (4.4.7).

No caso das matrizes de transformação homogênea, a sub-matriz jR_i de dimensão 3x3, que define a orientação do referencial i em relação ao referencial j , necessariamente deve respeitar a condição de ortogonalidade $({}^jR_i)^T \cdot {}^jR_i = I_3$, em que I_3 é a matriz identidade 3x3. Da mesma forma, para os dual quaternions a condição de normalização deve ser respeitada.

Desta forma diz-se que cada ponto do espaço projetivo de dimensão igual a sete, mapeado pelo método de Study, pertence a quádrlica de Study definida pela Eq. (4.5.2). Isto quer dizer que os elementos do dual quaternion mapeado devem sempre

satisfazer as Eqs. (4.5.1) e (4.5.2) de maneira a representar a orientação de um corpo rígido no espaço. Observa-se portanto que os elementos x_0, x_1, x_2 e x_3 não podem ser todos iguais a zero, de maneira que seja satisfeita a Eq. (4.5.1), dada pela condição de normalização.

Um estudo da quádrlica de Study é apresentado em Selig (2005), e uma maior descrição da interpretação geométrica de espaços projetivos pode ser encontrada em Cox *et al.* (2007).

De maneira a satisfazer as condições descritas, o mapeamento de Study é feito segundo as Eqs. (4.5.3) e (4.5.4). Observa-se que a quádrupla (x_0, x_1, x_2, x_3) pode ser calculada por qualquer uma das 4 maneiras apresentadas. Caso, por exemplo, utilizando-se as primeiras equações, verifica-se a igualdade $1+a_{11}+a_{22}+a_{33}=0$, o resultado da quádrupla levaria a $(0, 0, 0, 0)$, o que é inválido. Neste caso deve-se utilizar as equações seguintes, a menos que $a_{22}+a_{33}=0$ não seja verificado. Caso contrário, utilizam-se as equações seguintes verificando se $a_{11}+a_{33}=0$. Se ambos os casos acontecerem, as últimas equações garantem que o mapeamento de Study seja feito satisfazendo as Eqs. (4.5.1) e (4.5.2). As relações apresentadas na Eq. (4.5.3) são obtidas a partir dos termos da matriz de rotação. Sendo conhecidos (x_0, x_1, x_2, x_3) , a quádrupla (y_0, y_1, y_2, y_3) é obtida a partir da Eq. (4.5.4).

$$\begin{aligned}
 x_0, x_1, x_2, x_3 &= 1 + a_{11} + a_{22} + a_{33}, a_{32} - a_{23}, a_{13} - a_{31}, a_{21} - a_{12} \\
 &= a_{32} - a_{23}, 1 + a_{11} - a_{22} - a_{33}, a_{12} + a_{21}, a_{31} + a_{13} \\
 &= a_{13} - a_{31}, a_{12} + a_{21}, 1 - a_{11} + a_{22} - a_{33}, a_{23} + a_{32} \\
 &= a_{21} - a_{12}, a_{31} + a_{13}, a_{23} + a_{32}, 1 - a_{11} - a_{22} + a_{33}.
 \end{aligned} \tag{4.5.3}$$

$$\begin{aligned}
 2y_0 &= a_{10}x_1 + a_{20}x_2 + a_{30}x_3, \\
 2y_1 &= -a_{10}x_0 + a_{30}x_2 - a_{20}x_3, \\
 2y_2 &= -a_{20}x_0 - a_{30}x_1 + a_{10}x_3, \\
 2y_3 &= -a_{30}x_0 + a_{20}x_1 - a_{10}x_2,
 \end{aligned} \tag{4.5.4}$$

no qual os elementos do dual quaternion $X = (x_0, x_1, x_2, x_3, y_0, y_1, y_2, y_3)$ são obtidos a partir da matriz de passagem homogênea

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{bmatrix}, \quad (4.5.5)$$

cujos elementos estão organizados conforme nomenclatura apresentada na Seção 3.1.

Para ilustrar o uso do mapeamento de Study, vamos considerar a matriz de transformação homogênea dada pela Eq. (4.5.6) que representa somente uma rotação u em torno do eixo coordenado Z_i :

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(u) & -\sin(u) & 0 \\ 0 & \sin(u) & \cos(u) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4.5.6)$$

A partir da Eq. (4.5.3), obtém-se a quádrupla (x_0, x_1, x_2, x_3) . Como não existe translação, a quádrupla (y_0, y_1, y_2, y_3) possui todos os elementos nulos. Logo, o dual quaternion $X = (x_0, x_1, x_2, x_3, y_0, y_1, y_2, y_3)$ é dado por

$$X = (1+\cos(u), 0, 0, \sin(u), 0, 0, 0, 0). \quad (4.5.7)$$

Visto que o robô humanoide, objeto desta tese, possui somente articulações de rotação, a seguir são analisados as equações para estruturas seriais que possuem somente articulações rotativas.

4.6. Equações de uma estrutura serial canônica

4.6.1. Estrutura serial 2R canônica

Obtidos os 8 parâmetros do dual quaternion através do Mapeamento Cinemático pelos Parâmetros de Study, é necessário obter as equações algébricas que descrevem a estrutura serial para resolver o problema de cinemática inversa. Pfulner (2006) define estas equações como hiperplanos que representam as superfícies de restrição do movimento de um mecanismo.

Em Pfuner (2006) é demonstrado que cada postura do mecanismo pode ser mapeada para um dual quaternion correspondente, o qual pode ser interpretado geometricamente como um ponto no espaço projetivo de sétima dimensão. Este mesmo ponto pode também ser interpretado como uma curva ou ainda superfícies, dependendo do espaço projetivo considerado. Desta maneira é possível obter equações algébricas analíticas que definem as superfícies. Estas equações são relacionadas à mobilidade do mecanismo, pois trata de uma representação do seu espaço de trabalho em um espaço projetivo de alta dimensão.

Com o objetivo de ilustrar a obtenção de um destes sistemas de equações algébricas que representam um mecanismo, é apresentado na sequência a análise de uma estrutura serial 2R canônica.

Uma estrutura serial é dita canônica quando o eixo coordenado Z_1 do referencial da primeira articulação coincide com o eixo Z_0 do referencial de base e o parâmetro de Denavit-Hartenberg d_1 seja nulo. Isso significa que $\Sigma_1 \equiv \Sigma_0$.

Selig (2005) e Pfuner (2006) mostram que a análise de uma estrutura serial 2R canônica pode ser feita a partir de uma estrutura 3R pelo travamento de uma articulação, transformando-a em uma estrutura 2R.

Como exemplo, visto que a perna do robô será dividida em duas estruturas 3R, uma representando o quadril e outra o tornozelo e joelho, vamos considerar a estrutura serial 2R canônica constituída pela articulação do joelho, representada pela variável articular u_3 , e a articulação anterior, pertencente ao tornozelo e definida por u_2 . Neste caso, u_1 é nula (pois foi considerada travada) e o referencial de base Σ_0 coincide com o referencial Σ_2 . Para permitir generalizar o procedimento, vamos considerar uma rotação α_2 entre os eixos Z_2 e Z_3 e os parâmetros de Denavit-Hartenberg de forma genérica. Para simplificar o exemplo, vamos considerar que o elemento terminal, ET, coincida com o centro da articulação 3, conforme Fig. 4.4.

Assim os parâmetros de Denavit-Hartenberg são apresentados na Tab. 4.1.

Tabela 4.1. Parâmetros de Denavit-Hartenberg de uma estrutura serial 2R canônica.

i	a_i	d_i	α_i	θ_i
0	0	0	0	0
2	a_2	0	α_2	u_2
3	0	0	0	u_3

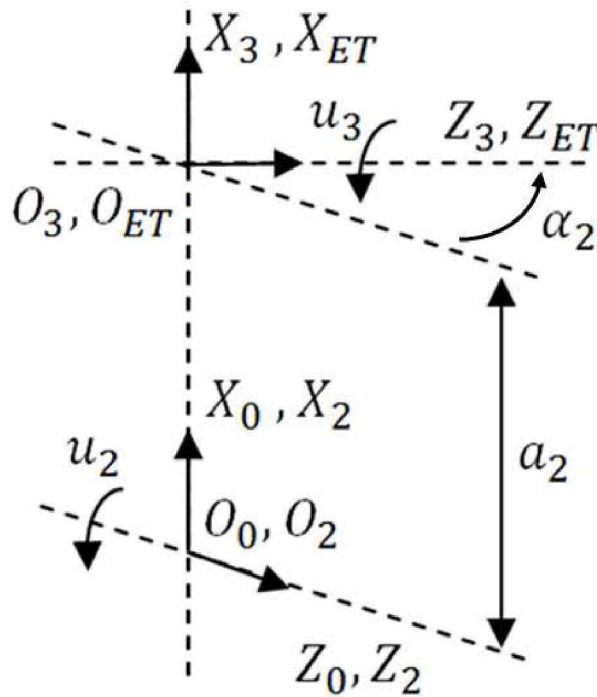


Figura 4.4. Estrutura serial 2R canônica.

A matriz de passagem homogênea do referencial de base ao elemento terminal (ET) é dada por

$${}^0T_{ET} = {}^2T_3 {}^3T_{ET}, \quad (4.6.1)$$

visto que a matriz 0T_2 é uma identidade de ordem 4. Pode-se verificar na Tab. 4.1 que o parâmetro d_2 é nulo, segundo a definição de uma estrutura canônica, visto que o primeiro referencial é o referencial relacionado à articulação 2.

A partir da matriz ${}^0T_{ET}$, é possível realizar o Mapeamento Cinemático obtendo os 8 elementos do dual quaternion correspondente. Neste exemplo, a Eq. (4.5.4) junto com qualquer uma das equações apresentadas na Eq. (4.5.3) podem ser usadas para o mapeamento, desde que o quaternion resultante não seja nulo.

Feito o Mapeamento Cinemático, o dual quaternion relativo à estrutura serial 2R canônica é dado pela Eq. (4.6.2):

$$\begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \\ y_0 \\ y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} (\cos(u_2)\cos(u_3) - \sin(u_2)\sin(u_3) + 1)(1 + \cos(\alpha_2)) \\ (\cos(u_2) + \cos(u_3))\sin(\alpha_2) \\ (\sin(u_2) - \sin(u_3))\sin(\alpha_2) \\ (\cos(u_2)\sin(u_3) + \sin(u_2)\cos(u_3))(1 + \cos(\alpha_2)) \\ \frac{1}{2}a_2(\cos(u_2)\cos(u_3) - \sin(u_2)\sin(u_3) + 1)\sin(\alpha_2) \\ -\frac{1}{2}a_2(\cos(u_2) + \cos(u_3))(1 + \cos(\alpha_2)) \\ -\frac{1}{2}a_2(\sin(u_2) - \sin(u_3))(1 + \cos(\alpha_2)) \\ \frac{1}{2}a_2(\cos(u_2)\sin(u_3) + \sin(u_2)\cos(u_3))\sin(\alpha_2) \end{pmatrix}. \quad (4.6.2)$$

A partir de uma avaliação direta e inspeção dos elementos do dual quaternion, Eq. (4.6.2), é possível verificar que a primeira linha multiplicada por $a_2 \sin(\alpha_2)$ é igual à quinta linha multiplicada por $2(1 + \cos(\alpha_2))$. Essa igualdade permite obter a Eq. (4.6.3),

$$a_2 \sin(\alpha_2) x_0 - 2(1 + \cos(\alpha_2)) y_0 = 0, \quad (4.6.3)$$

Esse mesmo produto pode ser feito entre as linhas 4 e 8 da Eq. (4.6.2).

De forma similar a segunda linha multiplicada por $a_2(1 + \cos(\alpha_2))$ é igual à sexta linha multiplicada por $2 \sin(\alpha_2)$, obtendo a Eq. (4.6.4),

$$a_2(1 + \cos(\alpha_2)) x_1 + 2 \sin(\alpha_2) y_1 = 0 \quad (4.6.4)$$

O mesmo produto pode ser feito entre as linhas 3 e 7.

Logo são obtidas quatro equações que podem ser usadas para resolver a cinemática inversa da estrutura serial 2R canônica, dadas por:

$$\begin{aligned} a_2 \sin(\alpha_2) x_0 - 2(1 + \cos(\alpha_2)) y_0 &= 0, \\ a_2(1 + \cos(\alpha_2)) x_1 + 2 \sin(\alpha_2) y_1 &= 0, \\ a_2(1 + \cos(\alpha_2)) x_2 + 2 \sin(\alpha_2) y_2 &= 0, \\ a_2 \sin(\alpha_2) x_3 - 2(1 + \cos(\alpha_2)) y_3 &= 0, \end{aligned} \quad (4.6.5)$$

Tem-se pela relação do arco metade que

$$\eta_2 = \tan\left(\frac{\alpha_2}{2}\right), \quad (4.6.6)$$

e, substituindo a Eq. (4.6.6) na Eq. (4.6.5), resulta no sistema

$$\begin{aligned} 2a_2\eta_2x_0 - 4y_0 &= 0, \\ 2a_2x_1 + 4\eta_2y_1 &= 0, \\ 2a_2x_2 + 4\eta_2y_2 &= 0, \\ 2a_2\eta_2x_3 - 4y_3 &= 0. \end{aligned} \quad (4.6.7)$$

Pode-se observar que o sistema de Eq. (4.6.7) é algébrico e depende dos Parâmetros de Study e dos parâmetros a_i e α_i de Denavit-Hartenberg que são conhecidos em função da estrutura, isto é, não depende explicitamente das variáveis articulares u_i . Em Pfurner (2006) é demonstrado que estas equações podem ser utilizadas para obter a solução do modelo cinemático inverso da estrutura.

O algoritmo utilizado para resolver o modelo cinemático inverso a partir do sistema de equações algébricas é descrito a seguir, expandindo a análise para estruturas seriais 3R e 6R.

4.6.2. Estrutura serial 3R canônica

No exemplo da estrutura serial 2R canônica descrito anteriormente, as equações algébricas que descrevem o mecanismo foram obtidas por inspeção direta do dual quaternion, calculado pelo método de Mapeamento Cinemático.

Em Husty *et al.* (2006) e Walter e Husty (2010) é apresentado um algoritmo para obter este conjunto de equações para diferentes tipos de estruturas seriais e paralelas. Em Pfurner (2006) este algoritmo, denominado *Implicitization Algorithm* é utilizado para obter as equações genéricas que descrevem uma estrutura serial 3R canônica, com os parâmetros de Denavit-Hartenberg apresentados na Tab. 4.2.

Tabela 4.2. Parâmetros de Denavit-Hartenberg de uma estrutura serial 3R canônica.

i	a_i	d_i	α_i	θ_i
0	0	0	0	0
1	a_1	0	α_1	u_1
2	a_2	d_2	α_2	u_2
3	a_3	d_3	α_3	u_3

Na Tabela 4.2 o parâmetro d_1 é nulo, segundo a definição de uma estrutura canônica, porém os demais parâmetros são deixados simbólicos de maneira a generalizar a aplicação do método. Além disso o elemento terminal ET não necessita estar sobre a origem do referencial Σ_3 .

Nas equações algébricas da estrutura serial é considerado ainda a substituição pelo arco metade apresentadas nas Eqs. (4.6.8) e (4.6.9):

$$\eta_i = \tan\left(\frac{\alpha_i}{2}\right), \quad (i=1,2,3), \quad (4.6.8)$$

$$v_i = \tan\left(\frac{u_i}{2}\right), \quad (i=1,2,3), \quad (4.6.9)$$

Cada conjunto de equações que representam uma estrutura serial 3R canônica depende não apenas dos parâmetros de Study e dos parâmetros a_i , d_i e α_i ($i=1,2,3$) de Denavit-Hartenberg, como observado na Eq. (4.6.7), mas também dependem explicitamente de um dos parâmetros v_i .

Visto que $i=1,2,3$, é possível representar a estrutura serial 3R canônica por três conjuntos distintos de equações, cada um em função de um dos parâmetros v_i .

Assim, o sistema algébrico que descreve a estrutura serial 3R canônica em função do parâmetro v_2 é dado pelas Eqs. (4.6.10) a (4.6.13), a seguir:

$$\begin{aligned}
& (-\eta_3 v_2 a_1 + \eta_3 \eta_1 \eta_2 a_2 v_2 - \eta_3 \eta_1 a_1 \eta_2 v_2 + \eta_3 a_2 v_2 - \eta_3 \eta_1 d_2 \\
& + \eta_3 \eta_2 d_2 + a_3 \eta_1 v_2 - \eta_2 a_3 v_2 - \eta_3 d_3 \eta_2 - \eta_3 d_3 \eta_1) x_0 \\
& + (a_1 v_2 - \eta_1 \eta_2 a_2 v_2 + \eta_1 a_1 \eta_2 v_2 - a_2 v_2 + \eta_1 d_2 - \eta_2 d_2 + \eta_3 a_3 \eta_1 v_2 \\
& - a_3 \eta_3 \eta_2 v_2 + d_3 \eta_2 + d_3 \eta_1) x_1 + (d_2 \eta_2 v_2 + \eta_1 d_2 v_2 - a_1 + a_2 \eta_1 \eta_2 \\
& - a_2 + a_1 \eta_1 \eta_2 + d_3 \eta_1 v_2 - d_3 \eta_2 v_2 - a_3 \eta_3 \eta_2 - a_3 \eta_3 \eta_1) x_2 \\
& + (-d_2 \eta_3 \eta_2 v_2 - \eta_1 d_2 \eta_3 v_2 + \eta_3 a_1 - \eta_3 \eta_1 \eta_2 a_2 + \eta_3 a_2 - \eta_3 \eta_1 \eta_2 a_1 \\
& - d_3 \eta_1 \eta_3 v_2 + \eta_3 d_3 \eta_2 v_2 - \eta_2 a_3 - \eta_1 a_3) x_3 - 2\eta_3 v_2 (\eta_1 - \eta_2) y_0 \\
& + 2(\eta_1 v_2 - \eta_2 v_2) y_1 - 2(\eta_2 + \eta_1) y_2 + 2\eta_3 (\eta_2 + \eta_1) y_3 = 0,
\end{aligned} \quad (4.6.10)$$

$$\begin{aligned}
& (\eta_2 a_2 v_2 + \eta_1 v_2 a_1 - \eta_1 v_2 a_2 - \eta_2 v_2 a_1 - d_2 - \eta_1 \eta_2 d_2 - \eta_3 a_3 \eta_1 \eta_2 v_2 \\
& - \eta_3 a_3 v_2 + d_3 \eta_1 \eta_2 - d_3) x_0 (a_2 \eta_3 \eta_2 v_2 + \eta_3 \eta_1 v_2 a_1 - \eta_3 \eta_1 v_2 a_2 \\
& - a_1 \eta_3 \eta_2 v_2 - \eta_3 d_2 - \eta_3 \eta_1 \eta_2 d_2 + a_3 \eta_1 \eta_2 v_2 + a_3 v_2 + \eta_3 d_3 \eta_1 \eta_2 \\
& - \eta_3 d_3) x_1 + (-\eta_1 d_2 \eta_3 \eta_2 v_2 + d_2 \eta_3 v_2 + a_2 \eta_3 \eta_2 + a_1 \eta_3 \eta_2 \\
& + a_2 \eta_3 \eta_1 + a_1 \eta_3 \eta_1 + \eta_3 d_3 \eta_1 \eta_2 v_2 + d_3 \eta_3 v_2 - a_3 \eta_1 \eta_2 + a_3) x_2 \\
& + (-\eta_1 d_2 \eta_2 v_2 + d_2 v_2 + \eta_2 a_2 + \eta_2 a_1 + \eta_1 a_2 + \eta_1 a_1 + d_3 \eta_1 \eta_2 v_2 \\
& + d_3 v_2 + \eta_3 \eta_1 \eta_2 a_3 - \eta_3 a_3) x_3 - 2(\eta_1 \eta_2 v_2 + v_2) y_0 \\
& - 2\eta_3 v_2 (1 + \eta_1 \eta_2) y_1 + 2\eta_3 (\eta_1 \eta_2 - 1) y_2 + 2(\eta_1 \eta_2 - 1) y_3 = 0,
\end{aligned} \tag{4.6.11}$$

$$\begin{aligned}
& (-d_2 \eta_3 \eta_2 v_2 - \eta_1 d_2 \eta_3 v_2 + \eta_3 a_1 - \eta_3 \eta_1 \eta_2 a_2 + \eta_3 a_2 - \eta_3 \eta_1 \eta_2 a_1 \\
& - d_3 \eta_1 \eta_3 v_2 + \eta_3 d_3 \eta_2 v_2 - \eta_2 a_3 - \eta_1 a_3) x_0 \\
& + (d_2 \eta_2 v_2 + \eta_1 d_2 v_2 - a_1 + a_2 \eta_1 \eta_2 - a_2 + a_1 \eta_1 \eta_2 + d_3 \eta_1 v_2 - d_3 \eta_2 v_2 \\
& - a_3 \eta_3 \eta_2 - a_3 \eta_3 \eta_1) x_1 + (\eta_1 \eta_2 a_2 v_2 - a_1 v_2 - \eta_1 a_1 \eta_2 v_2 + a_2 v_2 \\
& - \eta_1 d_2 + \eta_2 d_2 - d_3 \eta_2 - d_3 \eta_1 + a_3 \eta_3 \eta_2 v_2 - \eta_3 a_3 \eta_1 v_2) x_2 \\
& + (-\eta_3 \eta_1 \eta_2 a_2 v_2 + \eta_3 v_2 a_1 + \eta_3 \eta_1 a_1 \eta_2 v_2 - \eta_3 a_2 v_2 + \eta_3 \eta_1 d_2 \\
& - \eta_3 \eta_2 d_2 + \eta_3 d_3 \eta_2 + \eta_3 d_3 \eta_1 + \eta_2 a_3 v_2 - a_3 \eta_1 v_2) x_3 + 2\eta_3 (\eta_2 + \eta_1) y_0 \\
& - 2(\eta_2 + \eta_1) y_1 + 2(\eta_2 v_2 - \eta_1 v_2) y_2 + 2\eta_3 v_2 (\eta_1 - \eta_2) y_3 = 0,
\end{aligned} \tag{4.6.12}$$

$$\begin{aligned}
& (\eta_1 d_2 \eta_2 v_2 - d_2 v_2 - \eta_2 a_1 - \eta_1 a_1 - \eta_2 a_2 - \eta_1 a_2 + \eta_3 a_3 - \eta_3 \eta_1 \eta_2 a_3 \\
& - d_3 \eta_1 \eta_2 v_2 - d_3 v_2) x_0 + (\eta_1 d_2 \eta_3 \eta_2 v_2 - d_2 \eta_3 v_2 - a_1 \eta_3 \eta_2 - a_1 \eta_3 \eta_1 \\
& - a_2 \eta_3 \eta_2 - a_2 \eta_3 \eta_1 - a_3 + a_3 \eta_1 \eta_2 - \eta_3 d_3 \eta_1 \eta_2 v_2 - d_3 \eta_3 v_2) x_1 \\
& + (a_2 \eta_3 \eta_2 v_2 + \eta_3 \eta_1 v_2 a_1 - \eta_3 \eta_1 v_2 a_2 - a_1 \eta_3 \eta_2 v_2 - \eta_3 d_2 \\
& - \eta_3 \eta_1 \eta_2 d_2 + a_3 \eta_1 \eta_2 v_2 + a_3 v_2 + \eta_3 d_3 \eta_1 \eta_2 - \eta_3 d_3) x_2 \\
& + (\eta_2 a_2 v_2 + \eta_1 v_2 a_1 - \eta_1 v_2 a_2 - \eta_2 v_2 a_1 - d_2 - \eta_1 \eta_2 d_2 - \eta_3 a_3 \eta_1 \eta_2 v_2 \\
& - \eta_3 a_3 v_2 + d_3 \eta_1 \eta_2 - d_3) x_3 + 2(1 - \eta_1 \eta_2) y_0 \\
& - 2\eta_3 (\eta_1 \eta_2 - 1) y_1 - 2\eta_3 v_2 (1 + \eta_1 \eta_2) y_2 - 2(\eta_1 \eta_2 v_2 + v_2) y_3 = 0.
\end{aligned} \tag{4.6.13}$$

O sistema algébrico que descreve a estrutura serial 3R canônica em função do parâmetro v_1 é dado pelas Eqs. (4.6.14) a (4.6.17).

$$\begin{aligned}
& (a_2\eta_2 - v_1d_2 - \eta_3a_1 - \eta_3a_3 - \eta_1a_1 - a_2\eta_2\eta_3\eta_1 - \eta_3v_1d_2\eta_1 - \eta_3d_3\eta_1v_1 \\
& - a_3\eta_1 - d_3v_1)x_0 + (-\eta_3v_1d_2 + a_2\eta_2\eta_3 + a_2\eta_2\eta_1 + a_1 + a_3 - \eta_3\eta_1a_1 \\
& + v_1d_2\eta_1 - \eta_3a_3\eta_1 + d_3\eta_1v_1 - \eta_3d_3v_1)x_1 + (a_1v_1 - d_2\eta_1 + \eta_3d_3 - d_3\eta_1 \\
& + a_2\eta_2\eta_1v_1 + \eta_3d_2 - \eta_3\eta_1a_1v_1 - \eta_3a_3\eta_1v_1 + a_3v_1 + a_2\eta_2\eta_3v_1)x_2 \\
& + (-a_3\eta_1v_1 + d_2 + d_3 - \eta_1a_1v_1 + a_2\eta_2v_1 - \eta_3a_1v_1 + \eta_3d_2\eta_1 + \eta_3d_3\eta_1 \\
& - a_2\eta_2\eta_3\eta_1v_1 - \eta_3a_3v_1)x_3 + 2(\eta_3\eta_1 - 1)y_0 \\
& - 2(\eta_3 + \eta_1)y_1 - 2(\eta_1v_1 + \eta_3v_1)y_2 + 2(\eta_3\eta_1v_1 - v_1)y_3 = 0,
\end{aligned} \tag{4.6.14}$$

$$\begin{aligned}
& (\eta_2a_1 + \eta_2v_1d_2\eta_1 - a_2\eta_1 - \eta_2\eta_3\eta_1a_1 - \eta_2\eta_3a_3\eta_1 - \eta_2\eta_3v_1d_2 - \eta_2d_3\eta_1v_1 \\
& - a_2\eta_3 + \eta_2a_3 + \eta_2\eta_3d_3v_1)x_0 + (-a_2\eta_3\eta_1 + \eta_2v_1d_2 - \eta_2\eta_3d_3\eta_1v_1 \\
& + \eta_2\eta_3a_1 + \eta_2\eta_3a_3 - \eta_2d_3v_1 + \eta_2\eta_1a_1 + \eta_2a_3\eta_1 + a_2 + \eta_2\eta_3v_1d_2\eta_1)x_1 \\
& + (\eta_2d_3 - \eta_2d_2 + \eta_2\eta_3a_1v_1 - \eta_2\eta_3d_2\eta_1 - a_2\eta_3\eta_1v_1 + \eta_2\eta_1a_1v_1 \\
& + \eta_2\eta_3a_3v_1 + \eta_2a_3\eta_1v_1 + \eta_2\eta_3d_3\eta_1 + a_2v_1)x_2 \\
& + (-a_2\eta_3v_1 + \eta_2a_1v_1 - \eta_2d_2\eta_1 + \eta_2\eta_3d_2 - \eta_2\eta_3d_3 + \eta_2a_3v_1 + \eta_2d_3\eta_1 \\
& - \eta_2\eta_3\eta_1a_1v_1 - \eta_2\eta_3a_3\eta_1v_1 - a_2\eta_1v_1)x_3 - 2(\eta_2\eta_3 + \eta_2\eta_1)y_0 \\
& + 2(-\eta_2\eta_3\eta_1 + \eta_2)y_1 + 2(\eta_2v_1 - \eta_2\eta_3\eta_1v_1)y_2 - 2(\eta_2\eta_3v_1 + \eta_2\eta_1v_1)y_3 = 0,
\end{aligned} \tag{4.6.15}$$

$$\begin{aligned}
& (-\eta_2a_3v_1 + \eta_2\eta_3d_3 + \eta_2a_1v_1 - \eta_2d_2\eta_1 - \eta_2\eta_3d_2 + \eta_2\eta_3\eta_1a_1v_1 \\
& - \eta_2\eta_3a_3\eta_1v_1 + \eta_2d_3\eta_1 - a_2\eta_1v_1 + a_2\eta_3v_1)x_0 \\
& + (-\eta_2\eta_3a_3v_1 + \eta_2\eta_3d_3\eta_1 - a_2v_1 + \eta_2d_2 - \eta_2d_3 - a_2\eta_3\eta_1v_1 + \eta_2\eta_3a_1v_1 \\
& - \eta_2\eta_3d_2\eta_1 - \eta_2\eta_1a_1v_1 + \eta_2a_3\eta_1v_1)x_1 + (a_2\eta_3\eta_1 + a_2 - \eta_2\eta_3v_1d_2\eta_1 \\
& + \eta_2v_1d_2 + \eta_2\eta_1a_1 + \eta_2\eta_3d_3\eta_1v_1 - \eta_2d_3v_1 + \eta_2\eta_3a_3 - \eta_2a_3\eta_1 - \eta_2\eta_3a_1)x_2 \\
& + (a_2\eta_1 - \eta_2a_1 + \eta_2a_3 - \eta_2\eta_3v_1d_2 - \eta_2\eta_3\eta_1a_1 - \eta_2v_1d_2\eta_1 + \eta_2d_3\eta_1v_1 \\
& + \eta_2\eta_3d_3v_1 - a_2\eta_3 + \eta_2\eta_3a_3\eta_1)x_3 + 2(\eta_2\eta_3v_1 - \eta_2\eta_1v_1)y_0 \\
& - 2(\eta_2v_1 + \eta_2\eta_3\eta_1v_1)y_1 + 2(\eta_2\eta_3\eta_1 + \eta_2)y_2 + 2(-\eta_2\eta_3 + \eta_2\eta_1)y_3 = 0,
\end{aligned} \tag{4.6.16}$$

$$\begin{aligned}
& (-d_2 + \eta_3a_3v_1 - d_3 - a_3\eta_1v_1 + \eta_3d_3\eta_1 - a_2\eta_2\eta_3\eta_1v_1 - a_2\eta_2v_1 \\
& + \eta_1a_1v_1 - \eta_3a_1v_1 + \eta_3d_2\eta_1)x_0 + (a_1v_1 - d_2\eta_1 - \eta_3d_2 - a_2\eta_2\eta_3v_1 \\
& + a_2\eta_2\eta_1v_1 + \eta_3\eta_1a_1v_1 - \eta_3a_3\eta_1v_1 - a_3v_1 - d_3\eta_1 - \eta_3d_3)x_1 \\
& + (a_2\eta_2\eta_3 + \eta_3a_3\eta_1 - v_1d_2\eta_1 - d_3\eta_1v_1 - a_2\eta_2\eta_1 - a_1 + a_3 - \eta_3v_1d_2 \\
& - \eta_3\eta_1a_1 - \eta_3d_3v_1)x_2 + (a_2\eta_2 + a_3\eta_1 - \eta_3a_3 - \eta_1a_1 + \eta_3a_1 - d_3v_1 \\
& + a_2\eta_2\eta_3\eta_1 + \eta_3v_1d_2\eta_1 + \eta_3d_3\eta_1v_1 - v_1d_2)x_3 + 2(\eta_3\eta_1v_1 + v_1)y_0 \\
& + 2(\eta_3v_1 - \eta_1v_1)y_1 + 2(\eta_1 - \eta_3)y_2 - 2(1 + \eta_3\eta_1)y_3 = 0.
\end{aligned} \tag{4.6.17}$$

O sistema algébrico que descreve a estrutura serial 3R canônica em função do parâmetro v_3 é dado pelas Eqs. (4.6.18) a (4.6.21).

$$\begin{aligned}
& (a_1\eta_1 - a_1\eta_1\eta_2\eta_3 - \eta_2a_2 - d_2v_3 - \eta_3a_2 - \eta_3\eta_2d_2v_3 - \eta_3a_3 - \eta_2a_3 \\
& + \eta_2v_3\eta_3d_3 - v_3d_3)x_0 + (a_1\eta_1\eta_3 + a_1\eta_1\eta_2 - \eta_3\eta_2a_2 - \eta_3d_2v_3 + a_2 + \eta_2d_2v_3 \\
& + a_3 - \eta_2\eta_3a_3 - \eta_2v_3d_3 - v_3\eta_3d_3)x_1 + (-a_1\eta_1\eta_2v_3 + a_1\eta_1v_3\eta_3 - \eta_3\eta_2a_2v_3 \\
& + \eta_3d_2 - a_2v_3 + \eta_2d_2 - \eta_2d_3 + \eta_3d_3 + v_3a_3 + \eta_2v_3\eta_3a_3)x_2 + (a_1\eta_1\eta_2v_3\eta_3 \\
& + a_1\eta_1v_3 + \eta_3a_2v_3 - \eta_3\eta_2d_2 - \eta_2a_2v_3 + d_2 + d_3 + \eta_2\eta_3d_3 + \eta_2v_3a_3 - v_3\eta_3a_3)x_3 \\
& + 2(-1 + \eta_2\eta_3)y_0 - 2(\eta_3 + \eta_2)y_1 + 2(\eta_2v_3 - v_3\eta_3)y_2 - 2(\eta_2v_3\eta_3 + v_3)y_3 = 0,
\end{aligned} \tag{4.6.18}$$

$$\begin{aligned}
& (-a_1\eta_2 - a_1\eta_3 + \eta_1a_2 + \eta_1\eta_2d_2v_3 - \eta_1\eta_3\eta_2a_2 - \eta_1\eta_3d_2v_3 - \eta_1\eta_2\eta_3a_3 \\
& + \eta_1a_3 - \eta_1v_3\eta_3d_3 - \eta_1\eta_2v_3d_3)x_0 + (a_1 - a_1\eta_2\eta_3 + \eta_1\eta_2a_2 + \eta_1d_2v_3 \\
& + \eta_1\eta_3a_2 + \eta_1\eta_3\eta_2d_2v_3 + \eta_1\eta_3a_3 + \eta_1\eta_2a_3 - \eta_1\eta_2v_3\eta_3d_3 + \eta_1v_3d_3)x_1 \\
& + (-a_1v_3 - a_1\eta_2v_3\eta_3 - \eta_1\eta_2a_2v_3 + \eta_1d_2 + \eta_1\eta_3a_2v_3 - \eta_1\eta_3\eta_2d_2 \\
& + \eta_1\eta_2\eta_3d_3 + \eta_1d_3 - \eta_1v_3\eta_3a_3 + \eta_1\eta_2v_3a_3)x_2 \\
& + (-a_1\eta_2v_3 + a_1v_3\eta_3 + \eta_1\eta_3\eta_2a_2v_3 - \eta_1\eta_3d_2 + \eta_1a_2v_3 - \eta_1\eta_2d_2 + \eta_1\eta_2d_3 \\
& - \eta_1\eta_3d_3 - \eta_1v_3a_3 - \eta_1\eta_2v_3\eta_3a_3)x_3 - 2(\eta_1\eta_2 + \eta_1\eta_3)y_0 \\
& + 2(\eta_1 - \eta_1\eta_2\eta_3)y_1 - 2(\eta_1v_3 + \eta_1\eta_2v_3\eta_3)y_2 + 2(-\eta_1\eta_2v_3 + \eta_1v_3\eta_3)y_3 = 0,
\end{aligned} \tag{4.6.19}$$

$$\begin{aligned}
& (a_1\eta_2v_3 - a_1v_3\eta_3 - \eta_1\eta_3\eta_2a_2v_3 + \eta_1\eta_3d_2 + \eta_1\eta_2d_2 - \eta_1a_2v_3 + \eta_1v_3a_3 \\
& + \eta_1\eta_2v_3\eta_3a_3 - \eta_1\eta_2d_3 + \eta_1\eta_3d_3)x_0 \\
& + (a_1\eta_2v_3\eta_3 + a_1v_3 - \eta_1\eta_3a_2v_3 + \eta_1\eta_3\eta_2d_2 + \eta_1\eta_2a_2v_3 - \eta_1d_2 - \eta_1d_3 \\
& - \eta_1\eta_2\eta_3d_3 - \eta_1\eta_2v_3a_3 + \eta_1v_3\eta_3a_3)x_1 + (a_1 - a_1\eta_2\eta_3 + \eta_1\eta_2a_2 + \eta_1d_2v_3 \\
& + \eta_1\eta_3a_2 + \eta_1\eta_3\eta_2d_2v_3 + \eta_1\eta_3a_3 + \eta_1\eta_2a_3 - \eta_1\eta_2v_3\eta_3d_3 + \eta_1v_3d_3)x_2 \\
& + (-a_1\eta_2 - a_1\eta_3 + \eta_1a_2 + \eta_1\eta_2d_2v_3 - \eta_1\eta_3\eta_2a_2 - \eta_1\eta_3d_2v_3 - \eta_1\eta_2\eta_3a_3 \\
& + \eta_1a_3 - \eta_1v_3\eta_3d_3 - \eta_1\eta_2v_3d_3)x_3 + 2(\eta_1\eta_2v_3 - \eta_1v_3\eta_3)y_0 \\
& + 2(\eta_1\eta_2v_3\eta_3 + \eta_1v_3)y_1 + 2(\eta_1 - \eta_1\eta_2\eta_3)y_2 - 2(\eta_1\eta_2 + \eta_1\eta_3)y_3 = 0,
\end{aligned} \tag{4.6.20}$$

$$\begin{aligned}
& (-a_1\eta_1v_3 - a_1\eta_1\eta_2v_3\eta_3 + \eta_2a_2v_3 - d_2 - \eta_3a_2v_3 + \eta_3\eta_2d_2 - \eta_2\eta_3d_3 \\
& - d_3 + v_3\eta_3a_3 - \eta_2v_3a_3)x_0 + (a_1\eta_1\eta_2v_3 - a_1\eta_1v_3\eta_3 + \eta_3\eta_2a_2v_3 \\
& - \eta_3d_2 - \eta_2d_2 + a_2v_3 - v_3a_3 - \eta_2v_3\eta_3a_3 + \eta_2d_3 - \eta_3d_3)x_1 \\
& + (a_1\eta_1\eta_3 + a_1\eta_1\eta_2 - \eta_3\eta_2a_2 - \eta_3d_2v_3 + a_2 + \eta_2d_2v_3 + a_3 - \eta_2\eta_3a_3 \\
& - \eta_2v_3d_3 - v_3\eta_3d_3)x_2 + (a_1\eta_1 - a_1\eta_1\eta_2\eta_3 - \eta_2a_2 - d_2v_3 - \eta_3a_2 \\
& - \eta_3\eta_2d_2v_3 - \eta_3a_3 - \eta_2a_3 + \eta_2v_3\eta_3d_3 - v_3d_3)x_3 + 2(v_3 + \eta_2v_3\eta_3)y_0 \\
& + 2(-\eta_2v_3 + v_3\eta_3)y_1 - 2(\eta_3 + \eta_2)y_2 + 2(-1 + \eta_2\eta_3)y_3 = 0.
\end{aligned} \tag{4.6.21}$$

4.7. Equações de uma estrutura serial 6R

A partir da representação de uma estrutura serial 3R pelo conjunto de equações apresentados, é possível avaliar uma estrutura serial 6R dividindo-a em duas estruturas seriais 3R. Deve-se salientar que a utilização de três estruturas seriais 2R para representar a estrutura serial 6R de cada uma das pernas, ao invés de duas estruturas seriais 3R, não poderia ser feita pois não permitiria resolver o sistema de equações polinomiais para obter os valores das variáveis articulares, já que nenhuma equação seria escrita explicitamente em função de uma das variáveis articulares.

Com esta abordagem, Pfurner (2006) propõe que um conjunto de 8 equações representem uma estrutura 6R de maneira que 4 destas equações sejam dadas em função de v_i ($i=1,2$ ou 3) e outras 4 equações em função de v_j ($j=4,5$ ou 6).

Neste trabalho cada perna do robô humanoide é modelada como uma estrutura serial 6R, conforme apresentado no Capítulo III. Desta forma cada perna pode ser representada por um conjunto de 8 equações. A divisão em duas estruturas 3R é feita de maneira a agrupar as articulações do tornozelo e do joelho em uma primeira estrutura serial 3R e as três articulações da cintura em uma segunda estrutura esférica.

De acordo com a demonstração apresentada em Pfurner (2006), para o caso de uma estrutura serial 3R constituindo uma articulação esférica, o sistema algébrico utilizado para representá-la é dado pelas Eqs. (4.7.1) a (4.7.4), o que define um caso particular da análise geral de uma estrutura serial 3R. Estas equações são usadas para qualquer estrutura esférica, ou seja, não é necessário que ela seja canônica.

A estrutura esférica, não sendo necessária ser canônica, seu referencial de base pode ser definido da forma mais conveniente possível para a solução do problema. Neste caso, como o interesse reside em resolver a cinemática inversa, sendo conhecidas a posição e a orientação do referencial do centro de massa do robô humanoide, é possível colocar esse referencial de base da estrutura esférica no referencial inercial do robô. Sendo assim, pode-se então mapear a localização da articulação esférica em relação ao referencial inercial por um dual quaternion T , Eq. (4.7.5), a partir da localização conhecida do centro de massa.

O elemento terminal (ET) não precisa estar, necessariamente, no centro da articulação esférica. Seu referencial é definido pelos parâmetros a_3 , d_3 e α_3 . Cabe salientar ainda que os parâmetros de Denavit-Hartenberg a_1 , a_2 e d_2 são nulos, por se tratar de uma estrutura esférica.

Destaca-se que a estrutura serial 6R definida para modelar cada uma das pernas do robô humanoide é representado por 8 equações. As 4 primeiras equações

relativas ao modelo 3R canônico, que representa as articulações do tornozelo e do joelho, e as 4 outras equações relativas à estrutura esférica. As 4 primeiras equações são dadas em função de um dos parâmetros v_i ($i=1,2$ ou 3). As outras 4 dependem apenas dos Parâmetros de Study do dual quaternion T e dos parâmetros de Denavit-Hartenberg, isto é, não depende dos parâmetros v_j ($j=4,5$ ou 6). Isto pode ser verificado nas Eqs. (4.7.1) a (4.7.4), que representam a estrutura esférica.

$$\begin{aligned}
& x_0(t_0\eta_3a_3 + t_1a_3 + t_2\eta_3d_3 + t_3d_3 + 2t_4 - 2t_5\eta_3) \\
& + x_1(t_1\eta_3a_3 - t_0a_3 + t_3\eta_3d_3 - t_2d_3 + 2t_5 + 2t_4\eta_3) \\
& + x_2(t_2\eta_3a_3 - t_3a_3 - t_0\eta_3d_3 + t_1d_3 + 2t_6 + 2t_7\eta_3) \\
& + x_3(t_3\eta_3a_3 + t_2a_3 - t_1\eta_3d_3 - t_0d_3 + 2t_7 - 2t_6\eta_3) + 2y_0(t_0 - t_1\eta_3) \\
& + 2y_1(t_1 + t_0\eta_3) + 2y_2(t_2 + t_3\eta_3) + 2y_3(t_3 - t_2\eta_3) = 0,
\end{aligned} \tag{4.7.1}$$

$$\begin{aligned}
& x_0(t_0a_3 - t_1\eta_3a_3 - t_2d_3 + t_3\eta_3d_3 - 2t_4\eta_3 - 2t_5) \\
& + x_1(t_1a_3 + t_0\eta_3a_3 - t_3d_3 - t_2\eta_3d_3 - 2t_5\eta_3 + 2t_4) \\
& + x_2(t_2a_3 + t_3\eta_3a_3 + t_0d_3 + t_1\eta_3d_3 - 2t_6\eta_3 + 2t_7) \\
& + x_3(t_3a_3 - t_2\eta_3a_3 + t_1d_3 - t_0\eta_3d_3 - 2t_7\eta_3 - 2t_6) - 2y_0(t_0\eta_3 + t_1) \\
& + 2y_1(t_0 - t_1\eta_3) + 2y_2(t_3 - t_2\eta_3) - 2y_3(t_3\eta_3 + t_2) = 0,
\end{aligned} \tag{4.7.2}$$

$$\begin{aligned}
& x_0(t_0\eta_3d_3 + t_1d_3 - t_2\eta_3a_3 - t_3a_3 - 2t_6 + 2t_7\eta_3) \\
& + x_1(t_1\eta_3d_3 - t_0d_3 - t_3\eta_3a_3 + t_2a_3 - 2t_7 - 2t_6\eta_3) \\
& + x_2(t_2\eta_3d_3 - t_3d_3 + t_0\eta_3a_3 - t_1a_3 + 2t_4 + 2t_5\eta_3) \\
& + x_3(t_3\eta_3d_3 + t_2d_3 + t_1\eta_3a_3 + t_0a_3 + 2t_5 - 2t_4\eta_3) + 2y_0(-t_2 + t_3\eta_3) \\
& - 2y_1(t_3 + t_2\eta_3) + 2y_2(t_0 + t_1\eta_3) + 2y_3(t_1 - t_0\eta_3) = 0,
\end{aligned} \tag{4.7.3}$$

$$\begin{aligned}
& x_0(t_0d_3 - t_1\eta_3d_3 + t_2a_3 - t_3\eta_3a_3 - 2t_6\eta_3 - 2t_7) \\
& + x_1(t_1d_3 + t_0\eta_3d_3 + t_3a_3 + t_2\eta_3a_3 - 2t_7\eta_3 + 2t_6) \\
& + x_2(t_2d_3 + t_3\eta_3d_3 - t_0a_3 - t_1\eta_3a_3 + 2t_4\eta_3 - 2t_5) \\
& + x_3(t_3d_3 - t_2\eta_3d_3 - t_1a_3 + t_0\eta_3a_3 + 2t_5\eta_3 + 2t_4) - 2y_0(t_3 + t_2\eta_3) \\
& + 2y_1(-t_3\eta_3 + t_2) + 2y_2(t_0\eta_3 - t_1) + 2y_3(t_0 + t_1\eta_3) = 0.
\end{aligned} \tag{4.7.4}$$

Sendo,

$$T = \begin{pmatrix} t_0 \\ t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \end{pmatrix}. \quad (4.7.5)$$

4.8. Algoritmo para solução da cinemática inversa

A partir das equações apresentadas para uma estrutura serial 3R, é possível dividir cada perna do robô humanoide em 2 estruturas 3R, e resolver o problema cinemático inverso de cada uma das pernas separadamente, conforme algoritmo apresentado a seguir.

Visto que as duas pernas possuem a mesma estrutura, é necessário calcular a cinemática inversa para apenas uma delas em posturas correspondentes às fases de apoio e de movimento. Conhecendo os ângulos das articulações para todas as posturas, o movimento das pernas é feito trocando as fases de uma perna para outra sequencialmente, isto é, enquanto uma perna está no apoio a outra está em movimento.

Para obter os valores das articulações em cada postura, é necessário conhecer as posições e orientações do pé e do centro de massa. A descrição das posturas utilizadas para as implementações numéricas e experimentais são apresentadas no Capítulo V, sendo aqui tratadas de modo genérico.

As manipulações algébricas apresentadas a seguir são para permitir que a estrutura 6R possa ser tratada como duas estruturas 3R interligadas.

Para isso, é possível considerar um novo referencial Σ_{6^*} , localizado após a articulação 6, com origem no centro da articulação esférica e eixos com a mesma orientação do referencial do centro de massa. A matriz de transformação homogênea entre o referencial Σ_1 e este novo referencial Σ_{6^*} é dada por

$${}^1T_{6^*} = {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_{6^*}. \quad (4.8.1)$$

Sendo os parâmetros de Denavit-Hartenberg apresentados na Tab. 4.3, e a matriz ${}^iT_{i+1}$, $i=1,\dots,6$, obtida pela Eq. (3.1.1).

Tabela 4.3. Parâmetros de Denavit-Hartenberg de uma perna do robô humanoide.

i	a_i	d_i	α_i	u_i
1	0	0	$\pi/2$	u_1
2	$a_2 = l_2$	0	0	u_2
3	$a_3 = l_3$	0	0	u_3
4	0	0	$-\pi/2$	u_4
5	0	0	$\pi/2$	$u_5 + \pi/2$
6	0	0	0	u_6

As matrizes de transformação homogêneas entre o referencial inercial, Σ_0 , e o referencial do pé de apoio, Σ_B , é dada por 0T_B , bem como entre os referenciais Σ_B e Σ_1 por BT_1 .

Considerando o novo referencial Σ_{6*} , a posição e a orientação do referencial do centro de massa Σ_{CM} em relação à ele é dada pela matriz de transformação homogênea ${}^{6*}T_{CM}$. Logo, a situação do centro de massa em relação ao referencial inercial pode ser dado por:

$${}^0T_{CM} = {}^0T_B {}^BT_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_{6*} {}^{6*}T_{CM}. \quad (4.8.2)$$

Pode-se pré multiplicar a Eq. (4.8.2) por BT_0 , o que resulta em

$${}^BT_0 {}^0T_{CM} = ({}^0T_B)^{-1} {}^0T_{CM} = {}^BT_{CM}. \quad (4.8.3)$$

Substituindo a inversa de ${}^BT_{CM}$ na Eq. (4.8.2), obtém-se

$$({}^BT_{CM})^{-1} {}^BT_1 {}^1T_2 {}^2T_3 {}^3T_4 {}^4T_5 {}^5T_6 {}^6T_{6*} {}^{6*}T_{CM} = I_4, \quad (4.8.4)$$

na qual I_4 é a matriz identidade 4x4. A Equação (4.8.4) pode ser manipulada pós multiplicando pelas inversas das matrizes, resultando em

$$({}^BT_{CM})^{-1} {}^BT_1 {}^1T_2 {}^2T_3 {}^3T_4 = ({}^{6*}T_{CM})^{-1} ({}^6T_{6*})^{-1} ({}^5T_6)^{-1} ({}^4T_5)^{-1}, \quad (4.8.5)$$

ou ainda,

$${}^1T_2 {}^2T_3 {}^3T_4 = ({}^BT_1)^{-1} ({}^BT_{CM})^{-1} ({}^{6*}T_{CM})^{-1} ({}^6T_{6*})^{-1} ({}^5T_6)^{-1} ({}^4T_5)^{-1}. \quad (4.8.6)$$

A matriz ${}^1T_{6*}$ pode então ser escrita na seguinte forma:

$${}^1T_{6*} = ({}^BT_1)^{-1} {}^BT_{CM} ({}^{6*}T_{CM})^{-1}, \quad (4.8.7)$$

a qual leva, pela sua substituição na Eq. (4.8.6), à

$${}^1T_2 {}^2T_3 {}^3T_4 = {}^1T_{6*} ({}^{6*}T_{6*})^{-1} ({}^5T_6)^{-1} ({}^4T_5)^{-1}. \quad (4.8.8)$$

Pela Equação (4.8.8), é possível verificar que o produto matricial dos termos do lado esquerdo da igualdade representam uma estrutura serial 3R canônica.

Já o produto matricial dos termos do lado direito representam uma estrutura esférica não canônica, visto que as direções de rotação das articulações se cruzam em um único ponto. Além disso, os termos são pré multiplicados pela matriz ${}^1T_{6*}$, que translada a estrutura do referencial inercial, tornando-a não canônica.

Conforme apresentado na Seção 4.7, tanto uma estrutura serial 3R canônica quanto uma esférica não canônica podem ser descritos por um conjunto de equações algébricas. Para isto as matrizes obtidas pelos produtos de ${}^iT_{i+1}$, $i=1,..,3$, do lado esquerdo, e ${}^iT_{i+1}$, $i=4,..,6$, do lado direito da Eq. (4.8.8), devem ser convertidas para a notação de dual quaternions, dadas pelas Eqs. (4.5.3) e (4.5.4).

Em seguida, as Eqs. (4.6.10) a (4.6.21) podem ser diretamente aplicadas para representar a estrutura 3R representada pelos termos do lado esquerdo da Eq. (4.8.8). Já as Equações (4.7.1) a (4.7.4) podem ser utilizadas para representar a estrutura esférica não canônica dada pelos termos do lado direito, com as seguintes considerações:

- a- A matriz ${}^1T_{6*}$, que representa a translação da estrutura esférica do referencial inercial, deve ser também convertida para o dual quaternion correspondente, conforme apresentado nas Eqs. (4.5.3) e (4.5.4). Esta transformação resulta no dual quaternion T, Eq. (4.7.5).
- b- Observa-se que a estrutura esférica representada pelo termo do lado direito da Eq. (4.8.8) é dada por matrizes inversas de ${}^iT_{i+1}$, $i=4,..,6$. Desta forma, os parâmetros das Eqs. (4.7.1) a (4.7.4), que representam a estrutura esférica, devem ser corrigidos de maneira a refletir essa inversão. Esta correção é demonstrada por Pfurner (2006), onde é provado que as mesmas Eqs (4.7.1) a (4.7.4) podem ser utilizadas, realizando as seguintes adequações nos termos:

$$\begin{aligned}
d_3 &\rightarrow -d_4, \\
\eta_1 &\rightarrow \eta_5, \\
\eta_2 &\rightarrow \eta_4, \\
a_3 &= 0, \\
\eta_3 &= 0.
\end{aligned} \tag{4.8.9}$$

Com estas substituições, torna-se possível representar a estrutura serial 6R de cada perna do robô humanoide através de um conjunto de 8 equações algébricas.

O algoritmo para a solução da cinemática inversa do robô humanoide pode ser descrito em 6 etapas, sendo que as etapas já apresentadas podem ser resumidas de acordo com os passos de 1 a 3 descritos a seguir:

1- O produto das matrizes de ${}^i T_{i+1}$, $i=1,\dots,3$, resulta em uma matriz que é convertida em seguida para um dual quaternion, conforme Eqs. (4.5.3) e (4.5.4). Para representar o estrutura serial 3R canônica resultante, utiliza-se inicialmente o conjunto das 4 equações em função do parâmetro v_2 , Eqs. (4.6.10) a (4.6.13).

2- A matriz ${}^1 T_{6*}$ é calculada pela Eq. (4.8.7) e convertida em seguida para um dual quaternion, conforme Eqs. (4.5.3) e (4.5.4). Essa transformação resulta no dual quaternion T.

3- O produto das matrizes de ${}^i T_{i+1}$, $i=4,\dots,6$, resulta em uma matriz que é convertida em seguida para um dual quaternion, conforme Eqs. (4.5.3) e (4.5.4). Para representar o estrutura esférica não canônica resultante, utiliza-se as Eqs. (4.7.1) a (4.7.4), e o dual quaternion T, obtido no passo 2, levando em conta as substituições expressas na Eq. (4.8.9).

Resultam, dos passos 1 a 3, um conjunto de 8 equações. Em seguida, os procedimentos para obter a solução do problema cinemático inverso, são apresentados.

As 4 equações polinomiais, ou hiperplanos, obtidas no passo 1 dependem dos parâmetros de Denavit-Hartenberg e de v_2 explicitamente. Estes hiperplanos possuem coordenadas definidas como sendo os termos que multiplicam respectivamente x_0 , x_1 , x_2 , x_3 , y_0 , y_1 , y_2 e y_3 . Estas coordenadas podem ser organizadas na forma de uma

matriz linha, para cada uma das 4 equações, sendo estas matrizes denominadas por $h_i(v_2)$, $i=1,..,4$.

Já os 4 hiperplanos obtidos no passo 3 dependem dos Parâmetros de Study do dual quaternion T, e dos parâmetros de Denavit-Hartenberg, isto é, não dependem explicitamente dos parâmetros v_j ($j=4,5$ ou 6). As coordenadas destes hiperplanos também são definidas como sendo os termos que multiplicam $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 , respectivamente. Estas coordenadas também podem ser organizadas na forma de uma matriz linha, para cada uma das 4 equações, sendo denominadas por h_j , $j=5,..,8$.

Estes 8 hiperplanos se interceptam se o determinante da matriz dada pela Eq. (4.8.10) for nulo, sendo cada uma das linhas do determinante iguais às matrizes linhas das coordenadas dos hiperplanos,

$$\det [(h_1(v_2), h_2(v_2), h_3(v_2), h_4(v_2), h_5, h_6, h_7, h_8)^T] = 0. \quad (4.8.10)$$

Da solução deste determinante obtém-se um polinômio de ordem 4 em v_2 , o que leva a 4 valores para v_2 . Os seguintes passos são então repetidos para cada um dos valores obtidos para v_2 :

4- Substituindo nas 8 equações dos hiperplanos um dos valores de v_2 obtido, é possível calcular os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 do quaternion S que representa a interseção dos hiperplanos. Esta interseção corresponde à postura em que as estruturas 3R canônica e esférica não canônica se encontram no espaço.

Uma das possíveis soluções para S é o dual quaternion com 8 elementos nulos, que não é uma solução válida. Para evitar esta solução, impõe-se a condição de normalidade de dual quaternions, dada por $x_0=1$, calculando assim os demais valores $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 .

5- Dado o valor de S, os valores de v_1 e v_3 são então calculados. A solução de v_1 é obtida substituindo os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 pelos valores de S em uma das Eqs. (4.6.14) a (4.6.17), que descrevem a estrutura 3R canônica em função do parâmetro v_1 .

A solução de v_3 é obtida substituindo os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 em uma das Eqs. (4.6.18) a (4.6.21), que descrevem a estrutura 3R canônica em função do parâmetro v_3 .

6- Visto que S corresponde à postura em que as estruturas 3R canônica e esférica não canônica se encontram, é possível igualar ambos os lados da Eq. (4.8.8). a S . Igualando o lado direito, correspondente à estrutura esférica não canônica, resulta em

$${}^4T_5 {}^5T_6 {}^6T_{6*} = (S)^{-1} {}^1T_{6*}, \quad (4.8.11)$$

na qual $(S)^{-1}$ é o dual quaternion inverso de S .

O termo do lado esquerdo da Eq. (4.8.11) pode então ser convertido no dual quaternion correspondente. Sendo T o dual quaternion correspondente à matriz ${}^1T_{6*}$, resulta do lado direito da Eq. (4.8.11) um produto entre dois quaternions, que pode ser resolvido pela Eq. (4.3.2). Tem-se portanto uma igualdade entre dois quaternions e, consequentemente, 8 equações nas 3 incógnitas v_4 , v_5 e v_6 . Este sistema de 8 equações com 3 incógnitas resulta em 2 soluções para cada uma das incógnitas.

Ao final do 6º passo tem-se as 8 possíveis soluções para os ângulos das articulações de cada perna para cada postura do robô humanoide. Isto ocorre pois são encontradas 4 soluções de v_2 , e para cada uma destas são encontradas 1 solução para v_1 e v_3 , e para cada solução de v_1 e v_3 são obtidas 2 soluções para v_4 , v_5 e v_6 . Os valores dos ângulos das articulações são obtidos a partir de v_i , $i=1,\dots,6$, sendo

$$u_i = 2 \cdot \text{atan}(v_i) . \quad (4.8.12)$$

Os passos apresentados foram então implementados em um código Matlab, presente no Apêndice B. No código utiliza-se também a biblioteca de operações com dual quaternion chamada *Dual Quaternion Symbolic Toolbox*, criada por Leclercq (2012) e disponível para *download* no site oficial do *software* Matlab.

No Capítulo V o algoritmo proposto é usado para obter os ângulos das articulações para um conjunto de posturas propostas ao longo da caminhada do robô humanoide, que serão utilizadas para exemplificar o uso do algoritmo proposto.

CAPÍTULO V

SOLUÇÃO DO PROBLEMA CINEMÁTICO INVERSO

Neste capítulo é apresentada a solução do problema cinemático inverso aplicada a uma trajetória proposta para o movimento do robô humanoide. Essa trajetória é dada pelos deslocamentos e orientações dos referenciais Σ_{CM} e Σ_B posicionados no centro de massa e nos pés, respectivamente, e dadas em função de suas componentes espaciais, posição e orientação, que variam ao longo de cada postura do movimento. Nesta tese, a orientação destes referenciais será denominada orientação do centro de massa e dos pés, respectivamente.

Os valores relativos à posição e à orientação do centro de massa e dos pés foram escolhidos não nulos para verificar a aplicabilidade do método. Além disso, eles foram escolhidos para que o movimento do robô seja o mais semelhante possível ao movimento da marcha humana, conforme descrito a seguir.

5.1. Trajetória Proposta

Nos estudos sobre a marcha humana, tais como em Kharb *et al.* (2011), Whittle (2007), Chou *et al.* (2000) e Zijlstra e Hof (2003), fica claro que a trajetória, tanto do centro de massa como dos pés, consistem em uma curva tridimensional, ou seja, existem componentes de deslocamento nas direções vertical, lateral e frontal.

Com base nos dados dos trabalhos citados, foi definida uma trajetória para o movimento do centro de massa e outra para os pés, tanto em termos de posição como de orientação, para o movimento do robô humanoide.

O método apresentado nesta tese permite analisar qualquer tipo de trajetória definida pela quantidade de posturas considerada conveniente para sua representação.

Nas próximas seções são apresentadas as definições adotadas para representar a trajetória do centro de massa e dos pés do robô humanoide, objeto desta tese.

5.1.1. Deslocamento do centro de massa e dos pés

Uma das vantagens do método proposto para a solução do problema de cinemática inversa, utilizando dual quaternions, é a possibilidade de considerar uma trajetória para os pés e centro de massa que possuam componentes de deslocamentos nas direções vertical, lateral e frontal do movimento a ser executado pelo robô humanoide. As componentes espaciais dos deslocamentos dos pés e do centro de massa são propostas em seguida, onde alguns parâmetros foram definidos para facilitar a caracterização do movimento.

Foi verificado em diversos experimentos com os protótipos disponíveis que quando o robô inicia seu movimento com as pernas esticadas ($u_2 = u_3 = u_4 = 0$), seus atuadores ficavam superaquecidos. Então, optou-se por definir uma posição inicial do robô onde $u_2 \neq 0$, $u_3 \neq 0$ e $u_4 \neq 0$, ou seja, ele parte de uma posição “levemente agachada” e mantém esse comportamento durante seu movimento. Isso também pode ser verificado em diversos robôs humanoides descritos no Capítulo II. Essa condição também evita a condição de singularidade onde ($u_2 = u_3 = u_4 = 0$).

Assim, é definido como *Agachamento* o deslocamento, para baixo, do centro de massa do robô a partir do qual ele vai se deslocar verticalmente. Para a trajetória proposta, foi definido um valor de 15 mm, significando que durante a caminhada do robô, o deslocamento vertical do centro de massa oscila a partir de sua altura original subtraída de 15 mm, o que deixa as pernas ligeiramente dobradas durante a caminhada.

Além de abaixar a altura do centro de massa, é possível abrir um pouco as pernas do robô, afastando os pés um do outro. Este parâmetro é definido como *Abertura das Pernas*, e para a trajetória proposta foi definido um valor de 10 mm.

Durante a fase de balanço, cada um dos pés possuem componentes definidos espacialmente, ou seja, simultaneamente se move verticalmente em relação ao solo, lateralmente em relação ao pé de apoio e frontalmente fazendo com que o robô caminhe para frente. As amplitudes máximas destes 3 parâmetros são denominadas *Altura Máxima dos Pés*, *Largura Lateral* e *Tamanho do Passo*, respectivamente. Para a trajetória proposta foram escolhidos os valores de 30 mm para a *Altura Máxima dos Pés* e *Largura Lateral* e 180 mm para o *Tamanho do Passo*.

Para o centro de massa também foram escolhidos os valores das Amplitudes Máximas dos Deslocamentos Lateral e Vertical iguais a 4 mm e 3 mm, respectivamente. A trajetória proposta é apresentada na Fig. 5.1, considerando que o centro de massa do robô humanoide com as pernas esticadas está a 433.5 mm do piso. Os demais dados dimensionais do robô são apresentados na Seção 5.3.

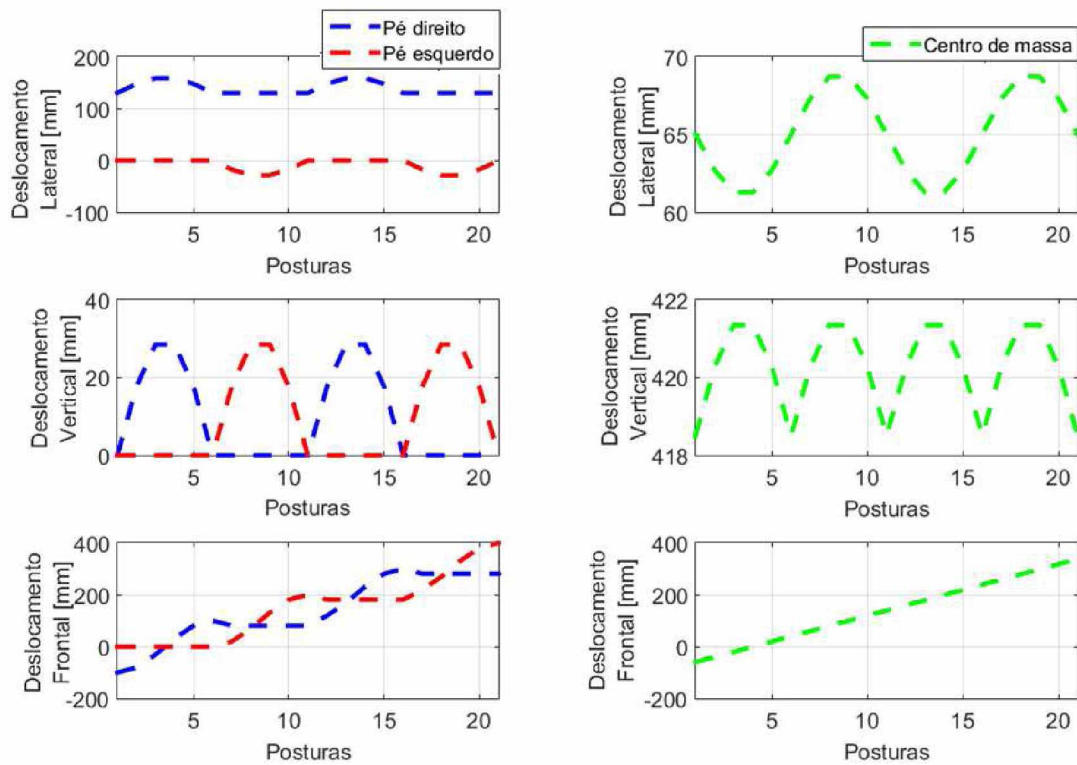


Figura 5.1. Deslocamento proposto para o centro de massa e para os pés.

A mesma trajetória proposta na Fig. 5.1 pode ser vista espacialmente na Fig. 5.2. Cabe salientar que o referencial inercial utilizado é definido na Seção 3.2.

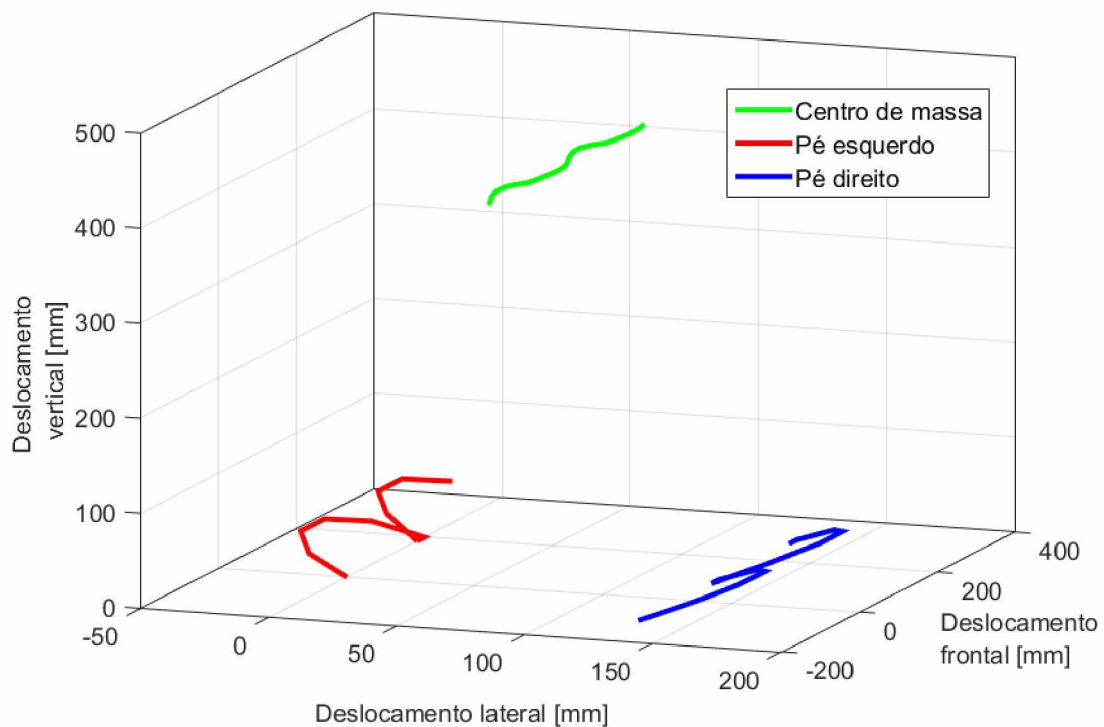


Figura 5.2. Visão espacial do deslocamento proposto para o centro de massa e pés.

Para a trajetória proposta, o centro de massa é posicionado entre os dois pés em ambas componentes frontal e lateral. Porém, é possível deslocar o centro de massa para frente ou para trás, fazendo com que se mova adiantado ou atrasado em relação aos pés. Além disso, é possível deslocar o centro de massa lateralmente, para que se mova sobre algum dos pés. Estas possibilidades de deslocamentos frontal e lateral do centro de massa em relação aos pés foram incluídos no modelo de maneira a generalizar a trajetória, permitindo que qualquer movimento específico seja definido. Um exemplo da necessidade de alterar a componente frontal ocorre quando o centro de massa precisa se deslocar atrasado em relação aos pés, necessário em casos que o robô tende a cair para frente, permitindo assim aumentar o equilíbrio.

A partir da trajetória proposta, são escolhidas as 10 primeiras posturas para a solução do problema cinemático inverso e aplicação da metodologia apresentada a seguir, visto que o movimento se repete conforme as Figs. 5.1 e 5.2. As coordenadas relativas às posturas estão apresentadas na Tab. 5.1.

Para resolver o problema cinemático inverso, além da definição dos deslocamentos, torna-se necessário definir as orientações dos pés e do centro de massa, assunto da próxima seção.

Tabela 5.1. Coordenadas dos deslocamentos do pé esquerdo e do centro de massa nas posturas da trajetória proposta.

	Coordenadas dos deslocamentos [mm] nas posturas:									
	1	2	3	4	5	6	7	8	9	10
Vertical-pé	0,0	0,0	0,0	0,0	0,0	0,0	17,6	30,0	30,0	17,6
Lateral-pé	0,0	0,0	0,0	0,0	0,0	0,0	-17,6	-30,0	-30,0	-17,6
Frontal-pé	0,0	0,0	0,0	0,0	0,0	0,0	19,1	69,1	130,9	180,0
Vertical-CM	418,5	420,3	421,5	421,5	420,3	418,5	420,3	421,5	421,5	420,3
Lateral-CM	65,0	61,3	61,0	61,0	61,3	65,0	68,7	69,0	69,0	68,7
Frontal-CM	-60,0	-40,1	-20,1	-0,2	19,6	39,5	59,4	79,3	99,2	120,0

5.1.2. Orientação do centro de massa e dos pés

De acordo com a Eq. (4.4.10), a orientação de um corpo rígido pode ser representada na forma de um dual quaternion, sendo necessário apenas definir o ângulo e a direção na qual ocorre a rotação. Desta forma a orientação dos pés e do centro de massa para a trajetória proposta, são apresentadas nas Tabs. 5.2 e 5.3, nas quais são definidos os ângulos de rotação em torno das direções lateral, vertical e frontal.

São apresentados apenas os valores das orientações do pé esquerdo. O pé direito assume as mesmas orientações do pé esquerdo: enquanto o pé esquerdo varia entre as posturas de 1 a 5, o pé direito varia entre as posturas de 6 a 10.

Tabela 5.2. Orientação do referencial do pé esquerdo.

Direção	Ângulos [Graus] nas posturas:									
	1	2	3	4	5	6	7	8	9	10
Lateral	0	0	0	0	0	0	1	0	-1	0
Vertical	0	0	0	0	0	0	0,2	0	-0,2	0
Frontal	0	0	0	0	0	0	0	0	0	0

Tabela 5.3. Orientação do referencial do centro de massa.

Direção	Ângulos [Graus] nas posturas:									
	1	2	3	4	5	6	7	8	9	10
Lateral	80	80	80	79	80	80	80	80	79	80
Vertical	90	89	88	89	90	90	91	92	91	90
Frontal	0	0	0	0	0	0	0	0	0	0

Observa-se que os valores de orientação do centro de massa possuem ângulos que variam próximos a 80 graus em torno da direção lateral e próximos a 90 graus em torno da direção vertical. Valores de 90 graus para ambas as orientações correspondem ao tronco totalmente ereto, conforme definição dos parâmetros de Denavit-Hartenberg, Fig. 3.3. Portanto, 80 graus em torno da direção lateral corresponde à uma pequena inclinação do tronco para frente, e variações do ângulo em torno da direção vertical correspondem à um pequeno giro do tronco para os lados.

5.3. Solução do modelo cinemático inverso para a trajetória proposta

Tendo sido proposta uma trajetória, é possível calcular os movimentos das articulações das pernas de maneira que resulte no movimento desejado para o robô humanoide.

A seguir o problema cinemático inverso do robô humanoide é resolvido aplicando o algoritmo proposto, sendo que todas as equações, utilizadas para realizar os cálculos desta seção, foram descritas no Capítulo IV.

Para cada uma das 10 posturas definidas, repete-se os passos apresentados e assim calcula-se o conjunto dos ângulos articulares para uma das pernas, sendo aqui

aplicado à perna esquerda. A perna direita segue na sequência que vai da 6ª até a 10ª postura e, em seguida, da 1ª à 5ª.

Para ilustrar a aplicação do algoritmo, são apresentados abaixo os valores numéricos obtidos para a solução da 7ª postura. Nesta postura tanto o pé esquerdo quanto o centro de massa se movem apoiados no pé direito.

Utilizando os parâmetros de Denavit-Hartenberg apresentados nas Tabs. 3.1 e 4.3, são definidos os comprimentos geométricos do robô, ilustrados na Fig. 3.3, como

$$a_1 = l_1 = 109.8 \text{ mm},$$

$$a_2 = l_2 = 202.5 \text{ mm},$$

$$a_3 = l_3 = 221 \text{ mm},$$

$$a_4 = l_4 = 65 \text{ mm},$$

$$a_5 = l_5 = 10 \text{ mm}.$$

Além do comprimento l_1 , que é a distância relativa do referencial Σ_B ao referencial Σ_1 , considera-se no modelo, para o cálculo da estabilidade do robô, a largura do pé, de 131 mm, e o comprimento total do pé, de 211.6 mm, que diz respeito à distância da ponta dos dedos à extremidade do calcanhar, como será discutido no Capítulo VII.

A posição e orientação do Σ_{CM} em relação a Σ_{6^*} é definida pela matriz de transformação ${}^{6^*}T_{CM}$, dada por

$${}^{6^*}T_{CM} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 65 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 10 & 0 & 0 & 1 \end{bmatrix}. \quad (5.3.1)$$

A matriz ${}^{6^*}T_{CM}$ é convertida para o dual quaternion $D_{CM/6^*}$ correspondente, a partir das Eqs. (4.5.3) e (4.5.4), resultando em

$$D_{CM/6^*} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ -32,5 \\ 0 \\ -5 \end{pmatrix}. \quad (5.3.2)$$

Como ambos os pés e o centro de massa possuem posições relativas ao referencial inercial, estando o pé direito parado em contato com o solo, obtém-se as matrizes 0T_B e ${}^0T_{CM}$, que fornecem a posição e orientação de Σ_B e Σ_{cm} ambos em relação ao referencial inercial Σ_0 . Os elementos das matrizes 0T_B e ${}^0T_{CM}$ são os valores definidos pela trajetória proposta para o pé de apoio e centro de massa respectivamente, conforme Tabs. 5.1, 5.2 e 5.3.

$${}^0T_B = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 17,6 & 1 & -0,0035 & 0 \\ -17,6 & 0,0035 & 1 & -0,0175 \\ 19,1 & 0 & 0,0175 & 1 \end{bmatrix}, \quad (5.3.3)$$

$${}^0T_{CM} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 420,26 & -0,01 & -0,170 & 0,9852 \\ 68,70 & 0,9998 & -0,0032 & 0,0185 \\ 59,42 & 0 & 0,9854 & 0,1703 \end{bmatrix}, \quad (5.3.4)$$

as quais podem ser representadas pelos dual quaternions

$$D_{B/0} = \begin{pmatrix} 1 \\ 0,0087 \\ 0 \\ 0,0017 \\ 0,0935 \\ -8,8009 \\ 8,7485 \\ -9,6258 \end{pmatrix}, \quad (5.3.5)$$

$$D_{CM/0} = \begin{pmatrix} 0,5358 \\ 0,4511 \\ 0,4597 \\ 0,5460 \\ 126,8153 \\ -117,6807 \\ 82,9118 \\ -97,0218 \end{pmatrix}. \quad (5.3.6)$$

Para a solução da cinemática inversa, o tamanho do pé não influencia o resultado, visto que a trajetória proposta para os pés é definida para o movimento do

tornozelo e não da ponta do pé. Desta forma, pode-se considerar que o referencial Σ_B coincide com o referencial Σ_1 , logo

$${}^B T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5.3.7)$$

o que resulta no dual quaternion,

$$D_{1/B} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \quad (5.3.8)$$

Posteriormente ao cálculo da cinemática inversa, pode-se alterar ${}^B T_1$ para calcular o movimento de Σ_B via cinemática direta. Esta mudança posterior de ${}^B T_1$ apenas translada Σ_1 em relação à Σ_B , já que não há nenhuma articulação entre estes referenciais. A transformação ${}^B T_1$ depende apenas do tamanho do pé, e desta forma a orientação de Σ_1 e Σ_B em relação ao referencial inercial são sempre iguais uma à outra.

Em seguida, calcula-se a matriz ${}^1 T_{6*}$, obtida pelas Eqs (4.8.3) e (4.8.7),

$${}^1 T_{6*} = ({}^B T_1)^{-1} ({}^0 T_B)^{-1} {}^0 T_{CM} ({}^{6*} T_{CM})^{-1}. \quad (5.3.9)$$

O dual quaternion T correspondente à matriz ${}^1 T_{6*}$, é dado por

$$T = \begin{pmatrix} 0,5407 \\ 0,4472 \\ 0,4637 \\ 0,5410 \\ 103,2209 \\ -103,1909 \\ 92,5049 \\ -97,1293 \end{pmatrix}, \quad (5.3.10)$$

o qual também pode ser obtido diretamente pelo produto dos quaternions

$$T = (D_{1/B})^{-1} (D_{B/0})^{-1} D_{CM/0} (D_{CM/6*})^{-1}. \quad (5.3.11)$$

Dividindo a estrutura serial 6R em duas estruturas seriais 3R, obtém-se o conjunto de 8 equações algébricas. As 4 equações que representam a estrutura 3R canônica são dadas pelas Eqs (4.6.10) a (4.6.13), que resultam em

$$221.0 v_2 x_0 - 221.0 x_3 - 2.0 y_2 - 202.5 x_2 - 202.5 v_2 x_1 + 2.0 v_2 y_1 = 0,$$

$$221.0 x_2 + 202.5 x_3 - 2.0 y_3 - 202.5 v_2 x_0 + 221.0 v_2 x_1 - 2.0 v_2 y_0 = 0,$$

$$202.5 v_2 x_2 - 202.5 x_1 - 2.0 y_1 - 221.0 x_0 - 221.0 v_2 x_3 - 2.0 v_2 y_2 = 0,$$

$$2.0 y_0 - 221.0 x_1 - 202.5 x_0 + 221.0 v_2 x_2 - 202.5 v_2 x_3 - 2.0 v_2 y_3 = 0, \quad (5.3.12)$$

as quais dependem explicitamente de v_2 .

Já as 4 equações que representam a estrutura esférica não canônica são dadas pelas Eqs. (4.7.1) a (4.7.4), que resultam em

$$206.44 x_0 - 206.39 x_1 + 184.98 x_2 - 194.25 x_3 + 1.08 y_0 + 0.89 y_1 + 0.92 y_2 + 1.08 y_3 = 0,$$

$$206.39 x_0 + 206.44 x_1 - 194.25 x_2 - 184.98 x_3 - 0.89 y_0 + 1.08 y_1 + 1.08 y_2 - 0.92 y_3 = 0,$$

$$194.25 x_1 - 184.98 x_0 + 206.44 x_2 - 206.39 x_3 - 0.92 y_0 - 1.08 y_1 + 1.08 y_2 + 0.89 y_3 = 0,$$

$$194.25 x_0 + 184.98 x_1 + 206.39 x_2 + 206.44 x_3 - 1.08 y_0 + 0.92 y_1 - 0.89 y_2 + 1.08 y_3 = 0,$$

(5.3.13)

as quais não dependem explicitamente de nenhum v_i , $i=4, \dots, 6$.

O determinante da matriz dos coeficientes dos hiperplanos representados pelas Eqs. (5.3.12) e (5.3.13), o qual foi definido pela Eq. (4.8.10), resulta em

$$- 51809550879.01 v_2^4 - 148185202142.55 v_2^3 + 1546312702760.45 v_2^2 - 148185202142.55 v_2 - 51809550879.01 = 0 . \quad (5.3.14)$$

O qual permite calcular os valores de v_2 . As 4 soluções são dadas por:

$$v_2 = \{ -0,1406 ; 4,1511 ; 0,2409 ; -7,1116 \} . \quad (5.3.15)$$

Para cada valor de v_2 são então calculados os valores de v_1, v_3, v_4, v_5 e v_6 .

A postura em que as estruturas serial 3R canônica e esférica não canônica se encontram, definida pelo dual quaternion S , é obtida pela solução do sistema formado pelas Eqs. (5.3.12) e (5.3.13), substituindo nestas equações um dos valores de v_2 .

Em seguida são apresentados os valores obtidos para $v_2 = -7,1116$. Os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 , são então calculados, o que resulta em

$$S = \begin{pmatrix} 1,0 \\ -0,98 \\ -0,24 \\ -0,19 \\ -201,06 \\ -199,77 \\ -30,03 \\ 19,63 \end{pmatrix} . \quad (5.3.16)$$

A solução de v_1 é obtida substituindo os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 pelos valores dos elementos de S na Eq. (4.6.14), o que resulta em

$$-10.44 v_1 - 402.66 = 0, \quad (5.3.17)$$

e, conseqüentemente,

$$v_1 = -38,5389 . \quad (5.3.18)$$

A solução de v_3 é obtida substituindo os valores de $x_0, x_1, x_2, x_3, y_0, y_1, y_2$ e y_3 pelos valores dos elementos de S na Eq. (4.6.18), o que resulta em

$$-383.83 v_3 - 144.34 = 0, \quad (5.3.19)$$

e assim

$$v_3 = -0,3760. \quad (5.3.20)$$

Os valores de v_4 , v_5 e v_6 são obtidos pela Eq. (4.8.11), a qual leva ao sistema de 8 equações, mas somente 3 linearmente independentes, ou seja:

$$\begin{aligned} \frac{v_5(v_4 - v_6)}{v_4 * v_6 - 1} &= -8.43, \\ \frac{-v_5(v_4 * v_6 + 1)}{v_4 * v_6 - 1} &= -1.23, \\ \frac{-(v_4 + v_6)}{v_4 * v_6 - 1} &= -8.1, \end{aligned} \quad (5.3.21)$$

e, conseqüentemente, as duas soluções para cada uma das incógnitas v_4 , v_5 e v_6 são

$$\begin{aligned} v_4 &= \{ -7,3889 ; 0,1353 \}, \\ v_5 &= \{ -1,0336 ; 1,0336 \}, \\ v_6 &= \{ -0,0117 ; 85,4213 \}. \end{aligned} \quad (5.3.22)$$

Sendo $v_i = \tan\left(\frac{u_i}{2}\right)$, $i=1,...,6$, é possível obter os ângulos das articulações em radianos, dados por u_i . As 8 soluções obtidas são apresentadas na Tab. 5.4.

Tabela 5.4. Soluções do modelo cinemático inverso para a sétima postura.

Articulação	Ângulos [Graus]:							
	Sol. 1	Sol. 2	Sol. 3	Sol. 4	Sol. 5	Sol. 6	Sol. 7	Sol. 8
u_1	3,0	3,0	-177,0	-177,0	3,0	3,0	-177,0	-177,0
u_2	-16,0	-16,0	152,9	152,9	27,1	27,1	-164,0	-164,0
u_3	41,2	41,2	41,2	41,2	-41,2	-41,2	-41,2	-41,2
u_4	164,6	-15,4	156,1	-23,9	-156,1	23,9	-164,6	15,4
u_5	-178,1	-1,9	-181,9	1,9	-178,1	-1,9	-181,9	1,9
u_6	178,7	-1,3	-1,3	178,7	178,7	-1,3	-1,3	178,7

É importante observar que na Tab. 5.4, ao valor do parâmetro u_5 está adicionado de 90° , desta forma, o valor do ângulo do giro da articulação é o valor encontrado como solução do modelo cinemático inverso subtraído de 90° , o que deve ser feito devido aos parâmetros de Denavit-Hartenberg obtidos conforme Tab. 4.3.

Além disto, é necessário implementar uma última etapa no algoritmo de solução do modelo cinemático inverso afim de selecionar uma das soluções encontradas para realizar os testes experimentais com o robô. Para tanto, observa-se que 4 das soluções encontradas apresentam um valor positivo para u_3 . Na prática, isto significa que a rotação do joelho ocorreria para frente, o que fisicamente não ocorre, e desta forma podem ser eliminadas 4 soluções. Um segundo critério adotado para escolher apenas uma das soluções se baseia no fato de que nenhuma articulação atinge um ângulo maior que 120° durante o movimento de marcha semelhante à humana. Desta forma, outras 3 soluções são eliminadas, restando somente uma, a solução final escolhida.

A solução escolhida para a 7ª postura portanto é a Solução 6 da Tab. 5.4. As soluções escolhidas para cada uma das 10 posturas, todas calculadas de maneira semelhante à 7ª postura, são apresentadas na Tab. 5.5.

Tabela 5.5. Solução do modelo cinemático inverso para a trajetória proposta.

Articulação	Ângulos [Graus] nas posturas:									
	1	2	3	4	5	6	7	8	9	10
u_1	0,0	-0,5	-0,5	-0,5	-0,5	0,5	3,0	4,8	3,4	0,0
u_2	4,7	8,7	12,1	15,4	18,2	20,3	27,1	26,6	17,0	4,2
u_3	-25,4	-27,7	-29,0	-30,0	-30,1	-28,7	-41,2	-48,5	-41,6	-25,0
u_4	30,5	28,9	26,7	24,8	21,7	18,2	23,9	31,6	34,7	30,6
u_5	-0,1	-0,7	-1,3	-0,7	0,4	-0,6	-1,9	-3,0	-2,2	-0,1
u_6	0,0	-0,1	-0,2	-0,1	0,1	-0,1	-1,3	-0,5	0,6	0,0

CAPÍTULO VI

INTERPOLAÇÃO DO MOVIMENTO

Dada a trajetória discretizada em posturas, é possível determinar os valores dos ângulos nas articulações das pernas do robô humanoide, aplicando o algoritmo proposto para solução do problema cinemático inverso. No entanto, é necessário conhecer o movimento dos atuadores responsáveis por movimentar as articulações durante toda a trajetória do robô ao longo do tempo.

Para tanto, além de valores discretos dos ângulos das articulações, deve-se calcular também os valores das velocidades e acelerações angulares dos atuadores.

Isto significa que é necessário definir as equações que descrevem o movimento dos atuadores para qualquer instante de tempo, ou seja, não apenas para as 10 posturas propostas.

As respostas para estas questões podem ser obtidas a partir da interpolação do movimento proposto ao robô humanoide a partir das posturas que definem seu movimento. A interpolação pode ser feita tanto no espaço Cartesiano quanto no espaço das juntas. No espaço Cartesiano, a interpolação consiste em definir funções temporais que definem os deslocamentos e orientações dos pés e do centro de massa. Já no espaço das juntas, a interpolação é feita definindo funções temporais para os ângulos, velocidades e acelerações angulares dos atuadores.

A seguir são apresentados dois métodos para interpolar o movimento do robô humanoide no espaço das juntas. Algumas vantagens e desvantagens de cada método são discutidas.

O método de interpolação no espaço das juntas é avaliado em relação à sua influência na aceleração do centro de massa, parâmetro importante para analisar o equilíbrio dinâmico do movimento proposto.

6.1. Interpolação no espaço Cartesiano

No espaço Cartesiano, a interpolação do movimento é feita definindo funções temporais que representam as componentes espaciais do deslocamento do centro de massa e dos pés. Essas funções interpolam os pontos que definem as trajetórias propostas, fazendo com que o robô passe pelas posturas definidas durante seu movimento.

Em Kajita *et al.* (2014) é proposto uma trajetória do tipo seno hiperbólico para o movimento lateral e frontal do centro de massa. A orientação é mantida sempre paralela ao solo, e é definida uma amplitude nula para o deslocamento vertical do centro de massa. Os pés se movem sempre paralelos ao solo, e as componentes dos deslocamentos frontal e vertical são interpoladas linearmente. Esta trajetória é proposta de maneira a simplificar o modelo cinemático inverso utilizado.

Uma outra possibilidade de interpolação do movimento no espaço Cartesiano é proposta em Gferrer (1999-a, 1999-b, 2000). Nestes trabalhos é apresentado um método denominado *QB-curves* para interpolar dual quaternions que estão contidos na Quádrica de Study, condição necessária para representar o movimento de corpos rígidos através de dual quaternions. Os resultados encontrados mostram uma das maiores vantagens de trabalhar com os dual quaternions, qual seja, permitem uma interpolação levando em conta simultaneamente tanto a translação quanto a orientação.

Klawitter (2010) implementa em Matlab um algoritmo para a realização da interpolação proposta por Gferrer, sendo possível aplica-lo para interpolar os pontos de uma trajetória após sua representação por dual quaternions.

Porém, a desvantagem de interpolar o movimento do robô humanoide no espaço Cartesiano se deve à necessidade de aplicar o modelo cinemático inverso para cada nova postura interpolada. Em caso de uma grande quantidade de novas posturas serem definidas, esta abordagem se torna ineficiente, visto que, em aplicações que requerem processamento em tempo real, o custo computacional para repetir os cálculos várias vezes é elevado.

Além disso, torna-se necessário realizar cálculos posteriores para obter as velocidades e acelerações angulares dos atuadores. Estas informações são necessárias para controlar os movimentos articulares, garantindo que o movimento executado seja igual ao especificado.

Considerando essas desvantagens, optou-se pela interpolação do movimento no espaço das juntas.

6.2. Interpolação no espaço das juntas

Interpolar o movimento das articulações consiste em definir funções para descrever os ângulos, velocidades e acelerações angulares dos atuadores em qualquer instante de tempo da trajetória proposta.

Esta abordagem é mais eficiente, pois basta aplicar o modelo cinemático inverso para as 10 posturas propostas, independente da quantidade desejada de posturas interpoladas.

Então, neste trabalho, os movimentos angulares das articulações são definidos de maneira a interpolar os valores apresentados na Tab. 5.5. As funções polinomiais são definidas para representar os ângulos das articulações durante o movimento, e podem ser algebricamente derivadas para obter as velocidades e acelerações angulares.

Duas propostas são apresentadas e comparadas, sendo a primeira uma interpolação linear e a segunda uma interpolação polinomial de 5ª ordem para os valores dos ângulos articulares.

6.2.1. Interpolação linear

A interpolação linear está sendo usada para comparação pois é comumente utilizada para o controle de robôs humanoides devido sua facilidade de implementação. Neste caso, considera-se que os ângulos nas articulações variam com velocidade constante entre duas posturas consecutivas.

O valor da velocidade angular da i -ésima articulação, entre duas posturas $u_i(t_0)$ e $u_i(t_f)$ consecutivas é dado por

$$\dot{u}_i = \frac{u_i(t_f) - u_i(t_0)}{\Delta t}. \quad (6.2.1)$$

O intervalo de tempo entre as duas posturas que se deseja interpolar é definido por Δt ,

$$\Delta t = t_f - t_0, \quad (6.2.2)$$

sendo $u_i(t_0)$ o valor de u_i , $i=1, \dots, 6$, calculado pelo modelo cinemático inverso para a primeira postura do intervalo que se deseja interpolar e $u_i(t_f)$ o valor de u_i calculado pelo modelo cinemático inverso para a postura seguinte do intervalo.

Desta forma, a equação que permite calcular os ângulos dos atuadores para qualquer instante de tempo entre duas posturas dadas é definida por

$$u_i(t) = u_i(t_0) + t \cdot \dot{u}_i, \quad (6.2.3)$$

sendo $t_0 < t < t_f$.

Apesar da articulação se mover teoricamente com aceleração nula entre duas posturas consecutivas, devido ao fato de existir descontinuidade na curva de velocidade, na realidade, essa transição tem que ser feita em um intervalo de tempo muito pequeno, provocando “picos” da curva de aceleração. Desta forma são introduzidas perturbações dinâmicas no sistema, que podem afetar a estabilidade dinâmica do robô humanoide.

6.2.2. Interpolação polinomial

A interpolação polinomial do movimento das articulações é proposta afim de evitar as perturbações dinâmicas provocadas pela interpolação linear e, desta forma, suavizar o movimento.

Um polinômio de 5ª ordem é utilizado para definir os ângulos dos atuadores em um instante de tempo entre duas posturas dadas, conforme Eq. (6.2.4). Consequentemente, as velocidades e acelerações angulares podem ser analiticamente definidas pelas Eqs. (6.2.5) e (6.2.6), respectivamente:

$$u_i = A_i t^5 + B_i t^4 + C_i t^3 + D_i t^2 + E_i t + F_i, \quad (6.2.4)$$

$$\dot{u}_i = 5A_i t^4 + 4B_i t^3 + 3C_i t^2 + 2D_i t + E_i, \quad (6.2.5)$$

$$\ddot{u}_i = 20A_i t^3 + 12B_i t^2 + 6C_i t + 2D_i. \quad (6.2.6)$$

A escolha de um polinômio de 5ª ordem se justifica pela possibilidade de interpolar o movimento das articulações utilizando as seis condições iniciais e finais de ângulo, velocidades e acelerações angulares. Desta forma, é possível calcular os coeficientes A_i , B_i , C_i , D_i , E_i e F_i , $i=1, \dots, 6$. As 6 condições são definidas por

$$1- \text{ Para } t = 0 \rightarrow u_i = u_i(t_0) = u_{i0}, \quad (6.2.7)$$

$$2- \text{ Para } t = \Delta t \rightarrow u_i = u_i(t_f) = u_{if}, \quad (6.2.8)$$

$$3- \text{ Para } t = 0 \rightarrow \dot{u}_i = \dot{u}_i(t_0) = \dot{u}_{i0}, \quad (6.2.9)$$

$$4- \text{ Para } t = \Delta t \rightarrow \dot{u}_i = \dot{u}_i(t_f) = \dot{u}_{if}, \quad (6.2.10)$$

$$5- \text{ Para } t = 0 \rightarrow \ddot{u}_i = \ddot{u}_i(t_0) = \ddot{u}_{i0}, \quad (6.2.11)$$

$$6- \text{ Para } t = \Delta t \rightarrow \ddot{u}_i = \ddot{u}_i(t_f) = \ddot{u}_{if}, \quad (6.2.12)$$

sendo $0 < t < \Delta t$, considerando $t_0 = 0$.

Os valores de $u_i(t_0)$ e $u_i(t_f)$ são calculados pelo modelo cinemático inverso, analogamente ao caso linear. Já os valores das acelerações angulares \ddot{u}_{i0} e \ddot{u}_{if} são definidos como nulos para as condições iniciais e finais. Esta escolha é feita para que os atuadores iniciem o movimento com uma aceleração nula e retornem à mesma condição no final do intervalo, garantindo que não haja descontinuidade na aceleração angular entre intervalos de posturas consecutivos.

As velocidades angulares podem, de maneira semelhante, ser definidas nulas para a primeira postura que o robô irá executar, saindo do repouso a partir de uma velocidade nula. O mesmo pode ser definido para a última postura realizada ao longo do movimento, e assim o robô finaliza a trajetória desacelerando até parar.

As velocidades angulares dos atuadores nos instantes das posturas a serem interpoladas são definidas iguais à velocidade média calculada pela Eq. (6.2.1). Observa-se que apesar dos valores instantâneos de $\dot{u}_i(t_0)$ e $\dot{u}_i(t_f)$ serem iguais aos valores da interpolação linear, tem-se que a interpolação polinomial garante que a transição da velocidade seja feita de maneira suave, definida por polinômios de 4ª ordem e não por uma velocidade constante.

Substituindo as condições definidas pelas Eqs. (6.2.7) a (6.2.12), é possível calcular os coeficientes A_i , B_i , C_i , D_i , E_i e F_i , $i=1, \dots, 6$, o que resulta em

$$\begin{aligned}
u_i = & \left[\frac{-3(\dot{u}_{if} + \dot{u}_{i0})}{\Delta t^4} + \frac{6(u_{if} - u_{i0})}{\Delta t^5} \right] t^5 + \left[\frac{(7\dot{u}_{if} + 8\dot{u}_{i0})}{\Delta t^3} - \frac{15(u_{if} - u_{i0})}{\Delta t^4} \right] t^4 \\
& + \left[\frac{(-4\dot{u}_{if} - 6\dot{u}_{i0})}{\Delta t^2} + \frac{10(u_{if} - u_{i0})}{\Delta t^3} \right] t^3 + \dot{u}_{i0}t + u_{i0},
\end{aligned}
\tag{6.2.13}$$

$$\begin{aligned}
\dot{u}_i = & \left[\frac{-15(\dot{u}_{if} + \dot{u}_{i0})}{\Delta t^4} + \frac{30(u_{if} - u_{i0})}{\Delta t^5} \right] t^4 + \left[\frac{(28\dot{u}_{if} + 32\dot{u}_{i0})}{\Delta t^3} - \frac{60(u_{if} - u_{i0})}{\Delta t^4} \right] t^3 \\
& + \left[\frac{(-12\dot{u}_{if} - 18\dot{u}_{i0})}{\Delta t^2} + \frac{30(u_{if} - u_{i0})}{\Delta t^3} \right] t^2 + \dot{u}_{i0},
\end{aligned}
\tag{6.2.14}$$

$$\begin{aligned}
\ddot{u}_i = & \left[\frac{-60(\dot{u}_{if} + \dot{u}_{i0})}{\Delta t^4} + \frac{120(u_{if} - u_{i0})}{\Delta t^5} \right] t^3 + \left[\frac{(84\dot{u}_{if} + 96\dot{u}_{i0})}{\Delta t^3} - \frac{180(u_{if} - u_{i0})}{\Delta t^4} \right] t^2 \\
& + \left[\frac{(-24\dot{u}_{if} - 36\dot{u}_{i0})}{\Delta t^2} + \frac{60(u_{if} - u_{i0})}{\Delta t^3} \right] t.
\end{aligned}
\tag{6.2.15}$$

A interpolação polinomial é então aplicada para a trajetória proposta, o que determina as equações que descrevem os ângulos, velocidades e acelerações angulares das articulações das pernas do robô humanoide. Estas são as equações utilizadas nos testes apresentados no Capítulo VII.

6.3. Interpolação do movimento do robô humanoide

Com o objetivo de comparar os métodos de interpolações linear e polinomial aplicados ao movimento das articulações, são apresentados na Fig. 6.1 os valores dos ângulos interpolados por ambos os métodos para o intervalo entre as 5 primeiras posturas da trajetória proposta. Os valores foram calculados utilizando as Eqs. (6.2.13) e (6.2.3), para valores de tempo incrementados de dt , sendo

$$dt = \frac{\Delta t}{10}, \tag{6.3.1}$$

o que acrescenta 10 intervalos entre duas posturas dadas, totalizando 41 pontos interpolados entre as 5 primeiras posturas da trajetória proposta.

A partir de experimentos com os protótipos dos robôs humanoides disponíveis no Laboratório de Automação e Robótica da Universidade Federal de Uberlândia, foi definido um período de 0,4 seg para um passo. Para análise da metade de um passo, correspondente às cinco primeiras posturas, têm-se $\Delta t = 0,05$ seg e $dt = 0,005$ segundos, conforme a Eq. (6.3.1).

Considerando o comprimento do passo de 180 mm, definido na Seção 5.1.1, tem-se que o robô humanoide se desloca a uma velocidade linear de 450 mm/seg.

Aparentemente, na Fig. 6.1-a, o perfil das variações angulares da interpolação linear e polinomial são coincidentes. Isso ocorre devido à escala de representação das 6 variáveis articulares u_i ($i=1, \dots, 6$). Para salientar a diferença dos perfis das curvas, na Fig. 6.1-b é apresentada isoladamente a variação angular de u_5 para as 5 primeiras posturas.

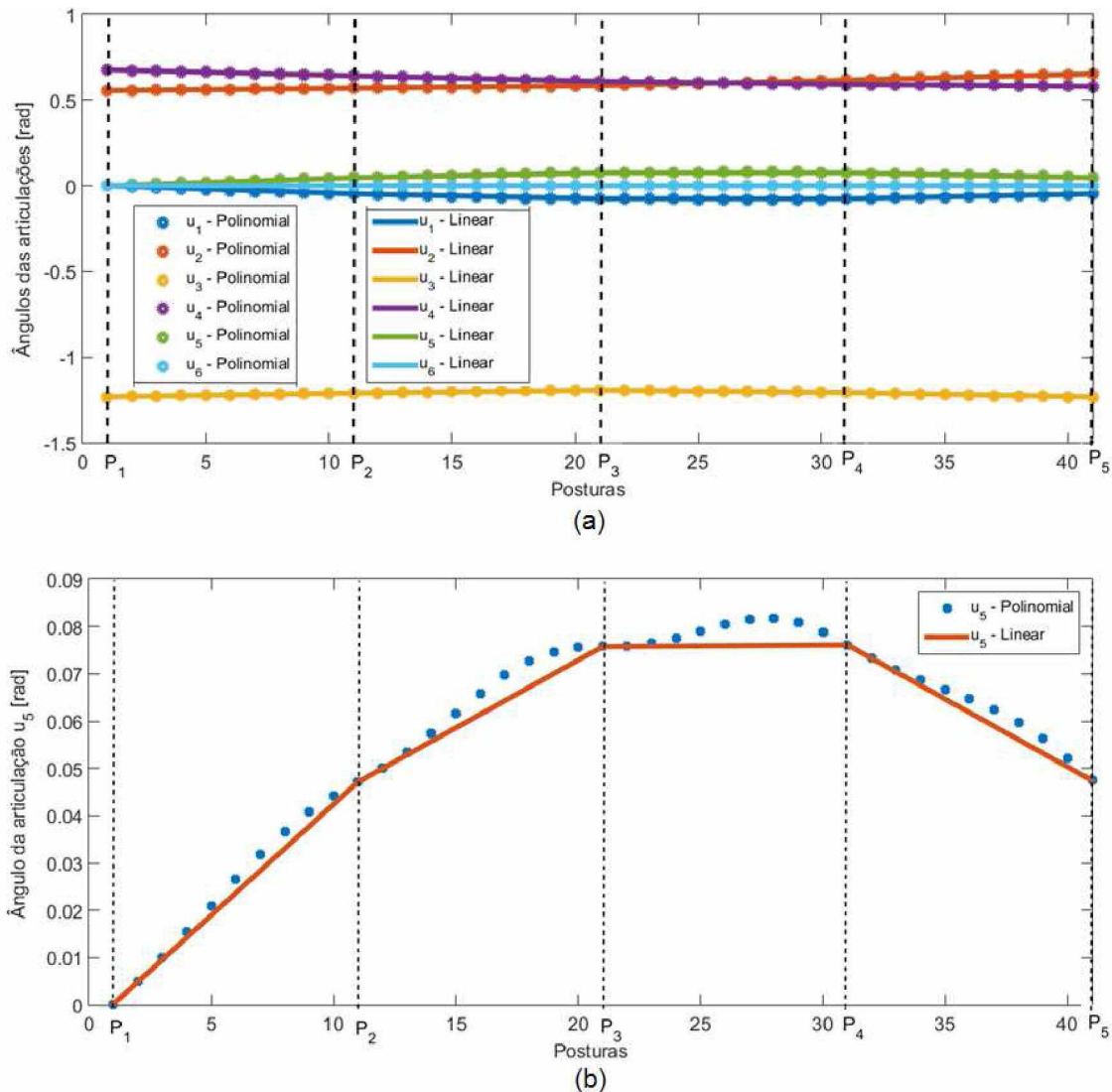


Figura 6.1. Comparação entre os ângulos das articulações para interpolação linear e polinomial: (a) Para todas variáveis articulares; (b) Para a variável articular u_5 .

De maneira semelhante, utilizando as Eqs. (6.2.1) e (6.2.14), foram interpoladas as velocidades angulares das articulações pelos métodos linear e polinomial, resultando nos valores apresentados na Fig. 6.2.

As acelerações angulares da interpolação polinomial são apresentadas na Fig. 6.3, utilizando a Eq. (6.2.15). Conforme já comentado anteriormente, as acelerações para a interpolação linear, definida pela Eqs. (6.2.1), são nulas.

As curvas das velocidades e acelerações angulares, apresentadas nas Figs 6.2 e 6.3, foram agrupadas de forma a facilitar a visualização.

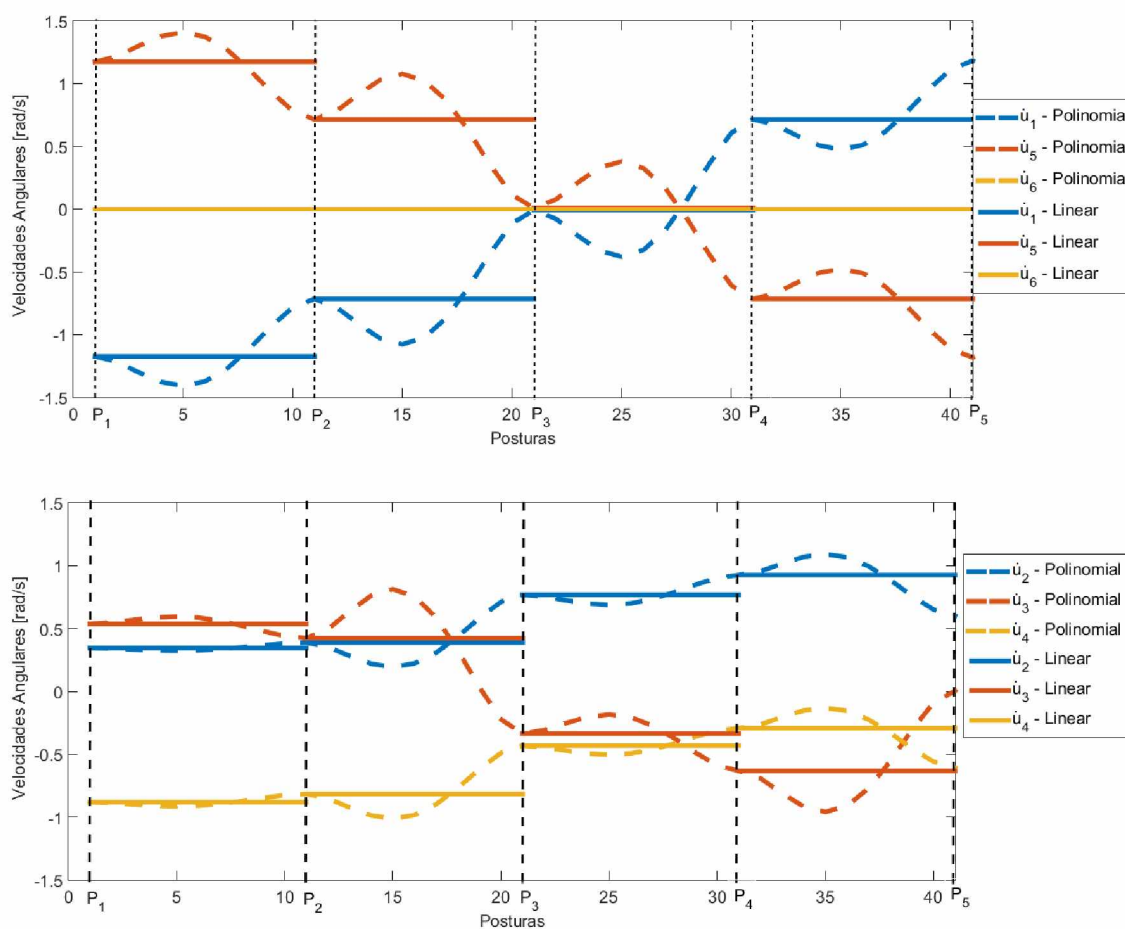


Figura 6.2. Comparação entre as velocidades angulares das articulações para interpolação linear e polinomial.

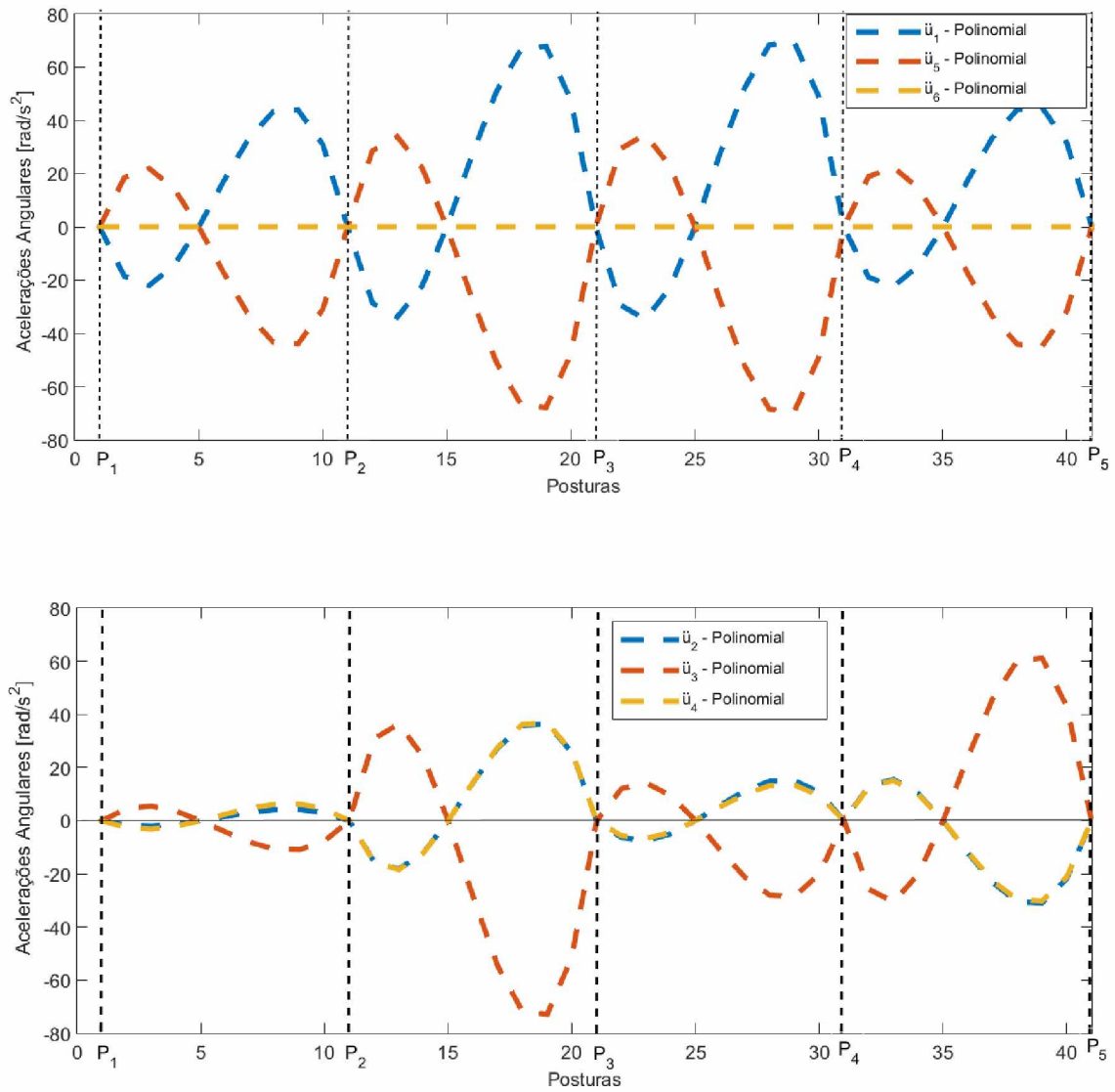


Figura 6.3. Comparação entre as acelerações angulares das articulações para a interpolação polinomial.

É possível ainda comparar as curvas de aceleração do centro de massa após interpolar os movimentos articulares linear e polinomialmente. Para o cálculo desta aceleração é utilizado o modelo cinemático descrito a seguir.

A aceleração linear do centro de massa escrito em relação ao referencial inercial é dada por

$${}^0\mathbf{a}_{CM} = {}^0\mathbf{a}_B + {}^0\mathbf{a}_{CM/B}, \quad (6.3.2)$$

sendo os vetores

${}^0\mathbf{a}_{CM}$: aceleração linear do centro de massa escrito em relação ao referencial inercial;
 ${}^0\mathbf{a}_B$: aceleração linear do referencial de base em relação ao referencial inercial;
 ${}^0\mathbf{a}_{CM/B}$: vetor da aceleração linear relativa do centro de massa do robô humanoide em relação ao referencial de base, escrito no referencial inercial.

Os vetores aceleração linear e posição do centro de massa e do pé de apoio, em relação ao referencial inercial, são apresentados na Fig. 6.4, nos instantes (t_1) e (t_2) .

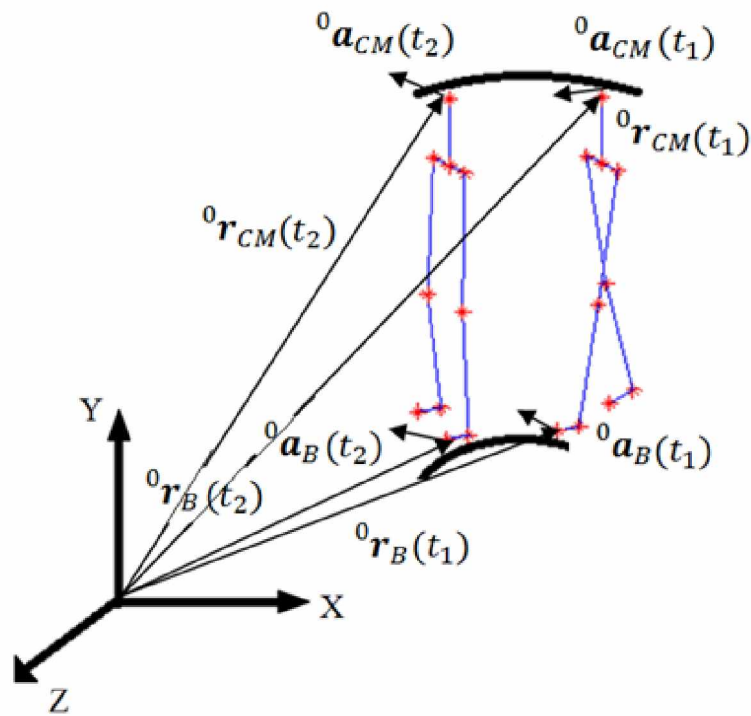


Figura 6.4. Posições e acelerações dos referenciais de base e do centro de massa escritos em relação ao referencial inercial.

Na Figura 6.4, os índices t_i indicam a relação temporal dos vetores das posições e acelerações lineares. De maneira semelhante à aceleração, tem-se

${}^0\mathbf{r}_{CM}$: posição do centro de massa escrito em relação ao referencial inercial;
 ${}^0\mathbf{r}_B$: posição do referencial de base escrito em relação ao referencial inercial.

A aceleração ${}^0\mathbf{a}_{CM/B}$ depende do movimento angular das articulações, isto é, dos ângulos, velocidades e acelerações angulares dos atuadores da perna de apoio. A

influência do movimento dos braços, da perna em movimento, assim como o efeito de esforços externos dados por folgas e atrito do pé de apoio com o solo, não são considerados nesta análise. O objetivo do modelo aqui apresentado é calcular o efeito que o movimento das articulações da perna de apoio tem sobre a aceleração linear do centro de massa escrito em relação a um referencial inercial posicionado fora do robô humanoide.

Ao observar a influência do movimento da perna de apoio na aceleração do centro de massa, torna-se possível identificar as diferenças obtidas com as interpolações linear e polinomial aplicadas aos ângulos, velocidades e acelerações angulares das articulações.

Para calcular a aceleração do centro de massa, é utilizado o método iterativo proposto por Craig (2005) para estruturas seriais, que no caso de possuir apenas articulações rotativas, é dado pelas equações

$${}^0\omega_i = {}^0\omega_{i-1} + {}^0\dot{u}_i, \quad (6.3.3)$$

$${}^{i-1}\mathbf{r}_i = l_{i-1}, \quad (6.3.4)$$

$${}^0\mathbf{r}_{i/i-1} = {}^0R_{i-1} {}^{i-1}\mathbf{r}_i, \quad (6.3.5)$$

$${}^0\dot{\omega}_i = {}^0\dot{\omega}_{i-1} + ({}^0\ddot{u}_i + {}^0\omega_{i-1} \times {}^0\dot{u}_i), \quad (6.3.6)$$

$${}^0\mathbf{a}_i = {}^0\mathbf{a}_{i-1} + {}^0\dot{\omega}_{i-1} \times {}^0\mathbf{r}_{i/i-1} + {}^0\omega_{i-1} \times ({}^0\omega_{i-1} \times {}^0\mathbf{r}_{i/i-1}), \quad (6.3.7)$$

no qual \times denota o produto vetorial, $i = 1, \dots, 6$, variando da primeira articulação posicionada no pé de apoio até a sexta articulação posicionada na cintura, conforme Fig. 3.3, e com os parâmetros de Denavit-Hartenberg dados na Tab. 4.3, onde

${}^0\mathbf{r}_{i/i-1}$: vetor da posição relativa da origem do referencial i , O_i , em relação à O_{i-1} , escrito no referencial inercial;

${}^{i-1}\mathbf{r}_i$: vetor que define a posição da origem do referencial i , O_i , escrito em relação ao referencial $i - 1$;

l_{i-1} : comprimento da normal comum aos eixos das articulações i e $i - 1$;

${}^0\omega_{i-1}$: vetor velocidade angular do corpo $i - 1$ escrito no referencial inercial.

${}^0\dot{\omega}_{i-1}$: vetor aceleração angular do corpo $i - 1$ escrito no referencial inercial.

0a_i : vetor que define a aceleração linear da origem do referencial i , O_i , escrito em relação ao referencial inercial.

Aplicando-se as Eqs. (6.3.3) a (6.3.7) para a primeira articulação, obtém-se

$${}^0a_1 = {}^0a_B + {}^0\dot{\omega}_B \times {}^0r_{1/B} + {}^0\omega_B \times ({}^0\omega_B \times {}^0r_{1/B}), \quad (6.3.8)$$

$${}^0\omega_1 = {}^0\omega_B + {}^0\dot{u}_1, \quad (6.3.9)$$

$${}^0\dot{\omega}_1 = {}^0\dot{\omega}_B + ({}^0\ddot{u}_1 + {}^0\omega_B \times {}^0\dot{u}_1). \quad (6.3.10)$$

Considera-se que o pé de apoio está parado em contato com o solo e com aceleração angular nula, o que resulta em

$${}^0a_1 = 0, \quad (6.3.11)$$

$${}^0\omega_1 = {}^0\dot{u}_1, \quad (6.3.12)$$

$${}^0\dot{\omega}_1 = {}^0\ddot{u}_1, \quad (6.3.13)$$

sendo ${}^0\dot{u}_1$ e ${}^0\ddot{u}_1$ os vetores velocidade e aceleração angulares da primeira articulação, cujos módulos são obtidos pelas Eqs. (6.2.14) e (6.2.15), através da interpolação polinomial, e as direções de ${}^0\dot{u}_1$ e ${}^0\ddot{u}_1$ dadas pela direção do eixo da articulação onde é posicionado o referencial Σ_1 , escritos em relação ao referencial inercial.

Para a segunda articulação, aplicando as Eqs. (6.3.3) a (6.3.7), obtém-se

$${}^0a_2 = {}^0a_1 + {}^0\dot{\omega}_1 \times {}^0r_{2/1} + {}^0\omega_1 \times ({}^0\omega_1 \times {}^0r_{2/1}), \quad (6.3.14)$$

$${}^0\omega_2 = {}^0\omega_1 + {}^0\dot{u}_2, \quad (6.3.15)$$

$${}^0\dot{\omega}_2 = {}^0\dot{\omega}_1 + ({}^0\ddot{u}_2 + {}^0\omega_1 \times {}^0\dot{u}_2), \quad (6.3.16)$$

o que leva a

$${}^0\mathbf{a}_2 = {}^0\mathbf{a}_1, \quad (6.3.17)$$

$${}^0\boldsymbol{\omega}_2 = {}^0\dot{\mathbf{u}}_1 + {}^0\dot{\mathbf{u}}_2, \quad (6.3.18)$$

$${}^0\dot{\boldsymbol{\omega}}_2 = {}^0\dot{\boldsymbol{\omega}}_1 + ({}^0\dot{\mathbf{u}}_2 + {}^0\dot{\mathbf{u}}_1 \times {}^0\dot{\mathbf{u}}_2). \quad (6.3.19)$$

As Eqsuações (6.3.3) a (6.3.7) são então aplicadas de maneira iterativa até se obter a velocidade angular e as acelerações lineares e angulares do centro de massa escritas no referencial inercial.

As componentes da aceleração do centro de massa calculadas a partir dos movimentos das articulações da perna de apoio, obtidos com as interpolações linear e polinomial, são apresentadas na Fig. 6.4.

Observa-se na Figura 6.4 que as componentes da aceleração do centro de massa sofrem descontinuidade em suas amplitudes quando utilizada uma interpolação linear para os movimentos das articulações. Cabe salientar que estas curvas são funções das velocidades consideradas constantes nos intervalos entre as posturas interpoladas. No entanto, se for considerado que a variação da velocidade, entre dois intervalos consecutivos interpolados, ocorre em um intervalo de tempo infinitesimal, consequentemente ocorrerá um pico instantâneo da aceleração articular. Já para a interpolação polinomial, a aceleração possui curvas suaves.

Apesar dos movimentos mais suaves, é necessário verificar se a trajetória obtida com a interpolação polinomial dos movimentos articulares é dinamicamente estável, isto é, se o robô se mantém em equilíbrio ao longo de sua caminhada. A avaliação de estabilidade é feita no Capítulo VII, comparando os resultados teóricos com os de simulações computacionais feitas com um *software* para análises multicorpos, além da comparação com os resultados de testes experimentais realizados.

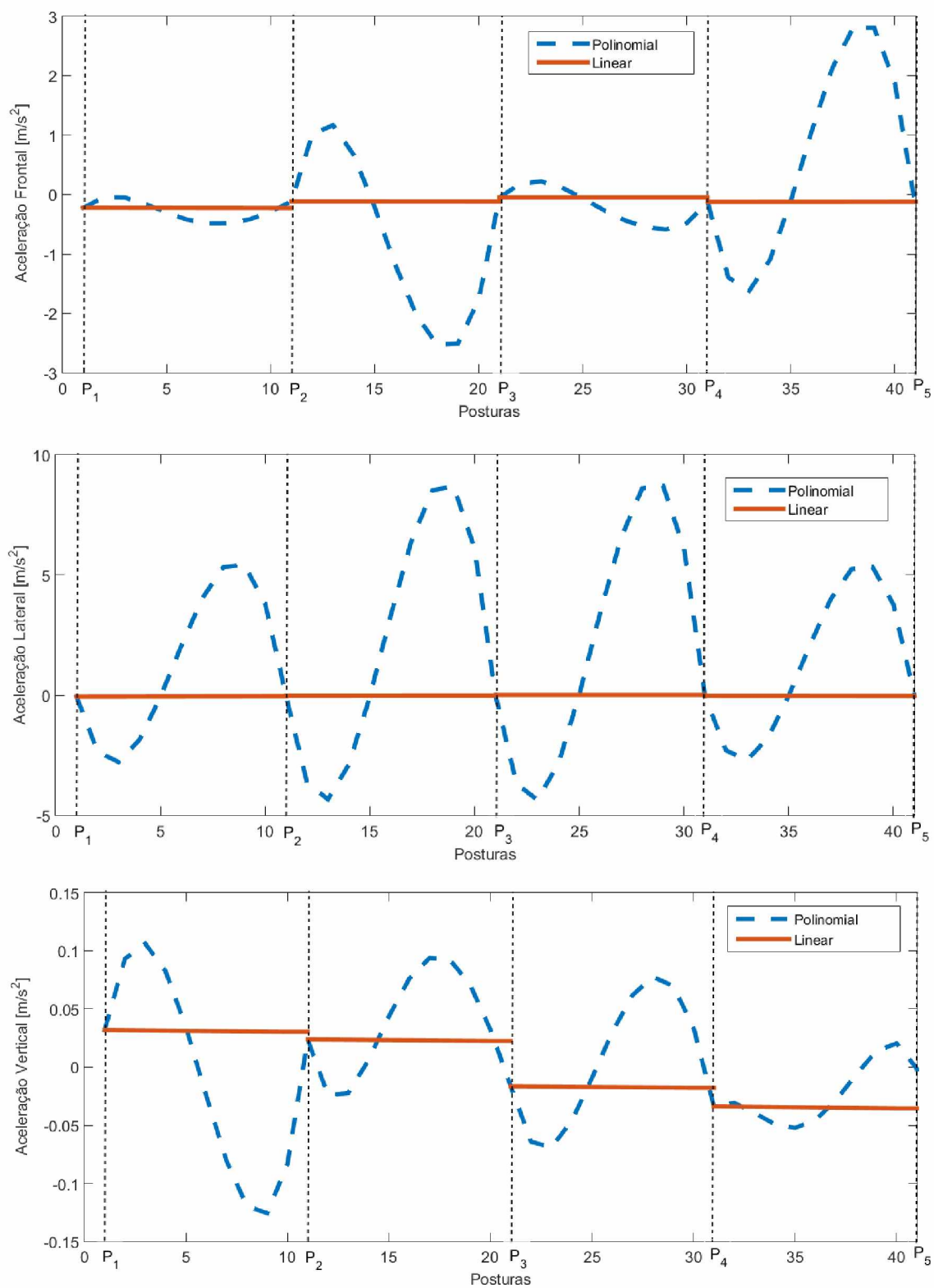


Figura 6.4. Comparação entre as acelerações do centro de massa para interpolação linear e polinomial dos movimentos articulares.

CAPÍTULO VII

ESTABILIDADE DA TRAJETÓRIA PROPOSTA

O objetivo deste capítulo é apresentar os resultados das avaliações teórica, computacional e experimental da estabilidade dos movimentos propostos ao robô humanoide.

É importante destacar que a estabilidade do movimento depende da trajetória dos pés e do centro de massa. No caso teórico, a trajetória do centro de massa é definida exclusivamente pelo movimento dos atuadores nas articulações da perna de apoio, conforme pode ser observado nas equações apresentadas na Seção 6.3.

No entanto, como será observado tanto nos resultados das simulações computacionais e quanto nos testes experimentais, esforços externos como o atrito entre o pé de apoio e o solo e também as folgas nos atuadores, contribuem para alterações na trajetória proposta. Desta forma, para uma trajetória ideal, todos os elementos que, de uma forma ou de outra, afetam o comportamento do robô, deveriam ser levados em consideração. Esta abordagem se torna inviável principalmente em casos de perturbações imprevistas vindas da interação do robô com o meio em que caminha. Uma proposta de análise teórica para esses casos é apresentada no Capítulo VIII.

Apesar disto, a verificação da estabilidade do movimento projetado pode ser observada diretamente durante os testes, a medida que o robô caminha sem cair. A seguir são apresentados os resultados obtidos.

7.1. Avaliação teórica

7.1.1. Método do Ponto de Momento Nulo

O Método do Ponto de Momento Nulo, do Inglês *Zero Moment Point* (ZMP), foi proposto por Vukobratovic´ em 1969 e, desde então, tornou-se um dos principais

métodos utilizados no projeto de robôs móveis, conforme é descrito pelo próprio Vukobratovic' e Borovac (2004).

Dado o caráter geral do método, é possível fazer algumas considerações no modelo dinâmico do robô humanoide conforme proposto em Han (2012), Choi *et al.* (2004), Hemami e Golliday (1976), e Erbatur e Kurt (2006). Nestes trabalhos é apresentado como calcular a trajetória do ZMP considerando a massa do sistema concentrada no centro de massa. Desta forma, basta conhecer o movimento do centro de massa e dos pés para que seja possível avaliar a estabilidade da trajetória proposta.

Para avaliar a estabilidade do movimento, deve-se comparar a trajetória calculada do ZMP com a posição do polígono de apoio do robô humanoide. A posição do polígono de apoio é definida pela região de contato entre o pé de apoio e o solo, na fase de simples apoio da caminhada do robô, ou ainda pela região definida pelos dois pés durante a fase de duplo apoio da caminhada do robô, apresentadas na Fig. 7.1.

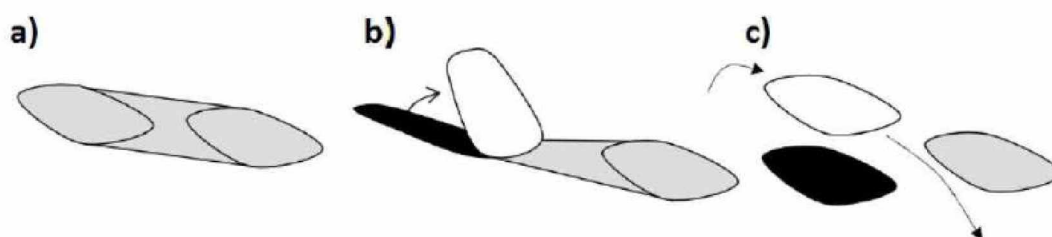


Figura 7.1. Polígono de apoio representado pela região cinza. (a) Fase de Duplo Apoio; (b) Parcialmente na Fase de Duplo Apoio e (c) Fase de Simples Apoio do pé esquerdo (DEKKER, 2009).

A Figura 7.2 apresenta uma ilustração dos esforços que agem sobre o pé de apoio durante a fase de simples apoio. A Figura 7.2-a apresenta a posição do ZMP, definido como ponto P , e o esforço de reação com o solo R . A Figura 7.2-b apresenta os esforços resultantes M_A e F_A , no ponto A , o qual coincide com as origens dos referenciais Σ_1 e Σ_2 , e que surgem como reação ao movimento do robô humanoide. Devido à componente vertical do momento M_A , o pé de apoio tende a girar em torno do ponto P pela ação de um momento M_z , indicados na Fig. 7.2-c. O referencial inercial $OXYZ$ é usado para os cálculos da posição do ZMP, Fig. 7.2-b, e as coordenadas do ponto P estão ilustradas na Fig. 7.2-d.

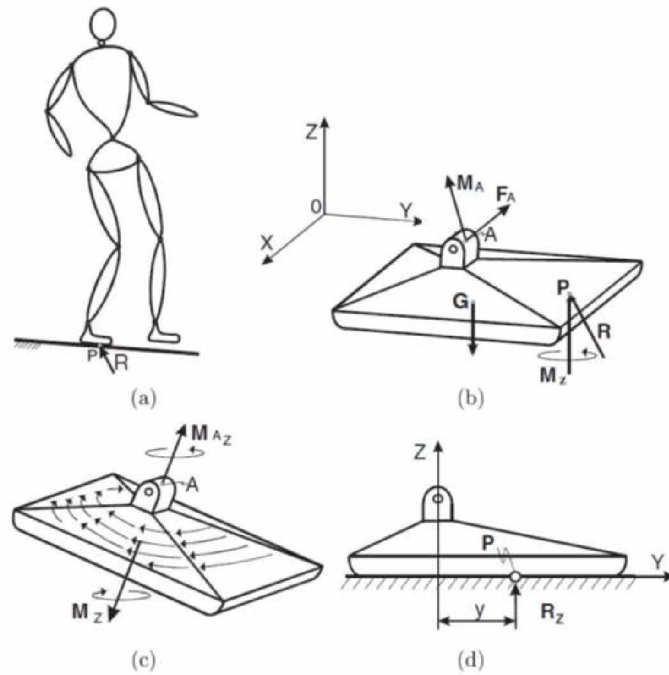


Figura 7.2. Esforços dinâmicos considerados no cálculo do ZMP (VUKOBRATOVIC'; BOROVARAC, 2004).

O momento resultante em relação ao ponto P, \mathbf{M}_P , é dado por

$$\sum_{i=1}^n \left[m_i \left({}^0\mathbf{r}_{CMi} - {}^0\mathbf{r}_P \right) \times \left(-{}^0\mathbf{a}_{CMi} + \mathbf{G} \right) - {}^0I_i {}^0\dot{\boldsymbol{\omega}}_i - {}^0\boldsymbol{\omega}_i \times \left({}^0I_i {}^0\boldsymbol{\omega}_i \right) \right] = \mathbf{M}_P, \quad (7.1.1)$$

no qual, $i=1, \dots, n$, sendo n o número total de corpos do sistema e

${}^0\mathbf{a}_{CMi}$: Vetor que define a aceleração linear do centro de massa do corpo i escrito em relação ao referencial inercial;

m_i : Massa do corpo i ;

${}^0\mathbf{r}_{CMi}$: Vetor que define a posição do centro de massa do corpo i escrito em relação ao referencial inercial;

${}^0\mathbf{r}_P$: Vetor que define a posição do Ponto de Momento Nulo escrito em relação ao referencial inercial;

\mathbf{G} : Vetor campo gravitacional;

0I_i : Matriz dos Momentos de Inércia do corpo i em relação ao referencial inercial;

${}^0\dot{\boldsymbol{\omega}}_i$: Vetor aceleração angular do corpo i escrito no referencial inercial;

${}^0\boldsymbol{\omega}_i$: Vetor velocidade angular do corpo i escrito no referencial inercial.

Nesta modelagem é assumido que toda inércia está concentrada no centro de massa do robô. Assim, os efeitos das inércias de cada corpo são avaliados a partir do movimento do centro de massa. Por outro lado, a velocidade e aceleração angular do centro de massa dependem da sua variação angular (variação da orientação). Sendo esta variação muito pequena, os termos da Eq. (7.1.1) que multiplicam a velocidade e aceleração angular do centro de massa podem ser desprezados quando comparados com os termos que dependem da aceleração linear. Logo, a Equação (7.1.1) pode ser reescrita como:

$$\sum_{i=1}^n \left[m_i \left({}^0\mathbf{r}_{CMi} - {}^0\mathbf{r}_P \right) \times \left(- {}^0\mathbf{a}_{CMi} + \mathbf{G} \right) \right] = \mathbf{M}_P, \quad (7.1.2)$$

no qual

$${}^0\mathbf{r}_{CMi} = \begin{Bmatrix} x_i \\ y_i \\ z_i \end{Bmatrix}, \quad {}^0\mathbf{a}_{CMi} = \begin{Bmatrix} \ddot{x}_i \\ \ddot{y}_i \\ \ddot{z}_i \end{Bmatrix},$$

$${}^0\mathbf{r}_P = {}^0\mathbf{r}_{ZMP} = \begin{Bmatrix} x_{zmp} \\ y_{zmp} \\ 0 \end{Bmatrix},$$

$$\mathbf{G} = \begin{Bmatrix} 0 \\ 0 \\ -g \end{Bmatrix},$$

$$\mathbf{M}_P = \mathbf{M}_{ZMP} = \begin{Bmatrix} 0 \\ 0 \\ M_z \end{Bmatrix}.$$

Isolando as coordenadas do ponto P na Eq. (7.1.2), obtém-se

$$x_{zmp} = \frac{\sum_{i=1}^n [m_i(\ddot{z}_i + g)x_i] - \sum_{i=1}^n [m_i\ddot{x}_i z_i]}{\sum_{i=1}^n [m_i(\ddot{z}_i + g)]},$$

$$y_{zmp} = \frac{\sum_{i=1}^n [m_i(\ddot{z}_i + g)y_i] - \sum_{i=1}^n [m_i\ddot{y}_i z_i]}{\sum_{i=1}^n [m_i(\ddot{z}_i + g)]}. \quad (7.1.3)$$

Por definição, o pé de apoio está em contato com o solo, de forma que a coordenada vertical da trajetória do ZMP, z_{zmp} , é igual a zero.

Considerando que a massa esteja concentrada na posição do centro de massa de todo o corpo do robô humanoide, dada pelas coordenadas x_{CM} , y_{CM} e z_{CM} , é possível simplificar a Eq. (7.1.3) eliminando os somatórios, o que resulta em

$$x_{zmp} = x_{CM} - \frac{z_{CM}}{(\ddot{z}_{CM} + g)} \ddot{x}_{CM},$$

$$y_{zmp} = y_{CM} - \frac{z_{CM}}{(\ddot{z}_{CM} + g)} \ddot{y}_{CM}. \quad (7.1.4)$$

A Equação (7.1.4) fornece as coordenadas lateral e frontal da posição do ZMP. Sendo conhecidas as posições do polígono de apoio, dadas pelas coordenadas dos pés durante a trajetória proposta, é possível avaliar a estabilidade dinâmica do movimento. Para isto basta que sejam comparadas as trajetórias do ZMP e do polígono de apoio. De acordo com o Método do Ponto de Momento Nulo, caso a trajetória do ZMP esteja sempre na região interna do polígono de apoio, o movimento proposto é estável dinamicamente.

O Método do Ponto de Momento Nulo é então aplicado para avaliar a estabilidade da trajetória proposta como segue.

7.1.2. Análise da estabilidade

A partir da trajetória proposta, e conhecendo a trajetória dos pés e suas dimensões, calcula-se a posição do polígono de apoio. Na Figura 7.3, são apresentadas as componentes lateral e frontal do passo do robô humanoide comparando a posição do polígono de apoio com a trajetória do ZMP. As curvas azuis e vermelhas indicam a trajetória dos pés direito e esquerdo, respectivamente. No momento em que ambos estão em contato com o solo, a região do polígono de apoio, representada pelo retângulo de cor preta, compreende a área interna aos pés, tanto na direção frontal quanto lateral. Na fase de simples apoio, a área do polígono de apoio corresponde a região delimitada por apenas um dos pés. A trajetória do ZMP é indicada pela curva verde e todas as curvas são apresentadas durante 2 ciclos do passo do robô. Para calcular as componentes da aceleração do centro de massa, \ddot{x}_{CM} , \ddot{y}_{CM} e \ddot{z}_{CM} , utilizou-se a Eq. (6.3.7) e o método proposto na Seção 6.3. O período do passo foi definido como sendo de 0.4 seg.

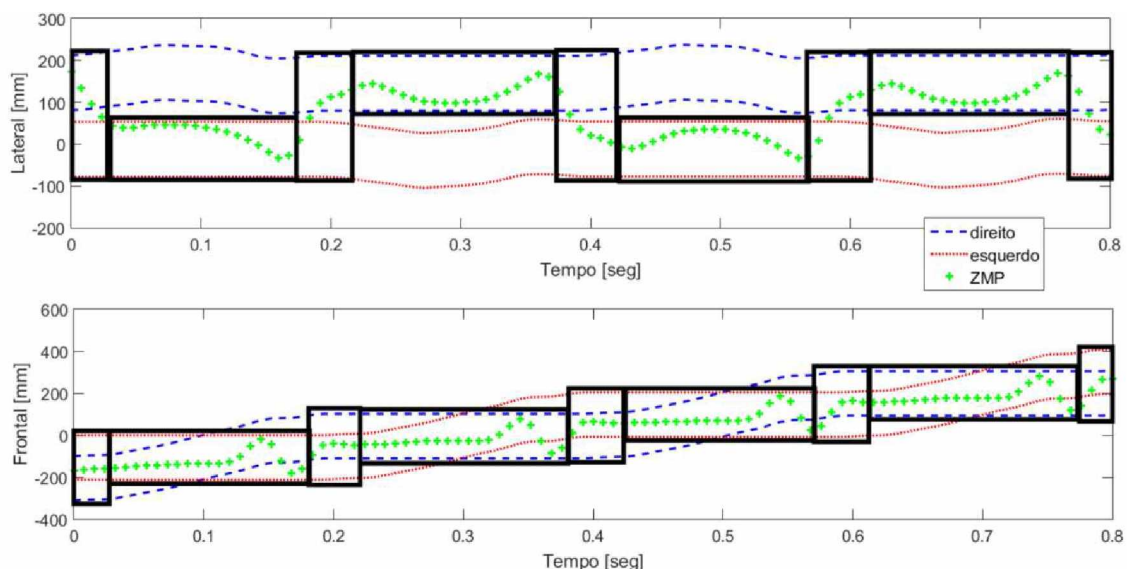


Figura 7.3. Estabilidade da trajetória proposta.

Observa-se na Figura 7.3 que a trajetória do ZMP está sempre contida dentro da região interna dos polígonos de apoio, em ambas direções lateral e frontal do movimento do robô humanoide. De acordo com o Método do Ponto de Momento Nulo, a trajetória proposta é dinamicamente estável, pois em nenhum instante a trajetória do ZMP fica de fora do polígono de apoio.

7.1.3. Influência dos parâmetros da trajetória

É possível ainda verificar a influência dos diferentes parâmetros definidos para a trajetória proposta.

Todos os parâmetros descritos na Seção 5.1, ou seja, as componentes dos deslocamentos nas direções vertical, lateral e frontal dos pés e do centro de massa, tem influência na trajetória do Ponto de Momento Nulo e, conseqüentemente, tem influência na estabilidade do movimento.

Um outro parâmetro que tem influência direta na estabilidade do movimento é o período do passo, o qual influencia na aceleração do centro de massa. Na Figura 7.4 é apresentada a trajetória do ZMP para um passo com período de 0.1 seg.

Observa-se que para um menor período do passo, a trajetória do ZMP não está sempre contida na região interna do polígono de apoio, o que indica que o movimento é dinamicamente instável. Este comportamento é esperado, visto que um período de passo muito pequeno indica que o robô caminha rapidamente. Com uma maior velocidade para o passo, o robô tende a se desequilibrar e cair. Como indicado na Fig.

7.4, observa-se ainda que o robô tenderia a cair tanto lateralmente quanto frontalmente, já que ambas as componentes lateral e frontal da trajetória do ZMP saem do polígono de apoio.

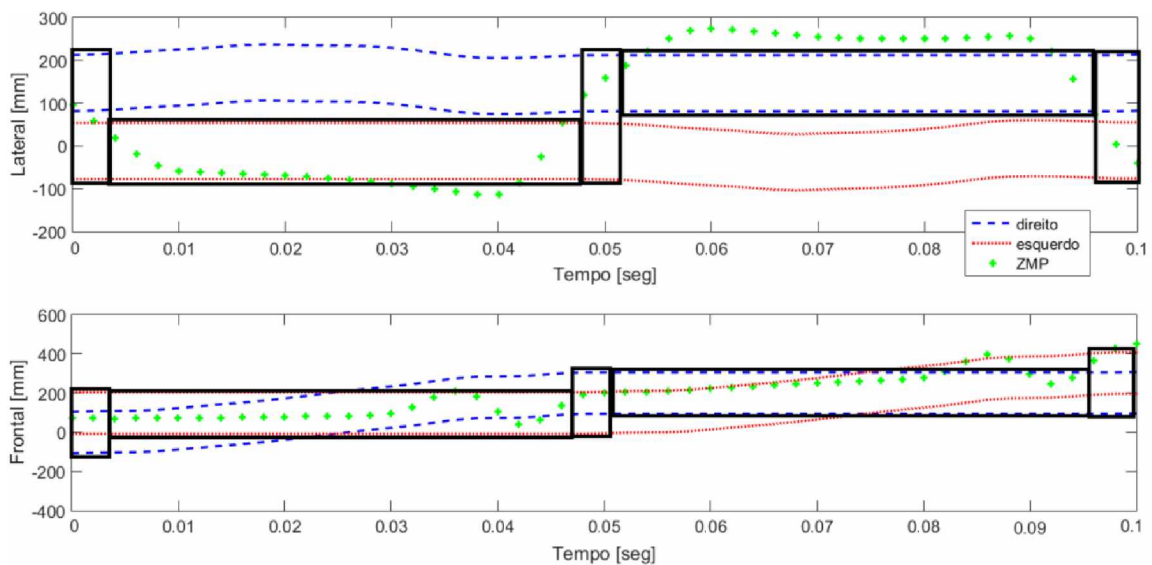


Figura 7.4. Estabilidade da trajetória com o período do passo alterado para 0.1 seg.

Na Figura 7.5 a trajetória volta a ter o período de passo igual a 0.4 seg., e neste caso, é alterada a Amplitude Máxima do Deslocamento Lateral do centro de massa para um valor de 2 mm. Um valor de 2 mm indica que o centro de massa se desloca lateralmente de uma quantidade menor em comparação com as trajetórias propostas nas Figs. 7.3 e 7.4.

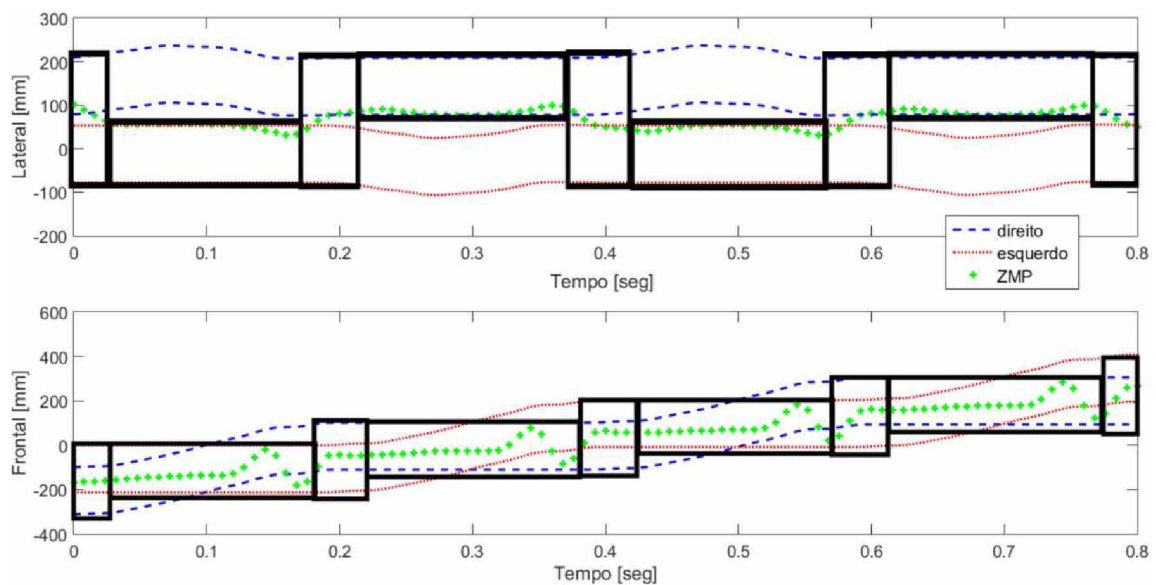


Figura 7.5. Estabilidade da trajetória com Amplitude Máxima do Deslocamento Lateral do centro de massa alterada para 2 mm.

A influência da alteração do deslocamento lateral do centro de massa é diretamente observada na componente lateral da trajetória do ZMP. Desta forma observa-se na Fig. 7.5 que o movimento se mantém estável, porém com a trajetória do ZMP ficando bem próxima aos limites da região interna do polígono de apoio.

Destaca-se portanto uma das principais contribuições deste trabalho. Com o algoritmo proposto para o cálculo dos movimentos articulares de robôs humanoides, é possível que seja simulada qualquer trajetória que se deseje para os pés e centro de massa, verificando se o movimento proposto é dinamicamente estável de acordo com o Método do Ponto de Momento Nulo. Caso o movimento não seja estável, é possível simular diferentes combinações dos parâmetros que definem a trajetória até que o sistema se estabilize. Com o algoritmo, são calculados os ângulos, velocidades e acelerações angulares das articulações das pernas do robô humanoide, possibilitando usar esses dados na simulação e na verificação experimental da estabilidade do movimento proposto.

7.2. Simulação computacional

Com o objetivo de verificar a estabilidade do movimento projetado, foram realizadas simulações do robô humanoide caminhando em um ambiente virtual para simulação de mecanismos multicorpos através do *software* Adams. Os mesmos parâmetros geométricos foram considerados para modelar o mecanismo multicorpo correspondente ao robô humanoide, acrescentando as massas e inércias dos membros do robô.

No Adams foram definidos os referenciais em cada um dos centros de massa dos membros do robô humanoide e nas articulações. Os valores dos momentos de inércia foram definidos em relação aos referenciais baricêntricos criados.

Em seguida foram adicionados os graus de liberdade para as articulações das pernas, visto que não foram atribuídas mobilidades aos ombros e a cabeça.

Para cada uma das articulações foram adicionados os motores responsáveis pela movimentação dos corpos.

As forças de contato entre os pés e solo foram especificadas, permitindo o movimento do robô atribuindo coeficientes de atrito estático e dinâmico iguais a 0.8.

Posicionou-se também os referenciais no centro de massa do robô e nas extremidades dos pés.

Para que fosse possível simular o movimento proposto no Adams, utilizou-se o Simulink, o qual calcula os valores angulares dos atuadores e comunica-se com o Adams, fornecendo os parâmetros obtidos. Concluído o modelo no Adams, foi então gerado o arquivo com as informações utilizadas na comunicação com o Simulink. O passo a passo para realizar a simulação no Adams podem ser vistos em Marques *et al.* (2015).

Durante a simulação algumas imagens foram capturadas afim de ilustrar os resultados obtidos. Foi possível confirmar durante a simulação que o robô se mantém estável durante todo o movimento, conforme previsto teoricamente. Destaca-se a grande influência dos parâmetros escolhidos para modelar o contato entre os pés e o solo. A partir de diversas simulações, verificou-se que o tamanho do passo possui grande sensibilidade em relação ao coeficiente de atrito escolhido, inviabilizando a comparação direta entre as curvas de deslocamento obtidas numericamente. De maneira a minimizar a diferença entre os modelos, é necessário identificar os parâmetros de atrito reais entre o pé e o piso utilizado nos testes experimentais para incluí-los na simulação, o que não faz parte do escopo deste trabalho, visto que o objetivo é verificar, numericamente, se o movimento é estável.

As imagens da Fig. 7.6 correspondem a um passo, seguindo da esquerda para a direita e de cima para baixo. Nas imagem superiores, as linhas de referência brancas destacam a posição do pé esquerdo, o qual está fixo no solo e apoia o corpo do robô. O mesmo é representado nas imagens inferiores, porém o pé direito é que está fixo e apoiando o corpo do humanoide enquanto o pé esquerdo se move.

Apesar do modelo teórico não levar em conta os esforços de atrito, o algoritmo teórico proposto é necessário para calcular o movimento das articulações de maneira a mover o robô em condições ideais, isto é, a simulação no Adams depende dos resultados do modelo cinemático inverso.

A grande vantagem em utilizar o modelo numérico implementado é a possibilidade de incluir na simulação a influência de esforços externos e simular seus efeitos sobre a estabilidade do movimento proposto.

As simulações realizadas permitiram confirmar a estabilidade do movimento projetado, visto que o robô não se desequilibra mesmo sobre influência de parâmetros tais como o atrito entre os pés e o solo. Em seguida foram realizados os testes experimentais.

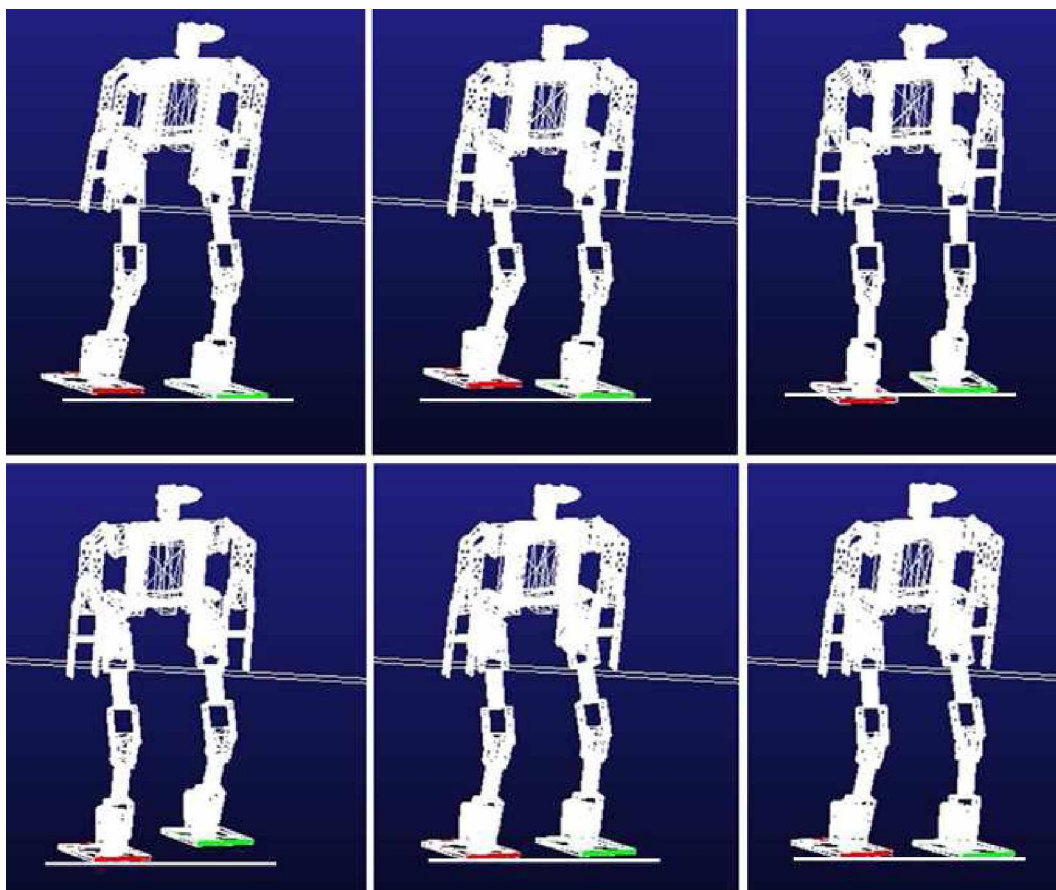


Figura 7.6. Simulação computacional do movimento proposto.

7.3. Testes experimentais

Foram feitos testes experimentais com o robô humanoide estudado ao longo desta tese. Os valores articulares calculados para a trajetória proposta foram programados para que o robô caminhe tendo como referência o mesmo movimento simulado. Assim como feito nas simulações computacionais, o objetivo dos testes experimentais é verificar a estabilidade da trajetória proposta. As comparações entre os deslocamentos espaciais dos pés e do centro de massa não foram feitas, visto que o modelo teórico apresentado não considera a influência de possíveis esforços externos na alteração da trajetória, tais como o atrito, impacto dos pés com o piso e folgas.

Destaca-se aqui dois fatores que mostraram grande efeito sobre o movimento do robô durante os testes experimentais. O primeiro é devido às folgas nos atuadores. Observou-se que para alguns dos motores as folgas permitiam um deslocamento

angular das articulações mesmo que travado eletronicamente em uma posição definida.

O segundo fator que influenciou os resultados é o atrito entre os pés e o piso. Para verificar a influência do atrito sobre o equilíbrio do robô, foram feitos testes sobre dois pisos: encarpetado e cerâmico. Apesar do deslocamento dos pés ser influenciado pelo atrito, o objetivo foi verificar a estabilidade do movimento proposto, mesmo com influência de fatores não previstos no modelo. A quantificação da influência destes parâmetros no equilíbrio do robô é discutida no Capítulo VIII. Nas Figuras 7.7 e 7.8 são apresentadas imagens do robô humanoide caminhando sobre o piso encarpetado. São apresentadas 6 diferentes posturas do robô durante um ciclo do passo.

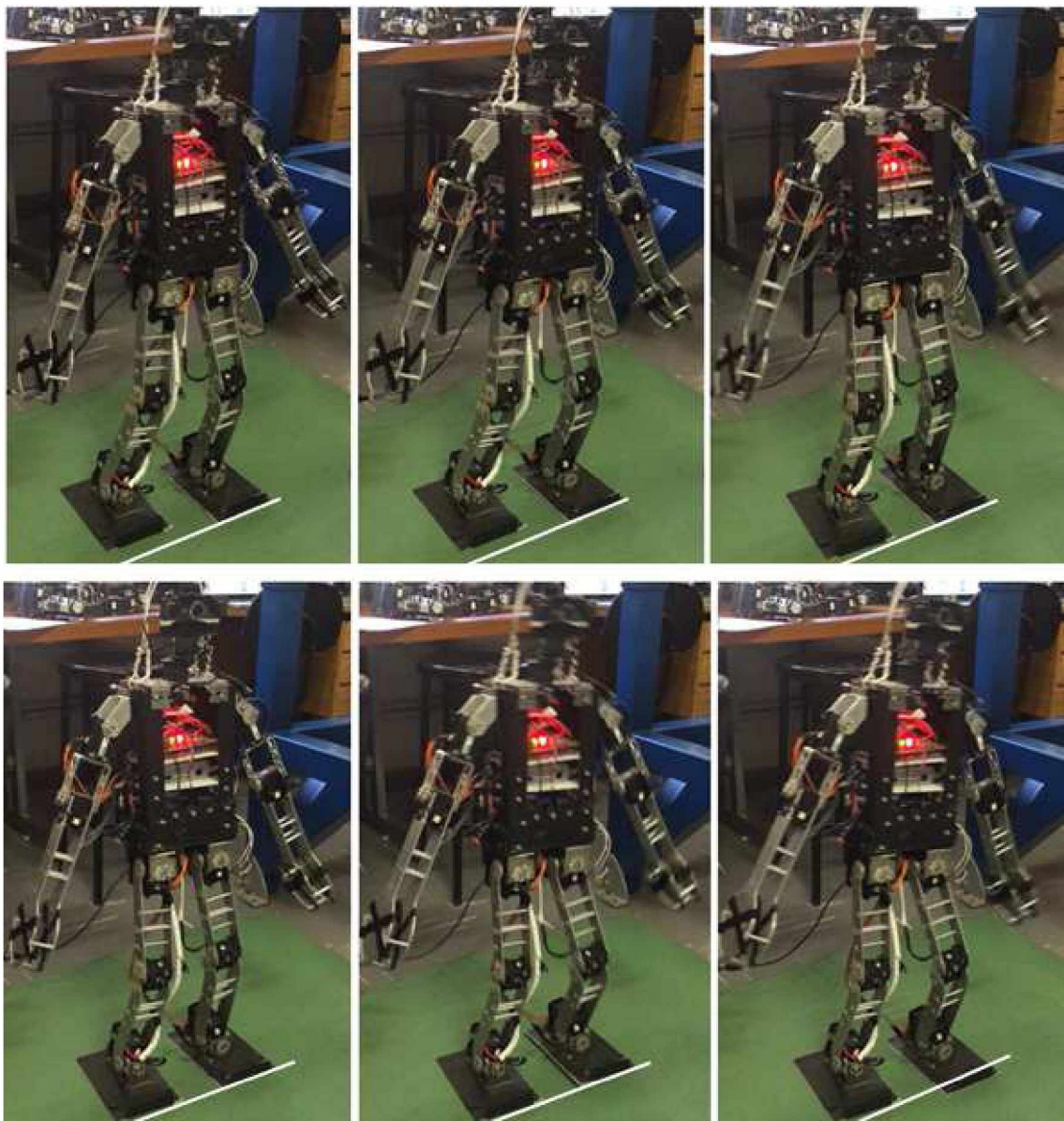


Figura 7.7. Testes experimentais do movimento proposto.

Na Figura 7.7 observa-se, da esquerda para a direita na parte superior, a evolução de um passo enquanto o pé esquerdo se mantém em contato com o piso. Na primeira postura, o pé esquerdo se encontra na frente do direito e ambos em contato com o piso definindo uma postura de duplo apoio. Em seguida, o pé direito se levanta e avança para frente. Na parte de baixo da Fig. 7.7, o pé direito se mantém em contato com o piso durante a segunda metade do passo, na qual o pé esquerdo se levanta e avança para frente, finalizando um ciclo do passo.

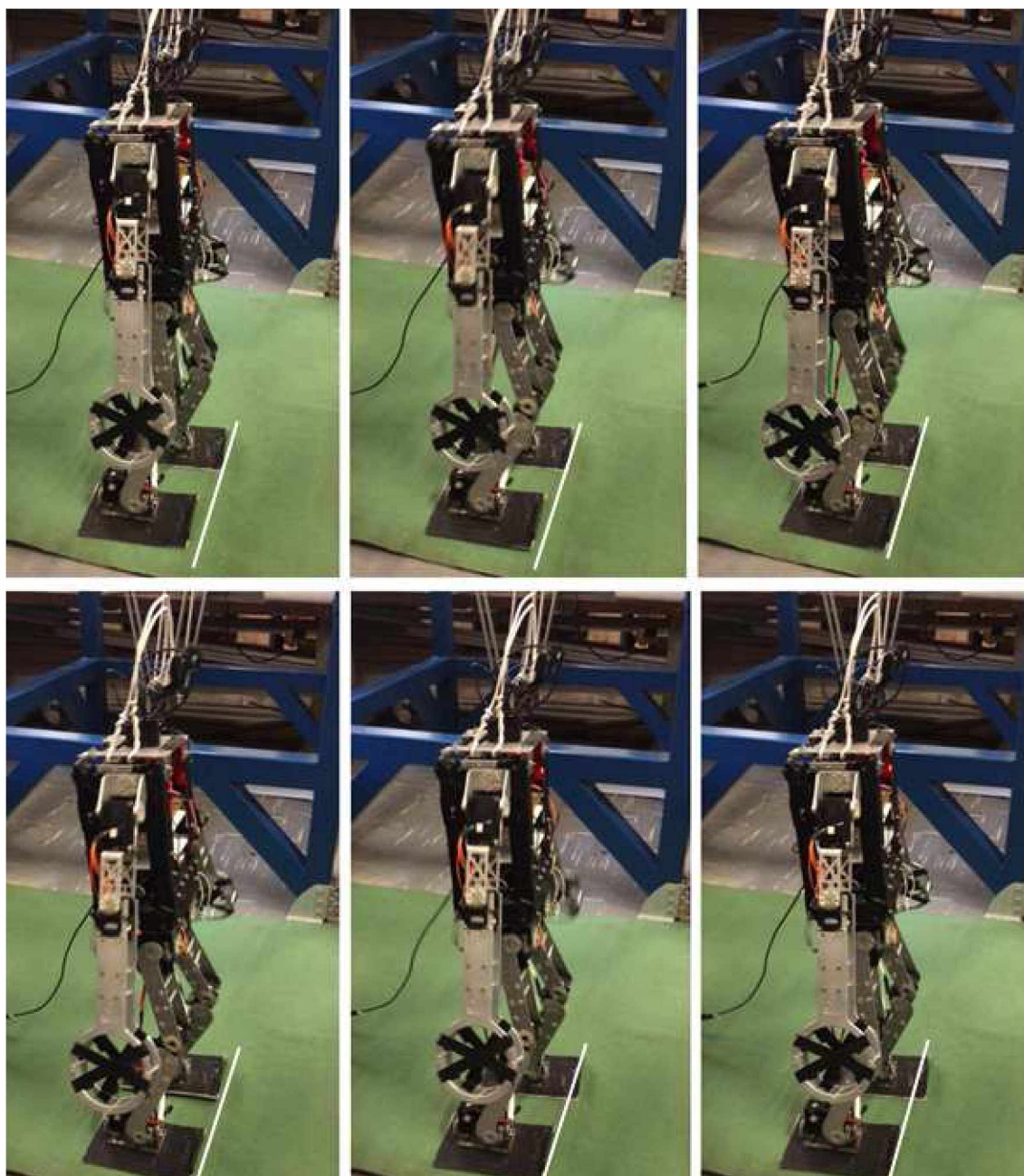


Figura 7.8. Testes experimentais do movimento proposto – vista lateral.

Assim como observado na Fig. 7.7, na Fig. 7.8 também é apresentado um ciclo completo do passo, com a primeira linha apresentando o movimento do pé direito para frente enquanto o corpo se apoia sobre a perna esquerda, e na segunda linha com a segunda metade do ciclo na qual a perna esquerda avança para frente.

Durante os testes, a estabilidade do movimento proposto foi confirmada, mesmo quando a caminhada se deu no piso cerâmico. Apesar da influência dos diferentes coeficientes de atrito entre os pés e o piso e das folgas nas articulações, o robô se manteve caminhando de forma estável.

CAPÍTULO VIII

INFLUÊNCIA DE FATORES DINÂMICOS

Conforme destacado anteriormente, o modelo teórico utilizado não prevê a influência de esforços dinâmicos na alteração da trajetória projetada. No entanto, é possível observar essa influência no equilíbrio através de experimentos.

Neste capítulo é apresentada a avaliação experimental da aceleração do centro de massa durante o movimento, comparada com a aceleração teórica. É possível então relacionar a influência de esforços dinâmicos sobre as variações das amplitudes observadas experimentalmente para a aceleração do centro de massa. Desta forma propõe-se neste capítulo definir quais são os limites de estabilidade do movimento mesmo em casos que se tenha a influência de esforços dinâmicos.

Assume-se na metodologia aqui proposta que a influência dos esforços dinâmicos resulta na alteração da aceleração do centro de massa. Também é considerado para a avaliação teórica do equilíbrio que a posição do polígono de apoio é mantida sem alteração devido aos esforços externos. Estas considerações permitem propor um método de previsão de estabilidade para o movimento, utilizado para simular diferentes trajetórias e projetar a necessária alteração da mesma para que o movimento se mantenha estável sob perturbações.

Sensores inerciais foram utilizados para mensurar a aceleração do centro de massa durante os testes experimentais. Os resultados são discutidos e assim é proposto um método para simulação do movimento do robô humanoide.

8.1. Sensores Inerciais

Durante os testes experimentais apresentados a seguir são utilizados sensores inerciais para medir e comparar as acelerações do centro de massa com os valores teoricamente calculados.

O sensor utilizado é conhecido como IMU (modelo PHIDGETS 1044), do inglês *Inertial Measurement Unit*, e possui em um único *hardware* um acelerômetro, uma bússola e um giroscópio (Disponível em: < http://www.phidgets.com/products.php?product_id=1044_0 >; Acesso em: 15/10/2016). Com este sensor é possível medir a aceleração e a velocidade angular em 3 direções, e ainda determinar a orientação em relação ao norte magnético da Terra.

O sinal do acelerômetro indica, através do nível DC do sinal, a aceleração gravitacional. O valor do nível DC do sinal varia de 0 a 9.8 m/s^2 de acordo com a orientação do corpo em relação à direção do campo gravitacional. A validação do correto funcionamento do IMU foi feita a partir de um teste de bancada utilizando um calibrador de acelerômetros o qual gera uma vibração de 9.8 m/s^2 de amplitude.

Os sinais do sensor são pós-processados para filtrar ruídos de alta frequência, utilizando filtros passa baixo. Nos testes realizados foram utilizados filtros *Butterworth* de 6ª ordem com frequência de corte de 5 Hz, sendo previamente avaliado numericamente que para nenhuma componente da trajetória proposta tenha movimentos em frequências acima deste valor. Esta frequência de corte foi definida com base no período do movimento proposto, dado por 0.4 seg, isto é 2.5 Hz.

Os valores lidos pelos sensores foram utilizados para calcular a orientação completa do corpo rígido em movimento. Neste trabalho, a orientação é dada por quaternions obtidos pelo Filtro de Madgwick, proposto em Madgwick (2010) e Madgwick *et al.* (2011).

O cálculo da orientação do corpo rígido é feito a partir dos sinais do acelerômetro, giroscópio e magnetômetro, conhecido como Método de *Sensor Fusion*. Existem diferentes combinações destas medições que levam à orientação do corpo em movimento. Kalman (1960) propôs um dos filtros mais referenciados em trabalhos da área. Porém, os resultados do Filtro de Kalman são expressos, em geral, na forma de ângulos de Euler, os quais apresentam problemas de singularidade de parametrização dos dados, como citado previamente neste trabalho. Além disso, Madgwick *et al.* (2011) destaca outras desvantagens do Filtro de Kalman como a necessidade de um grande número de variáveis de estado em sua formulação, o fato de requerer uma elevada taxa de aquisição dos dados e ainda depender do ajuste empírico de vários parâmetros.

O Filtro de Madgwick é baseado em um algoritmo de otimização que depende do ajuste de um único parâmetro empiricamente. Para os testes realizados foi utilizada uma baixa taxa de aquisição, de 125 Hz, e para pós-processar os dados, foi utilizado um código em MatLab apresentado no Apêndice C, tendo sido adaptado dos trabalhos originais de Madgwick, escritos em linguagem C.

Após mensurar a orientação do corpo em movimento, é possível escrever a aceleração experimental do centro de massa no referencial inercial, Σ_0 . Afim de comparar os resultados experimentais com os teóricos, todos os valores de aceleração são descritos no referencial inercial.

8.2. Comparação teórico-experimental da aceleração do centro de massa

Foram realizados testes experimentais para que fossem adquiridos os sinais de aceleração do centro de massa. Durante estes testes foi utilizado um segundo robô humanoide, com o objetivo de verificar a aplicabilidade do algoritmo proposto para projetar o movimento de diferentes tamanhos de robôs. O robô utilizado foi modelado seguindo as mesmas etapas apresentadas anteriormente, com número de articulações e parâmetros de Denavit-Hartenberg idênticos. As diferenças são as dimensões do robô, e desta forma o algoritmo foi alterado apenas para permitir calcular os valores angulares das articulações com os novos parâmetros, ou seja:

$$a_1 = l_1 = 40 \text{ mm},$$

$$a_2 = l_2 = 70 \text{ mm},$$

$$a_3 = l_3 = 80 \text{ mm},$$

$$a_4 = l_4 = 30 \text{ mm},$$

$$a_5 = l_5 = 10 \text{ mm}.$$

Além do comprimento l_1 , que é a distância relativa do referencial Σ_B ao referencial Σ_1 , considera-se no modelo, para o cálculo da estabilidade do robô, a largura do pé, de 60 mm, e o comprimento total do pé, de 100 mm, que é a distância da ponta dos dedos à extremidade do calcanhar.

Na Figura 8.1 é apresentada uma imagem do robô humanoide utilizado. Durante os testes utilizou-se um sensor inercial posicionado no centro de massa do robô, e uma fonte de tensão para alimentar os motores, conforme ilustrado na Fig. 8.2.

Na Figura 8.3 são apresentadas as acelerações teóricas (linhas tracejadas) e experimentais (linhas contínuas) do centro de massa do robô humanoide. Com base no período do passo, 0.4 seg, e na frequência de aquisição utilizada, 125 Hz, têm-se 125 pontos para os dados experimentais e para cada curva teórica, com base na interpolação aplicada, tem-se 102 pontos, no intervalo de 1 seg apresentado na Fig. 8.3.



Figura 8.1. Robô humanoide utilizado nos testes experimentais para medição das acelerações do centro de massa.

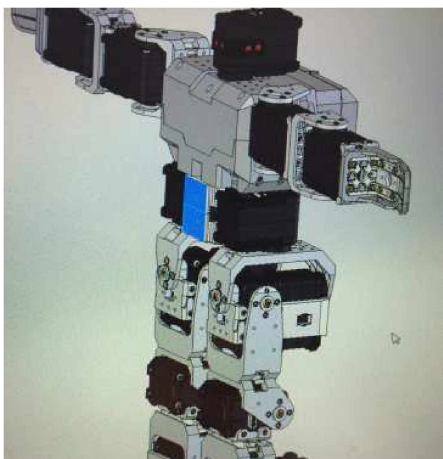


Figura 8.2. Posicionamento do sensor inercial no centro de massa.

Os resultados obtidos permitiram quantificar as diferenças entre os valores previstos e mensurados para a aceleração do centro de massa. As maiores diferenças foram observadas na componente vertical, cujo valor é de 400 mm/s^2 .

Apesar das diferenças observadas, novamente verificou-se que o robô caminhou de maneira estável, ou seja, não se desequilibrou. Além disso, pode-se verificar uma semelhança nos perfis das curvas teóricas e experimentais.

Mesmo com as diferenças observadas foi possível verificar teoricamente que o passo seria estável para o movimento projetado. Porém, deseja-se prever teoricamente qual o comportamento da estabilidade para o movimento do robô na presença de diferentes fatores não previstos no modelo. O valor da máxima diferença observada, 400 mm/s^2 , foi considerado para definir um valor numérico como referência para simular o comportamento da estabilidade, como mostrado na próxima seção.

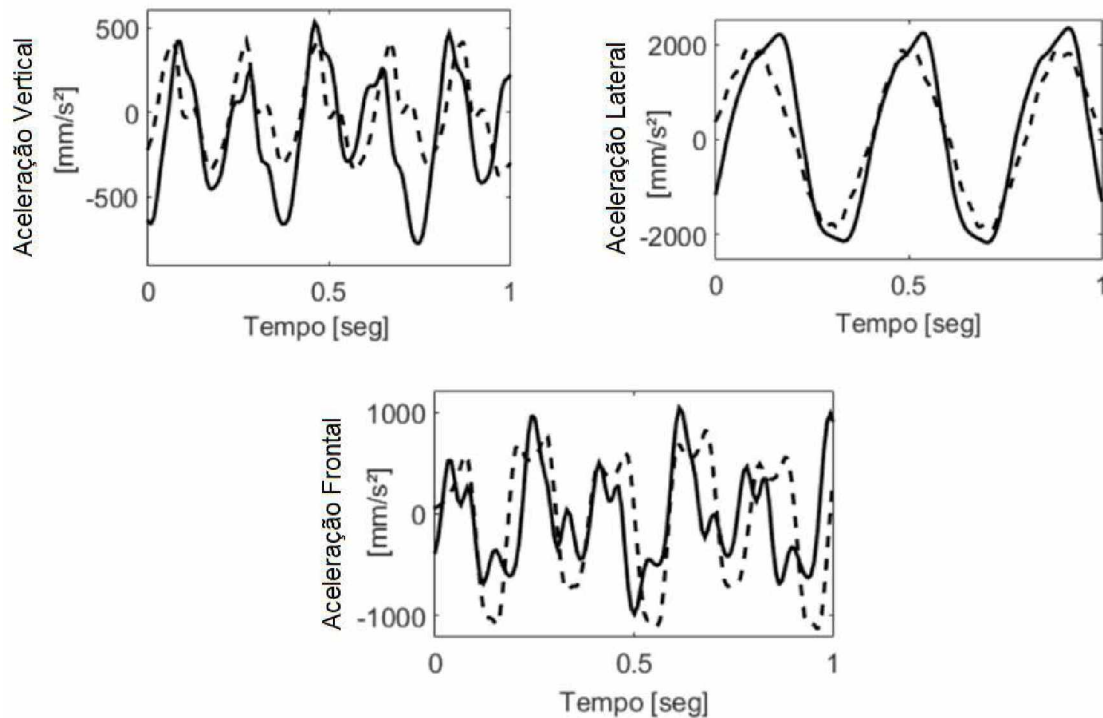


Figura 8.3. Comparação das acelerações teóricas (linhas tracejadas) e experimentais (linhas contínuas) do centro de massa.

8.3. Método de avaliação teórica da estabilidade

Durante os testes experimentais verificou-se que o movimento é estável mesmo na presença de fatores que provocam uma variação de 400 mm/s^2 no valor da aceleração teórica do centro de massa, conforme Fig. 8.3. Esse valor não é um valor limite, porém ainda garantiu a estabilidade durante a caminhada do robô humanoide.

Sendo assim, novas simulações teóricas da trajetória do ZMP devem resultar em um movimento estável mesmo incluindo uma variação de 400 mm/s^2 nos valores teóricos das componentes da aceleração do centro de massa. Isto é feito a partir da Eq. (7.1.4), a qual permite calcular a trajetória do ZMP em função das componentes espaciais da aceleração do centro de massa, \ddot{x}_{CM} , \ddot{y}_{CM} e \ddot{z}_{CM} .

Para o novo cálculo da trajetória do ZMP são consideradas as acelerações teóricas acrescidas de valores aleatórios, os quais representam o efeito não previsto de possíveis esforços externos. Estes valores aleatórios são gerados com média nula e desvio padrão igual a 400 mm/s^2 , valor correspondente às diferenças observadas. Na Figura 8.4 são apresentadas as componentes teóricas da aceleração do centro de

massa acrescida das variações aleatórias consideradas para simular o efeito dos esforços externos.

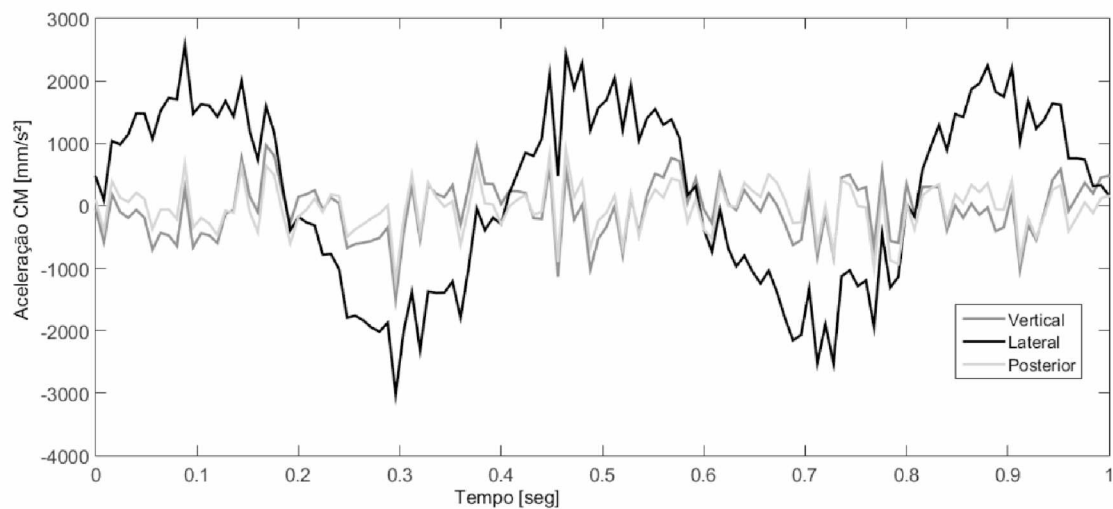


Figura 8.4. Aceleração do centro de massa considerada para a avaliação da estabilidade.

Em seguida, as novas acelerações são utilizadas para recalcular a trajetória do ZMP. Essa nova trajetória é comparada com as posições do polígono de apoio obtidas para a trajetória proposta. Na Figura 8.5 são apresentados os resultados obtidos.

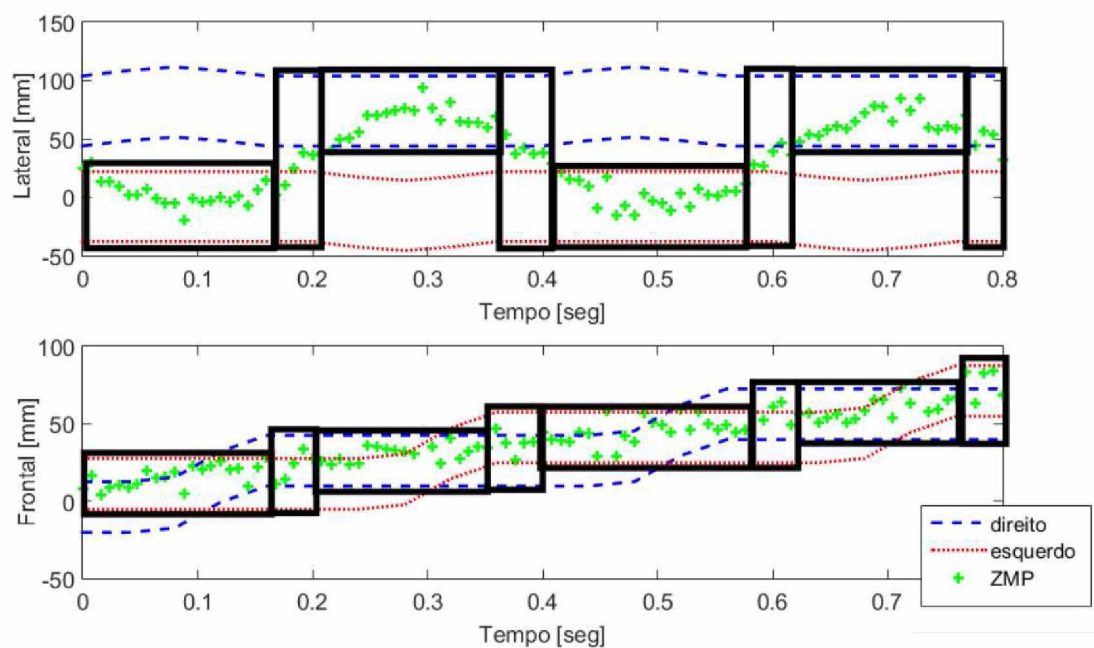


Figura 8.5. Estabilidade da trajetória com aceleração do centro de massa acrescida de uma variação de 400 mm/s^2 .

Observa-se que mesmo considerando variações nos valores de aceleração do centro de massa, a trajetória do ZMP se mantém dentro da região do polígono de apoio, indicando teoricamente que o movimento é estável.

A inclusão dos efeitos dinâmicos no modelo através de variações na aceleração do centro de massa permite que seja avaliada a influência tanto de esforços internos quanto de esforços externos ao sistema. Independente da natureza do esforço dinâmico, seus efeitos podem ser considerados sobre a aceleração do centro de massa e, desta forma, é possível verificar a estabilidade do movimento em condições mais realistas.

Com a análise proposta, pode-se ainda obter os limites de estabilidade do movimento, o que é feito aumentando o valor da aceleração até que a trajetória do ZMP fique fora da região do polígono de apoio.

Conclui-se que durante os testes feitos a trajetória proposta se mostrou estável mesmo na presença de esforços que provocam uma variação de 400 mm/s^2 na aceleração do centro de massa. Isso pôde ser verificado teoricamente pelo método do Ponto de Momento Nulo e ainda pelos testes experimentais realizados.

Caso se queira, pode-se simular diferentes trajetórias definindo os limites de estabilidade para cada caso. É possível ainda, experimentalmente, mensurar a aceleração do centro de massa e alterar o movimento das articulações em tempo real para que resulte numa trajetória estável ao se perceber uma perturbação externa. Caso isso ocorra, a leitura do sensor inercial indicará um aumento nas amplitudes da aceleração do centro de massa e, para o valor lido, pode-se previamente estabelecer qual deve ser a nova trajetória para manter a estabilidade.

CAPÍTULO IX

CONCLUSÕES

Um dos grandes desafios na construção de robôs humanoides consiste em fazer com que ele tenha uma mobilidade durante seu caminhar semelhante à dos humanos. Isso se deve às dificuldades da modelagem de seu comportamento e sua implementação prática.

Neste sentido, este trabalho consistiu em resolver a cinemática inversa de um robô humanoide utilizando dual quaternions, associado ao método do ZMP (*Zero Moment Point*) para verificar seu equilíbrio durante o movimento. O uso de dual quaternions tem sido estudado pois não são afetados pelos problemas de singularidade na parametrização dos dados e podem ser implementados com grande eficiência computacional.

Pode-se verificar na literatura que dual quaternions tem sido amplamente aplicados na solução da cinemática direta robôs humanoides. Assim, uma importante contribuição deste trabalho consiste em utilizar dual quaternions também para a solução da cinemática inversa de robôs humanoides, a qual permite obter uma solução global para a trajetória prevista para o robô.

Neste sentido, foi apresentado inicialmente uma revisão bibliográfica sobre a evolução dos robôs humanoides desde o século 19 até os projetos atuais, desenvolvidos por grandes centros de pesquisa.

As principais características do movimento da marcha humana foram analisadas para serem utilizadas como referência no estudo do movimento de robôs humanoides. Além disso, foram avaliadas as limitações dos métodos encontrados na literatura para resolver a cinemática inversa de humanoides, bem como os problemas de parametrização do movimento através dos ângulos de Euler, que são usualmente aplicados na definição da orientação de corpos rígidos.

O modelo cinemático direto do robô humanoide foi resolvido utilizando os parâmetros de Denavit-Hartenberg. Um código computacional foi implementado recebendo como parâmetros de entrada os valores dos ângulos das articulações de suas pernas, permitindo simular o movimento espacial do robô humanoide. Cabe

salientar que foram consideradas as inércias dos membros superiores porém, sem movimentação.

Um algoritmo foi então proposto para resolver o problema da cinemática inversa de robôs humanoides utilizando dual quaternions através do mapeamento cinemático empregando os parâmetros de Study. Assim, o problema inverso foi formulado por um conjunto de equações algébricas que descrevem a estrutura das pernas, sendo cada perna do robô humanoide representada por uma estrutura serial 6R composta por uma estrutura serial 3R, representando o pé, o tornozelo e o joelho, e uma estrutura esférica, representando o quadril. O sistema de equações obtidos algebricamente é resolvido para uma trajetória desejada resultando nas posições articulares.

A trajetória proposta define o deslocamento e orientação dos pés e do centro de massa do robô. Para tanto, foram definidos parâmetros para descrever o movimento do robô de maneira mais similar possível ao movimento humano. Esses parâmetros permitem definir as trajetórias nas quais, tanto os pés quanto o centro de massa, possam alterar suas orientações e se deslocarem espacialmente durante a caminhada do robô. Desta forma, o método proposto nesta tese mostrou-se capaz de calcular os movimentos articulares sem as limitações dos métodos anteriormente apresentados na literatura.

O algoritmo proposto foi inicialmente aplicado para calcular os valores dos ângulos das articulações em 10 posturas discretas. Em seguida, os movimentos foram interpolados por funções polinomiais que descrevem os valores de ângulos, velocidades e acelerações angulares das articulações em qualquer instante de tempo ao longo da trajetória.

Foram avaliadas as possibilidades de interpolação do movimento no espaço Cartesiano e no espaço das juntas. No espaço Cartesiano destaca-se a necessidade de aplicar o modelo cinemático inverso para computar os valores articulares para cada nova postura interpolada ao longo da trajetória. Além do elevado custo computacional, são necessários cálculos extras para obter as velocidades e acelerações angulares das articulações.

A interpolação no espaço das juntas foi feita comparando a escolha de funções linear e polinomial de 5ª ordem para descrever as variações angulares. A escolha da função linear para comparação com a polinomial se deve ao fato de ser de fácil implementação prática e comumente utilizada para o controle dos atuadores. Já a polinomial de 5ª ordem permite que o movimento seja interpolado considerando condições iniciais e finais de posição, velocidade e aceleração articulares.

A função linear considerada para o perfil de velocidade não é adequada, pois sabe-se que sua variação instantânea não é viável, tendo que ser realizada em um pequeno intervalo de tempo. Desta forma, embora sua derivada (aceleração) seja nula, na prática ocorrem picos instantâneos na aceleração, considerando que a velocidade varia em um intervalo de tempo infinitesimal no momento de transição entre os trechos de velocidades constantes. Estes picos instantâneos na aceleração provocam instabilidade no sistema. Por outro lado, observou-se que as funções polinomiais de 5ª ordem resultam em movimentos mais suaves, sendo então escolhida para ser utilizada durante os testes de estabilidade do movimento proposto ao robô humanoide.

No algoritmo apresentado para definição do movimento do robô humanoide é possível alterar diferentes parâmetros da trajetória, como por exemplo o período do ciclo do passo, de maneira a se obter um movimento dinamicamente estável para a sua caminhada. A avaliação teórica foi feita utilizando o Método do Ponto de Momento Nulo (ZMP). Neste trabalho foi considerado que os atuadores possuem energia suficiente para a movimentação do robô. O ZMP não foi aplicado na geração de trajetórias, mas somente para verificar a estabilidade do robô durante seu movimento.

Simulações computacionais foram feitas utilizando o *software* Adams, o qual permitiu avaliar a dinâmica de mecanismos multicorpos. O robô humanoide foi simulado caminhando sob a ação da gravidade e com atrito entre os pés e o solo. Verificou-se que o movimento das articulações calculado pelo algoritmo proposto também resultou em uma caminhada estável nas simulações realizadas com o Adams.

Testes experimentais foram então feitos para confirmar a estabilidade do movimento simulado. Foi verificado que mesmo na presença de esforços não previstos no modelo, obteve-se um movimento estável para o robô humanoide.

De maneira a prever teoricamente a influência destes esforços foi proposto por fim um método para calcular a trajetória do ZMP adicionando variações de amplitude na aceleração do centro de massa. Essas variações representam no modelo a influência de esforços tanto externos quanto internos ao sistema, tais como atrito, efeito de folgas e o impacto dos pés com o solo.

Outra grande contribuição deste trabalho é que é possível identificar os limites de estabilidade do movimento em função das variações máximas da amplitude da aceleração do centro de massa, que pode ser mensurada experimentalmente por sensores inerciais, e assim, a trajetória pode ser alterada ao ser percebido uma perturbação no sinal de aceleração. Deste forma, o robô se move seguindo os movimentos pré definidos como estáveis mesmo com variações da aceleração do centro de massa.

Durante os testes realizados, os sinais lidos pelo sensor inercial foram pós-processados para calcular a orientação do corpo utilizando quaternions, evitando problemas de parametrização também na aquisição de dados experimentais.

A possibilidade de projetar e simular os movimentos de robôs com trajetórias mais fidedignas às executadas pelos humanos é a grande contribuição deste trabalho. Os dual quaternions e os métodos algébricos apresentados fornecem uma avaliação geral do movimento, e mostrou ser eficaz para os projetos de robôs humanoides.

CAPÍTULO X

TRABALHOS FUTUROS

Como trabalhos futuros propõe-se avaliar a interação do método proposto com diferentes algoritmos de controle do movimento do robô humanoide em malha fechada, visto que é necessário alterar a trajetória em função de sinais lidos pelos sensores.

É proposto também unir o método apresentado com um algoritmo de planejamento de trajetória em tempo real. Esse algoritmo deverá ser capaz de alterar instantaneamente a trajetória dos pés e do centro de massa, para que em seguida sejam calculados os movimentos articulares como descrito neste trabalho.

Os movimentos dos braços também podem ser incluídos através de sua contribuição na aceleração do centro de massa. A partir do modelo dinâmico que relacione tal contribuição, deve ser possível mover os braços de maneira a alterar a aceleração do centro de massa e assim otimizar seu equilíbrio.

Uma outra linha para trabalhos futuros está relacionada à aplicação da modelagem apresentada nesta tese para planejar o movimento de subir e descer escadas, levantar-se a partir de uma posição em que o robô esteja deitado, deitar-se, e de maneira geral caminhar em terrenos desconhecidos. Para tanto será necessário conectar o algoritmo proposto a outros algoritmos de processamento de imagens de câmeras e outros tipos de sensores que forneçam as informações referentes ao ambiente.

Deve-se também analisar a máxima velocidade de deslocamento do robô humanoide, utilizando a técnica do ZMP, para a qual o robô mantém sua estabilidade. Além disso, deve-se verificar o consumo energético para sua otimização quando do uso do ZMP.

A metodologia apresentada neste trabalho pode ainda ser unida à algoritmos de controle adaptativo, gerando os movimentos articulares em casos que se tenha uma alteração do sistema como por exemplo a falha de algum atuador. De maneira geral o desenvolvimento de robôs humanoides está condicionado à técnicas de inteligência artificial, necessária para que tomem decisões em casos não previstos

anteriormente. A modelagem dinâmica de sistemas multicorpos utilizando-se dual quaternions também tem sido objeto de estudo, e desta forma há a possibilidade de realizar toda a modelagem e controle de robôs humanoides baseando-se apenas em técnicas algébricas.

REFERÊNCIAS BIBLIOGRÁFICAS

ALAN, W.; MART, W. **Advanced Animation and Rendering Techniques: Theory and Practice**. Adison-Wesley, 1992.

ALDEBARAN-ROBOTICS, Disponível em: < www.aldebaranrobotics.com >, Acesso em: 30/09/2016.

ALI, M. A.; PARK, H. A.; LEE, C. S. G. **Closed-Form Inverse Kinematic Joint Solution for Humanoid Robots**. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, October 18-22, 2010.

ANGELES, J. **Fundamentals of robotic mechanical systems: Theory, Methods, and Algorithms**. New York: Springer-Verlag, 2.ed, 521p., 2003.

ARAKAWA, T.; FUKUDA, T. **Natural motion generation of biped locomotion robot using hierarchical trajectory generation method consisting of GA, EP layers**. In: Proceedings of the IEEE International Conference on Robotics and Automation, pages 211-216, New Mexico, USA, 1997.

ATLAS, Disponível em: < http://www.bostondynamics.com/robot_Atlas.html >, Acesso em: 15/10/2016.

BOSTON DYNAMICS, Disponível em: < <http://www.bostondynamics.com/> >, Acesso em: 15/10/2016.

BOTTEMA, O.; ROTH, B. **Theoretical Kinematics**. North-Holland Series in Applied Mathematics and Mechanics. North-Holland Publishing Company, Amsterdam, volume 24, New York, Oxford, 1979.

BUSCHMANN, T.; LOHMEIER, S.; ULBRICH, H. **Biped Walking Control Based on Hybrid Position/Force Control**. In: The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, USA, 2009.

CAPI, G.; NASU, Y.; BAROLLI, L.; YAMANO, M.; MITOBE, K.; TAKEDA, K. **A neural network implementation of biped robot optimal gait during walking generated by genetic algorithm**. In: Proceedings of the Mediterranean Conference on Control and Automation, Dubrovnik, Croatia, 2001.

CARO, S.; NURAHMI, L.; WENGER, P.; SCHADLBAUER, J.; HUSTY, M. L. **Reconfiguration analysis of a 4-RUU parallel manipulator**. Mechanism and Machine Theory, 96(2):269-289, 2015.

CHEVALLEREAU, C.; ABBA, G.; AOUSTIN, Y.; PLESTAN, F.; WESTERVELT, E.R.; CANUDAS-DE-WIT, C.; GRIZZLE, J.W. **Rabbit: a testbed for advanced control theory**. In: IEEE Contr. Syst. Mag. 23(5), 57–79. 2003.

CHOI, Y.; YOU, B.J.; OH, S.R. **On the Stability of Indirect ZMP Controller for Biped Robot Systems**. In: Proceedings of International Conference on Intelligent Robots and Systems, pp: 1966 - 1971, vol. 2, Sendai, Japan, June 2004.

CHOU, L. -S.; KAUFMAN, K. R.; BREY, R. H.; DRAGANICH, L. F. **Motion of the whole body's center of mass when stepping over obstacles of different heights**. Gait and Posture Journal, 2000.

CHROBOTICS, Disponível em: < <http://www.chrobotics.com/library/understanding-euler-angles> >, Acesso em: 30/09/2016.

COLLINS, S.; RUINA, A.; TEDRAKE, R.; WISSE, M. **Efficient Bipedal Robots Based on Passive-Dynamic Walkers**. Science, Vol. 307, 2005.

COX, D.; LITTLE, J.; O'SHEA, D. **Ideals, Varieties and Algorithms**. Springer, 3rd. edition, 2007.

CRAIG, J. J. **Introduction to Robotics: Mechanics and Control**. Prentice Hall, 3th edition, ISBN-13: 9780201543612, 408 pp, 2005.

DARPA, Disponível em: < <http://www.darpa.mil/program/darpa-robotics-challenge> >, Acesso em: 15/10/2016.

DEKKER, M. H. P. **Zero-moment point method for stable biped walking**. Master's Internship Report of Eindhoven University of Technology, Netherlands, 2009.

DENAVIT, J.; HARTENBERG, R. S. **A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices**. Journal of Applied Mechanics, 77:215–221, 1955.

EDROM, Disponível em: < <https://ufuedrom.wordpress.com/> >, Acesso em: 15/10/2016.

ERBATUR, K.; KURT, O. **Humanoid Walking Robot Control with Natural ZMP References**. In: IEEE Industrial Electronics, IECON, 2006.

EULER, L. **Formulae generales pro translatione quacunque corpum rigidorum**. Novi Commentari Acad. Petropolitanae, 20:189–207, 1776.

GFRERRER, A. **Rational interpolation on a hypersphere**. Computer-Aided Geometric Design, 16:21-37, 1999-a.

GFRERRER, A. **On the Construction of Rational Curves on Hyperquadrics**. Grazer Mathematische Berichte, Bericht Nr. 340, pp. 1-69, 1999-b.

GFRERRER, A. **Study's kinematic mapping - a tool for motion design**. Recent Advances in Robot Kinematics, pp. 7-16, 2000.

GILBERT, S. **Introduction to Applied Mathematics**. Wellesley-Cambridge Press, 1986.

GOLUB, G. H.; VAN LOAN, C. F. **Matrix Computations**. Baltimore: Johns Hopkins University Press, 3ª ed., 1996.

HAN, J. **Bipedal Walking for a Full-sized Humanoid Robot Utilizing Sinusoidal Feet Trajectories and Its Energy Consumption**. Ph.D. dissertation, Blacksburg, Virginia, EUA, 2012.

HEMAMI, JR. H.; GOLLIDAY, C. L. **The inverted pendulum and biped stability**. Math. Biosci, pp. 95–110, 1976.

HONDA, Disponível em: < <http://asimo.honda.com/> >, Acesso em: 15/10/2016.

HORAK, F.B.; NASHNER, L.M. **Central programming of postural movements: adaptation to altered support-surface configurations**. J Neurophysiol, vol. 55, no. 6, pp. 1369-1381, 1986.

HUSTY, M. L. **An Algorithm for Solving the Direct Kinematic of General Stewart-Gough Platforms**. Mechanism and Machine Theory, vol. 31, No. 4, pp. 365-380, 1996.

HUSTY, M. L.; KARGER, A.; SACHS, H.; STEINHILPER, W. **Kinematik und Robotik**. Springer, Berlin, Heidelberg, New York, 1997.

HUSTY, M. L.; KARGER, A. **Self-Motions of Griffis-Duffy Type Platforms**. In: Proc. of IEEE conference on Robotics and Automation (ICRA 2000), San Francisco, pp. 7–12, 2000.

HUSTY, M. L.; KARGER, A. **Self motions of Stewart-Gough platforms, an overview**. In: Proc. of the workshop on fundamental issues and future research directions for parallel mechanisms and manipulators, Quebec City, pp. 131 – 141, 2002.

HUSTY, M. L.; PFURNER, M.; SCHRÖCKER, H. -P. **A new and efficient algorithm for the inverse kinematics of a general serial 6R manipulator**. Mechanism and Machine theory, 2006.

HUSTY, M. L.; PFURNER, M.; SCHRÖCKER, H. -P.; BRUNNTHALER, K. **Algebraic methods in mechanism analysis and synthesis**. Robotica, 25(6):661-675, 2007.

HUSTY, M. L.; GOSSELIN, C.. **On the singularity surface of planar 3-RPR parallel mechanisms**. Mech. Based Design of Structures and Machines, 36:411-425, 2008.

HUSTY, M. L.; SCHRÖCKER, H. -P. **Algebraic geometry and kinematics**. Nonlinear Computational Geometry, volume 151 of The IMA Volumes in Mathematics and its Applications, chapter Algebraic Geometry and Kinematics, pp. 85-107, Springer, 2010.

JOHNNIE, Disponível em: < <https://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/johnnie/> >, Acesso em: 15/10/2016.

KAGAMI, S. **A Fast Dynamically Equilibrated Walking Trajectory Generation Method of Humanoid Robot**. Autonomous Robots 12, 71–82, Kluwer Academic Publishers, 2002.

KAJITA, S.; MORISAWA, M.; MIURA, K.; NAKAOKA, S.; HARADA, K.; KANEKO, K.; KANEHIRO, F.; YOKOI, K. **Biped Walking Stabilization Based on Linear Inverted Pendulum Tracking**. In: The 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 2010.

KAJITA, S.; HIRUKAWA, H.; HARADA, K.; YOKOI, K. **Introduction to Humanoid Robotics**. Springer, ISSN 1610-7438, 2014.

KALMAN, R. E. **A new approach to linear filtering and prediction problems**. Journal of Basic Engineering, 82:35-45, 1960.

KATIC, D.; VUKOBRATOVIC, M. **Survey of intelligent control techniques for humanoid robots**. Journal of Intelligent and Robotic Systems, 37:117-141, 2003.

KATO, I.; OHTERU, S.; KOBAYASHI, H.; SHIRAI, K.; UCHIYAMA, A. **Information-Power Machine With Senses And Limbs**. On Theory And Practice Of Robots And Manipulators. Volume 1. UDINE 1974 SPRINGER - VERLAG, pp12-24, Ro-man 1973, IFToMM Sy. 1973.

KAWADA, Disponível em: < <http://global.kawada.jp/mechatronics/hrp4.html> >, Acesso em: 15/10/2016.

KENWRIGTH, B. **A Beginners Guide to Dual-Quaternions: What They Are, How They Work, and How to Use Them for 3D Character Hierarchies**. In: The 20th International Conference on Computer Graphics, Visualization and Computer Vision, WSCG Communication Proceedings, pp.1-13, 2012.

KENWRIGTH, B. **Inverse Kinematics with Dual-Quaternions, Exponential-Maps, and Joint Limits**. International Journal on Advances in Intelligent Systems, vol 6 no 1 & 2. 2013.

KHARB, A.; SAINI, V.; JAIN, Y. K.; DHIMAN, S. **A Review Of Gait Cycle And Its Parameters**. IJCEM International Journal of Computational Engineering & Management, Vol. 13, July 2011.

KIM, J. -Y.; PARK, ILL-W.; OH, J. -H. **Walking Control Algorithm of Biped Humanoid Robot on Uneven and Inclined Floor**. Journal of Intelligent and Robotic Systems, Volume 48, Issue 4 , pp 457-484, 2007.

KLAWITTER, D. **Bewegungsdesign mit der kinematischen Abbildung**. Master's thesis, Technische Universität Dresden, 2010.

KOFINAS, N.; ORFANOUDAKIS, E.; LAGOUDAKIS, M. **Complete Analytical Inverse Kinematics for NAO**. In: Proc. of the 13th International Conference on Autonomous Robot Systems and Competitions, Lisbon, Portugal, pp. 1-6, 2013.

KOFINAS, N. **Forward and Inverse Kinematics for the NAO Humanoid Robot**. Ph.D. dissertation, Department of Electronic and Computer Engineering, Technical University of Crete, 2012.

KRISHNAMOORTHY, V.; GOODMAN, S.; ZATSIORSKY, V.; LATASH, M. L. **Muscle synergies during shifts of the center of pressure by standing persons: identification of muscle modes**. Biol Cybern., vol. 89, no. 2, pp. 152-161, 2003.

LECLERCQ, G. **Dual Quaternion Symbolic Toolbox**. 2012. Disponível em: < <http://www.mathworks.com/matlabcentral/fileexchange/39288-dual-quaternion-toolbox> >, Acesso em: 24/03/2015.

LI, Z.; TSAGARAKIS, N. G.; CALDWELL, D. G. **Walking Trajectory Generation for Humanoid Robots with Compliant Joints: Experimentation with COMAN Humanoid**. In: IEEE International Conference on Robotics and Automation, Minnesota, USA, 2012.

LOLA, Disponível em: < <http://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/lola/> >, Acesso em: 15/10/2016.

LUO, J.; ZHANG, Y.; HAUSER, K.; PARK, H. A.; PALDHE, M.; LEE, C. S. G.; GREY, M.; STILMAN, M.; OH, J. -H.; LEE, J.; KIM, I.; OH, P. **Robust ladder-climbing with a humanoid robot with application to the DARPA Robotics Challenge**. In: Proceedings - IEEE International Conference on Robotics and Automation., 6907259, Institute of Electrical and Electronics Engineers Inc., pp. 2792-2798, ICRA 2014, Hong Kong, China, 2014.

MADGWICK, S. O. H. **An efficient orientation filter for inertial and inertial/magnetic sensor arrays**. Disponível em: < https://www.samba.org/tridge/UAV/madgwick_internal_report.pdf, 2010 >, Acesso em: 24/03/2015.

MADGWICK, S. O. H.; HARRISON, A. J. L.; VAIDYANATHAN, R. **Estimation of IMU and MARG orientation using a gradient descent algorithm**. In: IEEE International Conference on Rehabilitation Robotics, Rehab Week Zurich, ETH Zurich Science City, Switzerland, 2011.

MARQUES, V. A. S.; VENANCIO, M. M.; JAPPE, L. A.; CARVALHO, J. C. M.; GONCALVES R. S. **Analysis and Simulation of humanoid gait**. In: IEEE proceedings of LARS/SBR - 12th Latin American Robotics Symposium and 2015 Third Brazilian Symposium on Robotics, 2015.

MCCARTHY, J. M. **Geometric Design of Linkages**. Interdisciplinary Applied Mathematics. Springer-Verlag, volume 320, New York, 2000.

MCGEER, T. **Passive dynamic walking**. Int. J. Robot. Res. 9(2), 62–82. 1990.

MEREDITH, M.; MADDOCK, S. **Real-Time Inverse Kinematics: The Return of the Jacobian**. Department of Computer Science, University of Sheffield, United Kingdom, 2015. Disponível em: < <http://www.dcs.shef.ac.uk/intranet/research/public/resmes/CS0406.pdf> >, Acesso em: 24/03/2015.

MISTRY, M.; NAKANISHI, J.; CHENG, G.; SCHAAL, S. **Inverse kinematics with floating base and constraints for full body humanoid robot control**. In: IEEE-RAS International Conference on Humanoid Robots, 2008.

MIT, Disponível em: < <https://robotics.mit.edu/> >, Acesso em: 15/10/2016.

MITOBE, K.; CAPI, G.; NASU, Y. **Control of walking robots based on manipulation of the zero moment point**. Robotica, Volume 18, Issue 06, pp 651 – 657, 2000.

MURRAY, R. M. **Optimization-Based Control**. Control and Dynamical Systems, California Institute of Technology, 2010.

MUROOKA, M.; NOZAWA, S.; KAKIUCHI, Y.; OKADA K.; INABA, M. **Whole-body Pushing Manipulation with Contact Posture Planning of Large and Heavy Object for Humanoid Robot**. IEEE International Conference on Robotics and Automation (ICRA), Washington, USA, 2015.

NATARAJ, R. **Feedback Control Of Standing Balance Using Functional Neuromuscular Stimulation Following Spinal Cord Injury**. Ph.D. dissertation, Department of Biomedical Engineering, Case Western Reserve University, 2011.

NUNEZ, J. V.; BRISENO, A.; RODRIGUEZ, D. A.; IBARRA, J. M.; RODRIGUEZ, V. M. **Explicit analytic solution for inverse kinematics of Bioloid humanoid robot**. In: Brazilian Robotics Symposium and Latin American Robotics Symposium, IEEE DOI 10.1109/SBR-LARS.2012.62, 2012.

NURAHMI, L.; SCHADLBAUER, J.; CARO, S.; HUSTY, M. L.; WENGER, P. **Kinematic Analysis of the 3-RPS Cube Parallel Manipulator**. ASME Journal of Mechanisms and Robotics, 7(1):1-11, 2015.

PARK, I. -W.; KIM, J. -Y.; LEE, J.; OH, J. -H. **Online Free Walking Trajectory Generation for Biped Humanoid Robot KHR-3 (HUBO)**. In: Proceedings of the 2006 IEEE International Conference on Robotics and Automation, Florida , USA, 2006.

PFURNER, M. **Analysis of spatial serial manipulators using kinematic mapping**. Ph.D. dissertation, University of Innsbruck. 2006.

PHIDGETS 1044 - Products for USB Sensing and Control. Disponível em: < http://www.phidgets.com/products.php?product_id=1044_0 >, Acesso em: 15/10/2016.

RAIBERT, M. H.; BROWN JR., H. B.; CHEPPONIS, M. **Experiments in balance with a 3D one-legged hopping machine**. International J. Robotics Research, V 3:75—92, 1984.

RAIBERT, M. H.; CHEPPONIS, M.; BROWN, JR. H. B. **Running on Four Legs As Though They Were One**. IEEE Journal of Robotics and Automation, Vol. RA-2, No. 2, 1986.

ROBOCUP, Disponível em: < <http://www.robocup.org.br/> >, Acesso em: 15/10/2016.

ROBOTIS, Disponível em: < www.robotis.com >, Acesso em: 30/09/2016.

ROKBANI, N.; ALIM, A. M. **IK-PSO, PSO Inverse Kinematics Solver with Application to Biped Gait Generation**. International Journal of Computer Application, (0975 – 8887) Volume 58– No.22, 2012.

SCHILLING, M. **Universally manipulable body models — dual quaternion representations in layered and dynamic MMCs**. Autonomous Robots, 2011.

SCHMITZ, A.; MISSURA, M.; BEHNKE, S. **Real-Time Trajectory Generation by Offline Footstep Planning for a Humanoid Soccer Robot**. Robocup 2011, LNCS 7416, pp. 198-209, Springer, 2012.

SELIG, J. M. **Geometric Fundamentals of Robotics**. Monographs in Computer Science, Springer, New York, 2005.

SICILIANO, B.; KHATIB, O. **Springer handbook of robotics**. Springer, Berlin, 2008.

SONY, Disponível em: < www.sony.net/SonyInfo/News/Press_Archive/200312/03-060E >, Acesso em: 15/10/2016.

STASSE, O.; VERRELST, B.; DAVISON, A.; MANSARD, N.; SAIDI, F.; VANDERBORGHT, B.; ESTEVES, C.; YOKOI, K. **Integrating Walking and Vision to increase Humanoid Autonomy**. International Journal of Humanoid Robotics, World Scientific Publishing Company, 2008.

STUDY, E. **Von den Bewegungen und Umlegungen**. Math. Ann. pp. 441-566. 1891.

STUDY, E. **Geometrie der Dynamen**. B. G. Teubner, Leipzig, 1903.

SUGIHARA, T. **Standing Stabilizability and Stepping Maneuver in Planar Bipedalism based on the Best COM-ZMP Regulator**. In: 2009 IEEE International Conference on Robotics and Automation, Kobe, Japan, 2009.

VARSHNEY, A.; GE, Q.; MENON, J. P.; CHANG, C. F. **Double quaternions for motion interpolation**. In: Proceedings of the ASME Design Engineering Technical Conference, 1998.

VUKOBRATOVIC', M.; BOROVAC, B. **Zero-moment point — thirty five years of its life**. International Journal of Humanoid Robotics, Vol. 1, No. 1, pp. 157–173, 2004.

WALDRON, K. J.; VOHNOUT, V. J.; PERY, A.; MCGHEE, R. B. **Configuration design of the adaptive suspension vehicle**. International Journal of Robotics Research, Vol. 3, N°2, 1984.

WALTER, D. R.; HUSTY, M. L.; PFURNER, M. **A Complete Kinematic Analysis of the SNU 3- UPU Parallel Manipulator**. Contemporary Mathematics, Vol. 496, American Mathematical Society, pp. 331-346, ISBN: 978-0-8218-4746-6, 2009.

WALTER, D. R.; HUSTY, M. L. **On Implicitization of Kinematic Constraint Equations**. Machine Design & Research, 26:218–226, 2010.

WAMPLER, C. W.; SOMMESE, A. J. **Numerical Algebraic Geometry and Algebraic Kinematics**. Acta Numerica, 20:469–567, DOI:10.1017/S0962492911000067, 2011.

WEBER, W. **Introduction to Robotics: Module Trajectory generation and robot programming**. Fachhochschule Darmstadt, Germany, Summer Term, 2000.

WITTLE, M. W. **Gait Analysis – An Introduction**. Elsevier, 4^a Edition, 2007.

WILLIAMS II, R. L. **Darwin-Op Humanoid Robot Kinematics**. In: Proc. of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, 2012.

WINTER, D. A. **Biomechanics and Motor Control of Human Movement**. Canada: John Wiley & Sons, 2nd ed. Toronto, 1990.

WISSE, M.; SCHWAB, L.; VAN DER HELM, F. L. T. **Passive walking dynamic model with upper body**. Robotica 22(6), 681–688. 2004.

WOLLHERR, D. **Design and Control Aspects of Humanoid Walking Robots**. Ph.D. dissertation, Technische Universität München, 2005.

ZADEH, S. J.; KHOSRAVI, A.; MOGHIMI, A.; ROOZMAND, N. **A review and analysis of the trajectory gait generation for humanoid robot using inverse kinematic**. In: 3rd International Conference on Electronics Computer Technology, 2011.

ZHE, T.; ZENGQI, S.; CHANGJIU, Z.; LINGYUN, H. **Reference Trajectory Generation for 3-Dimensional Walking of a Humanoid Robot**. Tsinghua Science And Technology, ISSN 1007-0214, 12/19, pp. 577-584, Volume 12, Number 5, 2007.

ZIJLSTRA, W.; HOF, A. L. **Assessment of spatio-temporal gait parameters from trunk accelerations during human walking**. Gait and Posture Journal, 2003.

ZORJAN, M.; HUGEL, V. **Generalized Humanoid Leg Inverse Kinematics to Deal With Singularities**. In: Proc. IEEE Int. Conf. on Robotics and Automation, 2013.

APÊNDICE A

Modelo Cinemático Direto

```
clc
clear all
close all

%
%PARAMETROS GEOMETRICOS
pe=85;
canela=202;
coxa=221;
cinturah=104;
cinturab=104;

%VISTA DOS GRAFICOS
vista=1; %1 para iso, 2 para topo, 3 para lateral, 4 para frontal
opcoes=[-120,20;
        180,90;
        90,0;
        -180,0];
az=opcoes(vista,1);
el=opcoes(vista,2);

%NUMERO DE PASSOS DURANTE SIMULACAO
npassos=2;
poseiniesq=1;
posefimesq=5;
poseinidir=1;
posefimdir=5;

%INTERVALO DE TEMPO ENTRE POSES
dt=2;

%POSICAO E ORIENTACAO INICIAL PARA O PE DE APOIO
dedop=[0;0;0]; %primeiro passo - depois atualizar para depop do ciclo
anterior (qdo o outro pe estava no chao - mgd do outro lado)
dedor=[1 0 0;
       0 1 0;
       0 0 1];
dedo=[1 0 0 0;
      dedop dedor]

%COORDENADAS ARTICULARES

u=[ 5    14    15    8    0    0    -5    -5    0    0
   20    30   -20   -20   -30   -20   10    20   25   30
  -10   -20   -10    0    0    0   -10   -10    0    0
  -20   -10    30    30    30    20    0   -10   -25   -30
   -5   -14   -15   -8    0    0    5    5    0    0
    0    0    0    0    0    1    2    3    4    5];

%
eiox=100;
eioy=100;
```



```

eixoz=100;

px=[1 0 0 0;
    eiox 1 0 0;
    0 0 1 0;
    0 0 0 1];

py=[1 0 0 0;
    0 1 0 0;
    eiox 0 1 0;
    0 0 0 1];

pz=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    eiox 0 0 1];

%


---


u=u.*pi./180;
indice=1;

for passos=1:npassos

    %%perna esquerda no chao

    for i=poseiniesq:posefimesq

        u1=u(1,i+5); %inclina pe
        u2=u(2,i+5); %levanta pe
        u3=u(3,i+5); %levanta joelho
        u4=u(4,i+5); %levanta perna
        u5=u(5,i+5); %inclina quadril
        u6=u(6,i+5); %gira quadril

        u7=-u(6,i); %gira quadril
        u8=u(5,i); %inclina quadril
        u9=-u(4,i); %levanta perna
        u10=-u(3,i); %levanta joelho
        u11=-u(2,i); %levanta pe
        u12=u(1,i); %inclina pe

    %
    

---


        m1=[1 0 0 0;
            0 1 0 0;
            0 0 1 0;
            0 0 0 1];

        g1=[1 0 0 0;
            0 1 0 0;
            0 0 1 0;
            -pe 0 0 1];

        calcanhar=dedo*m1*g1;
        calcanharp=calcanhar(2:4,1);
        calcanharr=calcanhar(2:4,2:4);

```

```

%
m2=[1 0 0 0;
    0 cos(u1) -sin(u1) 0;
    0 sin(u1) cos(u1) 0;
    0 0 0 1];

g2=[1 0 0 0;
    0 1 0 0;
    0 0 cos(pi/2) -sin(pi/2);
    0 0 sin(pi/2) cos(pi/2)];

tornozele=calcanhar*m2*g2;
tornozelep=tornozele(2:4,1);
tornozelel=tornozele(2:4,2:4);

```

```

%
m3=[1 0 0 0;
    0 cos(u2) -sin(u2) 0;
    0 sin(u2) cos(u2) 0;
    0 0 0 1];

g3=[1 0 0 0;
    canela 1 0 0;
    0 0 1 0;
    0 0 0 1];

joelho=tornozele*m3*g3;
joelhople=joelho(2:4,1);
joelhoplel=joelho(2:4,2:4);

```

```

%
m4=[1 0 0 0;
    0 cos(u3) -sin(u3) 0;
    0 sin(u3) cos(u3) 0;
    0 0 0 1];

g4=[1 0 0 0;
    coxa 1 0 0;
    0 0 1 0;
    0 0 0 1];

cintural1=joelho*m4*g4;
cintural1p=cintural1(2:4,1);
cintural1l=cintural1(2:4,2:4);

```

```

%
m5=[1 0 0 0;
    0 cos(u4) -sin(u4) 0;
    0 sin(u4) cos(u4) 0;
    0 0 0 1];

```

```

g5=[1 0 0 0;
    0 1 0 0;
    0 0 cos(-pi/2) -sin(-pi/2);
    0 0 sin(-pi/2) cos(-pi/2)];

```

```

cintura2=cintura1*m5*g5;
cintura2p=cintura2(2:4,1);
cintura2r=cintura2(2:4,2:4);

```

%

```

m6=[1 0 0 0;
    0 cos(u5+pi/2) -sin(u5+pi/2) 0;
    0 sin(u5+pi/2) cos(u5+pi/2) 0;
    0 0 0 1];

```

```

g6=[1 0 0 0;
    0 1 0 0;
    0 0 cos(pi/2) -sin(pi/2);
    0 0 sin(pi/2) cos(pi/2)];

```

```

cintura3=cintura2*m6*g6;
cintura3p=cintura3(2:4,1);
cintura3r=cintura3(2:4,2:4);

```

%

```

m7b=[1 0 0 0;
    0 cos(u6) -sin(u6) 0;
    0 sin(u6) cos(u6) 0;
    0 0 0 1];

```

```

g7b=[1 0 0 0;
    cinturab 1 0 0;
    0 0 1 0;
    0 0 0 1];

```

```

cintura4=cintura3*m7b*g7b;
cintura4p=cintura4(2:4,1);
cintura4r=cintura4(2:4,2:4);

```

%

```

m7=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    0 0 0 1];

```

```

g7=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    cinturah 0 0 1];

```

```

cm=cintura4*m7*g7
cmp=cm(2:4,1);
cmr=cm(2:4,2:4);

```

```

%
m8b=[1 0 0 0;
      0 1 0 0;
      0 0 1 0;
      0 0 0 1];

g8b=[1 0 0 0;
      0 1 0 0;
      0 0 1 0;
      -cinturah 0 0 1];

cintura5=cm*m8b*g8b;
cintura5p=cintura5(2:4,1);
cintura5r=cintura5(2:4,2:4);

%
m9=[1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1];

g9=[1 0 0 0;
     cinturab 1 0 0;
     0 0 1 0;
     0 0 0 1];

cintura6=cintura5*m9*g9;
cintura6p=cintura6(2:4,1);
cintura6r=cintura6(2:4,2:4);

%
m10=[1 0 0 0;
      0 cos(u7) -sin(u7) 0;
      0 sin(u7) cos(u7) 0;
      0 0 0 1];

g10=[1 0 0 0;
      0 1 0 0;
      0 0 cos(-pi/2) -sin(-pi/2);
      0 0 sin(-pi/2) cos(-pi/2)];

cintura7=cintura6*m10*g10;
cintura7p=cintura7(2:4,1);
cintura7r=cintura7(2:4,2:4);

%
m11=[1 0 0 0;
      0 cos(u8-pi/2) -sin(u8-pi/2) 0;
      0 sin(u8-pi/2) cos(u8-pi/2) 0;
      0 0 0 1];

```

```

g11=[1 0 0 0;
      0 1 0 0;
      0 0 cos(pi/2) -sin(pi/2);
      0 0 sin(pi/2) cos(pi/2)];

cintura8=cintura7*m11*g11;
cintura8p=cintura8(2:4,1);
cintura8r=cintura8(2:4,2:4);

```

%

```

m12=[1 0 0 0;
      0 cos(u9-pi) -sin(u9-pi) 0;
      0 sin(u9-pi) cos(u9-pi) 0;
      0 0 0 1];

g12=[1 0 0 0;
      coxa 1 0 0;
      0 0 1 0;
      0 0 0 1];

joelho2=cintura8*m12*g12;
joelho2p=joelho2(2:4,1);
joelho2r=joelho2(2:4,2:4);

```

%

```

m13=[1 0 0 0;
      0 cos(u10) -sin(u10) 0;
      0 sin(u10) cos(u10) 0;
      0 0 0 1];

g13=[1 0 0 0;
      canela 1 0 0;
      0 0 1 0;
      0 0 0 1];

tornozelo2=joelho2*m13*g13;
tornozelo2p=tornozelo2(2:4,1);
tornozelo2r=tornozelo2(2:4,2:4);

```

%

```

m14=[1 0 0 0;
      0 cos(u11+pi) -sin(u11+pi) 0;
      0 sin(u11+pi) cos(u11+pi) 0;
      0 0 0 1];

g14=[1 0 0 0;
      0 1 0 0;
      0 0 cos(-pi/2) -sin(-pi/2);
      0 0 sin(-pi/2) cos(-pi/2)];

calcanhar2=tornozelo2*m14*g14;
calcanhar2p=calcanhar2(2:4,1);
calcanhar2r=calcanhar2(2:4,2:4);

```

%

```

m15=[1 0 0 0;
      0 cos(u12) -sin(u12) 0;
      0 sin(u12) cos(u12) 0;
      0 0 0 1];

```

```

g15=[1 0 0 0;
      0 1 0 0;
      0 0 1 0;
      pe 0 0 1];

```

```

dedo2=calcanhar2*m15*g15;
dedo2p=dedo2(2:4,1);
dedo2r=dedo2(2:4,2:4);

```

%

```

x=[dedop(1) calcanharp(1) tornozelop(1) joelhop(1)
  cintura1p(1) cintura2p(1) cintura3p(1) cintura4p(1) cmp(1) ...
  cintura5p(1) cintura6p(1) cintura7p(1) cintura8p(1)
  joelho2p(1) tornozelo2p(1) calcanhar2p(1) dedo2p(1)];
y=[dedop(2) calcanharp(2) tornozelop(2) joelhop(2)
  cintura1p(2) cintura2p(2) cintura3p(2) cintura4p(2) cmp(2) ...
  cintura5p(2) cintura6p(2) cintura7p(2) cintura8p(2)
  joelho2p(2) tornozelo2p(2) calcanhar2p(2) dedo2p(2)];
z=[dedop(3) calcanharp(3) tornozelop(3) joelhop(3)
  cintura1p(3) cintura2p(3) cintura3p(3) cintura4p(3) cmp(3) ...
  cintura5p(3) cintura6p(3) cintura7p(3) cintura8p(3)
  joelho2p(3) tornozelo2p(3) calcanhar2p(3) dedo2p(3)];

```

```

figure(1)
plot3(y,z,x)
hold on
plot3(y,z,x,'*r')
xlabel('medio-lateral')
ylabel('antero-posterior')
zlabel('vertical')
view(az,el)
axis([-2500 2500 -2500 2500 -100 600])
pause(dt)
hold off

```

```

cmx(indice)=cmp(1);
cmx(indice)=cmp(2);
cmz(indice)=cmp(3);

```

```

esquerdox(indice)=calcanharp(1);
esquerdoy(indice)=calcanharp(2);
esquerdoy(indice)=calcanharp(3);

```

```

direitox(indice)=calcanhar2p(1);
direitoy(indice)=calcanhar2p(2);
direitoz(indice)=calcanhar2p(3);

```

```

aux=cm*px;
aux1(:,indice)=aux(2:4,1);

```

```

    aux=cm*py;
    aux2(:,indice)=aux(2:4,1);
    aux=cm*pz;
    aux3(:,indice)=aux(2:4,1);

    aux=calcanhar*px;
    aux4(:,indice)=aux(2:4,1);
    aux=calcanhar*py;
    aux5(:,indice)=aux(2:4,1);
    aux=calcanhar*pz;
    aux6(:,indice)=aux(2:4,1);

    aux=calcanhar2*px;
    aux7(:,indice)=aux(2:4,1);
    aux=calcanhar2*py;
    aux8(:,indice)=aux(2:4,1);
    aux=calcanhar2*pz;
    aux9(:,indice)=aux(2:4,1);

    indice=indice+1;
end

%
%
%%perna direita no chao

dedop=dedo2p; %agora o mgd vai no sentido contrario
dedor=dedo2r;
dedo=[1 0 0 0;
      dedop dedor];

%
%
for i=poseinidir:posefimdir

    u1=-u(1,i+5); %inclina pe
    u2=u(2,i+5); %levanta pe
    u3=u(3,i+5); %levanta joelho
    u4=u(4,i+5); %levanta perna
    u5=-u(5,i+5); %inclina quadril
    u6=u(6,i+5); %gira quadril

    u7=-u(6,i); %gira quadril
    u8=-u(5,i); %inclina quadril
    u9=-u(4,i); %levanta perna
    u10=-u(3,i); %levanta joelho
    u11=-u(2,i); %levanta pe
    u12=-u(1,i); %inclina pe

%
%
m1=[1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1];

```

```

g1=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    -pe 0 0 1];

calcanhar=dedo*m1*g1;
calcanharp=calcanhar(2:4,1);
calcanharr=calcanhar(2:4,2:4);

```

%

```

m2=[1 0 0 0;
    0 cos(u1) -sin(u1) 0;
    0 sin(u1) cos(u1) 0;
    0 0 0 1];

g2=[1 0 0 0;
    0 1 0 0;
    0 0 cos(pi/2) -sin(pi/2);
    0 0 sin(pi/2) cos(pi/2)];

tornozele=calcanhar*m2*g2;
tornozelep=tornozele(2:4,1);
tornozeleor=tornozele(2:4,2:4);

```

%

```

m3=[1 0 0 0;
    0 cos(u2) -sin(u2) 0;
    0 sin(u2) cos(u2) 0;
    0 0 0 1];

g3=[1 0 0 0;
    canela 1 0 0;
    0 0 1 0;
    0 0 0 1];

joelho=tornozele*m3*g3;
joelhoop=joelho(2:4,1);
joelhoor=joelho(2:4,2:4);

```

%

```

m4=[1 0 0 0;
    0 cos(u3) -sin(u3) 0;
    0 sin(u3) cos(u3) 0;
    0 0 0 1];

g4=[1 0 0 0;
    coxa 1 0 0;
    0 0 1 0;
    0 0 0 1];

cintural1=joelho*m4*g4;
cintural1p=cintural1(2:4,1);
cintural1r=cintural1(2:4,2:4);

```


%

```

m5=[1 0 0 0;
    0 cos(u4) -sin(u4) 0;
    0 sin(u4) cos(u4) 0;
    0 0 0 1];

g5=[1 0 0 0;
    0 1 0 0;
    0 0 cos(-pi/2) -sin(-pi/2);
    0 0 sin(-pi/2) cos(-pi/2)];

cintura2=cintura1*m5*g5;
cintura2p=cintura2(2:4,1);
cintura2r=cintura2(2:4,2:4);

```

%

```

m6=[1 0 0 0;
    0 cos(u5+pi/2) -sin(u5+pi/2) 0;
    0 sin(u5+pi/2) cos(u5+pi/2) 0;
    0 0 0 1];

g6=[1 0 0 0;
    0 1 0 0;
    0 0 cos(pi/2) -sin(pi/2);
    0 0 sin(pi/2) cos(pi/2)];

cintura3=cintura2*m6*g6;
cintura3p=cintura3(2:4,1);
cintura3r=cintura3(2:4,2:4);

```

%

```

m7b=[1 0 0 0;
    0 cos(u6) -sin(u6) 0;
    0 sin(u6) cos(u6) 0;
    0 0 0 1];

g7b=[1 0 0 0;
    -cinturab 1 0 0;
    0 0 1 0;
    0 0 0 1];

cintura4=cintura3*m7b*g7b;
cintura4p=cintura4(2:4,1);
cintura4r=cintura4(2:4,2:4);

```

%

```

m7=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    0 0 0 1];

```

```

g7=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    cinturah 0 0 1];

```

```

cm=cintura4*m7*g7
cmp=cm(2:4,1);
cmr=cm(2:4,2:4);

```

%

```

m8b=[1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1];

```

```

g8b=[1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     -cinturah 0 0 1];

```

```

cintura5=cm*m8b*g8b;
cintura5p=cintura5(2:4,1);
cintura5r=cintura5(2:4,2:4);

```

%

```

m9=[1 0 0 0;
     0 1 0 0;
     0 0 1 0;
     0 0 0 1];

```

```

g9=[1 0 0 0;
    -cinturab 1 0 0;
    0 0 1 0;
    0 0 0 1];

```

```

cintura6=cintura5*m9*g9;
cintura6p=cintura6(2:4,1);
cintura6r=cintura6(2:4,2:4);

```

%

```

m10=[1 0 0 0;
     0 cos(u7) -sin(u7) 0;
     0 sin(u7) cos(u7) 0;
     0 0 0 1];

```

```

g10=[1 0 0 0;
     0 1 0 0;
     0 0 cos(-pi/2) -sin(-pi/2);
     0 0 sin(-pi/2) cos(-pi/2)];

```

```

cintura7=cintura6*m10*g10;

```

```
cintura7p=cintura7(2:4,1);
cintura7r=cintura7(2:4,2:4);
```

```
%
```

```
m11=[1 0 0 0;
      0 cos(u8-pi/2) -sin(u8-pi/2) 0;
      0 sin(u8-pi/2) cos(u8-pi/2) 0;
      0 0 0 1];
```

```
g11=[1 0 0 0;
      0 1 0 0;
      0 0 cos(pi/2) -sin(pi/2);
      0 0 sin(pi/2) cos(pi/2)];
```

```
cintura8=cintura7*m11*g11;
cintura8p=cintura8(2:4,1);
cintura8r=cintura8(2:4,2:4);
```

```
%
```

```
m12=[1 0 0 0;
      0 cos(u9-pi) -sin(u9-pi) 0;
      0 sin(u9-pi) cos(u9-pi) 0;
      0 0 0 1];
```

```
g12=[1 0 0 0;
      coxa 1 0 0;
      0 0 1 0;
      0 0 0 1];
```

```
joelho2=cintura8*m12*g12;
joelho2p=joelho2(2:4,1);
joelho2r=joelho2(2:4,2:4);
```

```
%
```

```
m13=[1 0 0 0;
      0 cos(u10) -sin(u10) 0;
      0 sin(u10) cos(u10) 0;
      0 0 0 1];
```

```
g13=[1 0 0 0;
      canela 1 0 0;
      0 0 1 0;
      0 0 0 1];
```

```
tornozelo2=joelho2*m13*g13;
tornozelo2p=tornozelo2(2:4,1);
tornozelo2r=tornozelo2(2:4,2:4);
```

```
%
```

```

m14=[1 0 0 0;
      0 cos(u11+pi) -sin(u11+pi) 0;
      0 sin(u11+pi) cos(u11+pi) 0;
      0 0 0 1];

g14=[1 0 0 0;
      0 1 0 0;
      0 0 cos(-pi/2) -sin(-pi/2);
      0 0 sin(-pi/2) cos(-pi/2)];

calcanhar2=tornozelo2*m14*g14;
calcanhar2p=calcanhar2(2:4,1);
calcanhar2r=calcanhar2(2:4,2:4);

%


---



m15=[1 0 0 0;
      0 cos(u12) -sin(u12) 0;
      0 sin(u12) cos(u12) 0;
      0 0 0 1];

g15=[1 0 0 0;
      0 1 0 0;
      0 0 1 0;
      pe 0 0 1];

dedo2=calcanhar2*m15*g15;
dedo2p=dedo2(2:4,1);
dedo2r=dedo2(2:4,2:4);

%


---



x=[dedop(1) calcanharp(1) tornozelop(1) joelhop(1)
cintura1p(1) cintura2p(1) cintura3p(1) cintura4p(1) cmp(1) ...
cintura5p(1) cintura6p(1) cintura7p(1) cintura8p(1)
joelho2p(1) tornozelo2p(1) calcanhar2p(1) dedo2p(1)];
y=[dedop(2) calcanharp(2) tornozelop(2) joelhop(2)
cintura1p(2) cintura2p(2) cintura3p(2) cintura4p(2) cmp(2) ...
cintura5p(2) cintura6p(2) cintura7p(2) cintura8p(2)
joelho2p(2) tornozelo2p(2) calcanhar2p(2) dedo2p(2)];
z=[dedop(3) calcanharp(3) tornozelop(3) joelhop(3)
cintura1p(3) cintura2p(3) cintura3p(3) cintura4p(3) cmp(3) ...
cintura5p(3) cintura6p(3) cintura7p(3) cintura8p(3)
joelho2p(3) tornozelo2p(3) calcanhar2p(3) dedo2p(3)];

%


---



figure(1)
plot3(y,z,x)
hold on
plot3(y,z,x,'*r')
xlabel('medio-lateral')
ylabel('antero-posterior')
zlabel('vertical')
view(az,el)

```

```

axis([-2500 2500 -2500 2500 -100 600])
pause(dt)
hold off

cmx(indice)=cmp(1);
cmx(indice)=cmp(2);
cmz(indice)=cmp(3);

direitox(indice)=calcanharp(1);
direitoy(indice)=calcanharp(2);
direitoz(indice)=calcanharp(3);

esquerdox(indice)=calcanhar2p(1);
esquerdoy(indice)=calcanhar2p(2);
esquerdoz(indice)=calcanhar2p(3);

aux=cm*px;
aux1(:,indice)=aux(2:4,1);
aux=cm*py;
aux2(:,indice)=aux(2:4,1);
aux=cm*pz;
aux3(:,indice)=aux(2:4,1);

aux=calcanhar*px;
aux7(:,indice)=aux(2:4,1);
aux=calcanhar*py;
aux8(:,indice)=aux(2:4,1);
aux=calcanhar*pz;
aux9(:,indice)=aux(2:4,1);

aux=calcanhar2*px;
aux4(:,indice)=aux(2:4,1);
aux=calcanhar2*py;
aux5(:,indice)=aux(2:4,1);
aux=calcanhar2*pz;
aux6(:,indice)=aux(2:4,1);

indice=indice+1;

end

%
dedop=dedo2p; %agora o mgd volta pro pe esquerdo no chao
dedor=dedo2r;
dedo=[1 0 0 0;
      dedop dedor];

end

%
%

figure(2)
plot3(cmy,cmz,cmx,'m','LineWidth',2)
hold on
plot3(esquerdoy,esquerdoz,esquerdox,'m','LineWidth',2)

```

```

plot3(direitoy,direitoz,direitox,'m', 'LineWidth', 2)
xlabel('medio-lateral')
ylabel('antero-posterior')
zlabel('vertical')
legend('cm','esquerdo','direito')
view(az,el)
axis([-1000 1000 -1000 1000 -100 600])
for i=1:indice-1
plot3([cmy(i) aux1(2,i)],[cmz(i) aux1(3,i)],[cmx(i) aux1(1,i)],'r',
'LineWidth', 2)
plot3([cmy(i) aux2(2,i)],[cmz(i) aux2(3,i)],[cmx(i) aux2(1,i)],'g',
'LineWidth', 2)
plot3([cmy(i) aux3(2,i)],[cmz(i) aux3(3,i)],[cmx(i) aux3(1,i)],'b',
'LineWidth', 2)

plot3([esquerdoy(i) aux4(2,i)],[esquerdoz(i) aux4(3,i)],[esquerdox(i)
aux4(1,i)],'r', 'LineWidth', 2)
plot3([esquerdoy(i) aux5(2,i)],[esquerdoz(i) aux5(3,i)],[esquerdox(i)
aux5(1,i)],'g', 'LineWidth', 2)
plot3([esquerdoy(i) aux6(2,i)],[esquerdoz(i) aux6(3,i)],[esquerdox(i)
aux6(1,i)],'b', 'LineWidth', 2)

plot3([direitoy(i) aux7(2,i)],[direitoz(i) aux7(3,i)],[direitox(i)
aux7(1,i)],'r', 'LineWidth', 2)
plot3([direitoy(i) aux8(2,i)],[direitoz(i) aux8(3,i)],[direitox(i)
aux8(1,i)],'g', 'LineWidth', 2)
plot3([direitoy(i) aux9(2,i)],[direitoz(i) aux9(3,i)],[direitox(i)
aux9(1,i)],'b', 'LineWidth', 2)
end

```

APÊNDICE B

Modelo Cinemático Inverso

```
clc
clear all
close all

syms x0 x1 x2 x3 y0 y1 y2 y3
R=[x0; x1; x2; x3; y0; y1; y2; y3];
syms v1 v2 v3 v4 v5 v6

%PARAMETROS GEOMETRICOS
pe=85;
canela=202;
coxa=221;
cinturah=104;
cinturab=104;

%poses

cm =[1.0000      0      0      0
     487.36      0.0000     -0.0000      1.0000
      70.47      0.9877     -0.1564     -0.0000
     517.14      0.1564      0.9877      0.0000];

foot =[1.0000      0      0      0
       35.221      0.9851     -0.0021      0.1720
      -126.752      0.0248      0.9912     -0.1302
       784.792     -0.1702      0.1325      0.9765];

EEmass=matrixtodual(cm);
EEfoot=matrixtodual(foot);

dual=EEmass;
cond1=dual(1,1)*dual(5,1)+dual(2,1)*dual(6,1)+dual(3,1)*dual(7,1)+dual(4,1)*dual(8,1);
cond2=dual(1,1)^2+dual(2,1)^2+dual(3,1)^2+dual(4,1)^2;

%pose-dual
EEfootinv=dqinv(EEfoot);
g1=[1 0 0 0;
    0 1 0 0;
    0 0 1 0;
    -pe 0 0 1];
Tb=matrixtodual(g1);
g7=[1 0 0 0;
    cinturab 1 0 0;
    0 0 1 0;
    cinturah 0 0 1];
Te=matrixtodual(g7);
Tbinv=dqinv(Tb);
Teinv=dqinv(Te);
EE=dqmult(Tbinv,EEfootinv);
EE=dqmult(EE,EEmass);
dual=(dqmult(EE,Teinv));
```

%ankle and knee - canonical

alfa1=pi/2;

alfa2=0;

alfa3=0;

a1=0;

a2=canela;

a3=coxa;

d1=0;

d2=0;

d3=0;

al1=tan(alfa1/2);

al2=tan(alfa2/2);

al3=tan(alfa3/2);

```
hc1v2 = [(-al3*v2*a1 + al3*al1*al2*a2*v2 - al3*al1*a1*al2*v2 +
al3*a2*v2 - al3*al1*d2 ...
+ al3*al2*d2 + a3*al1*v2 - al2*a3*v2 - al3*d3*al2 - al3*d3*al1); ...
+ (a1*v2 - al1*al2*a2*v2 + al1*a1*al2*v2 - a2*v2 + al1*d2 - al2*d2 +
al3*a3*al1*v2...
- a3*al3*al2*v2 + d3*al2 + d3*al1); + (d2*al2*v2 + al1*d2*v2 - a1 +
a2*al1*al2...
- a2 + al1*al1*al2 + d3*al1*v2 - d3*al2*v2 - a3*al3*al2 -
a3*al3*al1);...
+(-d2*al3*al2*v2 -al1*d2*al3*v2 +al3*a1 -al3*al1*al2*a2 +al3*a2 -
al3*al1*al2*a1...
- d3*al1*al3*v2 + al3*d3*al2*v2 - al2*a3 - al1*a3); ...
- 2*al3*v2*(al1 - al2); + 2*(al1*v2 - al2*v2); - 2*(al2 + al1); +
2*al3*(al2 + al1)];
```

```
hc2v2 = [(al2*a2*v2 + al1*v2*a1 - al1*v2*a2 - al2*v2*a1 - d2 -
al1*al2*d2 - al3*a3*al1*al2*v2...
- al3*a3*v2 + d3*al1*al2 - d3);+(a2*al3*al2*v2 + al3*al1*v2*a1 -
al3*al1*v2*a2...
- al1*al3*al2*v2 - al3*d2 - al3*al1*al2*d2 + a3*al1*al2*v2 + a3*v2 +
al3*d3*al1*al2...
- al3*d3); (-al1*d2*al3*al2*v2 + d2*al3*v2 + a2*al3*al2 +
al1*al3*al2...
+ a2*al3*al1 + al1*al3*al1 + al3*d3*al1*al2*v2 + d3*al3*v2 - a3*al1*al2
+ a3);...
+ (-al1*d2*al2*v2 + d2*v2 + al2*a2 + al2*a1 + al1*a2 + al1*a1 +
d3*al1*al2*v2...
+ d3*v2 + al3*al1*al2*a3 - al3*a3); - 2*(al1*al2*v2 + v2);...
- 2*al3*v2*(1 + al1*al2); + 2*al3*(al1*al2 - 1); + 2*(al1*al2 - 1)];
```

```
hc3v2 = [(-d2*al3*al2*v2 - al1*d2*al3*v2 + al3*a1 - al3*al1*al2*a2 +
al3*a2 - al3*al1*al2*a1...
- d3*al1*al3*v2 + al3*d3*al2*v2 - al2*a3 - al1*a3);...
+(d2*al2*v2 +al1*d2*v2 -al1 +a2*al1*al2 -a2 +al1*al1*al2 +d3*al1*v2 -
d3*al2*v2...
- a3*al3*al2 - a3*al3*al1); + (al1*al2*a2*v2 - a1*v2 - al1*a1*al2*v2 +
a2*v2...
- al1*d2 + al2*d2 - d3*al2 - d3*al1 + a3*al3*al2*v2 -
al3*a3*al1*v2);...
+ (-al3*al1*al2*a2*v2 + al3*v2*a1 + al3*al1*a1*al2*v2 - al3*a2*v2 +
al3*al1*d2...
-al3*al2*d2+al3*d3*al2+al3*d3*al1+al2*a3*v2-
a3*al1*v2);+2*al3*(al2+al1);...
- 2*(al2 + al1); + 2*(al2*v2 - al1*v2); + 2*al3*v2*(al1 - al2)];
```



```

hc4v2 = [(a11*d2*a12*v2 -d2*v2 -a12*a1 -a11*a1 -a12*a2 -a11*a2 +a13*a3
-a13*a11*a12*a3...
-d3*a11*a12*v2 -d3*v2); +(a11*d2*a13*a12*v2 -d2*a13*v2 -a1*a13*a12 -
a1*a13*a11...
-a2*a13*a12 - a2*a13*a11 - a3 + a3*a11*a12 - a13*d3*a11*a12*v2 -
d3*a13*v2);...
+ (a2*a13*a12*v2 + a13*a11*v2*a1 - a13*a11*v2*a2 - a1*a13*a12*v2 -
a13*d2...
- a13*a11*a12*d2 + a3*a11*a12*v2 + a3*v2 + a13*d3*a11*a12 -
a13*d3);...
+(a12*a2*v2 +a11*v2*a1 -a11*v2*a2 -a12*v2*a1 -d2 -a11*a12*d2 -
a13*a3*a11*a12*v2...
- a13*a3*v2 + d3*a11*a12 - d3); + 2*(1 - a11*a12);...
- 2*a13*(a11*a12 - 1); - 2*a13*v2*(1 + a11*a12); - 2*(a11*a12*v2 +
v2)];

```

```
%spherical joint
```

```
alfa4=-pi/2;
```

```
alfa5=pi/2;
```

```
alfa6=0;
```

```
a4=0;
```

```
a5=0;
```

```
a6=0;
```

```
d4=0;
```

```
d5=0;
```

```
d6=0;
```

```
al4=tan(alfa4/2);
```

```
al5=tan(alfa5/2);
```

```
al6=tan(alfa6/2);
```

```
% cos(u4)=(1-v4^2)/(1+v4^2)
```

```
% sin(u4)=(2*v4)/(1+v4^2)
```

```
% v4=tan(u4/2);
```

```
m4=[1 0 0 0;
```

```
0 (1-v4^2)/(1+v4^2) - (2*v4)/(1+v4^2) 0;
```

```
0 (2*v4)/(1+v4^2) (1-v4^2)/(1+v4^2) 0;
```

```
0 0 0 1];
```

```
g4=[1 0 0 0;
```

```
a4 1 0 0;
```

```
0 0 (1-al4^2)/(1+al4^2) - (2*al4)/(1+al4^2);
```

```
d4 0 (2*al4)/(1+al4^2) (1-al4^2)/(1+al4^2)];
```

```
m5=[1 0 0 0;
```

```
0 (1-v5^2)/(1+v5^2) - (2*v5)/(1+v5^2) 0;
```

```
0 (2*v5)/(1+v5^2) (1-v5^2)/(1+v5^2) 0;
```

```
0 0 0 1];
```

```
g5=[1 0 0 0;
```

```
a5 1 0 0;
```

```
0 0 (1-al5^2)/(1+al5^2) - (2*al5)/(1+al5^2);
```

```
d5 0 (2*al5)/(1+al5^2) (1-al5^2)/(1+al5^2)];
```

```
m6=[1 0 0 0;
```

```
0 (1-v6^2)/(1+v6^2) - (2*v6)/(1+v6^2) 0;
```

```
0 (2*v6)/(1+v6^2) (1-v6^2)/(1+v6^2) 0;
```

```
0 0 0 1];
```

```
g6=[1 0 0 0;
```

```

a6 1 0 0;
0 0 (1-a16^2)/(1+a16^2) -(2*a16)/(1+a16^2);
d6 0 (2*a16)/(1+a16^2) (1-a16^2)/(1+a16^2)];

dualM4 = matrixtodual(m4);
dualG4 = matrixtodual(g4);
dualM5 = matrixtodual(m5);
dualG5 = matrixtodual(g5);
dualM6 = matrixtodual(m6);
dualG6 = matrixtodual(g6);
dualM=dqmult(dualM4,dualG4);
dualM=dqmult(dualM,dualM5);
dualM=dqmult(dualM,dualG5);
dualM=dqmult(dualM,dualM6);
dualM=dqmult(dualM,dualG6);
dualM=simplify(dualM./dualM(1));

%right side
a3=0;
d3=-d4;
a1=a5;
a2=a4;
a3=0;

%pose-dual
t0=dual(1);
t1=dual(2);
t2=dual(3);
t3=dual(4);
t4=dual(5);
t5=dual(6);
t6=dual(7);
t7=dual(8);

hw1 = [(t0*a13*a3 + t1*a3 + t2*a13*d3 + t3*d3 + 2*t4 - 2*t5*a13);...
+ (t1*a13*a3 - t0*a3 + t3*a13*d3 - t2*d3 + 2*t5 + 2*t4*a13);...
+ (t2*a13*a3 - t3*a3 - t0*a13*d3 + t1*d3 + 2*t6 + 2*t7*a13);...
+ (t3*a13*a3 + t2*a3 - t1*a13*d3 - t0*d3 + 2*t7 - 2*t6*a13);...
+ 2*(t0 - t1*a13); + 2*(t1 + t0*a13); + 2*(t2 + t3*a13); + 2*(t3 -
t2*a13)];

hw2 = [(t0*a3 - t1*a13*a3 - t2*d3 + t3*a13*d3 - 2*t4*a13 - 2*t5);...
+ (t1*a3 + t0*a13*a3 - t3*d3 - t2*a13*d3 - 2*t5*a13 + 2*t4);...
+ (t2*a3 + t3*a13*a3 + t0*d3 + t1*a13*d3 - 2*t6*a13 + 2*t7);...
+ (t3*a3 - t2*a13*a3 + t1*d3 - t0*a13*d3 - 2*t7*a13 - 2*t6);...
- 2*(t0*a13 + t1); + 2*(t0 - t1*a13); + 2*(t3 - t2*a13); -2*(t3*a13 +
t2)];

hw3 = [(t0*a13*d3 + t1*d3 - t2*a13*a3 - t3*a3 - 2*t6 + 2*t7*a13);...
+ (t1*a13*d3 - t0*d3 - t3*a13*a3 + t2*a3 - 2*t7 - 2*t6*a13);...
+ (t2*a13*d3 - t3*d3 + t0*a13*a3 - t1*a3 + 2*t4 + 2*t5*a13);...
+ (t3*a13*d3 + t2*d3 + t1*a13*a3 + t0*a3 + 2*t5 - 2*t4*a13);...
+ 2*(-t2 + t3*a13); - 2*(t3 + t2*a13); + 2*(t0 + t1*a13); + 2*(t1 -
t0*a13)];

hw4 = [(t0*d3 - t1*a13*d3 + t2*a3 - t3*a13*a3 - 2*t6*a13 - 2*t7);...
+ (t1*d3 + t0*a13*d3 + t3*a3 + t2*a13*a3 - 2*t7*a13 + 2*t6);...
+ (t2*d3 + t3*a13*d3 - t0*a3 - t1*a13*a3 + 2*t4*a13 - 2*t5);...
+ (t3*d3 - t2*a13*d3 - t1*a3 + t0*a13*a3 + 2*t5*a13 + 2*t4);...

```

```

- 2*(t3 + t2*a13); + 2*(-t3*a13 + t2); + 2*(t0*a13 - t1); + 2*(t0 +
t1*a13)];

%solution for v2
H=[hc1v2,hc2v2,hc3v2,hc4v2,hw1,hw2,hw3,hw4];
deth=det(H)==0;
junta2=solve(deth,v2);
u2=2*atan(double(junta2))*180/pi;

alfa1=pi/2;
alfa2=0;
alfa3=0;
a1=0;
a2=canela;
a3=coxa;
d1=0;
d2=0;
d3=0;
al1=tan(alfa1/2);
al2=tan(alfa2/2);
al3=tan(alfa3/2);

aux=0;
for i=1:length(junta2)

    %R
    H1=subs(H,v2,junta2(i));
    system1=H1'*R==0;
    R1 =
    solve(system1(1),system1(2),system1(3),system1(4),system1(5),system1(6)
),system1(7),x0==1);

    %solution for v1
    hc2v1=(al2*a1+al2*v1*d2*al1-a2*al1-al2*al3*al1*a1-al2*al3*a3*al1-
al2*al3*v1*d2-al2*d3*al1*v1...
        - a2*a13 + al2*a3 + al2*al3*d3*v1)*R1.x0 + (-a2*al3*al1 +
al2*v1*d2 - al2*al3*d3*al1*v1...
        + al2*al3*a1 + al2*al3*a3 - al2*d3*v1 + al2*al1*a1 +
al2*a3*al1 + a2 + al2*al3*v1*d2*al1)*R1.x1...
        + (al2*d3 - al2*d2 + al2*al3*a1*v1 - al2*al3*d2*al1 -
a2*al3*al1*v1 + al2*al1*a1*v1...
        + al2*al3*a3*v1 + al2*a3*al1*v1 + al2*al3*d3*al1 +
a2*v1)*R1.x2...
        + (-a2*al3*v1 + al2*a1*v1 - al2*d2*al1 + al2*al3*d2 -
al2*al3*d3 + al2*a3*v1 + al2*d3*al1...
        - al2*al3*al1*a1*v1 - al2*al3*a3*al1*v1 - a2*al1*v1)*R1.x3 -
2*(al2*al3 + al2*al1)*R1.y0...
        +2*(-al2*al3*al1+al2)*R1.y1+2*(al2*v1-al2*al3*al1*v1)*R1.y2-
2*(al2*al3*v1+al2*al1*v1)*R1.y3 == 0;
    junta1=solve(hc2v1);

    %solution for v3
    hc4v3=(-a1*al1*v3-a1*al1*al2*v3*al3+al2*a2*v3-d2-
al3*a2*v3+al3*al2*d2-al2*al3*d3...
        - d3 + v3*al3*a3 - al2*v3*a3)*R1.x0 + (a1*al1*al2*v3 -
a1*al1*v3*al3 + al3*al2*a2*v3...
        - al3*d2 - al2*d2 + a2*v3 - v3*a3 - al2*v3*al3*a3 + al2*d3 -
al3*d3)*R1.x1...

```

```

        +(a1*a11*a13+a1*a11*a12-a13*a12*a2-a13*d2*v3+a2+a12*d2*v3+a3-
al2*a13*a3...
        - al2*v3*d3 - v3*a13*d3)*R1.x2 + (a1*a11 - a1*a11*a12*a13 -
al2*a2 - d2*v3 - a13*a2...
        -a13*a12*d2*v3 -a13*a3 -a12*a3 +a12*v3*a13*d3 -v3*d3)*R1.x3
+2*(v3 +a12*v3*a13)*R1.y0...
        + 2*(-a12*v3 + v3*a13)*R1.y1 - 2*(a13 + a12)*R1.y2 + 2*(-1 +
al2*a13)*R1.y3 == 0;
    junta3=solve(hc4v3);

    %solution for spherical
    S=[R1.x0; R1.x1; R1.x2; R1.x3; R1.y0; R1.y1; R1.y2; R1.y3];

    Sinv=dqinv(S);
    D=dqmult(Sinv,dual);
    Direito=double(D./D(1));

    system2=dualM==Direito;

    resp1=solve(system2(2),system2(3),system2(4));

    u1=2*atan(double(junta1))*180/pi;

    u3=2*atan(double(junta3))*180/pi;

    u4=2*atan(double(resp1.v4))*180/pi;

    u5=(2*atan(double(resp1.v5))*180/pi)-90;

    u6=2*atan(double(resp1.v6))*180/pi;

    u(:,i+aux:i+1+aux)=[u1 u1; u2(i) u2(i); u3 u3; u4'; u5'; u6']
    aux=aux+1;

end

```

APÊNDICE C

C.1- Sensor Fusion

%Adapted from implementation of Madgwick's IMU and AHRS algorithms

```
clear;
close all;
clc;
addpath('Quaternions');
```

```
load spatial_data.csv
teste=spatial_data;
clear spatial_data;
```

```
for i=2:length(teste(:,1))
    if teste(i,8)==0
        teste(i,8)=teste(i-1,8);
    end
    if teste(i,9)==0
        teste(i,9)=teste(i-1,9);
    end
    if teste(i,10)==0
        teste(i,10)=teste(i-1,10);
    end
end
```

```
fs=125;
samplePeriod =1/fs;
```

```
time=teste(:,1)-teste(1,1);
startTime = 0;
stopTime = time(end);
staticTime=0.1;
```

```
% -----
% Import data
```

```
gyrX = teste(:,5);
gyrY = teste(:,6);
gyrZ = teste(:,7);
accX = teste(:,2);
accY = teste(:,3);
accZ = teste(:,4);
magX = teste(:,8);
magY = teste(:,9);
magZ = teste(:,10);
```

```
% -----
% Manually frame data
```

```
indexSel = find(sign(time-startTime)+1, 1) : find(sign(time-
stopTime)+1, 1);
indexSel2 = find(sign(time-startTime)+1, 1) : find(sign(time-
staticTime)+1, 1);
time = time(indexSel);
gyrX = gyrX(indexSel, :);
```

```

gyrY = gyrY(indexSel, :);
gyrZ = gyrZ(indexSel, :);
accX = accX(indexSel, :);
accY = accY(indexSel, :);
accZ = accZ(indexSel, :);
magX = magX(indexSel, :);
magY = magY(indexSel, :);
magZ = magZ(indexSel, :);

accXfilt = accX;
accYfilt = accY;
accZfilt = accZ;

% -----
% % HP filter gyro data
% filtCutOff = 0.2;
% [b, a] = butter(6, (2*filtCutOff)/(1/samplePeriod), 'high');
% gyrX = filtfilt(b, a, gyrX);
% gyrY = filtfilt(b, a, gyrY);
% gyrZ = filtfilt(b, a, gyrZ);

% LP filter accelerometer data
filtCutOff = 5;
[b, a] = butter(1, (2*filtCutOff)/(1/samplePeriod), 'low');
gyrX = filtfilt(b, a, gyrX);
gyrY = filtfilt(b, a, gyrY);
gyrZ = filtfilt(b, a, gyrZ);
accXfilt = filtfilt(b, a, accXfilt);
accYfilt = filtfilt(b, a, accYfilt);
accZfilt = filtfilt(b, a, accZfilt);

% -----
% Plot data raw sensor data and stationary periods

figure('Position', [9 39 900 600], 'Number', 'off', 'Name', 'Sensor
Data');
ax(1) = subplot(3,1,1);
    hold on;
    plot(time, gyrX, 'r');
    plot(time, gyrY, 'g');
    plot(time, gyrZ, 'b');
    title('Gyroscope');
    xlabel('Time (s)');
    ylabel('Angular velocity (^{\circ}/s)');
    legend('X', 'Y', 'Z');
    hold off;
ax(2) = subplot(3,1,2);
    hold on;
    plot(time, accXfilt, 'r');
    plot(time, accYfilt, 'g');
    plot(time, accZfilt, 'b');
    title('Accelerometer');
    xlabel('Time (s)');
    ylabel('Acceleration (g)');
    legend('X', 'Y', 'Z');
    hold off;
ax(3) = subplot(3,1,3);
    hold on;

```

```

    plot(time, magX, 'r');
    plot(time, magY, 'g');
    plot(time, magZ, 'b');
    title('Magnetometer');
    xlabel('Time (s)');
    ylabel('Magnetometer');
    legend('X', 'Y', 'Z');
    hold off;
linkaxes(ax,'x');

% -----
% Compute orientation

quat = zeros(length(time), 4);
zeta=1;
beta=0.1;
AHRSalgorithm = MadgwickAHRS2('SamplePeriod', 1/fs, 'Beta', beta,
'Zeta',zeta);
% Initial convergence
for i = 1:2000
    % AHRSalgorithm.Update([0 0 0], [mean(accXfilt(indexSel2))
mean(accYfilt(indexSel2)) mean(accZfilt(indexSel2))],...
    % [mean(magX(indexSel2)) mean(magY(indexSel2))
mean(magZ(indexSel2))]);
    AHRSalgorithm.UpdateIMU([0 0 0], [mean(accXfilt(indexSel2))
mean(accYfilt(indexSel2)) mean(accZfilt(indexSel2))]);
end
% For all data
AHRSalgorithm.Zeta = 1;
AHRSalgorithm.Beta =0.1;
for t = 1:length(time)
    %AHRSalgorithm.Update(deg2rad([gyrX(t) gyrY(t) gyrZ(t)]),
[accXfilt(t) accYfilt(t) accZfilt(t)], [magX(t) magY(t) magZ(t)]);
    AHRSalgorithm.UpdateIMU(deg2rad([gyrX(t) gyrY(t) gyrZ(t)]),
[accXfilt(t) accYfilt(t) accZfilt(t)]);
    quat(t,:) = AHRSalgorithm.Quaternion;
end

% -----
% Compute translational accelerations

% Rotate body accelerations to Earth frame
acc = quaternRotate([accXfilt accYfilt accZfilt], quaternConj(quat));

% Convert acceleration measurements to m/s/s
acc = acc * 9.81;

% -----
% High pass filter in acceleration
fc=1;
RC=1/(2*pi*fc);
alfa=RC/(RC+(1/fs));
aux(1,1:3) =[0 0 0];
for i=2:length(time)
    aux(i,1:3) = alfa*aux(i-1,1:3)+alfa*(acc(i,1:3)-acc(i-1,1:3));
end
acc(:,1:3) = aux(:,1:3);

```

```

% LP filter accelerometer data
filtCutOff = 5;
[b, a] = butter(1, (2*filtCutOff)/(1/samplePeriod), 'low');
acc(:,1:3) = filtfilt(b, a, acc(:,1:3));

% Plot translational accelerations
figure('Position', [9 39 900 300], 'Number', 'off', 'Name',
'Accelerations');
hold on;
plot(time, acc(:,1), 'r');
plot(time, acc(:,2), 'g');
plot(time, acc(:,3), 'b');
title('Acceleration');
xlabel('Time (s)');
ylabel('Acceleration (m/s/s)');
legend('Lateral', 'Posterior', 'Vertical');
hold off;

% -----
% Plot Euler Angles

euler = quatern2euler(quaternConj(quat)) * (180/pi); % use
conjugate for sensor frame relative to Earth and convert to degrees.
rollAngle = atan2(accYfilt, accZfilt);
pitchAngle = atan(-accXfilt ./ ((accYfilt .* sin(rollAngle)) +
(accZfilt .* cos(rollAngle))));
yawAngle = atan2((magZ .* sin(rollAngle)) - (magY .*
cos(rollAngle)), (magX .* cos(pitchAngle)) + (magY .* sin(pitchAngle) .*
sin(rollAngle)) + (magZ .* sin(pitchAngle) .* cos(rollAngle)));
figure('Name', 'Euler Angles');
hold on;
plot(time, euler(:,1), 'r');
plot(time, euler(:,2), 'g');
plot(time, euler(:,3), 'b');
title('Euler angles');
xlabel('Time (s)');
ylabel('Angle (deg)');
legend('\phi', '\theta', '\psi');
hold on;
plot(time, rollAngle*180/pi, '.r');
plot(time, pitchAngle*180/pi, '.g');
plot(time, yawAngle*180/pi, '.b');

```


C.2- MadgwickAHRS2

```

classdef MadgwickAHRS2 < handle
%MADGWICKAHRS Implementation of Madgwick's IMU and AHRS algorithms
%
%   For more information see:
%   http://www.x-io.co.uk/node/8#open\_source\_ahrs\_and\_imu\_algorithms
%
%   Date           Author           Notes
%   28/09/2011     SOH Madgwick     Initial release

    %% Public properties
    properties (Access = public)
        SamplePeriod = 1/256;
        Quaternion = [1 0 0 0];    % output quaternion describing the
Earth relative to the sensor
        Beta = 1;                  % algorithm gain
        Zeta = 1;
        wb=[0 0 0];
    end

    %% Public methods
    methods (Access = public)
        function obj = MadgwickAHRS2(varargin)
            for i = 1:2:nargin
                if strcmp(varargin{i}, 'SamplePeriod'),
obj.SamplePeriod = varargin{i+1};
                elseif strcmp(varargin{i}, 'Quaternion'),
obj.Quaternion = varargin{i+1};
                elseif strcmp(varargin{i}, 'Beta'), obj.Beta =
varargin{i+1};
                elseif strcmp(varargin{i}, 'Zeta'), obj.Zeta =
varargin{i+1};
                elseif strcmp(varargin{i}, 'wb'), obj.wb =
varargin{i+1};
                else error('Invalid argument');
            end
        end;
        end
        function obj = Update(obj, Gyroscope, Accelerometer,
Magnetometer)
            q = obj.Quaternion; % short name local variable for
readability
            wbb = obj.wb;

            % Normalise accelerometer measurement
            if(norm(Accelerometer) == 0), return; end % handle NaN
            Accelerometer = Accelerometer / norm(Accelerometer); %
normalise magnitude

            % Normalise magnetometer measurement
            if(norm(Magnetometer) == 0), return; end % handle NaN
            Magnetometer = Magnetometer / norm(Magnetometer); %
normalise magnitude

            % Reference direction of Earth's magnetic feild
            h = quaternProd(q, quaternProd([0 Magnetometer],
quaternConj(q)));
            b = [0 norm([h(2) h(3)]) 0 h(4)];

```

```

% Gradient decent algorithm corrective step
F = [2*(q(2)*q(4) - q(1)*q(3)) - Accelerometer(1)
      2*(q(1)*q(2) + q(3)*q(4)) - Accelerometer(2)
      2*(0.5 - q(2)^2 - q(3)^2) - Accelerometer(3)
      2*b(2)*(0.5 - q(3)^2 - q(4)^2) + 2*b(4)*(q(2)*q(4) -
q(1)*q(3)) - Magnetometer(1)
      2*b(2)*(q(2)*q(3) - q(1)*q(4)) + 2*b(4)*(q(1)*q(2) +
q(3)*q(4)) - Magnetometer(2)
      2*b(2)*(q(1)*q(3) + q(2)*q(4)) + 2*b(4)*(0.5 - q(2)^2
- q(3)^2) - Magnetometer(3)];
J = [-2*q(3),          2*q(4),
      -2*q(1),          2*q(2),
      2*q(2),          2*q(1),
      2*q(4),          2*q(3),
      0,               -4*q(2),
      -4*q(3),          0
      -2*b(4)*q(3),      2*b(4)*q(4),
      -4*b(2)*q(3)-2*b(4)*q(1), -4*b(2)*q(4)+2*b(4)*q(2)
      -2*b(2)*q(4)+2*b(4)*q(2), 2*b(2)*q(3)+2*b(4)*q(1),
      2*b(2)*q(2)+2*b(4)*q(4), -2*b(2)*q(1)+2*b(4)*q(3)
      2*b(2)*q(3),        2*b(2)*q(4)-4*b(4)*q(2),
      2*b(2)*q(1)-4*b(4)*q(3), 2*b(2)*q(2)];
step = (J'*F);
step = step / norm(step); % normalise step magnitude

% % Compute rate of change of quaternion
% qDot = 0.5 * quaternProd(q, [0 Gyroscope(1) Gyroscope(2)
Gyroscope(3)]) - obj.Beta * step';
% % Integrate to yield quaternion
% q = q + qDot * obj.SamplePeriod;
% obj.Quaternion = q / norm(q); % normalise quaternion

%compute angular estimated direction of the gyroscope
error
w_err_x = 2*q(1) * step(2) - 2*q(2) * step(1) - 2*q(3)
* step(4) + 2*q(4) * step(3);
w_err_y = 2*q(1) * step(3) + 2*q(2) * step(4) - 2*q(3)
* step(1) - 2*q(4) * step(2);
w_err_z = 2*q(1) * step(4) - 2*q(2) * step(3) + 2*q(3)
* step(2) - 2*q(4) * step(1);
%compute and remove the gyroscope biases
wbb(1) = wbb(1) + w_err_x * obj.SamplePeriod *
obj.Zeta;
wbb(2) = wbb(2) + w_err_y * obj.SamplePeriod *
obj.Zeta;
wbb(3) = wbb(3) + w_err_z * obj.SamplePeriod *
obj.Zeta;
Gyroscope(1) = Gyroscope(1) - wbb(1);
Gyroscope(2) = Gyroscope(2) - wbb(2);
Gyroscope(3) = Gyroscope(3) - wbb(3);
%compute the quaternion rate measured by gyroscopes
SEqDot_omega_1 = -0.5*q(2) * Gyroscope(1) - 0.5*q(3) *
Gyroscope(2) - 0.5*q(4) * Gyroscope(3);
SEqDot_omega_2 = 0.5*q(1) * Gyroscope(1) + 0.5*q(3) *
Gyroscope(3) - 0.5*q(4) * Gyroscope(2);
SEqDot_omega_3 = 0.5*q(1) * Gyroscope(2) - 0.5*q(2) *
Gyroscope(3) + 0.5*q(4) * Gyroscope(1);
SEqDot_omega_4 = 0.5*q(1) * Gyroscope(3) + 0.5*q(2) *
Gyroscope(2) - 0.5*q(3) * Gyroscope(1);
%compute then integrate the estimated quaternion rate

```

```

        q(1) = q(1) + (SEqDot_omega_1 - (obj.Beta * step(1)))
* obj.SamplePeriod;
        q(2) = q(2) + (SEqDot_omega_2 - (obj.Beta * step(2)))
* obj.SamplePeriod;
        q(3) = q(3) + (SEqDot_omega_3 - (obj.Beta * step(3)))
* obj.SamplePeriod;
        q(4) = q(4) + (SEqDot_omega_4 - (obj.Beta * step(4)))
* obj.SamplePeriod;
        %normalise quaternion
        norma = sqrt(q(1) * q(1) + q(2) * q(2) + q(3) * q(3) +
q(4) * q(4));
        q(1) = q(1) /norma;
        q(2) = q(2) /norma;
        q(3) = q(3) /norma;
        q(4) = q(4) /norma;
        obj.Quaternion = q;
        obj.wb = wbb;

end
function obj = UpdateIMU(obj, Gyroscope, Accelerometer)
    q = obj.Quaternion; % short name local variable for
readability

    % Normalise accelerometer measurement
    if(norm(Accelerometer) == 0), return; end % handle NaN
    Accelerometer = Accelerometer / norm(Accelerometer); %
normalise magnitude

    % Gradient decent algorithm corrective step
    F = [2*(q(2)*q(4) - q(1)*q(3)) - Accelerometer(1)
        2*(q(1)*q(2) + q(3)*q(4)) - Accelerometer(2)
        2*(0.5 - q(2)^2 - q(3)^2) - Accelerometer(3)];
    J = [-2*q(3),    2*q(4),    -2*q(1), 2*q(2)
        2*q(2),    2*q(1),    2*q(4), 2*q(3)
        0,        -4*q(2),    -4*q(3), 0    ];
    step = (J'*F);
    step = step / norm(step); % normalise step magnitude

    % Compute rate of change of quaternion
    qDot = 0.5 * quaternProd(q, [0 Gyroscope(1) Gyroscope(2)
Gyroscope(3)]) - obj.Beta * step';

    % Integrate to yield quaternion
    q = q + qDot * obj.SamplePeriod;
    obj.Quaternion = q / norm(q); % normalise quaternion
end
end
end
end

```