

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

**DESENVOLVIMENTO DE DISPOSITIVOS NÃO
MOTORIZADOS PARA REABILITAÇÃO DO MEMBRO
INFERIOR HUMANO**



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA MECÂNICA
2017**

LUCAS ANTÔNIO OLIVEIRA RODRIGUES

DESENVOLVIMENTO DE DISPOSITIVOS NÃO MOTORIZADOS PARA REABILITAÇÃO DO MEMBRO INFERIOR HUMANO

Dissertação apresentada ao Programa de Pós-graduação em Engenharia Mecânica da Universidade Federal de Uberlândia, como parte dos requisitos para a obtenção do título de **MESTRE EM ENGENHARIA MECÂNICA**.

Área de concentração: Mecânica dos Sólidos e Vibrações.

Orientador: Prof. Dr. João Carlos Mendes Carvalho

Co-orientador: Prof. Dr. Rogério Sales Gonçalves

Uberlândia - MG

2017

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

R696d Rodrigues, Lucas Antônio Oliveira, 1990-
2017 Desenvolvimento de dispositivos não motorizados para reabilitação
do membro inferior humano / Lucas Antônio Oliveira Rodrigues. - 2017.
103 f. : il.

Orientador: João Carlos Mendes Carvalho.
Coorientador: Rogério Sales Gonçalves.
Dissertação (mestrado) - Universidade Federal de Uberlândia,
Programa de Pós-Graduação em Engenharia Mecânica.
Inclui bibliografia.

1. Engenharia mecânica - Teses. 2. Membros inferiores -
Reabilitação - Teses. I. Carvalho, João Carlos Mendes, 1954- II.
Gonçalves, Rogério Sales, 1981- III. Universidade Federal de
Uberlândia. Programa de Pós-Graduação em Engenharia Mecânica. IV.
Título.

"Feliz é a pessoa que acha a sabedoria e que consegue compreender as coisas, pois isso é melhor do que a prata e tem mais valor do que o ouro. A sabedoria é mais preciosa do que as joias; tudo o que a gente deseja não se pode comparar com ela. A sabedoria oferece uma vida longa e também riquezas e honras. Ela torna a vida agradável e guia a pessoa com segurança em tudo o que faz. Os que se tornam sábios são felizes, e a sabedoria lhes dará vida. "

Livro de Provérbios 3:13-18 [NTLH]

AGRADECIMENTOS

À Deus, pela oportunidade de viver este momento, através de todas as dificuldades e alegrias do caminho.

À Universidade Federal de Uberlândia e à Faculdade de Engenharia Mecânica pela oportunidade de realizar este Curso.

À minha noiva, familiares e colegas pelo apoio e pelo incentivo moral, desde o princípio até o final desta importante jornada de nossas vidas.

Ao CNPq pelo apoio financeiro e desenvolvimento total deste projeto

Aos meus orientadores, prof. Rogério Sales e prof. João Carlos, pela paciência e sabedoria para me direcionar e orientar ao longo de muitos anos, desde o início da graduação, até a realização deste projeto.

RODRIGUES, L. A. O., **Desenvolvimento de dispositivos não motorizados para reabilitação do membro inferior humano** 2017. 119 f. Dissertação de Mestrado, Universidade Federal de Uberlândia, Uberlândia.

RESUMO

O objetivo deste trabalho é desenvolver três dispositivos não-motorizados para a reabilitação de cada uma das articulações do membro inferior humano. Este tipo de sistema está disponível no comércio, porém seu acionamento é motorizado e automático. A construção dos sistemas aqui propostos visa explorar mecanismos puramente mecânicos, esperando obter sistemas mais facilmente operados por pacientes e profissionais da saúde. A modelagem matemática dos mecanismos é obtida a partir da análise de mecanismos de quatro barras e do equilíbrio estático do sistema. Os mecanismos transmitem movimento angular gerado pelo braço do paciente para um movimento oscilatório, direcionado às articulações da perna a serem trabalhadas. Os mecanismos são dimensionados com o auxílio de um algoritmo genético denominado evolução diferencial e um algoritmo para a análise de singularidades a partir da matriz geométrica dos sistemas. A partir deste modelo foi construído um protótipo para o movimento do joelho, a fim de validar a saída angular dos mecanismos propostos, averiguar a existência de singularidades e realizar testes utilizando um boneco de madeira antropométrico. Utilizou-se também o protótipo para o desenvolvimento de um jogo controlado pelo paciente como incentivo para a realização dos exercícios.

Palavras Chave: mecanismos de quatro barras, sistemas de reabilitação, membro inferior, jogos para terapia, análise de singularidades.

RODRIGUES, L. A. O., **Development of non-motorized mechanisms for lower limb rehabilitation** 2017. 119 s. Master thesis, Federal University of Uberlândia, Uberlândia, Brazil.

ABSTRACT

The objective of this work is to present the development of three non-motorized mechanisms for rehabilitation of each joint of the human lower limb. This type of system already exists but it is motorized and automatic. The structures here are mechanical devices, in order to obtain the system proposed whose operation is simpler for both patients and health professionals. The mathematical modeling of the mechanisms is based on the four-bar linkages and the static equilibrium of the system. The mechanism transmits angular movement generated by the patient's arm to an oscillatory movement on the respective joint to be moved. For designing each bar of the mechanisms, it is used a genetic algorithm called differential evolution and singularity analysis based on the geometrical matrix of the systems. To validate the system, a prototype for the knee was constructed to verify the angular outputs, check the existence of singularities, and to execute exercises with a wooden test dummy. This prototype was also used for the development and implementing a game, controlled by the patient, helping its motivation exercises.

Keywords: *four-bar linkage, rehabilitation systems, lower limb, therapy games, singularity analysis.*

Lista de Figuras

1.1	À esquerda, foto de uma prancha oscilante usado em tratamento de lesões do tornozelo. À direita, um exemplo de exercício usando a prancha (Figura adaptada de: http://www.fitter1.com).	2
1.2	Paciente utilizando um jogo desenvolvido para reabilitação usando o Wii Balance Board (GIL-GOMÉZ, 2011).	3
2.1	Tipos de estruturas robóticas utilizando (a) esteiras, (b) base programável para os pés, (c) base robótica móvel e (d) base robótica estacionária (Adaptada de DIAZ, GIL e SANCHÉZ, 2011).	6
2.2	Foto de um paciente utilizando a estrutura ReoAmbulator (Figura adaptada de: http://motorika.com/reoambulator).	7
2.3	Foto de um paciente utilizando a estrutura LOKOMAT (JEZERNIK; COLOMBO; MORARI, 2004).	7
2.4	Estrutura não-motorizada para mobilidade de pacientes com deficiência (TSUGE; McCARTHY, 2016).	8
2.5	(a) Projeção ortogonal isométrica do modelo computacional estrutura LLRR e,(b) Detalhe dos cursores atuadores nas bases para os pés e barras para transmissão do movimento (CHEN et al., 2011).	9
2.6	Imagen de um paciente operando o <i>Gangtrainer GT I</i> (HESSE; UHLENBROCK, 2000).	10
2.7	Imagen de um paciente operando o <i>G-EO System</i> (adaptada de: https://www.rehatechnology.com/)	10
2.8	Foto de paciente utilizando a estrutura de Rutgers, enquanto pratica um jogo terapêutico gerado pela mesma (BOIAN et al., 2004).	11

2.9	Modelo ilustrando o ReWalk (GOFFER,2006)	12
2.10	Foto de um paciente utilizando a WalkTrainer (STAUFFER et al., 2009)	13
2.11	Estrutura não-motorizada para mobilidade de pacientes com deficiência (SOARES JÚNIOR, 2015)	13
2.12	Modelo do Physioterabot, e foto de um paciente utilizando um protótipo do mesmo (AKDOGAN; ADLI, 2011)	14
2.13	Imagen de operação da estrutura MotionMaker TM (adaptada de: http://www.swortec.ch/index.php/products/motionmaker)	15
2.14	Foto de um paciente utilizando a estrutura vi-RABT (FARJATIAN et al., 2014)	15
2.15	(a) Planos de referência do corpo humano (DANGELO; FATTINI,2007); e (b) (XX') eixo transverso, (YY') eixo ântero-posterior,(OZ) eixo vertical, (OR) eixo longitudinal do quadril (KAPANDJI, 2000)	16
2.16	(a) Movimentos de flexão do quadril; (b) movimentos de extensão do quadril (KAPANDJI, 2000)	17
2.17	a) Movimentos de abdução do quadril; (b) exemplos de movimentos de adução relativa (KAPANDJI, 2000)	18
2.18	(a) Rotação interna do Quadril; (b) posição de referência; (c) rotação externa do quadril (KAPANDJI, 2000)	18
2.19	(XX') Eixo transverso, (YY') eixo longitudinal do joelho, (ZZ') eixo de referência (KAPANDJI, 2000)	19
2.20	Movimentos de flexão-extensão do joelho (KAPANDJI, 2000)	20
2.21	Movimentos de rotação longitudinal do joelho (KAPANDJI, 2000)	20
2.22	(XX') eixo transversal, (Y) eixo longitudinal da perna (Z) eixo longitudinal do pé (KAPANDJI, 2000)	21
2.23	(a)A: posição anatômica a 0°, B: Flexão do tornozelo, C: extensão do tornozelo; (b) Amplitudes da flexão-extensão do tornozelo (KAPANDJI, 2000)	22
2.24	Decomposição de um mecanismo em elemento condutor (2) e dois grupos de Assur (3 e 4, 5 e 6)	23
2.25	Proporções para mecanismos de quatro barras segundo a Lei de Grashof.	23

3.14	Variação dos autovalores não nulos da montagem para o tornozelo.	52
3.15	Variação dos autovalores não nulos da montagem para o joelho.	52
3.16	Variação dos autovalores não nulos da montagem para o quadril.	52
3.17	Sequência da simulação cinemática gerada pelo modelo tridimensional do tornozelo: (a) posição anatômica, (b) Extensão parcial, (c) extensão total, (d) posição anatômica, (e) flexão total.	53
3.18	Sequência da simulação cinemática gerada pelo modelo tridimensional do joelho: (a) posição anatômica, (b) flexão parcial, (c) flexão total, (d) flexão parcial, (e) posição anatômica.	53
3.19	Sequência da simulação cinemática gerada pelo modelo tridimensional do quadril: (a,b) flexão parcial, (c) posição próxima a anatômica, (d) extensão total,(e) posição próxima a anatômica.	54
4.1	captura de tela da plataforma utilizada para gerar o jogo para reabilitação. .	56
4.2	Captura de tela de programação do objeto do personagem principal.	57
4.3	Captura de tela de programação da <i>sprite</i> utilizada para as plataformas do jogo.	58
4.4	Captura de tela de programação do objeto de controle dos eventos principais das salas do jogo.	59
4.5	Captura de tela de programação do objeto de marcação de pontos, evidenciando os blocos de programação utilizados no lugar de um código escrito.	60
4.6	Captura de tela da interface de construção de sala usada na criação da sala principal do jogo.	62
4.7	Captura de tela da execução da versão final do jogo produzido, para operar em conjunto com o a estrutura construída.	62
4.8	Fluxograma básico da lógica de funcionamento do jogo.	63
5.1	(a) Montagem do protótipo para o joelho realizada no Laboratório de Automação e Robótica da Universidade Federal de Uberlândia, e (b) montagem do protótipo em um boneco de testes.	65

5.2	Detalhe na conexão das barras r_1 , r_2 , a_1 e a polia conduzida; e (b) detalhe na conexão da barra a_1 , barra de acionamento a_2 e polia condutora.	66
5.3	(a) Detalhe na conexão das barras r_2 e r_3 ; e (b) detalhe entre a conexão das barras r_3 e r_4	66
5.4	Detalhe na conexão das barras r_1 e r_4	67
5.5	Conexão do protótipo com a perna do boneco de testes através de órtese. . .	68
5.6	Sequência de movimentos captados da movimentação contínua do protótipo construído: (a,b,c) posição em flexão, (d) posição em extensão, (e) posição anatômica.	68
5.7	Esquema de conexão utilizado para conectar o encoder ao Arduino (adaptado de: http://www.arduinoecia.com.br/).	70
5.8	Detalhamento do encoder associado ao protótipo.	71
5.9	Imagens de um jogador realizando um teste de integração jogo-estrutura. . .	71

Lista de Tabelas

3.1	Parâmetros utilizados para encontrar os comprimentos das barras, e ângulos de saída selecionados para cada mecanismo	40
3.2	Massa relativa dos segmentos do corpo humano (CLAUSER; McCONVILLE; YOUNG, 1969)	43
3.3	Direções adotadas para a aceleração gravitacional, em relação a barra r_1 , nos modelos construídos.	43
3.4	Valores máximos de tensão obtidos durante a análise estática	47
3.5	Relações de transmissão escolhidas para as polias dentadas	47
5.1	Medições de amplitude de saída do protótipo.	69
5.2	Análise estatística das medições realizadas.	69

Notação

- r_i → Comprimento da barra de índice i ;
 \mathbf{F}_{ij} → Força exercida pela barra i sob a barra j ;
 $[A]$ → Matriz quadrada;
 $[G_e]$ → Matriz geométrica elementar;
 $[\bar{G}_e]$ → Matriz geométrica elementar expandida;
 $[G]$ → Matriz geométrica global;
 A → Ponto no espaço;
 θ_i → Posição angular da barra i ;
 σ_i → Tensão combinada na barra i ;
 σ_{N_i} → Tensão normal à seção da barra i devido a carregamentos de tração ou compressão;
 σ_{F_i} → Tensão normal à seção da barra i devido a forças de flexão;
 ${}^i\mathbf{a}, {}^i\mathbf{F}$, → Vetores escritos em relação ao referencial i : notação em negrito;
 $\{A\}$ → Vetor matricial;
 ${}^j\mathbf{a}_P$ → Vetor posicionado no ponto P em relação ao referencial j .

SUMÁRIO

Lista de Figuras	vii
Lista de Tabelas	xii
Capítulo 1 INTRODUÇÃO	1
Capítulo 2 REVISÃO BIBLIOGRÁFICA	5
2.1 Estruturas robóticas aplicadas à reabilitação do membro inferior	5
2.1.1 Estruturas robóticas com esteiras	5
2.1.2 Estruturas robóticas com base programável para os pés	8
2.1.3 Estruturas robóticas com base móvel	11
2.1.4 Estruturas robóticas com base estacionária	14
2.2 Movimentos do membro inferior humano	16
2.2.1 O quadril	16
2.2.2 O Joelho	18
2.2.3 O Tornozelo	21
2.3 Mecanismos articulados	21
2.3.1 Modelagem matemática para mecanismos planos de quatro barras . .	24
2.4 Análise de singularidade em mecanismos	27
2.4.1 Equação das velocidades nodais	28
2.4.2 Análise das singularidades	31
2.5 Transmissão por correia dentada e correntes	33
2.6 Otimização por algoritmo genético	35
Capítulo 3 MODELAGEM MATEMÁTICA DOS MECANISMOS	38
3.1 Modelagem matemática	39

3.1.1	Simulação gráfica dos mecanismos	40
3.1.2	Amplitudes de saída	42
3.2	Análise de carregamento	42
3.3	Dimensionamento das polias	46
3.4	Análise de singularidades	49
3.5	Modelagem CAD	53
Capítulo 4	APLICAÇÃO COM JOGOS ELETRÔNICOS	55
4.1	Plataforma de Desenvolvimento	55
4.2	Elementos do jogo	56
Capítulo 5	PROCEDIMENTOS EXPERIMENTAIS	64
5.1	Construção do protótipo	64
5.2	Testes com o boneco de madeira	67
5.3	Interface com o protótipo	69
5.4	Teste de integração jogo-estrutura	70
CONCLUSÕES	73
Referências Bibliográficas	74
APÊNDICE A.1	79
APÊNDICE A.2	83
APÊNDICE A.3	87
APÊNDICE A.4	97
ANEXO	98

CAPÍTULO I

INTRODUÇÃO

A robótica possui uma vasta área de aplicação dentro das quais pode-se incluir a terapia de movimento para diversos tipos de pacientes, tais como vítimas de Acidente Vascular Cerebral (AVC) (LUM et al., 2002), Lesão Encefálica Adquirida (RIBERTO et al., 2007), e lesões medulares prejudiciais aos movimentos necessários para a marcha humana (JEZERNIK et al., 2003). Dentro destas aplicações, o uso de estruturas robóticas gerou benefícios como a redução de custos com mão-de-obra ativa para tratamentos de reabilitação baseados em movimentação, bem como a expansão da gama de exercícios realizados, auxiliando pacientes crônicos a manter a mobilidade através de terapia contínua (LUM et al., 2002).

A proposta de se desenvolver estruturas não motorizadas contribui para que a redução de custos acima citada possa ser aplicada também a custos com fabricação, material e manutenção dos equipamentos, uma vez que os sistemas são robustos e compostos por elementos mecânicos simples e de fácil manuseio.

Tal importância destes equipamentos pode ser complementada com o fato de o acidente vascular cerebral causar, em grande parte dos casos, sequelas na capacidade motora das vítimas, além de ser a terceira causa de morte mais frequente nos Estados Unidos e Europa (LLOYD-JONES et al., 2010).

O trabalho apresentado por Bateni e Maki (2005) evidencia a capacidade de sistemas auto operados de gerar efeitos positivos no tratamento de lesões. Os autores demonstram em seu estudo que o uso deste tipo de dispositivo trouxe benefícios funcionais aos pacientes, bem como um benefício psicossocial, uma vez que, ao manterem sua mobilidade, estes podem manter suas atividades com maior autonomia e independência.

Dispositivos tradicionais, como a cadeira de rodas, por gerarem autonomia ao paciente, também podem representar uma transição para uma vida mais independente, a partir de algum momento do processo de reabilitação e pela convivência cotidiana com os equipamentos que o auxiliam em suas atividades diárias (COSTA et al., 2010).

O resgate de funções que geram independência funcional para os pacientes contribui

para a autoestima, gosto pela vida e sensação de superação de dificuldades do usuário, implicando uma melhoria na qualidade de vida (CAMARGOS et al., 2012). Na Figura 1.1 é ilustrado um exemplo de dispositivo auto operado tradicionalmente utilizado para tratamento de lesões no tornozelo, denominado prancha oscilante. Espera-se que benefícios similares possam ser obtidos pelos mecanismos aqui desenvolvidos.



Figura 1.1 – À esquerda, foto de uma prancha oscilante usado em tratamento de lesões do tornozelo. À direita, um exemplo de exercício usando a prancha (Figura adaptada de: <http://www.fitter1.com>).

Outro benefício gerado pela independência funcional é a redução do tempo ocioso de cuidadores de pacientes com Lesão Encefálica Adquirida, apontado como uma das principais fontes de estresse de seu trabalho. Eles reportam que constantemente auxiliam pacientes em tarefas que os mesmos são capazes de realizar sozinhos para reduzir o tempo da atividade, liberando os cuidadores para cuidar de assuntos próprios (RIBERTO et al., 2007).

O uso dos sistemas não motorizados propostos, devido à simplicidade de operação, pode contribuir também para esse aspecto, não apenas pelo resgate de independência funcional dos pacientes oriundo do resultado do tratamento, como também para a realização dos exercícios, que demandariam menos esforço de cuidadores e terapeutas.

Há ainda evidências clínicas que o treinamento ativamente assistido pode gerar redução significativa ao dano causado à movimentação do membro, conforme estudo realizado em 100 pacientes de AVC agudo (FEYS et al., 1998).

A inserção de ambientes virtuais e jogos ao tratamento aplicado ajuda a minimizar o efeito tedioso gerado pelos procedimentos repetitivos da fisioterapia tradicional, melhorando ativamente a motivação dos pacientes (MIREALMAN et al., 2010). Na Figura 1.2 é exemplificado tal alternativa com um jogo desenvolvido utilizando uma prancha oscilante eletrônica.

O estudo realizado por Lohse et al. (2013) mostra também que o uso de videogames no tratamento de pacientes vítimas de AVC melhora a adesão ao tratamento devido ao aumento da motivação e resulta em eventos como o aumento da frequência dos pacientes, o prolongamento das sessões de terapia e a participação voluntária dos pacientes, onde eles continuam a fazer os exercícios sem serem solicitados.



Figura 1.2 – Paciente utilizando um jogo desenvolvido para reabilitação usando o Wii Balance Board (GIL-GOMÉZ, 2011).

Então, o uso do jogo em conjunto com um sistema auto operado é capaz de associar mais aspectos positivos ao tratamento proporcionado.

Dessa forma, este trabalho procura associar as vantagens do uso de estruturas não motorizadas, acionadas pelo próprio paciente, com jogos motivacionais para acelerar o processo de recuperação de pacientes com redução da mobilidade dos membros inferiores.

Este trabalho apresenta os procedimentos para o desenvolvimento de três mecanismos, sendo um para cada articulação do membro inferior humano, acionados diretamente pelo paciente que os utiliza, gerando movimentos para uma terapia de movimentação contínua.

Além disso, foi elaborado um "jogo eletrônico" para motivar o paciente para a execução das atividades terapêuticas.

A construção de um protótipo para um dos mecanismos propostos permitiu realizar testes experimentais afim de comprovar a capacidade real das estruturas de gerar uma saída angular compatível com a movimentação das articulações para a terapia dos pacientes. Além disso, também associou-se um jogo eletrônico ao protótipo para complementar e incentivar o paciente ao operar o equipamento.

Para descrever as atividades desenvolvidas, esta dissertação está dividida em quatro capítulos, sendo o primeiro esta introdução.

No Capítulo II é apresentada a revisão bibliográfica com a descrição dos conhecimentos básicos necessários para o desenvolvimento das estruturas para reabilitação, objeto de estudo desta dissertação. Inclui-se neste capítulo a revisão das estruturas robóticas aplicadas à reabilitação do membro inferior; os movimentos do membro inferior humano; mecanismos articulados mais especificamente sobre mecanismos planos de quatro barras; o estudo sobre singularidade de mecanismos; transmissão por correntes e correias sincronizadoras e otimização pelo algoritmo genético por Evolução Diferencial.

No Capítulo III é apresentada a modelagem matemática dos três mecanismos propostos,

incluindo a análise de forças; estudo de singularidades e o respectivo dimensionamento mecânico.

No Capítulo IV é apresentado o jogo eletrônico e sua associação ao mecanismo construído, e no Capítulo V são detalhadas a construção do protótipo e os procedimentos experimentais para a validação do sistema mecanismo+jogo.

Finalmente tem-se as conclusões com as proposições para a continuidade da pesquisa.

CAPÍTULO II

REVISÃO BIBLIOGRÁFICA

O Desenvolvimento de estruturas robóticas e mecanismos para aplicações clínicas requer um estudo prévio da cinesiologia dos membros inferiores humanos, bem como os métodos para análise e síntese de mecanismos e estruturas multicorpos. Nesta seção será apresentada uma breve revisão acerca destes tópicos.

2.1 Estruturas robóticas aplicadas à reabilitação do membro inferior

Ao abordar a literatura existente para as estruturas robóticas aplicadas na reabilitação do membro inferior, pode-se identificar quatro tipos de estruturas já bem definidas, baseadas no estudo de Diaz, Gil e Sánchez (2011), sendo: estruturas reabilitadoras por (i) esteiras; (ii) base programável para os pés; (iii) base robótica móvel; e (iv) base estacionária. A Figura 2.1 mostra um esquema ilustrativo de cada um destes tipos de estrutura.

Adiante são detalhados alguns exemplos de projetos que se enquadram dentro dos grupos apresentados.

2.1.1 *Estruturas robóticas com esteiras*

O uso de esteiras na fisioterapia de pacientes é uma prática tradicional para a reabilitação da marcha humana (WERNIG et al., 1995). A partir deste princípio, várias estruturas robóticas que utilizam esteiras e exoesqueletos para auxiliar a movimentação dos pacientes foram desenvolvidas com o objetivo de diminuir o uso de recursos humanos para a execução dos exercícios (GALVEZ; REINKENSMEYER, 2005).

A presença de sistemas robóticos deste tipo no mercado, tais como a Lokomat (JEPERZENIK; COLOMBO; MORARI, 2004) e ReoAmbulator (WEST, 2004) resultou em diversos estudos acerca da eficiência e aplicações deste tipo de estrutura.

As estruturas deste grupo caracterizam-se essencialmente pela presença de atuadores associados aos membros inferiores dos pacientes (exoesqueleto) com ações precisamente sincro-

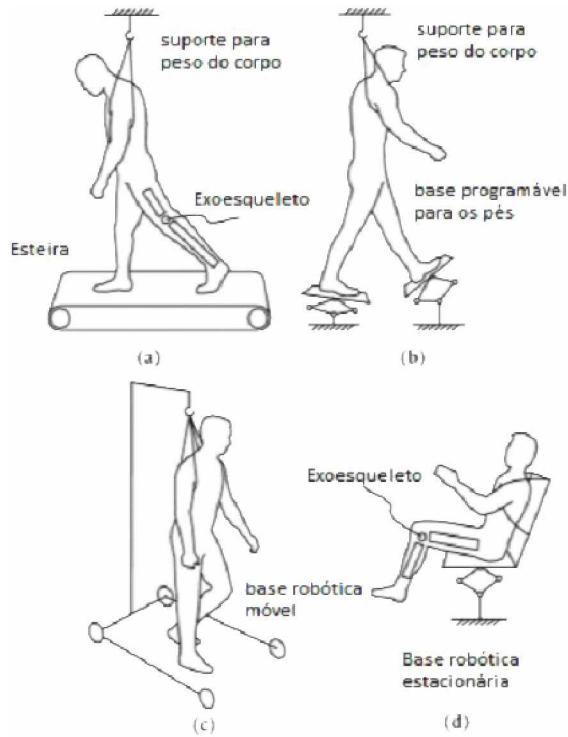


Figura 2.1 – Tipos de estruturas robóticas utilizando (a) esteiras, (b) base programável para os pés, (c) base robótica móvel e (d) base robótica estacionária (Adaptada de DIAZ, GIL e SANCHÉZ, 2011).

nizadas com a velocidade de uma esteira (DIAZ; GIL; SANCHÉZ, 2011). O ReoAmbulator utiliza um exoesqueleto associado diretamente às pernas dos pacientes para auxiliar o movimento sobre a esteira, Fig. 2.2. Para o caso da Lokomat, os atuadores são três motores capazes de mover o joelho e o quadril do usuário através do exoesqueleto, Fig. 2.3.

Ambos os sistemas foram testados em pacientes para levantamento experimental de resultados. O Lokomat foi testado em simulações computacionais e em voluntários saudáveis e portadores de lesão medular, obtendo boa adaptação dos algoritmos de controle aplicados, com melhores resultados para a trajetória do quadril. O ReoAmbulator já se encontra disponível no mercado norte-americano, sendo distribuído pela Motorika®.

Recentemente, Tsuge e McCarthy (2016) apresentaram a modelagem de uma estrutura de um grau de liberdade combinando mecanismos de seis barras e cames para gerar uma saída similar à trajetória do calcanhar durante a marcha humana, o qual é operado sobre uma esteira. O modelo obtido está ilustrado na Fig. 2.4.

Nessa abordagem, o mecanismo proposto visa auxiliar o processo natural de movimentação do membro inferior durante a marcha, utilizando um tipo inovador de mecanismo para acompanhar a movimentação do paciente. Esta proposta encontra-se atualmente em testes para determinar a viabilidade deste modelo para testes clínicos (TSUGE; McCARTHY, 2016).

Um trabalho recente de Stoller et al. (2015) apresenta um estudo sobre a influência de novas técnicas de controle para estruturas robóticas com esteiras, e as respectivas influências



Figura 2.2 – Foto de um paciente utilizando a estrutura ReoAmbulator (Figura adaptada de: <http://motorika.com/reoambulator>).

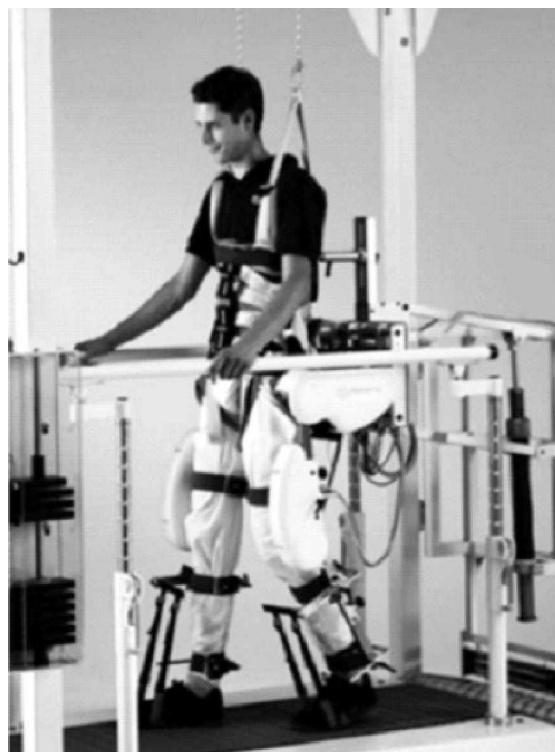


Figura 2.3 – Foto de um paciente utilizando a estrutura LOKOMAT (JEZERNIK; COLOMBO; MORARI, 2004).

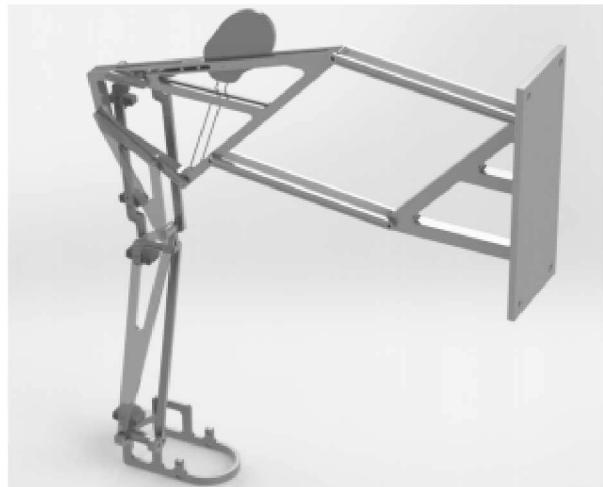


Figura 2.4 – Estrutura não-motorizada para mobilidade de pacientes com deficiência (TSUGE; McCARTHY, 2016).

no tratamento dos pacientes. Foi concluído que as estruturas que utilizam sinais de *feedback* dos usuários são capazes de elevar a intensidade dos exercícios, porém não resultaram em melhorias significativas quando comparadas com os exercícios tradicionais. Tal resultado é um importante indicativo da necessidade de melhorias nos algoritmos de controle para este tipo de estrutura.

Observa-se então que este tipo de estrutura já está se consolidando no mercado de equipamentos médicos e que a pesquisa nesta área ainda é necessária para que melhores resultados sejam alcançados. A proposta apresentada nesta dissertação de criar estruturas não-motorizadas faz com que a melhoria apontada por Stoller et al. (2015) não necessite ser levada em consideração neste projeto, uma vez que não há necessidade de um controle ativo do acionamento da estrutura.

2.1.2 *Estruturas robóticas com base programável para os pés*

Neste tipo de estrutura, encontram-se características similares às apresentadas no tópico anterior, porém destacam-se pela presença de mecanismos ativos que atuam diretamente nos pés do paciente enquanto ele permanece suspenso, mantendo os membros tratados livres para movimentação.

A estrutura LLRR, proposta por Chen et al., (2011), exemplifica este conceito. Seu projeto consiste em uma base fixa, sob a qual o peso do paciente é apoiado por meio de um cabo de suspensão, deixando assim as pernas livres para movimentação. A estrutura consiste em um sistema de três barras para cada perna, uma base para os pés e dois motores DC. A estrutura está ilustrada na Figura 2.5(a).

O princípio básico de funcionamento desta estrutura consiste em um cursor acionado através do motor DC que impulsiona a base de apoio do pé para cima. A partir deste movimento, as demais barras transmitem o movimento para todo o membro inferior, trabalhando assim o movimento do calcane, joelho e quadril. Na Figura 2.5(b) é possível observar o cursor e a

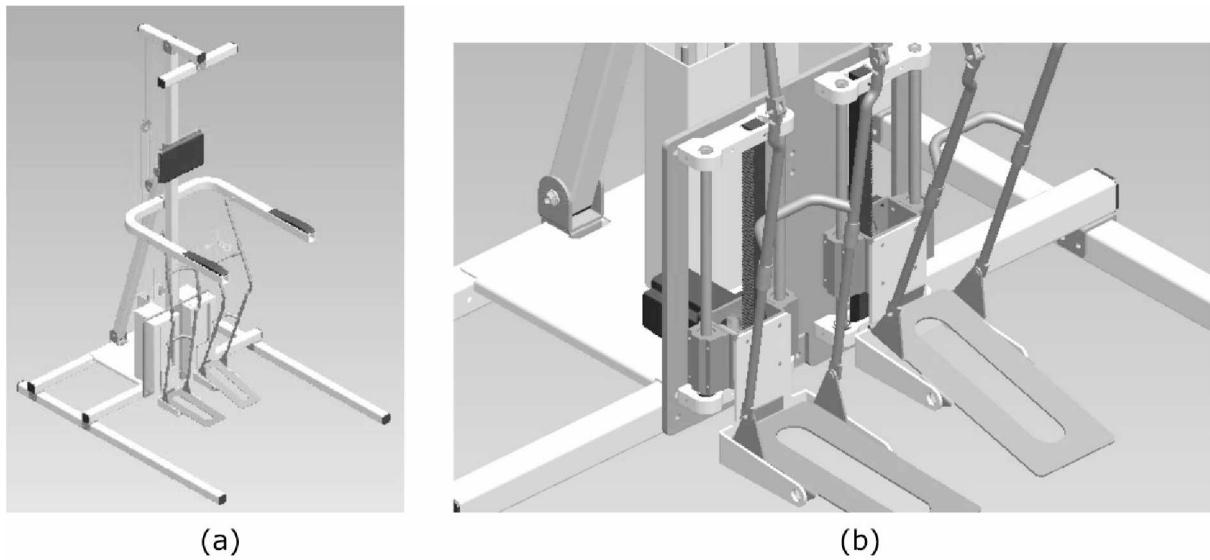


Figura 2.5 – (a) Projeção ortogonal isométrica do modelo computacional estrutura LLRR e,(b) Detalhe dos cursores atuadores nas bases para os pés e barras para transmissão do movimento (CHEN et al., 2011).

projeção das barras.

O projeto possui como vantagens a atuação em ambas as pernas do paciente e o fato de ser composto por mecanismos simples, facilitando seu controle e minimizando as possibilidades de falhas estruturais, o que implica em uma maior segurança operacional. Contudo, por se tratar de um mecanismo que atua com apenas um motor por perna do paciente, o mesmo se limita a replicar apenas movimentos elementares nos membros inferiores, podendo ser aplicado apenas para exercícios básicos de movimentação.

Outros exemplos para este tipo de estrutura são a *Gangtrainer GT I*, distribuída pela Reha-Stim®, e a *G-EO System*, comercializada pela Reha Technologies®. Nestes equipamentos, o peso do paciente é suportado por um sistema de suspensão. As bases programáveis se ajustam para o movimento natural dos pés. Na estrutura *Gangtrainer GT I*, cordas simulam a movimentação lateral do corpo para reproduzir a movimentação de forma natural (HESSE; UHLENBROCK, 2000). Uma figura ilustrativa do *Gangtrainer GT I* é apresentada na Fig. 2.6.

Estudos clínicos já foram realizados com esse equipamento e apontam que sua eficiência é, no mínimo, similar a de tratamentos que utilizam esteiras, porém promovendo mais independência para o fisioterapeuta durante as sessões de exercícios (WERNER et al., 2002), (PEURAL; AIRAKSINEN; HUUSKONEN, 2009).

Na estrutura *G-EO System*, a base atua nos pés através de dois braços robóticos que replicam o movimento da marcha humana. Durante testes comparativos em pacientes com Mal de Parkinson, a estrutura obteve resultados estatísticos mais eficientes (Galli et al., 2016). Uma figura ilustrativa da *G-EO System* é apresentada na Fig. 2.7.

Algumas estruturas dedicam-se apenas para a reabilitação do tornozelo, pois esta é uma das articulações mais comumente lesionadas do corpo humano, sendo estimada uma lesão



Figura 2.6 – Imagem de um paciente operando o *Gangtrainer GT I* (HESSE; UHLENBROCK, 2000).



Figura 2.7 – Imagem de um paciente operando o *G-EO System* (adaptada de: <https://www.rehatechnology.com/>)

para cada trinta pessoas anualmente (WATERMAN et al., 2010). Devido à complexidade de sua estrutura óssea e ao volume de ocorrências, tornou-se destacável o desenvolvimento de estruturas específicas para o seu tratamento (SUI et al., 2009).

Uma das estruturas para este tipo de tratamento é a Rutgers, que consiste em uma estrutura robótica paralela, atuada por cilindros pneumáticos, com 6 graus de liberdade para o sistema, que é controlado pelo monitoramento de suas variáveis cinemáticas. O sistema também possui uma interface gráfica virtual utilizada para gerar jogos terapêuticos que interagem com o paciente durante as sessões de tratamento (BOIAN et al., 2004). A estrutura é ilustrada na Figura 2.8.



Figura 2.8 – Foto de paciente utilizando a estrutura de Rutgers, enquanto pratica um jogo terapêutico gerado pela mesma (BOIAN et al., 2004).

As estruturas com base programável apresentam uma maior simplicidade para controle dos movimentos gerados nos pés dos pacientes, uma vez que a ação de movimentação é realizada diretamente nos pés. Elas são limitadas a atuarem apenas sobre um ponto de contato com o membro inferior, o que faz com que o movimento sobre as articulações seja indireto. Os mecanismos propostos nesta dissertação irão atuar diretamente sobre cada articulação, o que irá garantir que a saída angular projetada seja trabalhada durante os exercícios.

2.1.3 *Estruturas robóticas com base móvel*

Estruturas com bases móveis consistem em sistemas robóticos capazes de permitir exercícios onde os usuários podem se deslocar em um determinado espaço. Para tanto, a estrutura fornece ao paciente um suporte para seu peso facilitando a movimentação. Estes sistemas

possuem a vantagem de permitir ao paciente se movimentar de forma livre, sem precisar seguir um padrão pré-programado.

Um exemplo para este tipo de estrutura é a ReWalk, proposta pela ARGO (GOFFER, 2006) e aprovada para uso comercial. Ela consiste em um exoesqueleto que possui uma estrutura mecânica associada a cada perna do usuário, os quais integram motores DC, sensores de movimento e baterias recarregáveis. Basicamente, o sistema monitora os movimentos superiores do corpo e aciona os motores DC de forma a manter a marcha de acordo com o desejado pelo usuário. O sistema é apresentado na Figura 2.9.

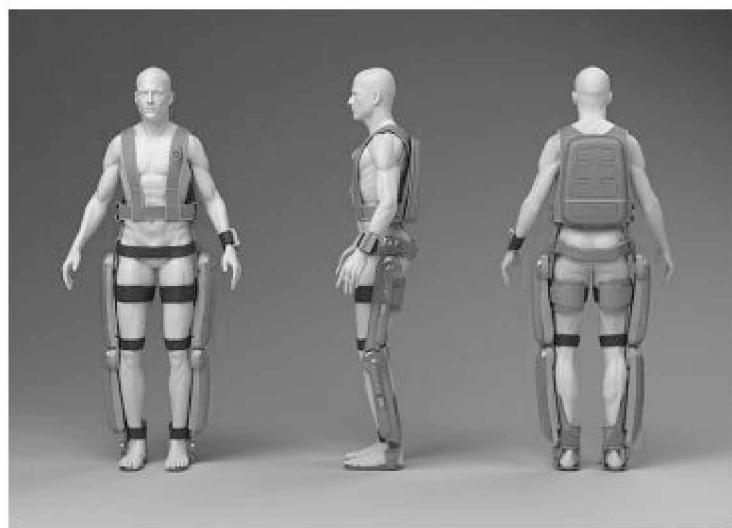


Figura 2.9 – Modelo ilustrando o ReWalk (GOFFER, 2006).

Outro exemplo para este tipo é a WalkTrainer, proposta por Stauffer et al. (2009), e distribuída pela Swortec S/A, que consiste em um exoesqueleto híbrido, combinando movimentos do quadril, joelho e tornozelo, utilizando um sistema de 6 graus de liberdade (GDL). Seu controle é realizado por um sistema denominado *Functional Electrical Stimulation* (FES), baseado nas medidas dos esforços entre o paciente e uma rede Proporcional Integral Derivativa (PID) em malha fechada (STAUFFER et al, 2009). Os primeiros testes clínicos desta estrutura foram conduzidos, demonstrando sua segurança e eficácia (HESSE;UHLENBROCK, 2000). Uma figura do protótipo desta estrutura é apresentada na Fig. 2.10.

Outro exemplo deste tipo de estrutura é o projeto proposto por Soares Júnior (2015), o qual descreve um exoesqueleto não-motorizado projetado para auxiliar a movimentação e prover exercícios para reabilitação da marcha humana.

O princípio de funcionamento se dá pela combinação de mecanismos manivela-balancim e came-seguidor para gerar as trajetórias de cada elemento do membro inferior de forma similar à marcha humana. O paciente opera este sistema girando uma manivela na altura de seus braços, acionando o exoesqueleto, o que faz com que seus membros inferiores acompanhem passivamente o movimento gerado pelo aparato, Fig. 2.11 (SOARES JÚNIOR, 2015).

Para este tipo de estruturas observa-se uma grande contribuição para a promoção da independência funcional dos pacientes, uma vez que elas possibilitam sua movimentação com



Figura 2.10 – Foto de um paciente utilizando a WalkTrainer (STAUFFER et al., 2009).



Figura 2.11 – Estrutura não-motorizada para mobilidade de pacientes com deficiência (SOARES JÚNIOR, 2015).

maior liberdade. Para uso clínico, estas estruturas podem requerer um amplo espaço de trabalho para que o paciente se exercente. Uma vez que a proposta dos mecanismos nesta dissertação é desenvolver exercícios elementares para cada articulação, não há uma demanda grande por espaço de trabalho como a aqui observada.

2.1.4 Estruturas robóticas com base estacionária

Define-se este tipo de estrutura robótica como sistemas construídos com uma base fixa, sem suporte ao peso do paciente, onde os movimentos são realizados por sistemas mecânicos conectados diretamente aos membros inferiores do usuário.

Normalmente, estes sistemas usam um tipo de poltrona ou maca adaptada como base estacionária. Assim, os mecanismos trabalham de forma similar aos apresentados na seção 2.1.2, mantendo a perna do paciente livre no ar.

A estrutura proposta por Akdogan e Adli (2011), denominada *Physiotherabot*, consiste em um exemplo clássico para este tipo de equipamento. Ela é composta por uma maca adaptada, podendo manter o paciente sentado ou deitado, e possui três atuadores dispostos conforme apresentado na Fig. 2.12.

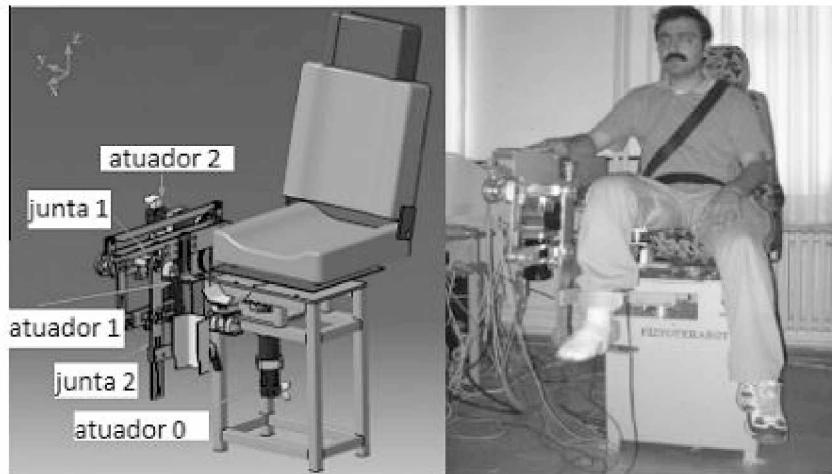


Figura 2.12 – Modelo do Physiotherabot, e foto de um paciente utilizando um protótipo do mesmo (AKDOGAN; ADLI, 2011).

O robô comercializado pela Swortec S/A, denominado MotionMaker[®] apresenta uma estrutura similar. Ele permite ao paciente que realize exercícios de musculação com membros paralisados, o que auxilia a evitar a atrofia da musculatura. Uma das principais vantagens deste equipamento é o uso da técnica FES (*Functional Electrical Stimulation*) sobre os músculos do paciente para se ajustar aos esforços gerados e aumentar a eficiência dos treinos (SCHMITT et al., 2004). Na Fig. 2.13 é apresentado uma imagem do MotionMakerTM.

Os primeiros testes clínicos dessa estrutura apontam melhorias na capacidade dos pacientes de realizar o exercício de *leg press*, onde o paciente empurra horizontalmente uma carga com as pernas (SCHMITT et al., 2004), (BOURI et al., 2006).



Figura 2.13 – Imagem de operação da estrutura MotionMaker™ (adaptada de: <http://www.swortec.ch/index.php/products/motionmaker>).

Outro exemplo recente foi proposto por Farjatian et al. (2014), denominada vi-RABT. A estrutura é composta por um sistema com 2 graus de liberdade, projetado para ser portátil, leve e de fácil manuseio para paciente e terapeuta.

O sistema é capaz de realizar os movimentos de flexão-extensão e eversão/inversão. Os eixos de rotação da estrutura foram alinhados de forma a coincidir com os eixos dos movimentos acima citados. O protótipo da estrutura é apresentado na Fig. 2.14.

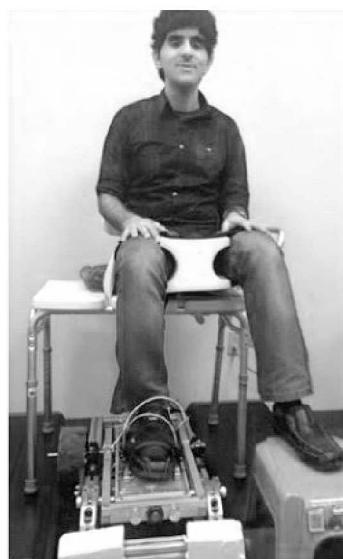


Figura 2.14 – Foto de um paciente utilizando a estrutura vi-RABT (FARJATIAN et al., 2014).

É possível notar que este tipo de estrutura se destaca por realizar os movimentos diretamente na perna do paciente sem a necessidade do uso de uma suspensão para suportar o seu peso. A proposta dos mecanismos desta dissertação se encaixa neste tipo de estrutura, o qual apresenta-se ideal para a execução de movimentos elementares de cada articulação.

2.2 Movimentos do membro inferior humano

Nesta seção são apresentados os movimentos básicos de cada articulação do membro inferior, abordando as características mecânicas e as limitações do mesmo, a fim de definir o espaço de trabalho das estruturas a serem projetadas.

2.2.1 O quadril

Conforme apresentado por Kapandji (2000), o quadril é a junta do membro inferior responsável pela sustentação do indivíduo e por sua locomoção. Possui 3 graus de liberdade, permitindo a movimentação da perna em três direções distintas, o que possibilita comparar esta conexão do corpo a uma articulação mecânica esférica.

Para referenciar adequadamente os movimentos aqui abordados, na Fig. 2.15 são apresentados os planos e eixos de referência do membro inferior utilizados. Admite-se como sendo o eixo transverso o eixo horizontal contido no plano frontal do corpo humano, eixo ântero-posterior o perpendicular a este mesmo plano, e o eixo vertical, como propriamente dito, contido no plano frontal. Admite-se que o ponto O representado na Figura 2.15 está localizado no centro da articulação e na interseção dos eixos transverso e ântero-posterior.

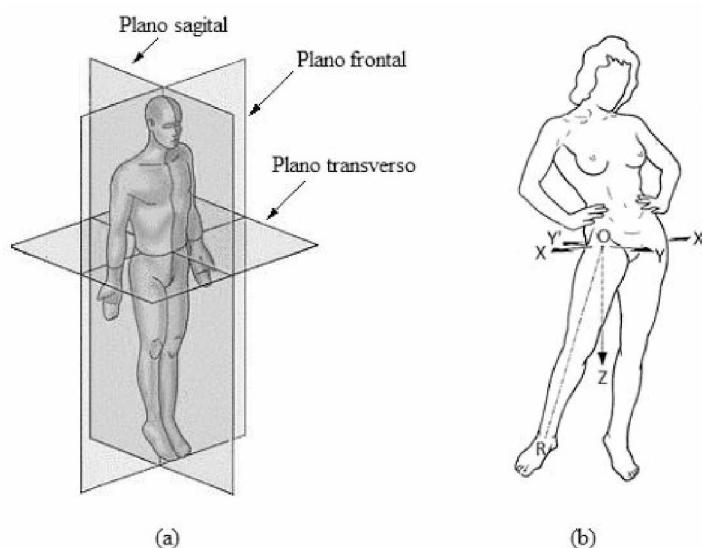


Figura 2.15 – (a) Planos de referência do corpo humano (DANGELO; FATTINI,2007); e (b) (XX') eixo transverso, (YY') eixo ântero-posterior, (OZ) eixo vertical, (OR) eixo longitudinal do quadril (KAPANDJI, 2000).

Flexão e extensão

O movimento de flexão-extensão é definido como o deslocamento da perna em torno do eixo transverso, fazendo com que o movimento ocorra no plano sagital, Fig. 2.16.

Durante a flexão, a perna se desloca para frente, possibilitando o contato entre coxa e tronco e promovendo a distensão dos ligamentos. A posição do joelho afeta a amplitude máxima

deste movimento, de forma que esta é inferior a 90° em relação à posição anatômica com o joelho estendido, e atinge até 120° com o joelho flexionado. Os limites também são influenciados pela forma o qual o movimento é executado.

A amplitude da flexão sofre ainda variações devido à ação de esforços externos à musculatura da perna. Os valores citados anteriormente correspondem a flexão ativa, realizada somente pela ação da perna. Quando a perna sofre a ação de uma força externa, a flexão é denominada passiva e sua amplitude máxima pode chegar a 145° com o joelho flexionado e 120° com o mesmo estendido, Fig. 2.16(a).

Na extensão o quadril realiza o movimento contrário à flexão, deslocando o membro para trás, gerando tensão nos ligamentos, numa amplitude de 20° ativos com o joelho estendido e aproximadamente 10° para o joelho flexionado. De forma passiva, a extensão pode alcançar aproximadamente 30° quando a perna se situa para trás e 20° com a mesma à frente, Fig. 2.16(b).

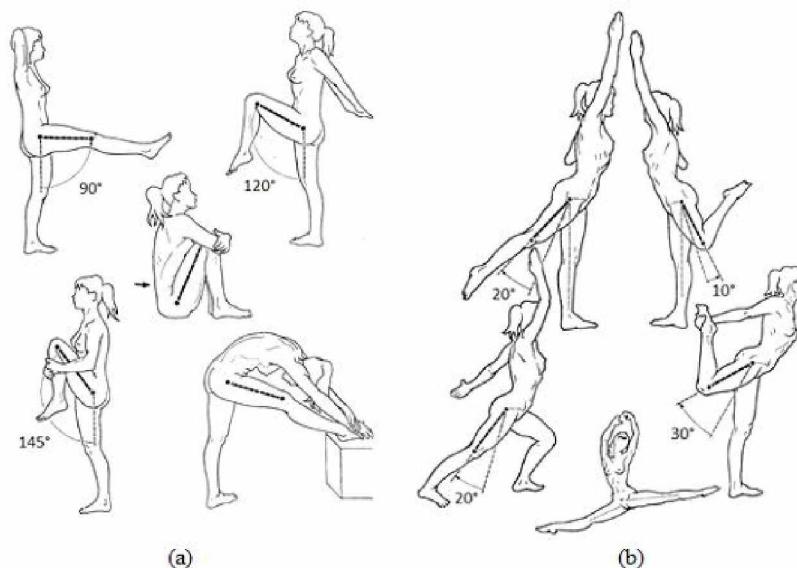


Figura 2.16 – (a) Movimentos de flexão do quadril; (b) movimentos de extensão do quadril (KAPANDJI, 2000).

Abdução e adução

Por definição, o movimento de abdução é o afastamento do membro do plano sagital. Este movimento ocorre no plano frontal em torno do eixo ântero-posterior e pode alcançar cerca de 90° entre uma perna e outra na abdução ativa, sendo que é comum se dizer que a amplitude máxima do quadril é de 30 a 45° em relação ao eixo de simetria do corpo, contido no plano sagital. Contudo, esta amplitude pode ser prolongada de forma passiva e alcançar amplitudes maiores, conforme ilustrado na Fig. 2.17(a).

A adução corresponde ao movimento inverso da abdução, onde o membro se desloca em direção ao eixo de simetria do corpo. Pelo fato dos membros inferiores estarem em contato na posição de referência, não existe um movimento de adução pura, mas sim movimentos de adução relativa, sendo a amplitude máxima de 30° , Fig. 2.17(b).

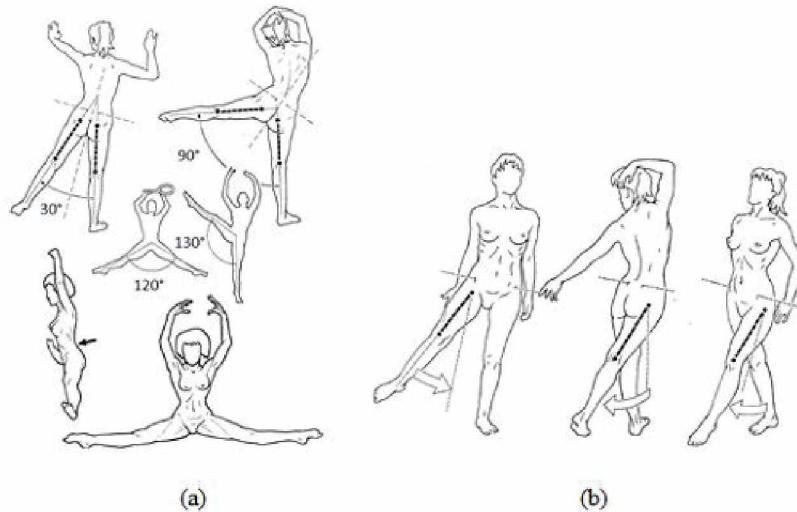


Figura 2.17 – a) Movimentos de abdução do quadril; (b) exemplos de movimentos de adução relativa (KAPANDJI, 2000).

Rotação do eixo longitudinal

Este movimento ocorre quando o joelho está em posição de flexão e em ângulo reto na vertical, Fig. 2.18(b). A partir desta posição, quando a perna se dirige para fora, mede-se a rotação interna, Fig. 2.18(a), cuja amplitude máxima é de 30 a 40°. Quando a perna se dirige para dentro, mede-se a rotação externa, Fig. 2.18(c), cuja amplitude máxima é de 60°.

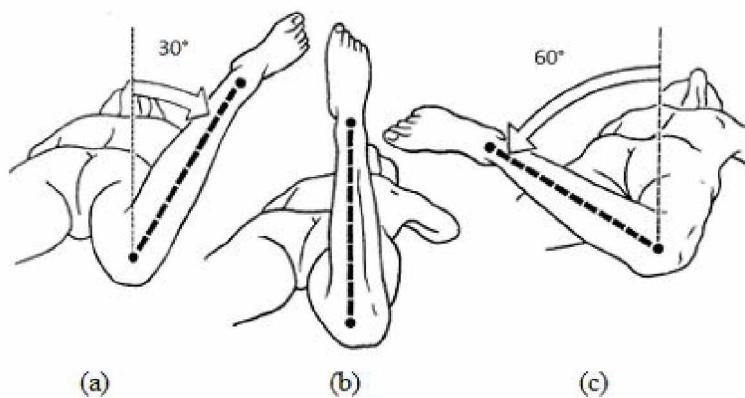


Figura 2.18 – (a) Rotação interna do Quadril; (b) posição de referência; (c) rotação externa do quadril (KAPANDJI, 2000).

2.2.2 *O Joelho*

O joelho é a articulação intermediária do membro inferior, sendo uma articulação com 1 grau de liberdade (flexão/extensão), que permite regular a distância do corpo em relação ao chão. A articulação do joelho, contudo, possui um segundo grau de liberdade (rotação sobre o eixo longitudinal da perna) o qual só existe quando o mesmo está flexionado.

Na Figura 2.19 estão indicados os eixos de referência utilizados para descrever os

movimentos que ocorrem no joelho.

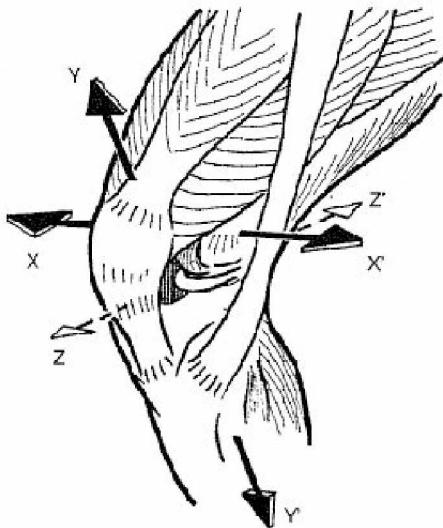


Figura 2.19 – (XX') Eixo transverso, (YY') eixo longitudinal do joelho, (ZZ') eixo de referência (KAPANDJI, 2000).

Flexão e extensão

O movimento de flexão-extensão é o principal movimento do joelho, sendo que sua amplitude é medida a partir da posição de referência definida a partir do eixo da perna que se situa no prolongamento do eixo da coxa, indicado como eixo ZZ', Fig. 2.19.

A extensão do joelho é definida como o afastamento da face posterior da perna da face posterior da coxa. Este movimento ocorre principalmente de forma passiva, variando de 5 a 10° a partir da posição de referência, Fig. 2.20(c).

Para a flexão do joelho, distingue-se também as amplitudes ativas e passivas deste movimento. A flexão de forma ativa atinge 140° caso o quadril se encontre previamente flexionado, Fig. 2.20(d), e pode chegar a 120° somente se o quadril estiver em extensão, Fig. 2.20(f).

A flexão passiva pode chegar a uma amplitude de 160°, Fig. 2.20(e), o que torna o calcanhar capaz de tocar a nádega nesta situação.

Rotação do eixo longitudinal

O movimento de rotação longitudinal do joelho corresponde a um caso particular que ocorre apenas quando este se encontra flexionado, possibilitando o pé a se deslocar em torno do eixo longitudinal do joelho.

A rotação interna leva a ponta do pé para dentro, intervindo no movimento de adução do pé, sendo a amplitude do movimento de 30°, Fig. 2.21(a). Já a rotação externa leva a ponta do pé para fora, com amplitude de movimento de 40°, Fig. 2.21(c). Quando realizados de forma passiva, estes movimentos podem aumentar suas amplitudes máximas de 5 a 10°, dependendo das características de cada indivíduo.

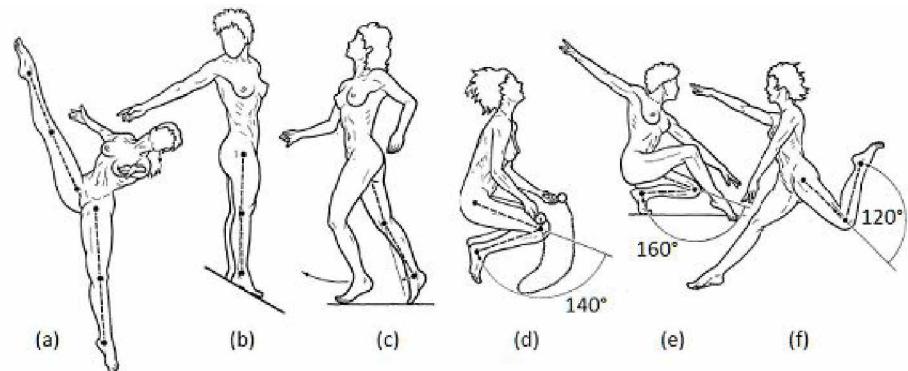


Figura 2.20 – Movimentos de flexão-extensão do joelho (KAPANDJI, 2000).

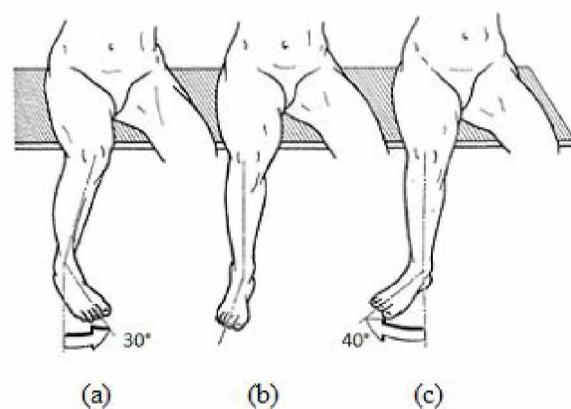


Figura 2.21 – Movimentos de rotação longitudinal do joelho (KAPANDJI, 2000).

2.2.3 O Tornozelo

O tornozelo é definido como a articulação distal do membro inferior, que condiciona os movimentos da perna com relação ao pé no plano sagital, e possui apenas um grau de liberdade, onde ocorre à flexão-extensão, sendo de vital importância para a marcha, seja em terreno plano ou acidentado (KAPANDJI, 2000).

O conjunto de articulações existentes no pé, auxiliado pela rotação axial do joelho, possui as mesmas funções de uma articulação com 3 graus de liberdade, permitindo orientar a abóbada plantar em todas as direções, fazendo com que ela se adapte aos diversos tipos de terreno.

Na Figura 2.22 são apresentados os eixos utilizados para referenciar os movimentos ocorridos no tornozelo.

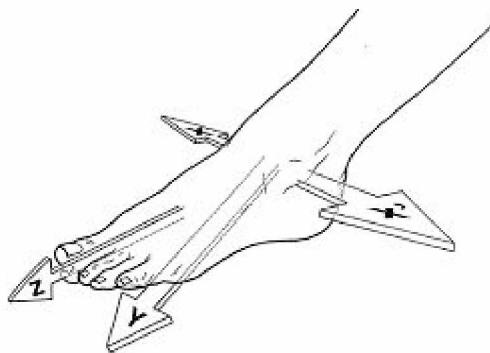


Figura 2.22 – (XX') eixo transversal, (Y) eixo longitudinal da perna (Z) eixo longitudinal do pé (KAPANDJI, 2000).

Flexão e extensão

Os movimentos de flexão-extensão do tornozelo ocorrem no plano sagital de maneira semelhante à flexão-extensão analisada nas articulações anteriores. Estes movimentos partem da posição anatômica em direções opostas.

A partir da posição de referência, Fig.2.23(a), posição A, pode-se afirmar que a flexão do tornozelo, também denominada de flexão dorsal ou dorsiflexão, é o movimento que aproxima o dorso do pé à face anterior da perna, Fig. 2.23(a), posição B.

A extensão da articulação do tornozelo (também denominada de flexão plantar) é o movimento que afasta o dorso do pé da face anterior da perna, Fig.2.23(a), posição C.

2.3 Mecanismos articulados

Originalmente os mecanismos foram definidos como dispositivos capazes de transformar um determinado tipo de movimento em outro, enquanto máquinas são o conjunto de dispositivos organizados de forma a transmitir força motora e realizar trabalho útil (HARTENBERG; DENA-

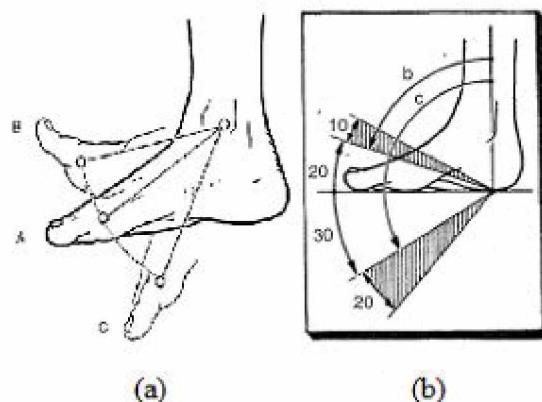


Figura 2.23 – (a) A: posição anatômica a 0° , B: Flexão do tornozelo, C: extensão do tornozelo; (b) Amplitudes da flexão-extensão do tornozelo (KAPANDJI, 2000).

VIT, 1964). Atualmente, esta definição pode ser expandida para dispositivos que são capazes de transmitir movimentos e/ou forças/torques.

As estruturas de multicorpos articuladas são comumente formadas pela montagem sucessiva de cadeias cinemáticas mais simples. Ao analisar, então, um mecanismo complexo, é possível aplicar o Princípio de Formação de Mecanismos Planos de Assur. Este princípio consiste em identificar o elemento condutor do mecanismo, e separar os demais em grupos, denominados grupos de Assur. Estes grupos caracterizam-se por não influenciarem no grau de mobilidade do mecanismo (NORTON, 2009).

Dessa forma, Assur define que todo mecanismo é formado pelo acréscimo sucessivo de grupos de Assur ao elemento condutor e ao chassi, ou a outros grupos previamente contidos (NORTON, 2009). Na Figura 2.24 é apresentado um exemplo de análise de um mecanismo decompondo-o em grupos de Assur.

Ao analisar os movimentos descritos na revisão da cinesiologia, é possível notar que para reproduzir os movimentos de cada articulação, é necessário propor sistemas que tenham, de forma geral, uma saída angular oscilatória compatível com as amplitudes de cada movimento a ser exercitado.

Para obter as estruturas de reabilitação, objeto desta dissertação, serão utilizados mecanismos de 4 barras através de sua análise cinemática e estática. Mais especificamente, serão utilizados mecanismos planos de 4 barras associados com correias e polias dentadas, a qual será discutida na seção 2.5.

Para a definição das estruturas, visto que estas serão construídas a partir de mecanismos de 4 barras, será utilizada a Lei de Grashof, a qual define que, de acordo com as proporções dos elementos que compõe um mecanismo de quatro barras plano, poderá ter entrada contínua e saída oscilatória (manivela-balancim), Fig. 2.25(a); entrada e saída oscilatórias (duplo-balancim), Fig. 2.25(b); ou entrada e saída contínuas (dupla-manivela), Fig. 2.25(c) (SHIGLEY e UICKER, 1995).

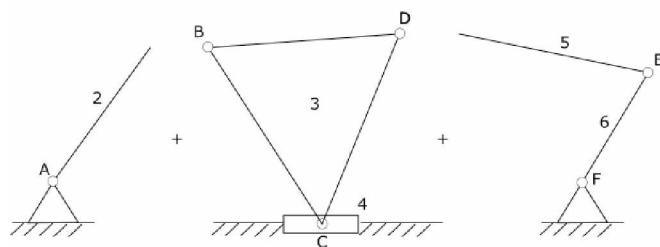
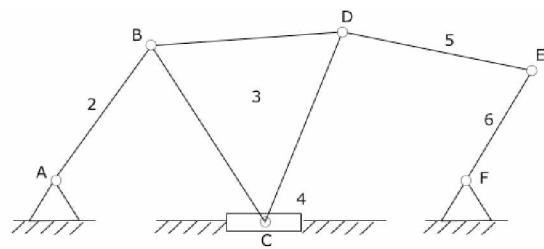


Figura 2.24 – Decomposição de um mecanismo em elemento condutor (2) e dois grupos de Assur (3 e 4, 5 e 6).

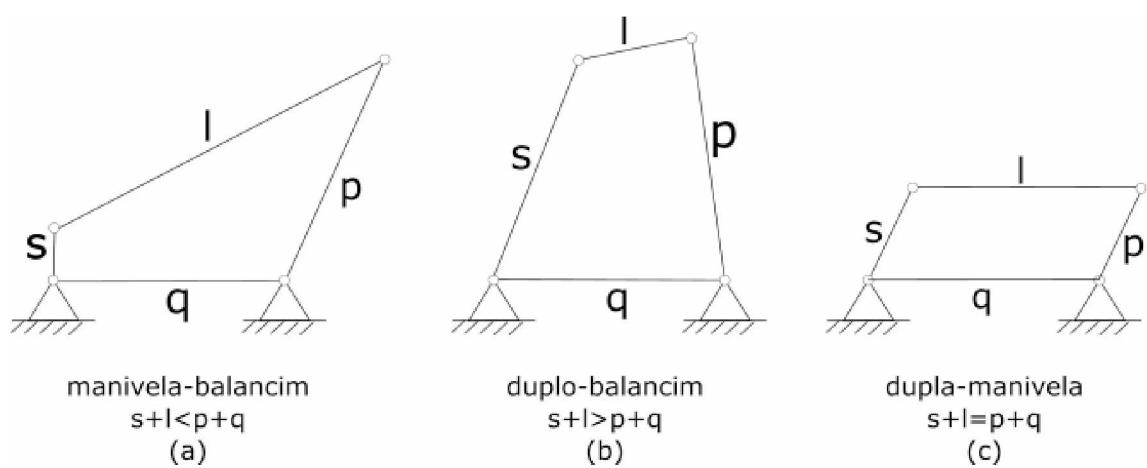


Figura 2.25 – Proporções para mecanismos de quatro barras segundo a Lei de Grashof.

2.3.1 Modelagem matemática para mecanismos planos de quatro barras

Para dimensionar adequadamente as peças que compõem os mecanismos desejados, é necessário aplicar um método analítico para modelagem dos sistemas a fim de encontrar as dimensões que possibilitam obter as saídas angulares desejadas para cada caso, bem como garantir que as tensões produzidas pelas reações em cada articulação sejam suportadas pelos componentes.

Para tanto, foi utilizado o método baseado no equilíbrio estático para mecanismos de quatro barras planos, composto por barras ideais, para desconsiderar o efeito de deformação, de massa desprezível e sem efeito de atritos e/ou folgas nas conexões entre os elementos, para simplificar os aspectos construtivos do sistema. Isso pode ser feito pois o movimento realizado tem baixa velocidade e aceleração devido à natureza dos exercícios de reabilitação. Dessa forma, estas podem ser desprezadas.

Os métodos apresentados adiante usam como referência a nomenclatura tradicional para análise cinemática de mecanismos de quatro barras, a qual é ilustrada na Fig. 2.26.

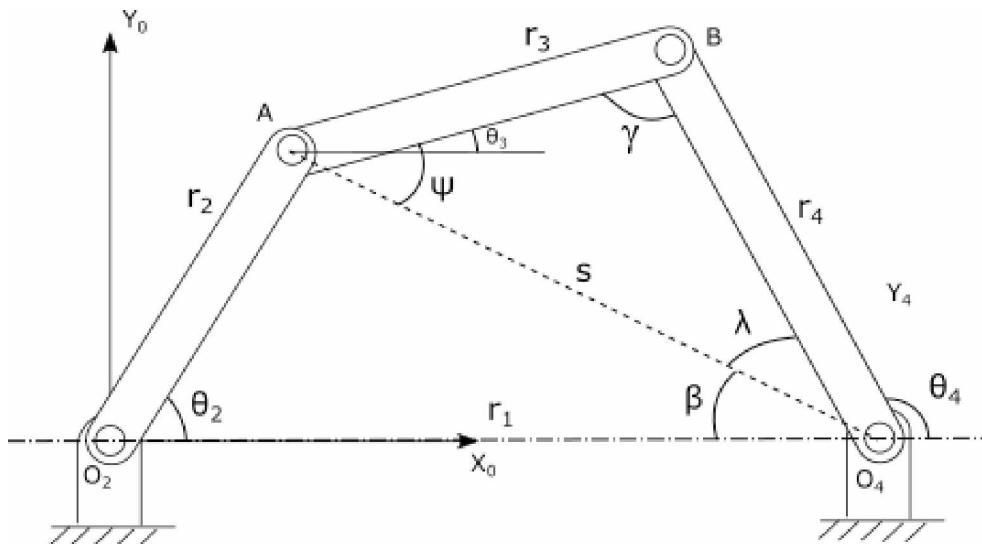


Figura 2.26 – Referência utilizada para os parâmetros de mecanismos de quatro barras.

Utiliza-se a ligação entre os pontos A e O_4 como recurso geométrico para auxiliar o cálculo dos parâmetros de interesse. A partir dos comprimentos das barras r_1 , r_2 , r_3 , r_4 , e do ângulo θ_2 , os valores de s e dos demais ângulos θ_3 , θ_4 , β , γ e ψ podem ser obtidos por meio das Eqs. (2.1) a (2.5) (UICKER Jr; PENNOCK; SHIGLEY, 2003).

$$s = (r_1^2 + r_2^2 - 2r_1r_2 \cos \theta_2)^{1/2} \quad (2.1)$$

$$\beta = \cos^{-1}[(r_1^2 - r_2^2 + s^2)/(2sr_1)] \quad (2.2)$$

$$\psi = \cos^{-1}[(r_3^2 - r_4^2 + s^2)/(2sr_3)] \quad (2.3)$$

$$\lambda = \cos^{-1}[(r_4^2 - r_3^2 + s^2)/(2sr_4)] \quad (2.4)$$

$$\gamma = \pm \cos^{-1}[(r_4^2 - s^2 + r_3^2)/(2r_3r_4)] \quad (2.5)$$

A partir destas equações, é possível obter os ângulos θ_3 e θ_4 , o qual corresponde à saída angular do sistema. Nas Equações (2.6) a (2.9) são apresentados os procedimentos para o cálculo destes ângulos.

$$\theta_3 = \phi - \beta \quad 0 \leq \theta_2 < 180^\circ \quad (2.6)$$

$$\theta_3 = \phi + \beta \quad 180^\circ \leq \theta_2 < 360^\circ \quad (2.7)$$

$$\theta_4 = 180^\circ - \lambda - \beta \quad 0 \leq \theta_2 < 180^\circ \quad (2.8)$$

$$\theta_4 = 180^\circ - \lambda + \beta \quad 180^\circ \leq \theta_2 < 360^\circ \quad (2.9)$$

Para a modelagem estática foi utilizado o referencial inercial $O_2X_0Y_0$, representado na Fig. 2.27(a) sendo que a somatória do momento foi calculada em relação as articulações O_2 , A e O_4 para o equilíbrio das barras r_2, r_3 e r_4 , respectivamente.

Para cada força \mathbf{F}_{ij} , o termo i indica o índice da barra que realiza a força, e j a barra que recebe o esforço.

Através das equações de equilíbrio estático de cada elemento, é possível obter a matriz de equilíbrio do sistema. As equações são obtidas através da aplicação da Segunda Lei de Newton nas componentes x e y, Eqs. (2.10) e (2.11), e do equilíbrio de momentos, Eq. (2.12). Assim, tem-se três equações para cada elemento do mecanismo, resultando em uma matriz de equilíbrio 9x9, a qual é representada na Eq. (2.13).

A construção da matriz de equilíbrio $[A]$ pode ser obtida a partir das equações (2.10) a (2.12). A força externa F_E deve-se apenas ao peso do membro a ser movimentado.

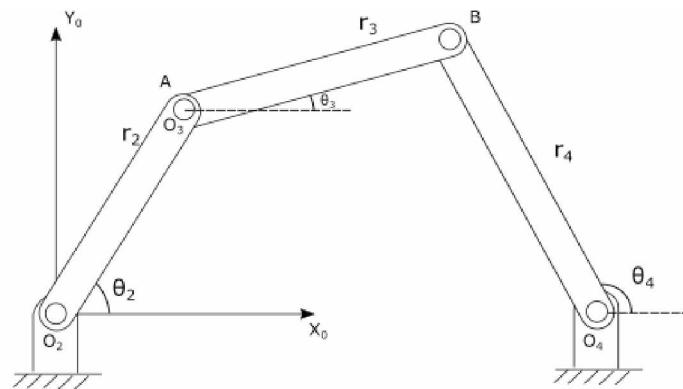
$$\sum F_x = 0 \quad (2.10)$$

$$\sum F_y = 0 \quad (2.11)$$

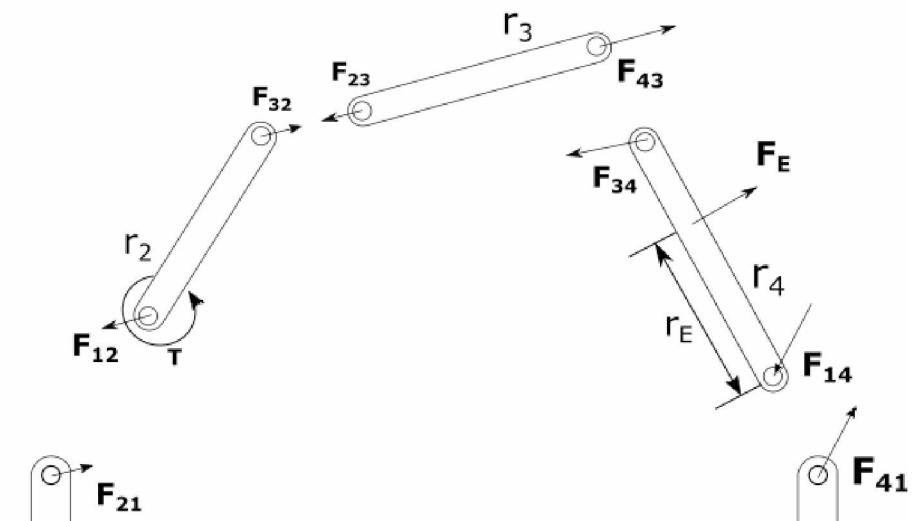
$$\sum M_z = 0 \quad (2.12)$$

Aplicando a forma matricial para equações 2.10 a 2.12, tem-se:

$$[A]\{F\} = \{B\}$$



(a)



(b)

Figura 2.27 – (a) Referenciais adotados para os elementos do mecanismo de quatro barras e;
 (b) Diagrama de corpo livre das barras.

$$\begin{bmatrix}
-1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -r_2 \sin \theta_2 & r_2 \cos \theta_2 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -r_3 \sin \theta_3 & r_3 \cos \theta_3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\
0 & 0 & 0 & 0 & -r_4 \sin \theta_4 & r_4 \cos \theta_4 & 0 & 0 & 0
\end{bmatrix}
\begin{pmatrix}
F_{12_x} \\
F_{12_y} \\
F_{32_x} \\
F_{32_y} \\
F_{43_x} \\
F_{43_y} \\
F_{14_x} \\
F_{14_y} \\
T
\end{pmatrix} = \\
= \begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
-F_{E_x} \\
-F_{E_y} \\
r_E \sin \theta_4 F_{E_x} - r_E \cos \theta_4 F_{E_x}
\end{pmatrix} \quad (2.13)$$

As componentes das forças \mathbf{F}_{ij} na Equação (2.13) serão utilizadas para determinar os esforços e as tensões nas barras.

As relações deste modelo também serão usadas para realizar uma busca otimizada para os melhores comprimentos de barra utilizando algoritmos genéticos, conforme apresentado na seção 2.6.

Se for de interesse considerar os efeitos dinâmicos do sistema, pode-se consultar o trabalho de Rezende et al. (2010).

2.4 Análise de singularidade em mecanismos

Para garantir o correto funcionamento de um mecanismo é de suma importância a realização de uma análise de mobilidade e de possíveis posições de singularidade. A metodologia utilizada é apresentada adiante, a qual é baseada na modelagem cinemática do mecanismo.

Altuzarra et al. (2004) apresenta a definição de singularidade como uma impossibilidade de solucionar o problema cinemático matematicamente em determinados pontos de um mecanismo, o que pode ser compreendido fisicamente como pontos que, de forma local ou global, alteram a mobilidade de um mecanismo levando-o à falta de controle de uma ou mais articulações durante seu ciclo de movimentação.

Para determinar as possíveis posições de singularidade em mecanismos de cadeia cinemática fechada será utilizada a metologia baseada nas relações da matriz geométrica do sistema e do problema de autovalores gerado à partir delas (ALTUZARRA et al., 2004,2006), (HERNAN-

DEZ et al., 2003).

2.4.1 Equação das velocidades nodais

Para aplicar a metodologia proposta é necessário inicialmente obter a matriz geométrica do sistema. Os procedimentos matemáticos para obtenção desta matriz estão descritos a seguir.

A modelagem aqui apresentada está direcionada a mecanismos planos com articulações rotacionais, objeto de estudo desta dissertação. Assim, considerando uma barra de comprimento r_i , inclinada de um ângulo θ_j em relação ao eixo X, e que seus extremos sejam denominados "nós", com índices i e j , conforme apresentado na Fig.2.28.

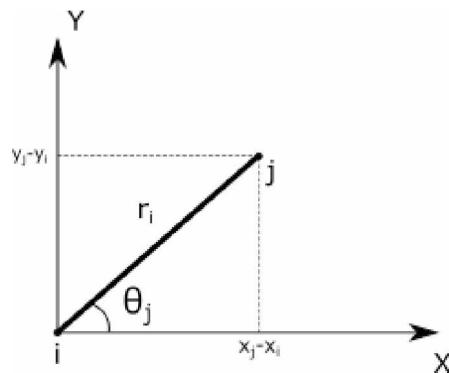


Figura 2.28 – Relações geométricas de uma barra elementar de comprimento r_i .

Da Figura 2.28 pode-se escrever:

$$r_i \cos \theta_j = x_j - x_i \quad (2.14)$$

$$r_i \sin \theta_j = y_j - y_i \quad (2.15)$$

Multiplicando a Equação (2.14) por $\cos \theta_j$ e a Eq. (2.15) por $\sin \theta_j$, e somando os resultados obtém-se:

$$r_i(\cos^2 \theta_j + \sin^2 \theta_j) = (x_j - x_i) \cos \theta_j + (y_j - y_i) \sin \theta_j$$

Ou ainda,

$$r_i = (x_j - x_i) \cos \theta_j + (y_j - y_i) \sin \theta_j \quad (2.16)$$

Reescrevendo a Equação 2.16 na forma matricial, tem-se:

$$r_i = \{x_i \ y_i \ x_j \ y_j\} \begin{pmatrix} -\cos \theta_j \\ -\sin \theta_j \\ \cos \theta_j \\ \sin \theta_j \end{pmatrix} = \{x_e\}^T \{h\} \quad (2.17)$$

Na Equação (2.17) o índice "e" representa o vetor com as coordenadas dos nós de um elemento (coordenadas dos nós i, j que definem a barra). O quadrado do comprimento da barra pode ser obtido tanto pela Eq. (2.16) como pela forma matricial usando a Eq. (2.17), ou seja:

$$r^2 = (x_j - x_i)^2 + (y_j - y_i)^2 = \{x_e\}^T \{h\} \{h\}^T \{x_e\}$$

Ou ainda,

$$r_i^2 = \{x_i \ y_i \ x_j \ y_j\} \begin{bmatrix} \cos^2 \theta_j & \cos \theta_j \sin \theta_j & -\cos^2 \theta_j & -\cos \theta_j \sin \theta_j \\ \cos \theta_j \sin \theta_j & \sin^2 \theta_j & -\cos \theta_j \sin \theta_j & -\sin^2 \theta_j \\ -\cos^2 \theta_j & -\cos \theta_j \sin \theta_j & \cos^2 \theta_j & \cos \theta_j \sin \theta_j \\ -\cos \theta_j \sin \theta_j & -\sin^2 \theta_j & \cos \theta_j \sin \theta_j & \sin^2 \theta_j \end{bmatrix} \begin{pmatrix} x_i \\ y_i \\ x_j \\ y_j \end{pmatrix} \quad (2.18)$$

Onde a forma matricial, Eq. (2.18), pode ser compactada em:

$$r_i^2 = \{x_e\}^T [G_e] \{x_e\} \quad (2.19)$$

Onde a matriz $[G_e]$ é a matriz geométrica elementar que é escrita em função do vetor de orientação da barra $\{h\}$, ou seja:

$$[G_e] = \{h\} \{h\}^T \quad (2.20)$$

Diferenciando a Eq. (2.17) em relação ao tempo, é possível observar as relações de velocidade das extremidades da barra, denominado de velocidades nodais $\{\dot{x}_e\}$, através da Eq. (2.21).

$$0 = \{\dot{x}_e\}^T \{h\} + \{x_e\}^T \{\dot{h}\} \quad (2.21)$$

Utilizando as propriedades dos cossenos diretores, pode-se demonstrar que o segundo

termo da Eq. (2.21) é nulo (ALTUZARRA et al., 2004,2006), (HERNANDEZ et al., 2003), ou seja:

$$\{x_e\}^T \{\dot{h}\} = 0 \quad (2.22)$$

Assim, a Equação (2.21) pode ser reescrita como:

$$0 = \{\dot{x}_e\}^T \{h\} \quad (2.23)$$

Sendo assim, pós-multiplicando a Equação (2.23) por $\{h\}^T$, obtém-se uma relação do vetor velocidade nodal $\{\dot{x}_e\}$ com a matriz geométrica elementar $[G_e]$, conforme apresentado na Eq. (2.24).

$$\begin{aligned} \{0\}^T &= \{\dot{x}_e\}^T \{h\} \{h\}^T = \{\dot{x}_e\}^T [G_e]^T \\ [G_e] \{\dot{x}_e\} &= \{0\} \end{aligned} \quad (2.24)$$

Para detalhamento do procedimento de obtenção da matriz geométrica para sistemas tridimensionais, o trabalho de Hernández et al. (2003) pode ser consultado.

A Equação (2.24) consiste em um sistema linear de equações relacionadas com as velocidades nodais do elemento considerado. Ela deve ser escrita para cada elemento do mecanismo.

Sendo o mecanismo composto por n nós, a quantidade de coordenadas nodais para o mecanismo plano é $2n$. Então, pode-se escrever o vetor $\{x\}$ agrupando todas as coordenadas nodais da seguinte forma:

$$\{x\} = \{x_1 \ y_1 \ x_2 \ y_2 \ \dots \ x_n \ y_n\}^T$$

Então, a matriz geométrica elementar $[G_e]$ pode ser expandida, para cada elemento, obtendo uma matriz quadrada de ordem $2n$, denominada matriz elementar expandida $[\bar{G}_e]$, de tal forma que:

$$\{r^2\} = \{x\}^T [\bar{G}_e] \{x\} \quad (2.25)$$

E pela Eq. (2.24) também pode-se escrever:

$$[\bar{G}_e] \{\dot{x}\} = 0 \quad (2.26)$$

Sendo o mecanismo formado por n elementos, a Eq. (2.24) deve ser escrita para cada um, ou seja:

$$[\bar{G}_{e_1}]\{\dot{x}\} = \{0\}$$

$$[\bar{G}_{e_2}]\{\dot{x}\} = \{0\}$$

:

$$[\bar{G}_{e_n}]\{\dot{x}\} = \{0\}$$

Que podem ser somadas para obter a matriz geométrica global $[G]$ do mecanismo, de tal forma que

$$[G]\{\dot{x}\} = \{0\} \quad (2.27)$$

$$[G] = \sum_{e=1}^M [\bar{G}_e] \quad (2.28)$$

Finalmente, as condições de restrições nas articulações fixas e duplicação de nós devem ser consideradas para simplificar a matriz geométrica global, pois as articulações fixas (nós fixos) possuem velocidade nula.

2.4.2 Análise das singularidades

A variação da quantidade de graus de liberdade (GDL) de um mecanismo, ao longo de sua movimentação está relacionada com a existência de posições singulares.

Então, a análise das configurações singulares pode ser feita a partir da equação das velocidades nodais escrita em termos da matriz geométrica do mecanismo, Eq. (2.27) (ALTUZARRA et al., 2004,2006), (HERNANDEZ et al., 2003). Isso equivale a resolver o problema de autovalores e autovetores, conforme a Eq. (2.29), permitindo obter os autovetores $\{v_i\}$ associados aos autovalores λ .

$$[G]\{\dot{x}\} = \lambda\{\dot{x}\} \quad (2.29)$$

Pode-se observar que a Equação (2.27) é um caso particular da Eq. (2.29) quando o valor de λ é nulo. Além disso, ela também é um caso particular da matriz Jacobiana do mecanismo.

O número de GDLs em um ciclo completo de um mecanismo é denominado por f e pode ser obtido através da análise de mobilidade do mecanismo. O procedimento para determinar as singularidades a partir dela são descritos por Altuzarra et al. (2004,2006) e Hernandez et

al.(2003).

Contagem dos autovalores nulos

Os l autovetores $\{v_i\}$ que correspondem aos autovalores λ nulos compõem vetores linearmente independentes que descrevem os possíveis movimentos do mecanismo. Dessa forma, estes l autovalores correspondem aos f GDLs do mecanismo, e se em algum ponto do ciclo, a quantidade de autovalores nulos l incrementar, tem-se uma configuração de GDL instantâneo incrementado.

$$\{\dot{x}\}_{2n \times 1} = \sum_{i=1}^l \tau_i \{v_i\}_{2n \times 1} = [V]_{2n \times l} * \{\tau\}_{l \times 1} \quad (2.30)$$

Onde $2n$ corresponde ao número de coordenadas nodais do mecanismo, $\{\dot{x}\}$ é o vetor de velocidades nodais global, τ_i são constantes para a ponderação dos autovetores $\{v_i\}$. $[V]$ e $\{\tau\}$ correspondem ao agrupamento dos autovetores $\{v_i\}$ e as constantes τ_i , respectivamente (ALTUZARRA et al., 2004,2006), (HERNANDEZ et al., 2003).

Portanto, se em qualquer ponto do mecanismo o valor de l incrementar, ocorrerá uma singularidade por mobilidade incrementada, onde a estrutura possui um grau de liberdade adicional não-controlado em uma posição específica. Ao analisar a proximidade do primeiro autovalor não-nulo, sua proximidade de zero indicará a aproximação do mecanismo à uma posição de singularidade.

Dependência das coordenadas de entrada

Outro fator que pode gerar singularidades é a ocorrência de dependências nas entradas do sistema. Este problema pode ser analisado separando as f coordenadas generalizadas $\{\dot{x}_{Ei}\}$ definidas como entradas do mecanismo, gerando a matriz $[V_E]$ a partir das coordenadas correspondentes nos autovetores $\{V\}$, conforme apresentado na Eq. (2.31).

$$\{\dot{x}_E\}_{f \times 1} = [V_E]_{f \times f} * \{\tau_E\}_{f \times 1} \quad (2.31)$$

O posto de $[V_E]$ deve ser igual ao número de coordenadas de entrada. Caso contrário, ocorrerá uma singularidade por dependência de entrada, o que poderá gerar falta de controle, instabilidade ou travamento do sistema.

Dependência das coordenadas de saídas

De forma similar ao procedimento apresentado anteriormente, analisa-se as saídas do mecanismo separando as f coordenadas $\{\dot{x}_{Si}\}$ definidas como saídas do mecanismo, gerando a matriz $[V_S]$ analogamente às entradas, de acordo com a Eq. (2.32).

$$\{\ddot{x}_S\}_{f \times 1} = [V_S]_{f \times f} * \{\tau_S\}_{f \times 1} \quad (2.32)$$

Novamente, verifica-se o posto de $[V_S]$ o qual deve ser igual ao número de coordenadas de saída para que não ocorra a singularidade por dependência de saídas, que possui características similares às dependências de entrada.

Para melhor visualização dos procedimentos descritos nesta subseção, foi elaborado um fluxograma da metologia de Altuzarra et al (2004), o qual é ilustrado na Fig. 2.29.

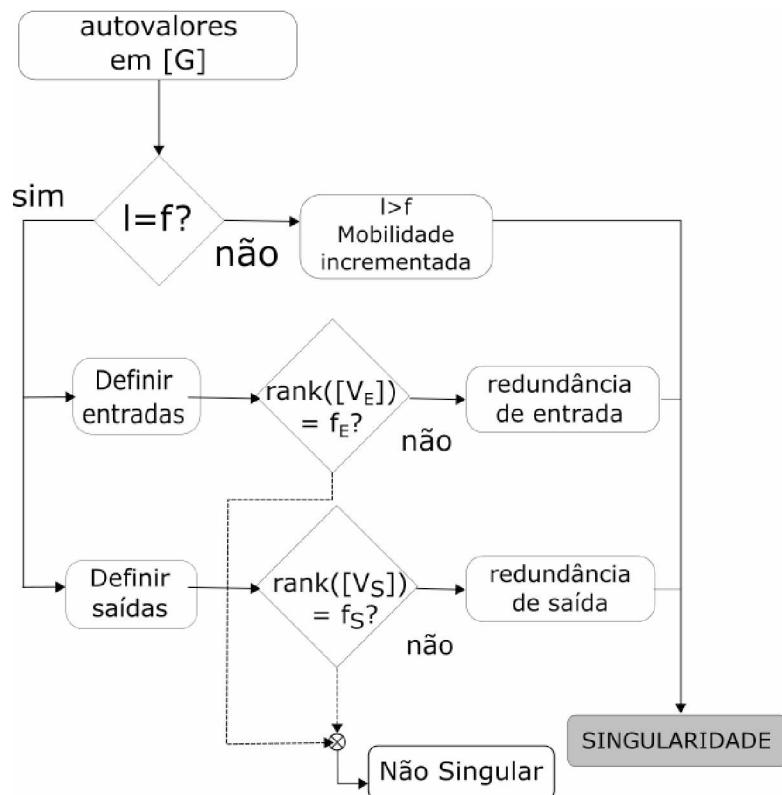


Figura 2.29 – Fluxograma simplificado para aplicação da metologia de Altuzarra et al. (2004) para singularidades.

Esta metologia foi aplicada aos mecanismos desenvolvidos e os resultados estão apresentados no Capítulo III.

2.5 Transmissão por correia dentada e correntes

Para que o movimento angular seja transmitido até o elemento condutor dos mecanismos propostos neste trabalho, é necessário o uso de correias e polias dentadas ou correntes.

Outras alternativas, tais como o paralelogramo articulado, que constitui um mecanismo do tipo dupla-manivela segundo a Lei de Grashof, são inviáveis para o projeto devido à presença de singularidades.

Transmissões por meio de correias planas consistem em um meio de transmissão indireta flexível e constituem boa opção para transmissões de até 1600kW de potência, velocidades angulares até 1800RPM e tangenciais de até 90 m/s, e uma relação de transmissão ideal de 1:5 (MELCONIAN, 2009). Correias planas podem possuir dentes visando aumentar o sincronismo e a aderência. Um exemplo de polia e correia dentada é ilustrado na Fig. 2.30.

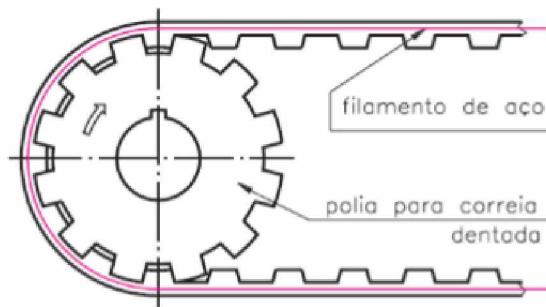


Figura 2.30 – Desenho de uma polia e uma correia dentada (adaptada de: blogdaengenharia.com/wp-content/uploads/2013/05/PoliaseCorreias.pdf).

Através deste tipo de sistema é possível transmitir o movimento angular entre as polias através das relações de velocidade e torque entre entrada e saída de forma que a velocidade periférica das polias seja a mesma (FRANCESCHI; ANTONELLO, 2014). Esta relação pode ser observada na Eq. (2.33).

$$\frac{\phi_2}{\phi_1} = \frac{\omega_2}{\omega_1} = i \quad (2.33)$$

Onde ϕ_1 e ϕ_2 representam os diâmetros primitivos das polias, e ω_1 e ω_2 as velocidades angulares de cada polia, indicando a relação de transmissão i para o acoplamento.

É valido ressaltar que a mesma Eq. (2.33) é válida para a quantidade de dentes z_1 e z_2 das polias dentadas, uma vez que esta quantidade é proporcional ao diâmetro primitivo de cada uma.

De forma análoga, a transmissão dos momentos M_1 e M_2 de uma polia para outra pode ser calculada proporcionalmente aos respectivos diâmetros primitivos, conforme apresentado na Eq. (2.34).

$$\frac{\phi_2}{\phi_1} = \frac{M_2}{M_1} \quad (2.34)$$

Para o dimensionamento, as Eqs. (2.33) e (2.34) são utilizadas para definir o diâmetro das polias a serem aplicadas na transmissão. Uma vez encontrados os diâmetros, é necessário calcular o comprimento da correia a ser aplicada.

Para este cálculo é necessário observar a relação de transmissão das polias. No caso

de transmissões por elementos de diâmetros iguais, o comprimento de correia L pode ser calculado de acordo com a Eq. (2.35), em função do diâmetro primitivo ϕ e da distância C entre os centros de cada polia. Para transmissões com a relação $i \neq 1$, é necessário utilizar a Eq. (2.36) devido a presença do ângulo gerado pela diferença entre os diâmetros das polias (FRANCESCHI; ANTONELLO, 2014). Exemplos das duas relações de transmissão são ilustrados na Fig. 2.31.

$$L = \pi\phi + 2C \quad (2.35)$$

$$L = \pi\left(\frac{\phi_1}{2} + \frac{\phi_2}{2}\right) + 2\sqrt{C^2 + \left(\frac{\phi_2}{2} - \frac{\phi_1}{2}\right)^2} \quad (2.36)$$

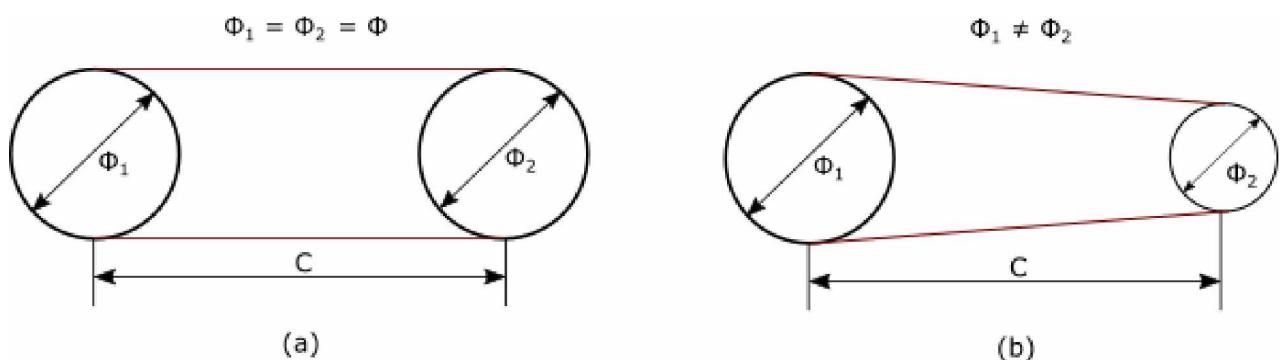


Figura 2.31 – Exemplo de uma transmissão para: (a) polias de diâmetros iguais; (b) de diâmetros diferentes.

2.6 Otimização por algoritmo genético

O método de otimização de sistemas é utilizado neste projeto para que os mecanismos projetados possuam comprimentos mínimos de barras, considerando as proporções médias de um indivíduo de 1,80m de altura, e os esforços que solicitarão o sistema.

O método selecionado consiste em uma otimização heurística baseada na simulação de um processo de seleção natural de indivíduos em um determinado ambiente, denominado método do algoritmo genético.

O algoritmo genético é baseado na teoria Darwiniana de sobrevivência e evolução das espécies. A busca se inicia através de uma população inicial de indivíduos (soluções candidatas) distribuídos aleatoriamente pelo espaço do projeto. Durante o processo evolutivo, cada indivíduo da população é avaliado, refletindo o seu grau de adaptação ao ambiente.

Alguns indivíduos da população são preservados enquanto outros são descartados, imitando o processo de seleção natural do Darwinismo. O grupo restante de indivíduos é emparelhado para que novos indivíduos possam ser gerados e assim substituir aqueles descartados durante a seleção.

Finalmente, alguns dos indivíduos são submetidos à reprodução e mutação, e como consequência, os cromossomos (vetor de variáveis de projeto) destes indivíduos são transmitidos e alterados. O processo como um todo é repetido até que uma solução satisfatória seja encontrada. Ainda que a proposta inicial para o algoritmo genético tenha sido dedicada a problemas com variáveis discretas, hoje em dia, melhorias estão disponíveis para o tratamento de variáveis contínuas e discretas simultaneamente. Para maiores detalhes, ver os trabalhos de Michalewicz (1994), Haupt e Haupt (1998), Michalewicz e Fogel (2000) e Goldberg (1989).

A seguir são descritas, de forma resumida, suas componentes.

Função-objetivo

Define-se como função-objetivo o objeto a ser minimizado ou maximizado durante a execução da rotina. Após a definição das condições iniciais e a execução do algoritmo, a função irá assumir o valor de interesse buscado de acordo com o problema em questão.

Uma das principais vantagens dos métodos heurísticos, como o algoritmo genético e suas variações, está no fato de não necessitar das características funcionais da função objetivo para otimizá-la, uma vez que o processo não é realizado analiticamente.

Indivíduo

O indivíduo consiste no artifício aplicado no método com a função meramente de armazenamento de seu código genético. Este, por sua vez, é definido como uma representação do espaço de busca do problema a ser resolvido, em geral na forma de sequências de bits, os quais definem o nível de adaptação de cada indivíduo em relação ao ambiente, objetivo da otimização.

Seleção

A seleção é a etapa do processo onde ocorre a escolha aleatória de indivíduos para se reproduzir e transmitir parte de seu código genético à próxima geração. O tipo mais comum de seleção é denominado roleta, onde os indivíduos são ordenados de acordo com a adaptação de seu código genético em relação à função-objetivo e lhes são atribuídas probabilidades decrescentes de serem escolhidos.

Dessa forma, é possível induzir a preferência aos mais adaptados e preservar a diversidade da população.

Reprodução

A reprodução, tradicionalmente, é dividida em três etapas distintas: acasalamento, recombinação e mutação. Durante a etapa de acasalamento, os indivíduos selecionados no processo anterior são ordenados em pares.

A partir dos pares formados, a etapa de recombinação simula o processo biológico de reprodução sexuada, combinando parte do código genético de seus ascendentes, o que irá

gradativamente gerar indivíduos mais aptos ao longo do processo.

A etapa final de mutação possui, geralmente, uma probabilidade baixa de ocorrência e consiste apenas na inversão aleatória de um dos bits do indivíduo que a sofre após a recombinação. Esta ação garante que o processo tenha menos chances de se limitar a pontos de mínimo locais.

O algoritmo a ser utilizado no projeto é do tipo evolução diferencial conforme ilustrado no fluxograma da Fig. 2.32, o qual também ilustra os conceitos apresentados nesta subseção.

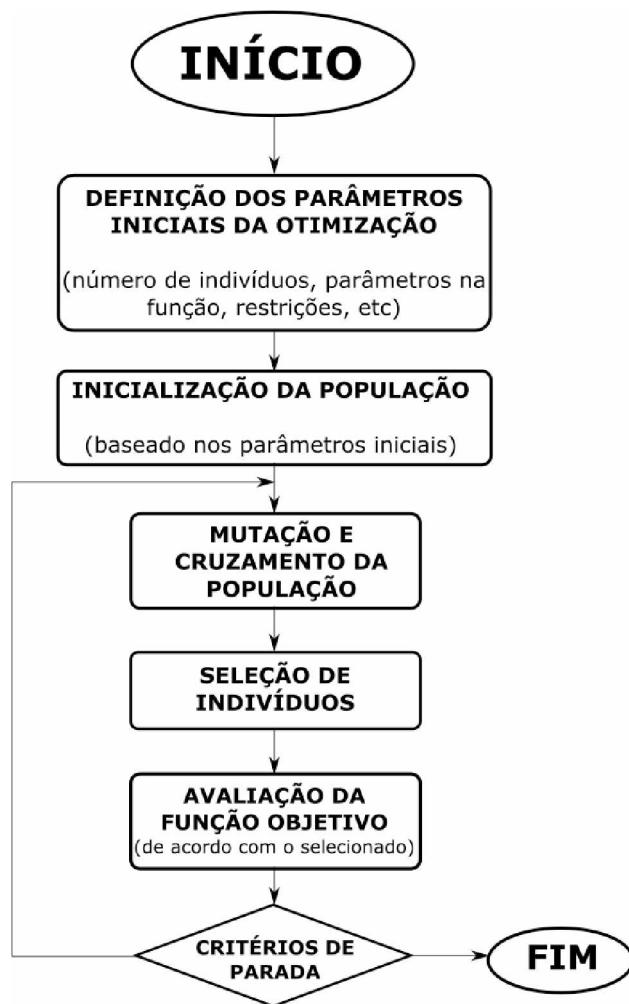


Figura 2.32 – Fluxograma detalhando o funcionamento do algoritmo de evolução diferencial aplicado neste trabalho (adaptado de Lobato, Steffen Jr e Oliveira-Lopes, 2007).

CAPÍTULO III

MODELAGEM MATEMÁTICA DOS MECANISMOS

Nessa dissertação é apresentado o desenvolvimento de três mecanismos articulados para serem aplicados em terapias de movimentação contínua dos membros inferiores. Cada mecanismo é direcionado à uma articulação ou seja: um mecanismo para movimentação do quadril; um para o joelho e outro para o tornozelo, conforme Fig. 3.1

Os mecanismos são construídos a partir de um mecanismo plano de quatro barras acionado pelo próprio paciente através de correias e polias sincronizadoras.

Neste capítulo é detalhado todo o processo de modelagem matemática dos três mecanismos propostos, bem como os resultados obtidos durante as simulações computacionais. Todos os procedimentos são baseados nas metodologias apresentadas no Capítulo II.

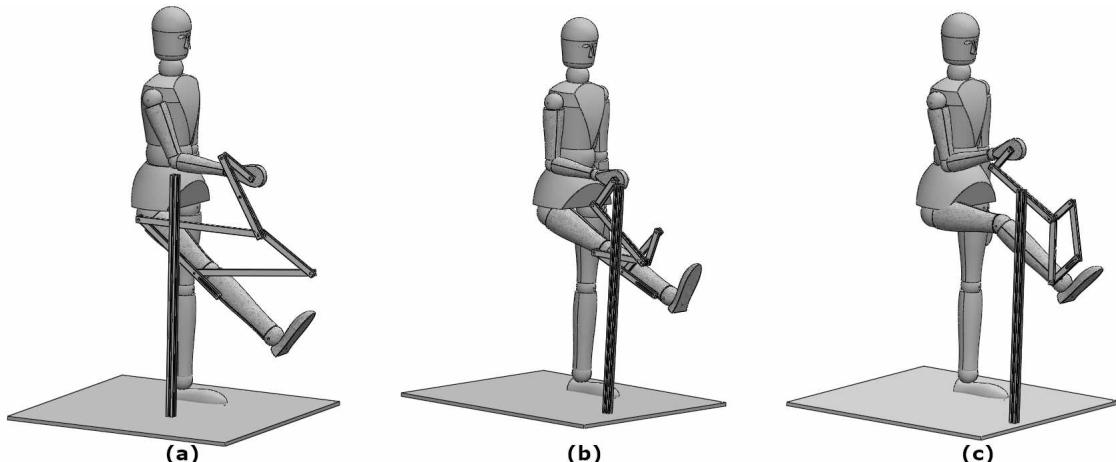


Figura 3.1 – Modelos computacionais gerados para os mecanismos do (a) quadril; (b) joelho; e (c) tornozelo.

Cada mecanismo é construído a partir da amplitude do movimento de flexão-extensão de sua respectiva articulação. O comprimento de cada barra é calculado com o auxílio do algoritmo de evolução diferencial.

O carregamento oriundo do peso da perna do paciente permite calcular os esforços, e as tensões nas barras para determinar o material e sua seção transversal.

Assim, com todas as dimensões definidas, é obtido o modelo computacional conforme ilustrado na Fig. 3.1.

3.1 Modelagem matemática

O primeiro passo para construção do modelo matemático dos mecanismos de 4 barras consiste em determinar o comprimento das barras, considerando as barras como unidimensionais, de forma que o centros de massa estejam sempre na metade do comprimento, e demais considerações conforme seção 2.3.1.

Para determinar os comprimentos das barras, utilizou-se inicialmente da busca otimizada proporcionada pelo algoritmo de evolução diferencial, utilizando o algoritmo elaborado por Lobato, Steffen Jr e Oliveira-Lopes (2007), transscrito no Anexo I, e aplicando as Eqs. (2.1) a (2.5) como restrições do problema, para formar a chamada função pseudo objetivo, que funciona como a função objetivo tradicional, porém aplicando penalidades quando as restrições são violadas. Também foi considerado a condição para que o ângulo de transmissão γ permaneça em uma faixa entre 30° e 150° , conforme recomendado por Brodell e Soni (1970) para evitar uma zona singular tradicionalmente conhecida em mecanismos de 4 barras planos. Uma captura de tela das montagens das restrições é apresentada na Fig. 3.2.

```

4 - theta_2= linspace(0,2*pi,2000);
5
6 % Características Geométricas (vide figura 2.22)
7
8 - S = sqrt(r(1)^2+r(2)^2-2*r(1)*r(2).*cos(theta_2));      %sequações 2.1 a 2.5
9 - beta = acos((r(1)+S.^2-r(2)^2)./(2*r(1).*S));
10 - phi = acos((r(3)+S.^2-r(4)^2)./(2*r(3).*S));
11 - lambda = acos((r(4)+S.^2-r(3)^2)./(2*r(4).*S));
12 - gama = acos((r(3)-S.^2+r(4)^2)./(2*r(3).*r(4)));
13
14 % Coordenadas dependentes
15
16 - theta_4 = pi-lambda-beta;
17
18 %restrições de igualdade:
19 - h1= (max(imag(gama))); %Elimina a possibilidade de ângulos intangíveis
20 - h2= abs(max(theta_4)-min(theta_4))/degtorad(80)-1; %garante o ângulo de saída desejado
21 - h3= abs(max(gama)-min(gama))/degtorad(120)-1; % garante a faixa do ângulo de transmissão
22
23 %restrições de desigualdade:
24 - g1= -min(gama)/degtorad(30)+1; % garante que o ângulo Gama seja maior que 30
25

```

Figura 3.2 – Restrições matemáticas construídas para gerar a função pseudo-objetivo e obter os comprimentos das barras à partir do algoritmo de evolução diferencial.

Inicialmente, define-se o número de variáveis que serão otimizadas pelo algoritmo, bem como os seus limites laterais, ou seja, o maior e o menor valor admissível dentro da busca a ser realizada. Para cada mecanismo, definiu-se os limites laterais de acordo com as medidas

médias de um indivíduo de 1,80m de altura. Um boneco de madeira antropométrico com medidas compatíveis foi utilizado posteriormente para testes em um protótipo de um dos mecanismos.

A Tabela 3.1 apresenta os dados aplicados para a busca pelo comprimento de cada uma das barras dos mecanismos propostos, bem como os resultados obtidos pelo algoritmo. O ambiente de desenvolvimento MATLAB® foi utilizado para executar o algoritmo de evolução diferencial.

Tabela 3.1 – Parâmetros utilizados para encontrar os comprimentos das barras, e ângulos de saída selecionados para cada mecanismo

	Parâmetro	Mínimo [mm]	Máximo[mm]	Selecionado[mm]
Tornozelo	r_1	100	300	273
	r_2	100	300	104
	r_3	100	300	235
	r_4	100	300	164
$\Delta\theta_4$ (ângulo de saída)		80°		
Joelho	r_1	150	400	280
	r_2	100	400	140
	r_3	150	400	263
	r_4	150	400	162
$\Delta\theta_4$ (ângulo de saída)		120°		
Quadril	r_1	250	565	273
	r_2	250	600	260
	r_3	250	600	235
	r_4	250	600	164
$\Delta\theta_4$ (ângulo de saída)		110°		

Cabe salientar que para os comprimentos das barras r_4 de cada mecanismo, foi utilizada uma extensão regulável para melhor acomodar a perna, com um rasgo na barra para realizar um ajuste contínuo, conforme observado na Fig. 3.3.

3.1.1 Simulação gráfica dos mecanismos

A partir dos comprimentos calculados para as barras dos mecanismos de cada articulação, a configuração de cada barra do mecanismo foi analisada em um intervalo de uma volta completa da entrada angular, oriunda da barra r_2 . Foi feita uma simulação gráfica computacional utilizando estes valores para observar o funcionamento de cada estrutura. Nas Figuras 3.4 a 3.6 estão ilustrados os movimentos animados usando as plotagens do MATLAB®. Nestas, as barras r_1 a r_4 estão indicadas em cada parte do movimento representado.

É importante ressaltar que as animações geradas não representam a modelagem cinemática das estruturas, as quais serão elaboradas adiante para eliminar as possibilidades de singularidades.

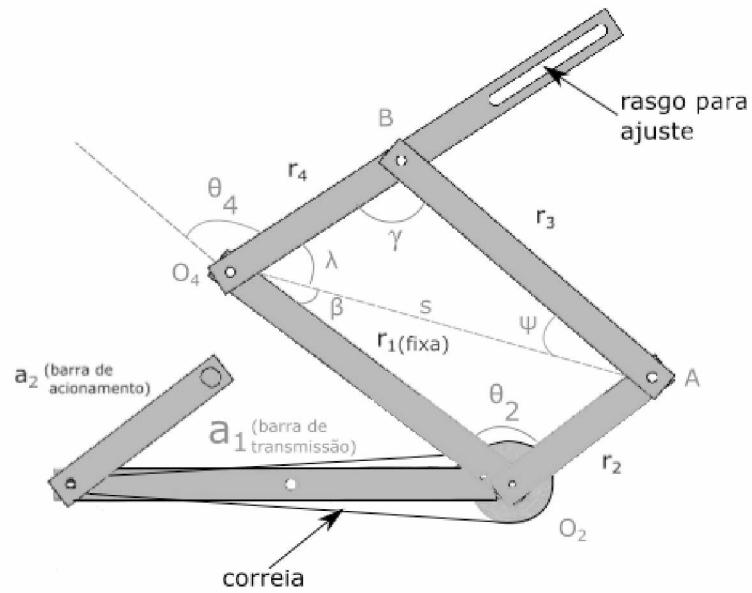


Figura 3.3 – Modelo computacional do mecanismo do joelho, detalhando as barras, o rasgo para ajuste e o sistema de acionamento.

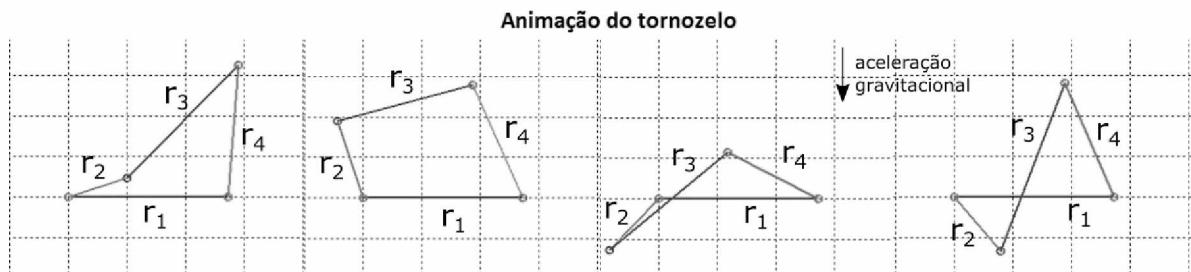


Figura 3.4 – Sequência de imagens gerada a partir da animação do movimento do tornozelo.

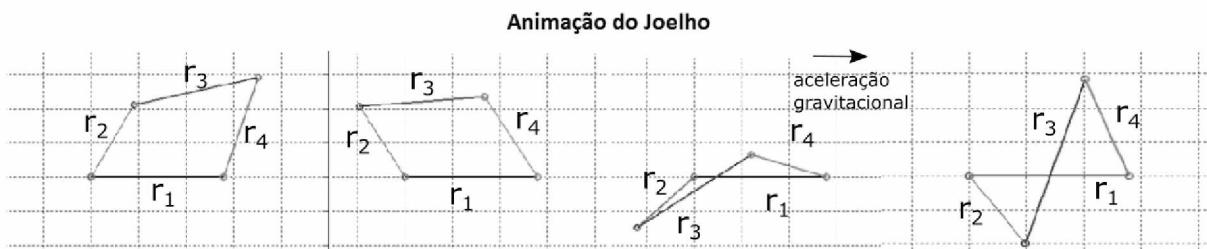


Figura 3.5 – Sequência de imagens gerada a partir da animação do movimento do joelho.

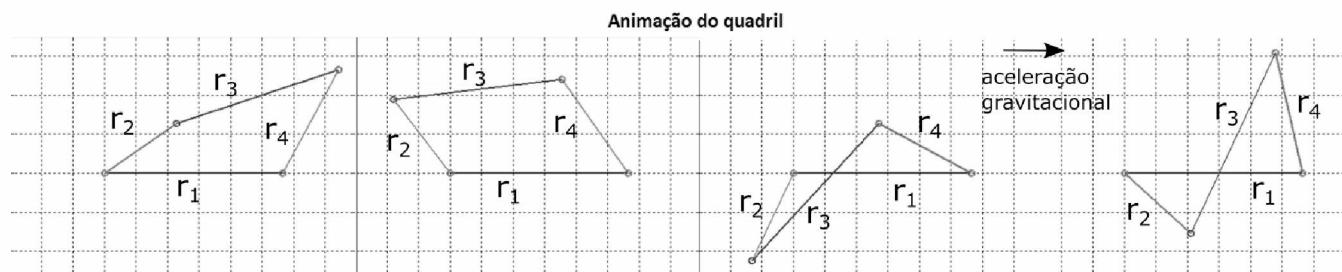


Figura 3.6 – Sequência de imagens gerada a partir da animação do movimento do quadril.

3.1.2 Amplitudes de saída

Após observar o movimento esperado para os mecanismos através das animações ilustradas nas Figs. 3.4 a 3.6, o próximo passo é medir a saída angular de cada mecanismo para garantir que as dimensões calculadas pelo algoritmo evolucionário estão próximas ao desejado.

Tendo em consideração o fato de que a metodologia aplicada para a otimização de sistemas por algoritmo genético está sujeita a variação, é necessário verificar os parâmetros impostos como restrição para garantir o correto funcionamento dos dispositivos.

A Figura 3.7 apresenta a variação do ângulo de saída θ_4 e do ângulo θ_3 encontrados para os comprimentos de barras obtidos para os mecanismos do tornozelo, joelho e quadril, respectivamente.

A amplitude angular obtida para o mecanismo do tornozelo é de 78.891° , contra 80° da articulação, com um erro relativo de 1.386%. Para o joelho, a saída obtida foi de 120.199° para 120° da articulação, resultando em 0.167% de erro relativo. Por fim, para o quadril foram obtidos 108.6899° para a amplitude de 110° , implicando um erro de 1.191%.

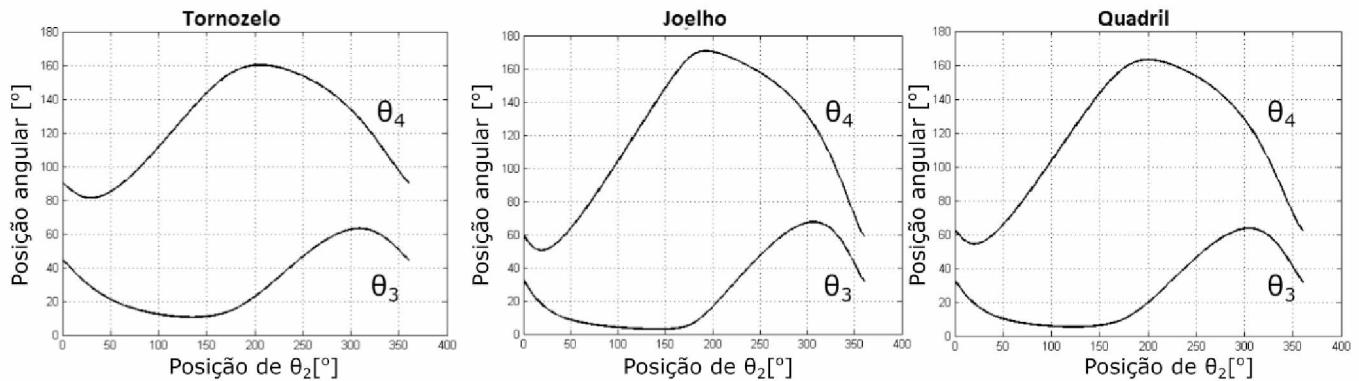


Figura 3.7 – Amplitudes angulares calculadas nos mecanismos gerados para o tornozelo, joelho e quadril.

3.2 Análise de carregamento

A seguir, antes de se iniciar a análise estática das posições do mecanismo, é necessário conhecer a natureza do carregamento o qual será submetido. Para a análise das estruturas propostas, Fig. 3.1, é possível utilizar uma estrutura simplificada para o estudo dos esforços existentes, conforme esquematizado na Fig. 3.8. Pode-se observar a posição relativa das barras onde r_2 corresponde à manivela de entrada do movimento e r_4 a saída, sendo a carga movimentada pela estrutura, correspondente ao peso do membro inferior, atuando como uma carga pontual \mathbf{F}_E atuando no rasgo utilizado para ajustes.

Para o mecanismo do tornozelo, o rasgo para ajuste foi posicionado entre as articulações B e O_4 , devido às proporções do pé humano, e nos mecanismos do quadril e do joelho no prolongamento da barra r_4 , de O_4 para B .

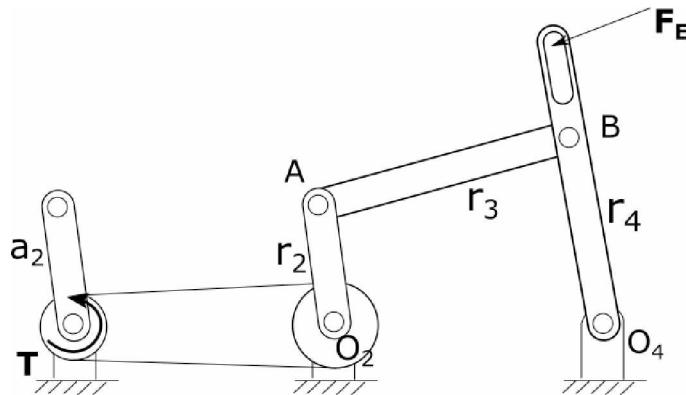


Figura 3.8 – Diagrama generalizado dos mecanismos propostos.

As cargas foram calculadas considerando as posições dos centros de massa locais e massa relativa de cada segmento do corpo humano de acordo com o estudo de Clauser, McConville e Young (1969), conforme apresentado na Tab. 3.2.

Tabela 3.2 – Massa relativa dos segmentos do corpo humano (CLAUSER; McCONVILLE; YOUNG, 1969)

Segmento	massa relativa
Pé	1.5%
Perna	4.4%
Coxa	10.5%
Pé e perna	5.8%
membro inferior	16.3%

A forma de se posicionar o mecanismo no membro inferior do paciente influencia a distribuição de cargas nas estruturas. Sendo assim, definiu-se a direção da aceleração gravitacional para aplicação dos carregamentos, utilizando a posição da barra r_1 como referência, de acordo com o apresentado nas animações das Figs. 3.4 a 3.6. Essas posições foram definidas de acordo com um pré-planejamento das posições de trabalho de cada mecanismo, e que serão evidenciadas ao final deste capítulo. As direções selecionadas estão apresentadas na Tab.3.3.

Tabela 3.3 – Direções adotadas para a aceleração gravitacional, em relação a barra r_1 , nos modelos construídos.

Articulação-alvo	direção
Tornozelo	0°
Joelho	90°
Quadril	90°

Utilizou-se como referência para o dimensionamento dos mecanismos, as medidas de um indivíduo de 1,80m de altura, e com um peso de até 150Kg.

Considerou-se também um fator de ponderação igual a 1,5 para compensar variações no carregamento da estrutura, como peso da órtese, atritos, variações bruscas na movimentação, etc.

Inicialmente, constrói-se a matriz de equilíbrio, Eq. (2.13), do sistema de cada mecanismo, de acordo com as Eqs. (2.10) a (2.12). A análise é feita em função do ângulo de entrada θ_2 , discretizado de 0 a 360° com intervalos de 1° , pela qual obtém-se os valores das reações em todas as posições de cada mecanismo.

Uma vez calculadas as forças e torques em todos os elementos do sistema, eles serão utilizados para calcular as tensões nas barras utilizando a definição de tensão combinada, Eq. (3.3), obtidas através dos carregamentos de tração/compressão σ_{Ni} e fletores σ_{Fi} , Eqs. (3.1) e (3.2), submetidas às barras r_i , onde A_i corresponde a área da seção transversal da barra, d_i é a distância da linha neutra da barra até a fibra mais tensionada e I_i o seu momento de inércia.

$$\sigma_{Ni} = \frac{F_{xi} \cos \theta_2}{A_i} + \frac{F_{yi} \sin \theta_2}{A_i} \quad (3.1)$$

$$\sigma_{Fi} = \frac{F_{xi} \sin \theta_2 d_i}{2I_i} + \frac{F_{yi} \cos \theta_2 d_i}{2I_i} \quad (3.2)$$

$$\sigma_{ri} = \sigma_{Ni} + \sigma_{Fi} \quad (3.3)$$

Os valores das tensões nas barras em cada posição percorrida no espaço de trabalho dos mecanismos deverá ser inferior ao limite de escoamento do material aplicado, multiplicado por um fator de segurança igual a 2 para compensar outros fatores como peso de elementos de conexão, erros estruturais, variações na posição da montagem do mecanismo, etc. O material utilizado é o alumínio, e seu respectivo limite de escoamento igual a 180 MPa.

O perfil inicial das barras foi um tubo retangular de 1" por $\frac{1}{2}$ " com espessura da parede de 3mm. No entanto, como pode ser verificado nos gráficos a seguir e na Tab. 3.4 que este perfil só poderia ser mantido para a estrutura do tornozelo, podendo ainda ser reduzido para 2mm. Para facilitar a construção, um mesmo perfil de barra foi utilizado para cada mecanismo.

Tornozelo

Para o mecanismo do tornozelo, as tensões foram calculadas de acordo com as Eqs. (3.1) a (3.3) individualmente por barra e para cada posição. Os resultados são apresentados na Fig. 3.9.

Os maiores valores calculados de tensão neste mecanismo para as barras r_1 a r_4 estão organizados na Tab. 3.4 para melhor visualização. Os valores de tensão calculados correspondem à espessura mínima considerada, de 3mm, o que implica que o mecanismo está apto a ser construído com segurança para esta dimensão, dentro das condições especificadas.

Joelho

Para o joelho, os resultados são detalhados nos gráficos da Fig. 3.10, e também foi possível manter as tensões estáveis dentro do limite de escoamento selecionado, considerando a espessura de 5mm do perfil. Os valores máximos estão destacados na Tab. 3.4.

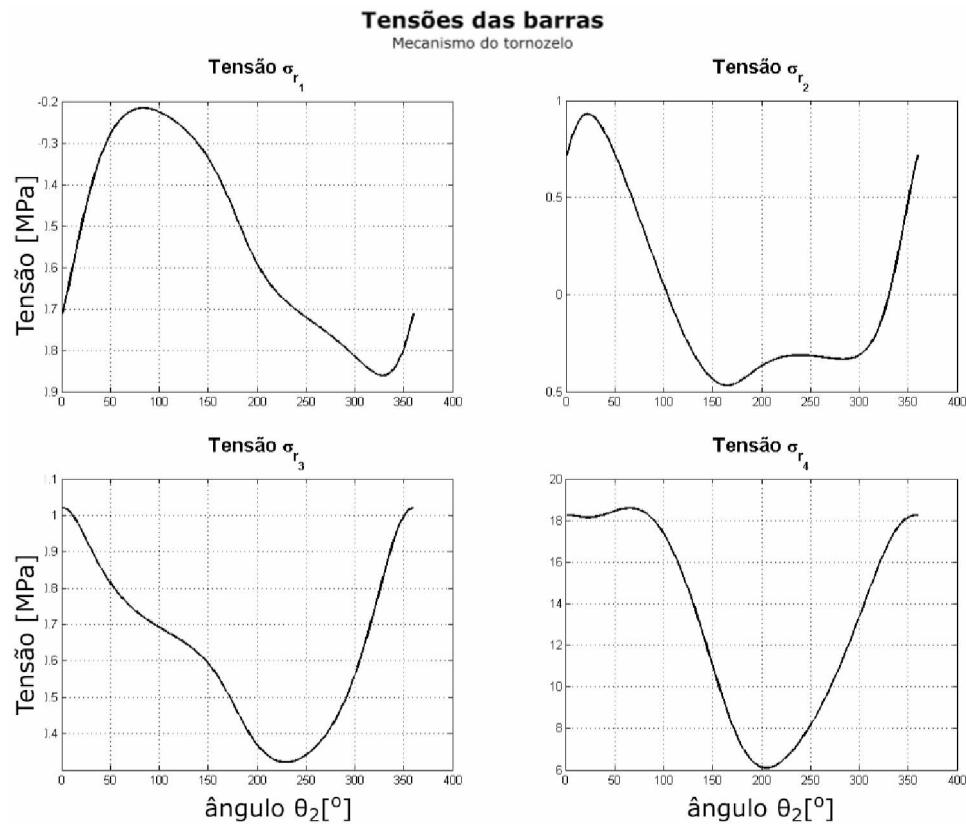


Figura 3.9 – Tensões calculadas nas barras r_1 a r_4 para o mecanismo do tornozelo.

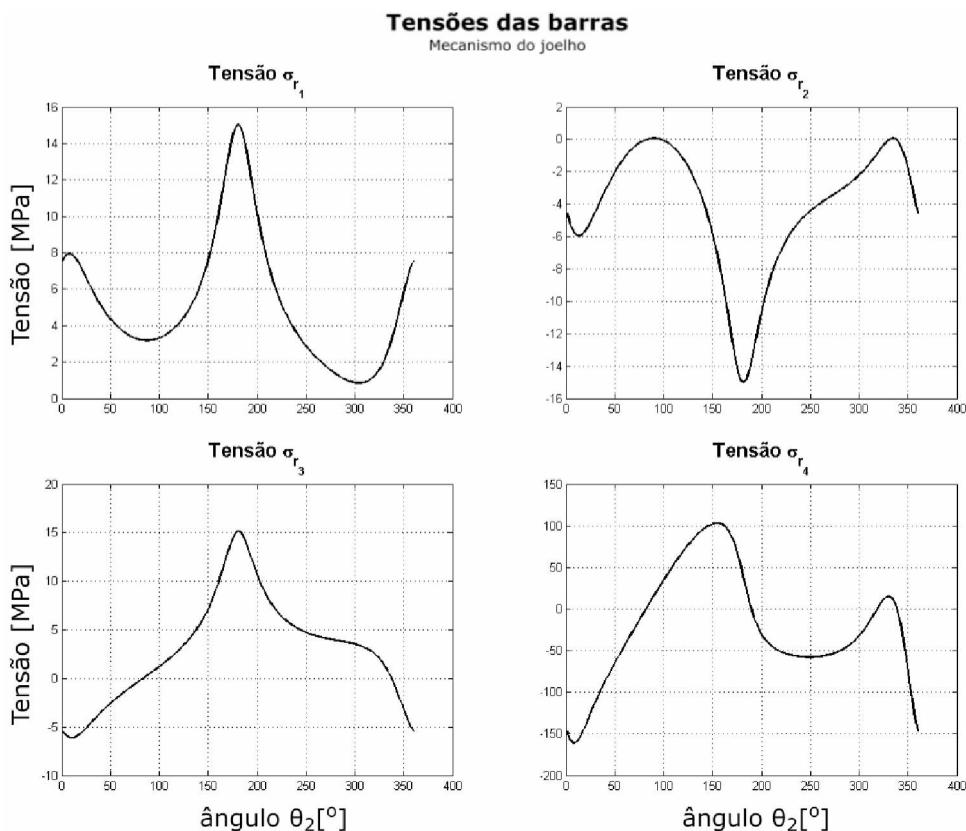


Figura 3.10 – Tensões calculadas nas barras r_1 a r_4 para o mecanismo do joelho.

Quadril

Para o quadril, são esperadas as maiores tensões, uma vez que sua estrutura deverá suportar o peso de todo o membro inferior do paciente durante a movimentação. Os dados são apresentados nos gráficos da Fig. 3.11.

Neste caso, foi necessário que as barras sejam maciças para suportar adequadamente os esforços, conforme valores de tensão máxima listados na Tab. 3.4.

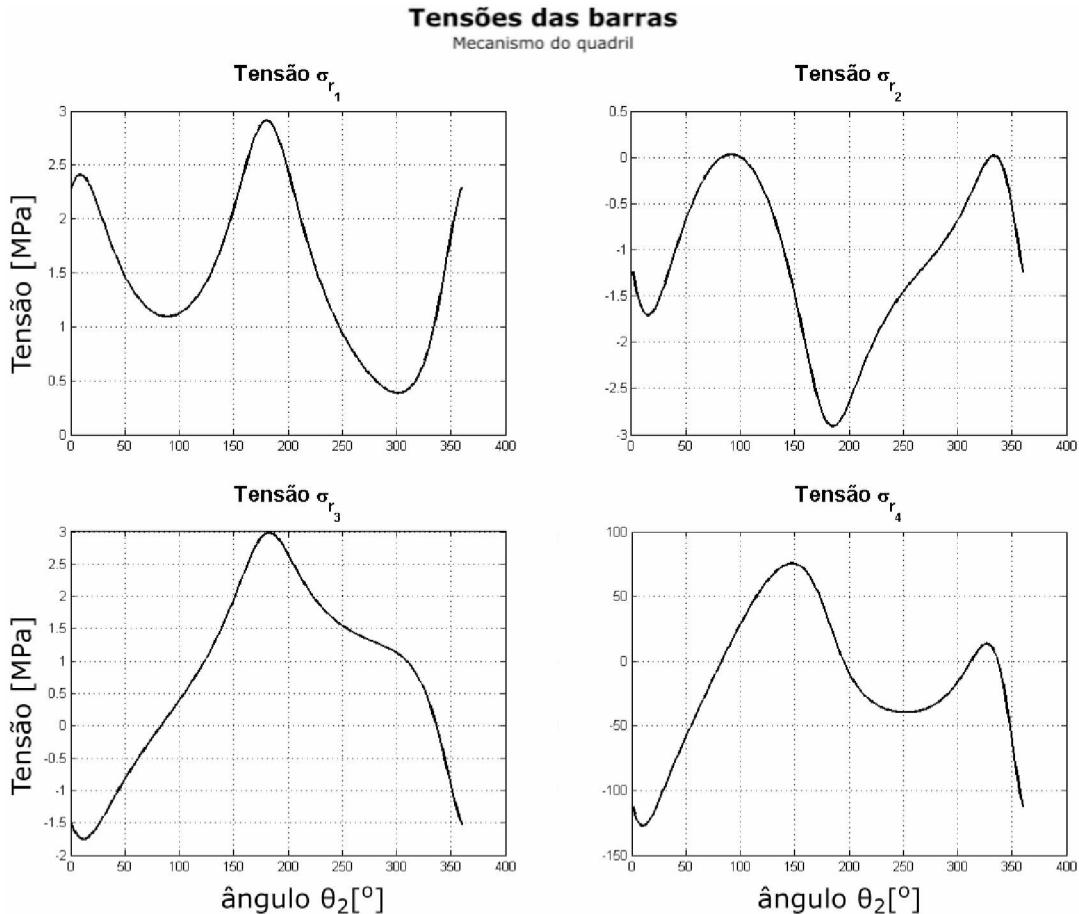


Figura 3.11 – Tensões calculadas nas barras r_1 a r_4 para o mecanismo do quadril.

3.3 Dimensionamento das polias

Para realizar o movimento previsto para os mecanismos, é preciso determinar a relação de redução necessária para que o usuário acione a manivela de entrada de forma adequada, garantindo que o torque seja transmitido de seu braço até a articulação do membro inferior a ser exercitado, e de forma suave.

Para isso, é feita uma análise do torque de entrada T exercido sobre a barra r_2 , o qual deverá ser transmitido por meio de um sistema de polias dentadas.

Como referência, os torques máximos para o braço humano foram escolhidos com base no trabalho de Neumann (2010), que apresenta estes valores como 725kgfcm para indivíduos

Tabela 3.4 – Valores máximos de tensão obtidos durante a análise estática

Limite de escoamento:(Alumínio)		180 [MPa]	
	Barra	σ_{max} [MPa]	Perfil especificado
Tornozelo	r_1	0,86103	
	r_2	0,93062	1”x1/2”x2mm
	r_3	1,0203	
	r_4	18,5972	
Joelho	r_1	15,0115	
	r_2	15,0052	1”x1/2”x5mm
	r_3	15,1471	
	r_4	160,8619	
Quadril	r_1	2,9121	
	r_2	2,9132	1”x1” - maciço
	r_3	2,9831	
	r_4	127,4755	

saudáveis de meia idade do sexo masculino, e $336kgfcm$ para o sexo feminino, o que corresponde a 71 e $33Nm$, respectivamente.

Portanto, as reduções devem ser construídas de forma a obter uma entrada menor que os valores acima apresentados. Buscando manter a ergonomia do sistema, dividiu-se os valores máximos de torque por um fator de ponderação igual a 2.

Os valores do torque de entrada para cada posição, ilustrado na Fig. 3.12, representam os torques calculados do tornozelo, joelho e quadril, de onde se extraí os valores extremos para comparação, mantendo o torque executado pelo paciente abaixo de $16,5Nm$.

Assim, como elaborado para a análise de tensões nas barras, os dados relativos ao dimensionamento das polias para a transmissão do movimento foram organizados conforme apresentado na Tab. 3.5. Conforme observado, a necessidade de redução do torque ocorre para o mecanismo do joelho e do quadril. Para a construção do protótipo a ser detalhada no próximo capítulo será utilizada uma polia dentada para uma redução de 3, para facilitar a fabricação da mesma.

Tabela 3.5 – Relações de transmissão escolhidas para as polias dentadas

Articulação	T_{max} calc. [Nm]	T_{max} permitido [Nm]	rel. de transmissão
Tornozelo	3.8241	16.5	1
Joelho	38.7391	16.5	2.8174
Quadril	78.5580	16.5	5.9514

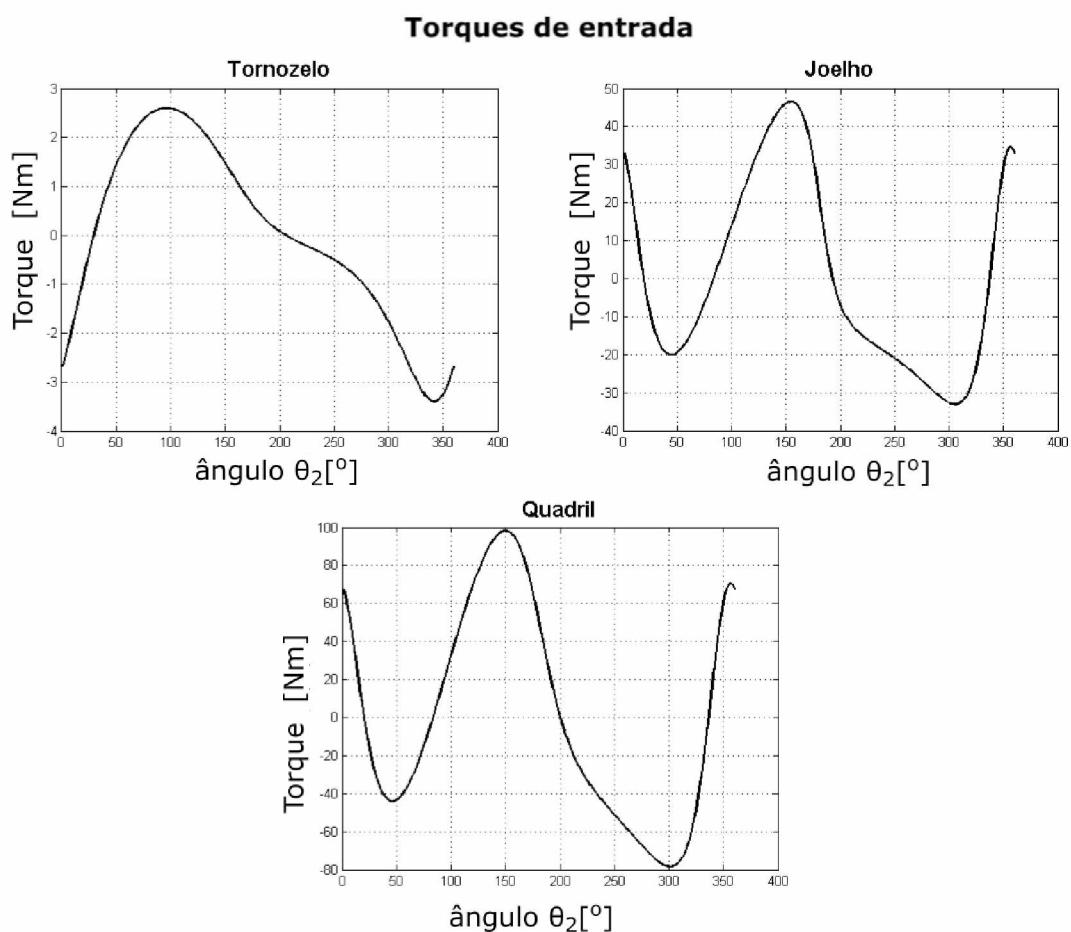


Figura 3.12 – Torques de entrada calculados para os mecanismos do tornozelo, joelho e quadril.

3.4 Análise de singularidades

Prosseguindo com a validação das dimensões geradas pelo algoritmo em questão, a última análise consiste no estudo das singularidades para garantir a ausência de travamento e/ou falta de controle nas estruturas, conforme detalhado no capítulo anterior.

O primeiro passo consiste em montar a matriz geométrica global $[G]$, que é a mesma para os três mecanismos. Para isso, a matriz geométrica elementar $[G_e]$, Eqs.(2.20) e (2.24), é obtida para cada barra r_2 , r_3 e r_4 cujos nós estão definidos na Fig. 3.13.

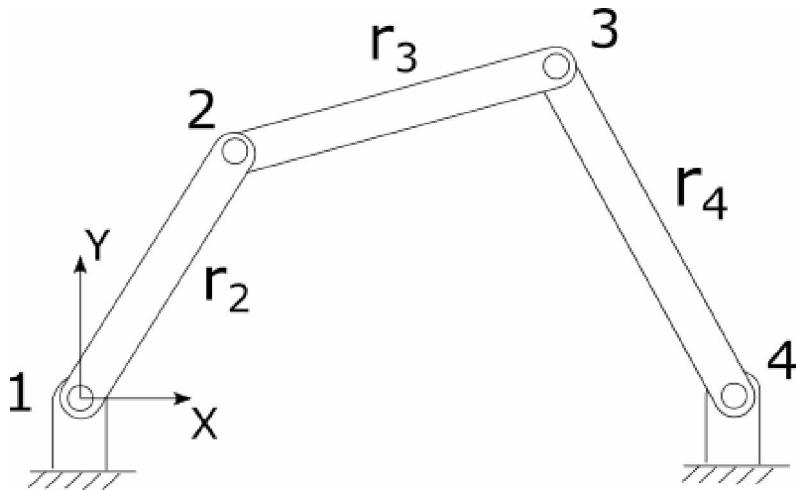


Figura 3.13 – Nós associados a cada barra para obter a matriz geométrica do mecanismo.

Assim, a barra r_2 é formada pelos nós 1 e 2; a barra r_3 pelos nós 2 e 3; e a barra r_4 pelos nós 3 e 4. Em seguida cada matriz geométrica é expandida, obtendo a matriz $[G_e]$, conforme Eqs. (2.25) e (2.26). Em seguida, a matriz geométrica global do mecanismo $[G]$ pode ser obtida a partir da Eq. (2.28).

Finalmente, são consideradas as restrições impostas pelas articulações fixas (nós 1 e 4), permitindo obter a equação das velocidades nodais, dada pela Eq. (2.27), permitindo resolver a cinemática do mecanismo.

Por exemplo, a matriz elementar relativa à barra r_2 , $[G_{r_2}]$ é dada pela Eq. (3.4).

$$\begin{aligned}
 [G_{r_2}]\{\dot{x}_{r_2}\} &= \\
 &= \begin{bmatrix} \cos^2 \theta_2 & \cos \theta_2 \sin \theta_2 & -\cos^2 \theta_2 & -\cos \theta_2 \sin \theta_2 \\ \cos \theta_2 \sin \theta_2 & \sin^2 \theta_2 & -\cos \theta_2 \sin \theta_2 & -\sin^2 \theta_2 \\ -\cos^2 \theta_2 & -\cos \theta_2 \sin \theta_2 & \cos^2 \theta_2 & \cos \theta_2 \sin \theta_2 \\ -\cos \theta_2 \sin \theta_2 & -\sin^2 \theta_2 & \cos \theta_2 \sin \theta_2 & \sin^2 \theta_2 \end{bmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \end{pmatrix} \quad (3.4)
 \end{aligned}$$

A matriz elementar $[G_{r_2}]$ pode então ser expandida para sua forma global, onde são inseridas as velocidades de todos os nós do mecanismo através do vetor $\{\dot{x}\}$. Os resultados para as barras r_2 , r_3 e r_4 são apresentados na Eqs. (3.5) a (3.7).

$$\begin{aligned}
& [\bar{G}_{r_2}] \{ \dot{x} \} = \\
& = \begin{bmatrix} -\cos^2 \theta_2 & -\cos \theta_2 \operatorname{sen} \theta_2 & 0 & 0 & 0 & 0 & \cos^2 \theta_2 & \cos \theta_2 \operatorname{sen} \theta_2 \\ -\cos \theta_2 \operatorname{sen} \theta_2 & -\operatorname{sen}^2 \theta_2 & 0 & 0 & 0 & 0 & \cos \theta_2 \operatorname{sen} \theta_2 & \operatorname{sen}^2 \theta_2 \\ \cos^2 \theta_2 & \cos \theta_2 \operatorname{sen} \theta_2 & 0 & 0 & 0 & 0 & -\cos^2 \theta_2 & -\cos \theta_2 \operatorname{sen} \theta_2 \\ \cos \theta_2 \operatorname{sen} \theta_2 & \operatorname{sen}^2 \theta_2 & 0 & 0 & 0 & 0 & -\cos \theta_2 \operatorname{sen} \theta_2 & -\operatorname{sen}^2 \theta_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{x}_3 \\ \dot{y}_3 \\ \dot{x}_4 \\ \dot{y}_4 \end{pmatrix} \quad (3.5)
\end{aligned}$$

$$\begin{aligned}
& [\bar{G}_{r_3}] \{ \dot{x} \} = \\
& = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \cos^2 \theta_3 & \cos \theta_3 \operatorname{sen} \theta_3 & -\cos^2 \theta_3 & -\cos \theta_3 \operatorname{sen} \theta_3 & 0 & 0 & 0 & 0 \\ \cos \theta_3 \operatorname{sen} \theta_3 & \operatorname{sen}^2 \theta_3 & -\cos \theta_3 \operatorname{sen} \theta_3 & -\operatorname{sen}^2 \theta_3 & 0 & 0 & 0 & 0 \\ -\cos^2 \theta_3 & -\cos \theta_3 \operatorname{sen} \theta_3 & \cos^2 \theta_3 & \cos \theta_3 \operatorname{sen} \theta_3 & 0 & 0 & 0 & 0 \\ -\cos \theta_3 \operatorname{sen} \theta_3 & -\operatorname{sen}^2 \theta_3 & \cos \theta_3 \operatorname{sen} \theta_3 & \operatorname{sen}^2 \theta_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{x}_3 \\ \dot{y}_3 \\ \dot{x}_4 \\ \dot{y}_4 \end{pmatrix} \quad (3.6)
\end{aligned}$$

$$\begin{aligned}
& [\bar{G}_{r_4}] \{ \dot{x} \} = \\
& = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \cos^2 \theta_4 & \cos \theta_4 \operatorname{sen} \theta_4 & -\cos^2 \theta_4 & -\cos \theta_4 \operatorname{sen} \theta_4 & 0 & 0 \\ 0 & 0 & \cos \theta_4 \operatorname{sen} \theta_4 & \operatorname{sen}^2 \theta_4 & -\cos \theta_4 \operatorname{sen} \theta_4 & -\operatorname{sen}^2 \theta_4 & 0 & 0 \\ 0 & 0 & -\cos^2 \theta_4 & -\cos \theta_4 \operatorname{sen} \theta_4 & \cos^2 \theta_4 & \cos \theta_4 \operatorname{sen} \theta_4 & 0 & 0 \\ 0 & 0 & -\cos \theta_4 \operatorname{sen} \theta_4 & -\operatorname{sen}^2 \theta_4 & \cos \theta_4 \operatorname{sen} \theta_4 & \operatorname{sen}^2 \theta_4 & 0 & 0 \end{bmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \dot{x}_2 \\ \dot{y}_2 \\ \dot{x}_3 \\ \dot{y}_3 \\ \dot{x}_4 \\ \dot{y}_4 \end{pmatrix} \quad (3.7)
\end{aligned}$$

Aplicando a Equação (2.28) obtém-se a matriz global $[G]$, Eq. (3.8), onde as funções seno e cosseno foram abreviadas por "s" e "c", respectivamente.

$$[G] = \sum_{i=1}^4 [\bar{G}_{r_i}] =$$

$$\begin{bmatrix} -c^2\theta_2 & -c\theta_2s\theta_2 & 0 & 0 & 0 & 0 & c\theta_2s\theta_2 & s^2\theta_2 \\ -c\theta_2s\theta_2 & -s^2\theta_2 & 0 & 0 & 0 & 0 & c^2\theta_2 & c\theta_2s\theta_2 \\ c^2\theta_2 + c^2\theta_3 & c\theta_2s\theta_2 + c\theta_3s\theta_3 & -c^2\theta_3 & -c\theta_3s\theta_3 & 0 & 0 & -c^2\theta_2 & -c\theta_2s\theta_2 \\ c\theta_2s\theta_2 + c\theta_3s\theta_3 & s^2\theta_2 + s^2\theta_3 & -c\theta_3s\theta_3 & -s^2\theta_3 & 0 & 0 & -c\theta_2s\theta_2 & -s^2\theta_2 \\ -c^2\theta_3 & -c\theta_3s\theta_3 & c^2\theta_3 + c^2\theta_4 & c\theta_3s\theta_3 + c\theta_4s\theta_4 & -c^2\theta_4 & -c\theta_4s\theta_4 & 0 & 0 \\ -c\theta_3s\theta_3 & -s^2\theta_3 & c\theta_3s\theta_3 + c\theta_4s\theta_4 & s^2\theta_3 + s^2\theta_4 & -c\theta_4s\theta_4 & -s^2\theta_4 & 0 & 0 \\ 0 & 0 & -c^2\theta_4 & -c\theta_4s\theta_4 & c^2\theta_4 & c\theta_4s\theta_4 & 0 & 0 \\ 0 & 0 & -c\theta_4s\theta_4 & -s^2\theta_4 & c\theta_4s\theta_4 & s^2\theta_4 & 0 & 0 \end{bmatrix} \quad (3.8)$$

Considerando as restrições impostas aos nós 1 e 4, referentes às articulações fixas na base e que possuem velocidades nulas, obtém-se as equações das velocidades nodais, ou seja:

$$[G]\{\dot{x}\} = \{0\} \Rightarrow$$

$$\Rightarrow \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ c^2\theta_2 + c^2\theta_3 & c\theta_2s\theta_2 + c\theta_3s\theta_3 & -c^2\theta_3 & -c\theta_3s\theta_3 & 0 & 0 & 0 & 0 \\ c\theta_2s\theta_2 + c\theta_3s\theta_3 & s^2\theta_2 + s^2\theta_3 & -c\theta_3s\theta_3 & -s^2\theta_3 & 0 & 0 & 0 & 0 \\ -c^2\theta_3 & -c\theta_3s\theta_3 & c^2\theta_3 + c^2\theta_4 & c\theta_3s\theta_3 + c\theta_4s\theta_4 & 0 & 0 & 0 & 0 \\ -c\theta_3s\theta_3 & -s^2\theta_3 & c\theta_3s\theta_3 + c\theta_4s\theta_4 & s^2\theta_3 + s^2\theta_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \\ x_4 \\ y_4 \end{pmatrix} = \{0\} \quad (3.9)$$

A matriz geométrica global $[G]$, Eq. (3.9) é calculada em todos os pontos do movimento e usada no problema de autovalores da Eq. (2.29) para gerar os parâmetros de avaliação necessários para investigar as possíveis singularidades no problema.

Conforme apresentado na seção 2.4, apenas os autovalores não nulos são usados para determinar a existência de pontos com mobilidade incrementada. Resolvendo a Equação (2.29) com a matriz $[G]$ da Eq. (3.9) obtém-se os autovalores λ e os respectivos autovetores $[v_i]$ associados. Foi observado que apenas dois autovalores não nulos variam conforme a progressão da entrada dos mecanismos, sendo que os restantes, permanecem com valores constantes. Os gráficos ilustrados nas Figuras 3.14 a 3.16 apresentam respectivamente a variação observada destes valores para as montagens para o tornozelo, joelho e quadril.

Observa-se que ambos os autovalores não se anulam durante a movimentação de nenhum dos mecanismos, evidenciando a ausência de incrementos da mobilidade dentro do espaço analisado.

Além disso, através das Eqs. (2.31) e (2.32), foi verificada a ausência de singularidades

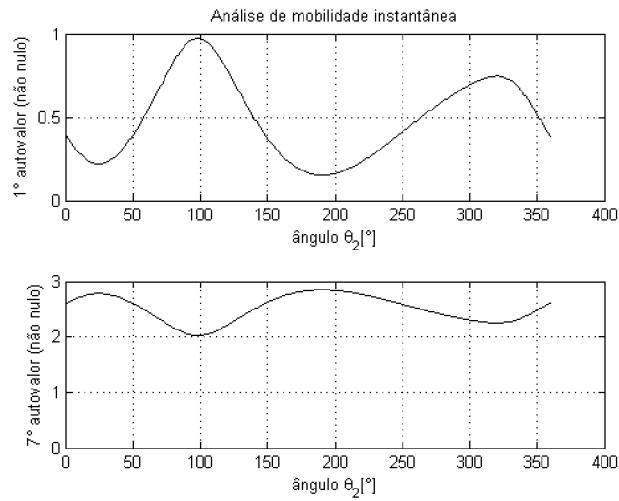


Figura 3.14 – Variação dos autovalores não nulos da montagem para o tornozelo.

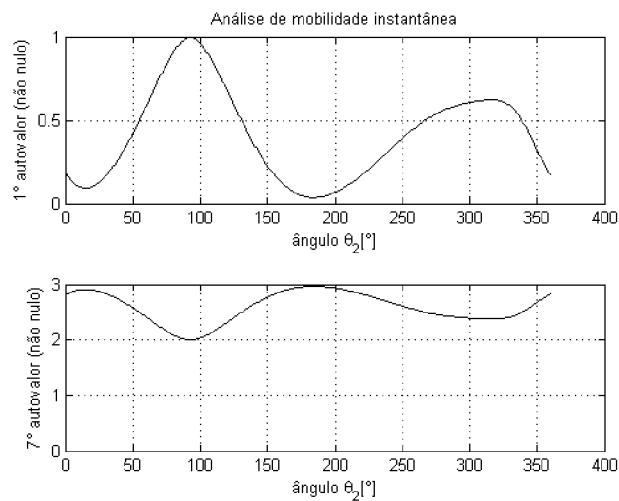


Figura 3.15 – Variação dos autovalores não nulos da montagem para o joelho.

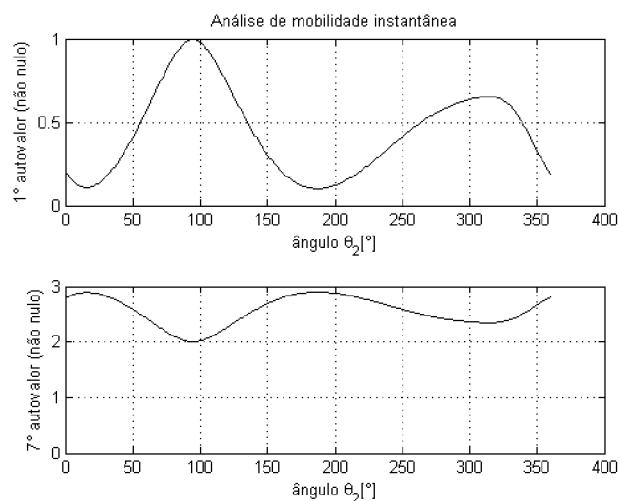


Figura 3.16 – Variação dos autovalores não nulos da montagem para o quadril.

de entrada e/ou saída nos mecanismos.

Sendo assim, as condições aqui impostas para o correto funcionamento dos mecanismos desenvolvidos foram atendidas, implicando que os mecanismos propostos estão aptos para gerar dispositivos capazes de obter saídas angulares adequadas para exercícios terapêuticos.

3.5 Modelagem CAD

Com os modelos matemáticos resolvidos e os comprimentos das barras definidos, a construção dos mecanismos em modelos tridimensionais pode ser realizada utilizando o software de CAD SolidWorks®.

Através destes modelos, foi possível adicionar a perspectiva do posicionamento de cada barra no espaço, forma de apoio e composição mecânica da transmissão por polias, além de tornar visível a direção da aceleração gravitacional selecionada para posicionar o carregamento, e a forma em que o paciente irá operar o sistema de entrada de movimento.

Com os modelos construídos é possível, então, executar uma simulação cinemática completa dos mecanismos e observar a movimentação esperada para o sistema. As Figuras 3.17 a 3.19 ilustram o resultado final obtido nas simulações realizadas.

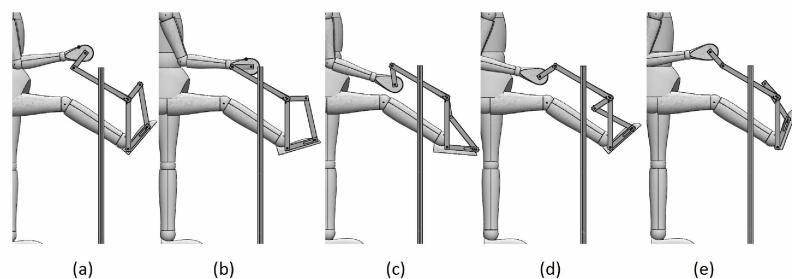


Figura 3.17 – Sequência da simulação cinemática gerada pelo modelo tridimensional do tornozelo: (a) posição anatômica, (b) Extensão parcial, (c) extensão total, (d) posição anatômica, (e) flexão total.

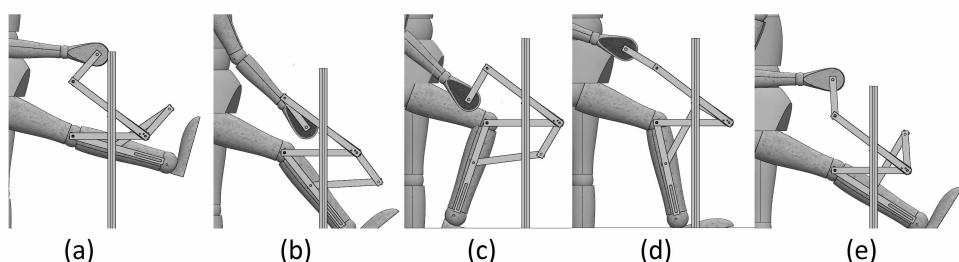


Figura 3.18 – Sequência da simulação cinemática gerada pelo modelo tridimensional do joelho: (a) posição anatômica, (b) flexão parcial, (c) flexão total, (d) flexão parcial, (e) posição anatômica.

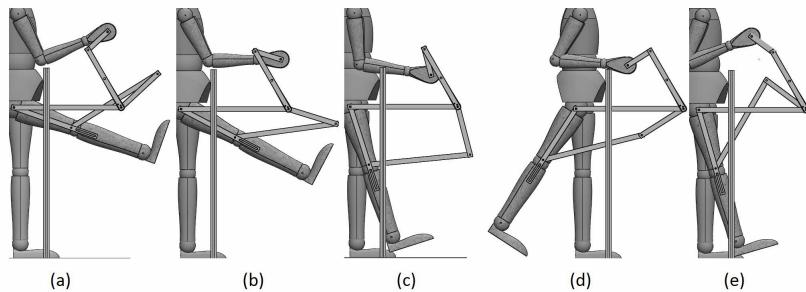


Figura 3.19 – Sequência da simulação cinemática gerada pelo modelo tridimensional do quadril: (a,b) flexão parcial, (c) posição próxima a anatômica, (d) extensão total,(e) posição próxima a anatômica.

Por fim, a partir dos modelos tridimensionais produzidos, construiu-se um protótipo para validação dos resultados e estudo de aplicações, o qual será detalhado no Capítulo V.

CAPÍTULO IV

APLICAÇÃO COM JOGOS ELETRÔNICOS

A seguir, são descritos os procedimentos utilizados para o desenvolvimento de um jogo eletrônico para funcionar em conjunto com os mecanismos propostos, a fim de motivar e proporcionar entretenimento para os pacientes, devido à natureza repetitiva, normalmente entediante, de tratamento por terapia passiva de movimento contínuo.

Nesta seção são apresentadas as ferramentas utilizadas para desenvolvimento da aplicação, a mecânica proposta para o jogo e o processo de interface homem-máquina aplicado à estrutura para interagir com o jogo desenvolvido.

4.1 Plataforma de Desenvolvimento

Para gerar a aplicação, foi utilizada a versão gratuita da plataforma para desenvolvimento de jogos Game Maker: Studio[®], distribuída pela Yoyo Games. A escolha deste ambiente de desenvolvimento se dá pela popularidade e funcionalidade da plataforma, além da agilidade no desenvolvimento de jogos 2D e do volume de conteúdos abertos gerado pela comunidade de desenvolvedores.

A plataforma selecionada combina uma interface interativa de programação por blocos gráficos, e uma linguagem própria orientada a objetos, baseada em C++, o que torna o processo de programação dos jogos intuitivo e simples.

O ambiente de desenvolvimento permite a escolha de imagens animadas para objetos e cenários, chamadas *sprites*, sons ambientais e efeitos sonoros, planos de fundo e fontes para textos. A partir destes recursos, pode-se programar os objetos propriamente ditos, que irão compor a mecânica do jogo, e posicioná-los em "salas", como são chamados os ambientes onde os objetos interagem entre si e com o usuário.

Uma imagem do ambiente de desenvolvimento da plataforma Game Maker: Studio[®] é apresentada na Fig. 4.1.

A mecânica operacional fornecida pela plataforma baseia-se na gestão de eventos que

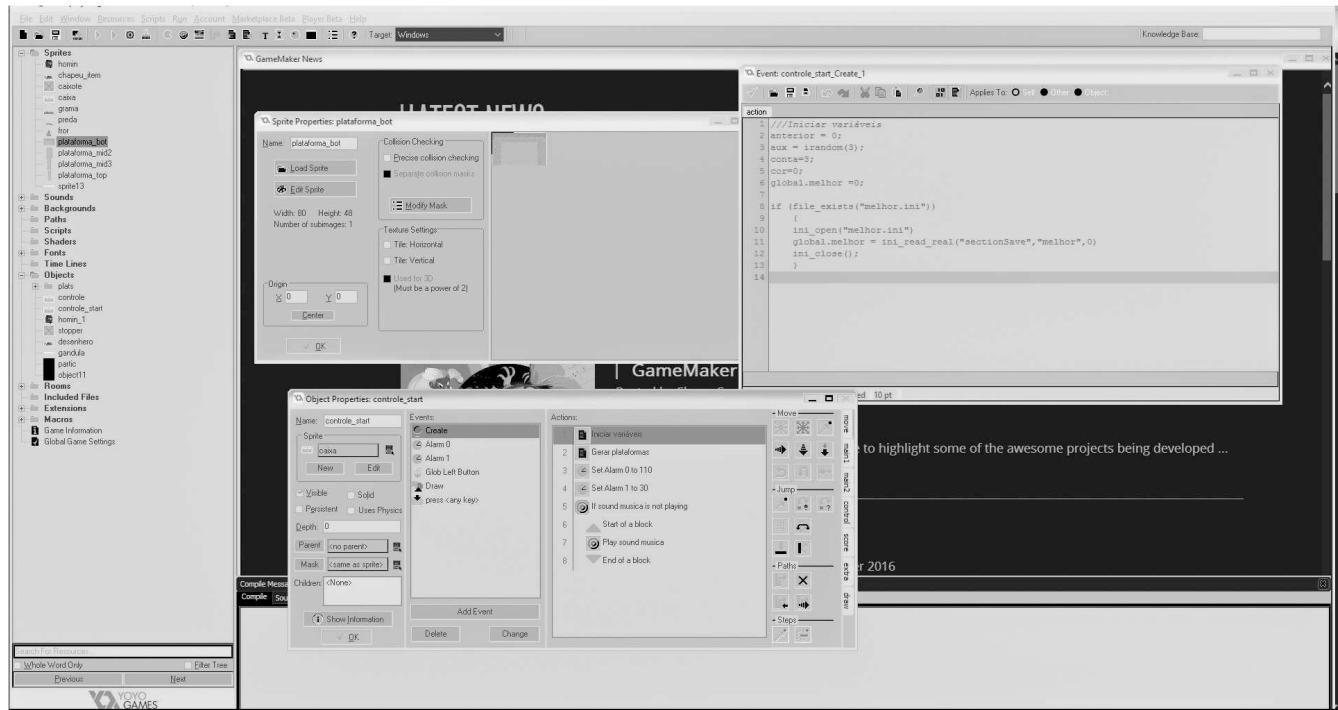


Figura 4.1 – captura de tela da plataforma utilizada para gerar o jogo para reabilitação.

ocorrem em uma sala. Define-se como evento todos os tipos de interação que os objetos possam ter entre si ou com o jogador, a criação ou destruição de um determinado objeto, seu posicionamento e colisões com outros objetos, temporizadores, botões de mouse e teclado, dentre outros.

Os eventos são processados a cada "passo", os quais correspondem a cada quadro processado no jogo, e que têm a velocidade controlada diretamente por cada sala. Todas as métricas do jogo são referenciadas pelos eventos e passos que ocorrem dentro do mesmo.

4.2 Elementos do jogo

O jogo foi desenvolvido orientado pelo gênero e intensidade adequados para ser associado ao mecanismo construído. Dessa forma, assim como a estrutura, ele possui apenas um grau de liberdade, o qual é controlado pela manivela que o paciente opera. Deve-se considerar que também que não há uma definição estrita para o público-alvo do jogo, uma vez que o mecanismos possam ser operados por qualquer paciente.

Sendo assim, o jogo tem um caráter simples, o que caracteriza jogos casuais clássicos. Optou-se, então, por desenvolver um jogo "plataforma", onde o personagem deve se mover em uma única direção, enquanto salta sobre plataformas. A pontuação do jogo é contada de acordo com o número de plataformas que o jogador conseguir saltar com sucesso.

Foram utilizados e adaptados doze *sprites*, três efeitos sonoros, uma música de ambiente, e um plano de fundo como recursos áudio-visuais para constituir os conteúdos do jogo.

Os objetos e suas respectivas funções são apresentados a seguir. Todos os recursos

utilizados para o desenvolvimento do jogo são de distribuição gratuita, livres de *royalties*, ou foram produzidos pelo autor.

Personagem principal

O personagem principal do jogo é uma caricatura pixerizada e sua única habilidade é saltar. Durante o jogo, ele vai se manter parado, enquanto o cenário restante se move em sua direção. Porém, sua *sprite* tem uma animação que simula movimentação, para dar a sensação de que o personagem se move.

Este personagem é definido com apenas um objeto, o qual possui três eventos gerenciados, sendo eles: o evento de própria criação, disparado no passo em que o objeto é criado na sala; um evento passo-a-passo, que ocorre no início de todos os passos; e um evento de colisão no fundo da sala. O objeto do personagem é apresentado na Fig. 4.2.

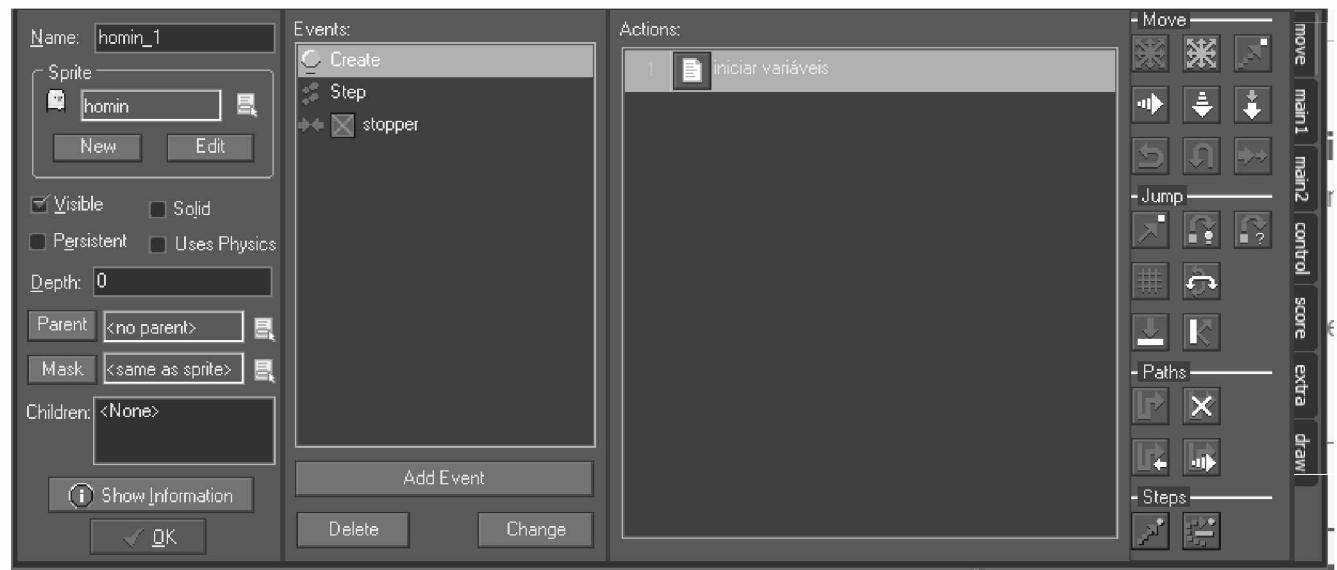


Figura 4.2 – Captura de tela de programação do objeto do personagem principal.

Durante o primeiro evento, o objeto dispara a execução de um código, que define as variáveis de movimentação do personagem. Estas variáveis controlam o efeito gravitacional a ser exercido sobre o personagem, as componentes horizontal e vertical de sua velocidade, e a velocidade inicial de impulso para realizar o pulo.

O segundo evento executa um código a cada passo e define a interface do objeto com o jogador. Neste código, inicialmente, efetua-se a leitura da tecla de interface para acionar o pulo, neste caso, a barra de espaço. Impõe-se também um limite vertical para a aceleração gravitacional, para simular o efeito de velocidade terminal de queda livre e facilitar o controle do personagem.

Também são definidos nesse evento a condição de pular apenas quando o personagem principal está sobre uma plataforma; as colisões suaves em ambas as direções; e a mecânica do pulo está conforme definido na Eq. (4.1), onde V_{sp} corresponde à velocidade vertical do objeto,

key_{jump} a variável binária que indica quando a barra de espaço é pressionada, e $(-jump_{speed})$ a velocidade inicial definida para o pulo.

$$V_{sp} = key_{jump}(-jump_{speed}) \quad (4.1)$$

O terceiro evento indica o fim do jogo, quando o personagem cai das plataformas e colide com os objetos delimitadores inferiores da sala. Quando iniciado, este evento toca um efeito sonoro indicando a queda do objeto, dispara um *timer* de 90 passos no objeto de controle da sala, e se autodestrói. Os delimitadores e o objeto de controle são definidos a seguir.

Plataformas

As plataformas pelas quais o personagem se apoia ao saltar são definidas com quatro objetos, um para cada altura. Cada objeto também possui uma *sprite* específica para sua altura.

Os únicos eventos gerenciados pelas plataformas ocorrem no passo de sua criação, onde estas iniciam um movimento para a esquerda a 1,5 pixel por segundo, e no passo em que colidem com os delimitadores de sala, quando se autodestroem se já estiverem fora da área da sala. O desenho usado como *sprite* para uma das plataformas é apresentado na Fig. 4.3.

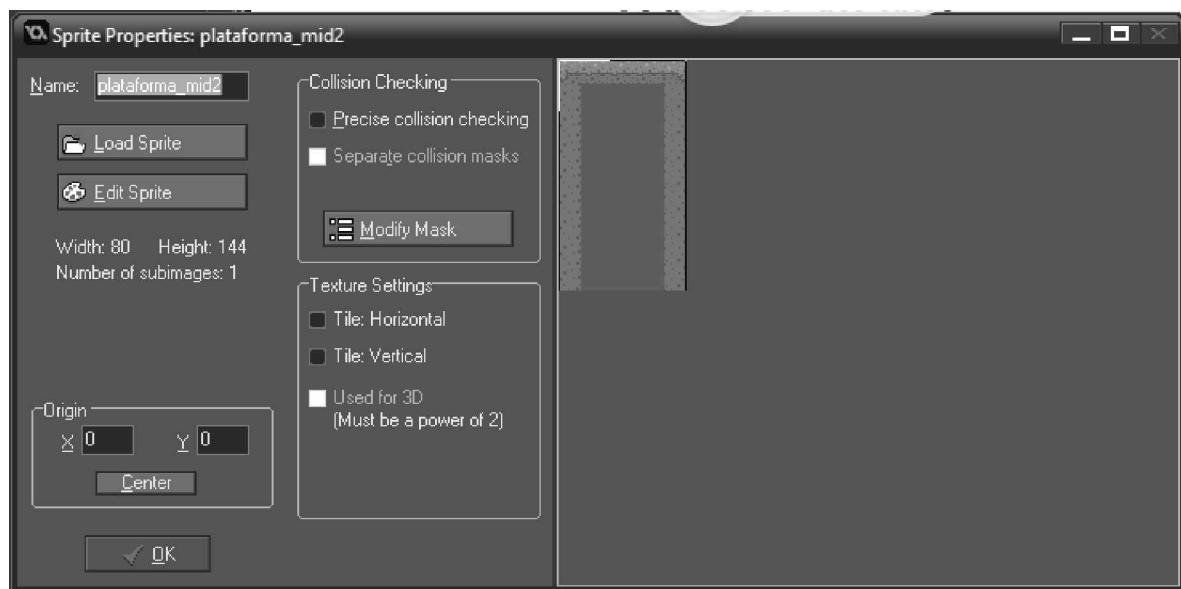


Figura 4.3 – Captura de tela de programação da *sprite* utilizada para as plataformas do jogo.

Controlador

Os demais objetos criados no jogo não são visíveis ao jogador, porém têm papel fundamental para seu correto funcionamento. O controlador do jogo se divide em 2 objetos, sendo um para a sala inicial do jogo, e outro para a sala principal, onde o tema principal acontece. Sua janela de programação é apresentada na Fig. 4.4.

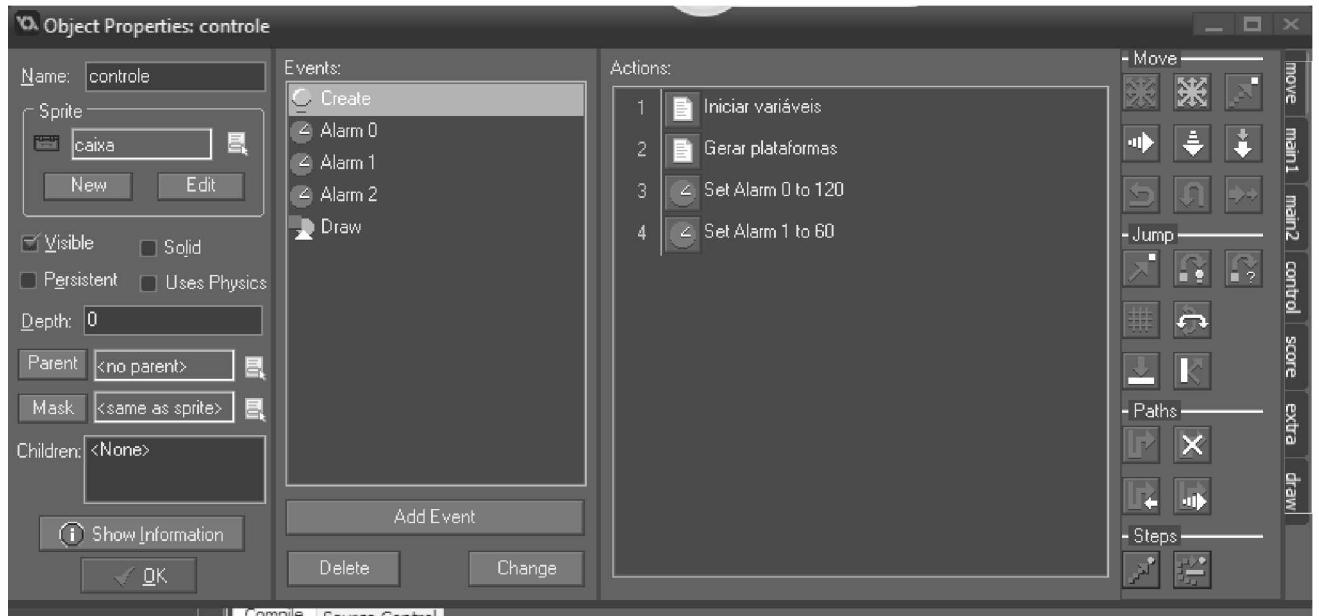


Figura 4.4 – Captura de tela de programação do objeto de controle dos eventos principais das salas do jogo.

Este objeto gerencia eventos de temporizadores e de apresentação de textos do jogo. No passo de sua criação, ele inicia as variáveis necessárias para controlar os textos a serem apresentados; a geração de plataformas; os dois primeiros temporizadores e o registro de melhor pontuação. Para o controle na sala inicial, este objeto também gerencia a música ambiente do jogo.

Para gerar as plataformas, o objeto executa um código que escolhe aleatoriamente uma posição entre 0 e 3, onde 0 corresponde à plataforma mais baixa, e 3 a mais alta. O código também possui um controle baseado nas variáveis definidas inicialmente para garantir que a próxima plataforma esteja ao alcance do personagem, ou seja, a próxima plataforma só poderá estar no máximo um nível acima da anterior. As plataformas são geradas imediatamente após a extremidade direita da sala.

O Primeiro temporizador, definido como "alarme 0", dispara a cada 120 passos, gerando a próxima plataforma. O segundo temporizador, no "alarme 1", é acionado a cada 60 passos e é utilizado apenas para alterar a variável de controle responsável por escrever uma contagem regressiva a partir do início do jogo. O terceiro temporizador é acionado pelo personagem principal quando este cai das plataformas e é utilizado para encerrar a rodada do jogo, registrando o placar do jogador e retornando à sala inicial.

O último evento presente neste objeto, chamado "draw", é responsável por executar continuamente, a cada passo, um código específico para apresentação de texto. Neste código, a variável do alarme 0 é analisada para gerar a contagem regressiva inicial, e em seguida escreve continuamente o placar atual do jogo e a melhor pontuação do jogo, que permanece salva em um arquivo .ini, gerado pelo próprio jogo.

Para o caso do objeto de controle na sala inicial, o evento *draw* é usado para escrever

o título do jogo e a mensagem de pressionar qualquer tecla para iniciar o jogo. Há também um evento de monitoramento do teclado para que, quando qualquer tecla for pressionada, o jogo avance para a próxima sala.

Delimitador de sala

A plataforma de desenvolvimento escolhida permite que os objetos extrapolem os limites impostos para a sala, e ainda oferece um evento disparado quando um objeto específico ultrapassa as dimensões definidas para a sala.

Porém, como as plataformas são geradas fora da sala, e em alguns casos o personagem principal pula acima do limite superior da sala, para controlar a sua queda e a saída das plataformas do cenário, criou-se um objeto chamado "delimitador de sala", cuja única função é permanecer fixo nas extremidades inferior e esquerda da sala para acionar eventos de colisão com outros objetos quando necessário.

Marcador de pontuação

Os pontos do jogo são registrados toda vez que o jogador saltar de uma plataforma a outra com sucesso. Para realizar esta ação, criou-se um objeto invisível, gerado conjuntamente com as plataformas, e posicionado imediatamente acima delas.

Embora não seja visível, ele possui uma *sprite* de uma linha com comprimento igual ao da base das plataformas. Assim, garante-se que ele estará cobrindo a área de contato correta para que o personagem pontue. Na Figura 4.5 está apresentada sua janela de programação, evidenciando o evento de colisão com o personagem.

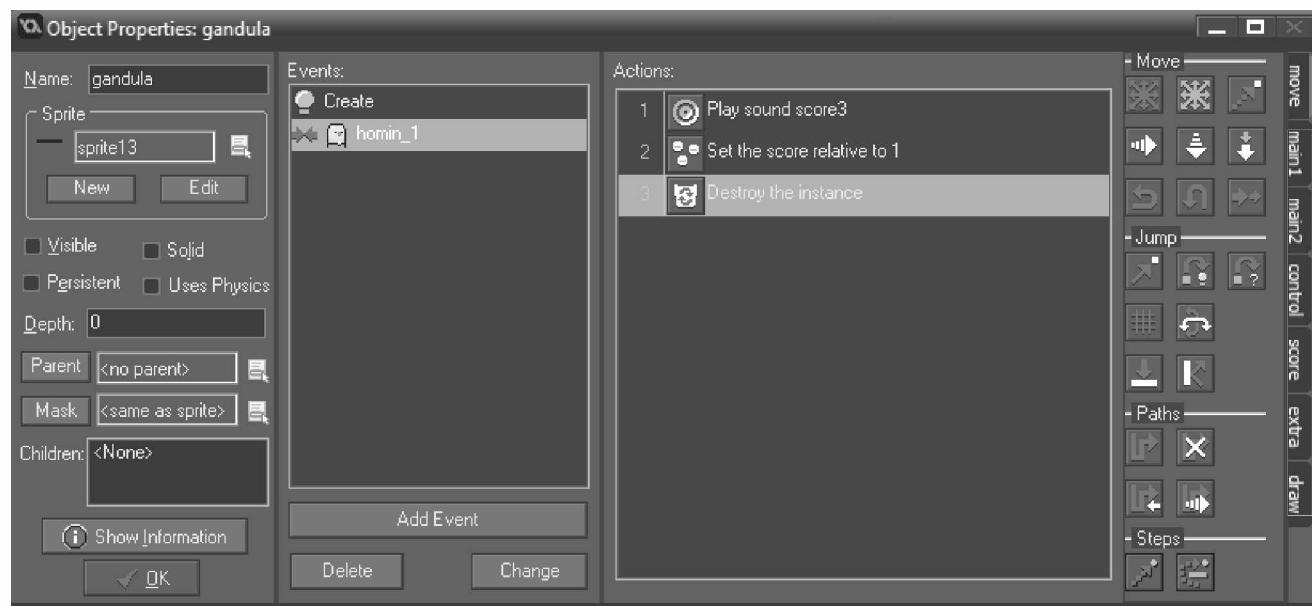


Figura 4.5 – Captura de tela de programação do objeto de marcação de pontos, evidenciando os blocos de programação utilizados no lugar de um código escrito.

Seus eventos gerenciados são para criação, onde inicia movimento idêntico ao das plataformas, e de colisão com o personagem, conforme mencionado acima, onde o objeto acrescenta um ponto ao placar, reproduz um efeito sonoro para pontuação e se autodestrói.

Emissor de partículas

O sistema de desenvolvimento do jogo possui um recurso estético interno usado para gerar partículas aleatórias a partir de um determinado ponto ou região, para tornar as animações do jogo mais dinâmicas e atraentes. Para utilizar esse recurso criou-se um objeto também invisível com o único propósito de gerar partículas simulando a emissão de faíscas.

Esse objeto foi posicionado no centro do placar, e seu evento gerenciado passo-a-passo, monitora a variação da pontuação, fazendo com que a transição de valor para outro no placar seja destacada com uma rajada de partículas. Esta operação é realizada com o auxílio de um código.

Obviamente, o papel do emissor de partículas é meramente estético e não possui papel fundamental no jogo elaborado.

Salas do jogo

Conforme já citado, o jogo possui apenas duas salas, sendo uma inicial, para funcionar como a capa do jogo, e sala principal do jogo, onde o personagem principal controlado pelo jogador é criado. Ambas as salas possuem as dimensões de 720 pixels de comprimento por 280 de altura e estão configuradas para executar 60 passos por segundo. A resolução de posicionamento de objetos das salas é de 16 por 16 pixels.

As salas usam uma imagem cíclica de "pano de fundo", o que significa que a imagem se repete continuamente conforme se movimenta. O "pano de fundo" se move a 1 pixel por segundo, o que gera uma sensação de profundidade por estar se movendo mais lentamente que as plataformas.

Na sala inicial estão previamente posicionados apenas o controlador de início e os delimitadores para colidir com as plataformas ilustrativas geradas. Para a sala principal, inclui-se o personagem principal, o controlador, o emissor de partículas e algumas plataformas são pré-posicionadas para que o jogo inicie suavemente. O construtor da sala principal está ilustrado na Fig. 4.6.

Finalmente, o arquivo executável do jogo é compilado para plataforma Windows. O jogo requer instalação para o sistema operacional e logo em seguida pode ser executado com segurança. A Fig. 4.7 apresenta o resultado final obtido.

Todos os códigos gerados para o jogo e aqui mencionados estão disponíveis nos apêndices deste trabalho, e a Fig. 4.8 apresenta o fluxograma do jogo.

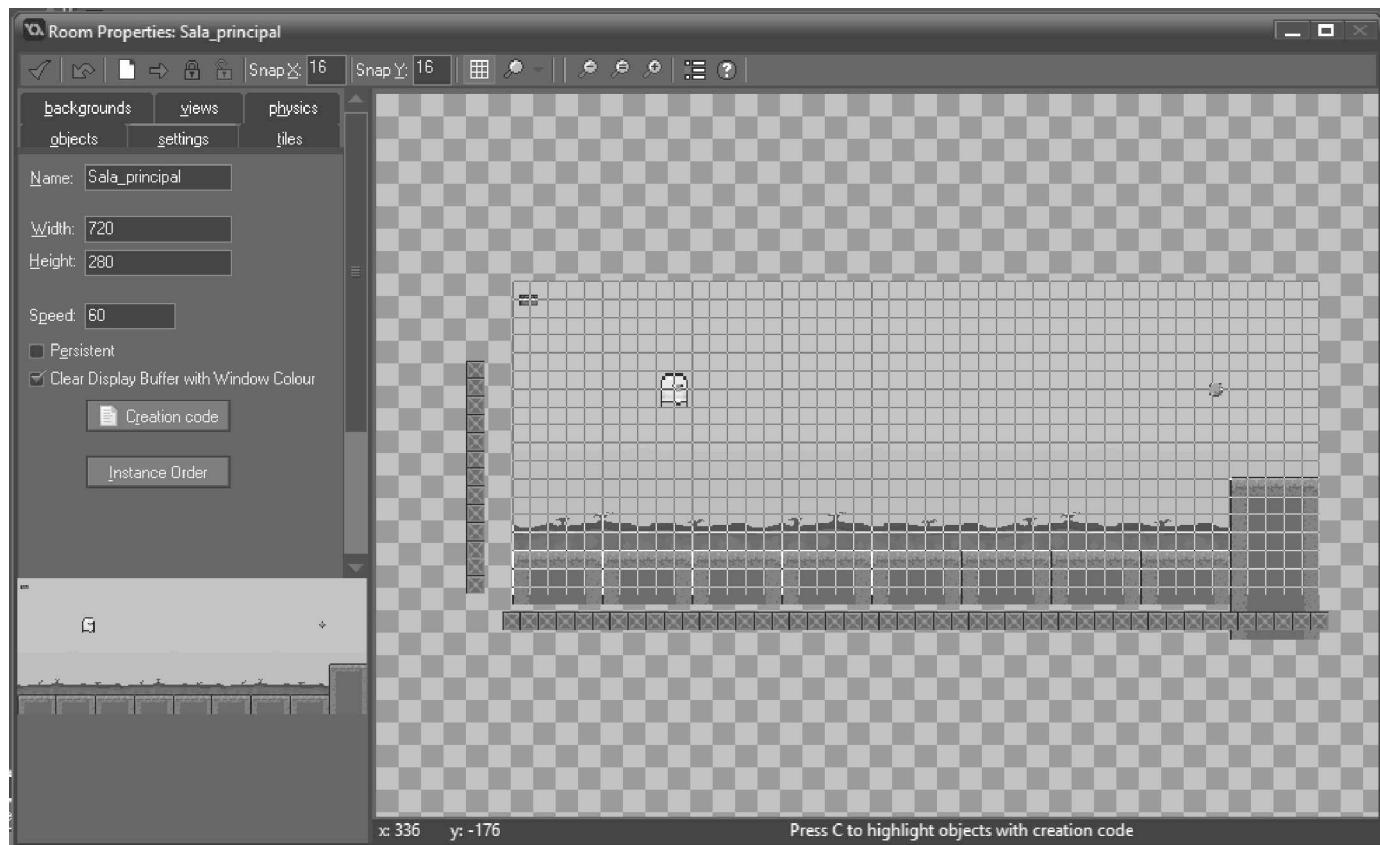


Figura 4.6 – Captura de tela da interface de construção de sala usada na criação da sala principal do jogo.



Figura 4.7 – Captura de tela da execução da versão final do jogo produzido, para operar em conjunto com a estrutura construída.

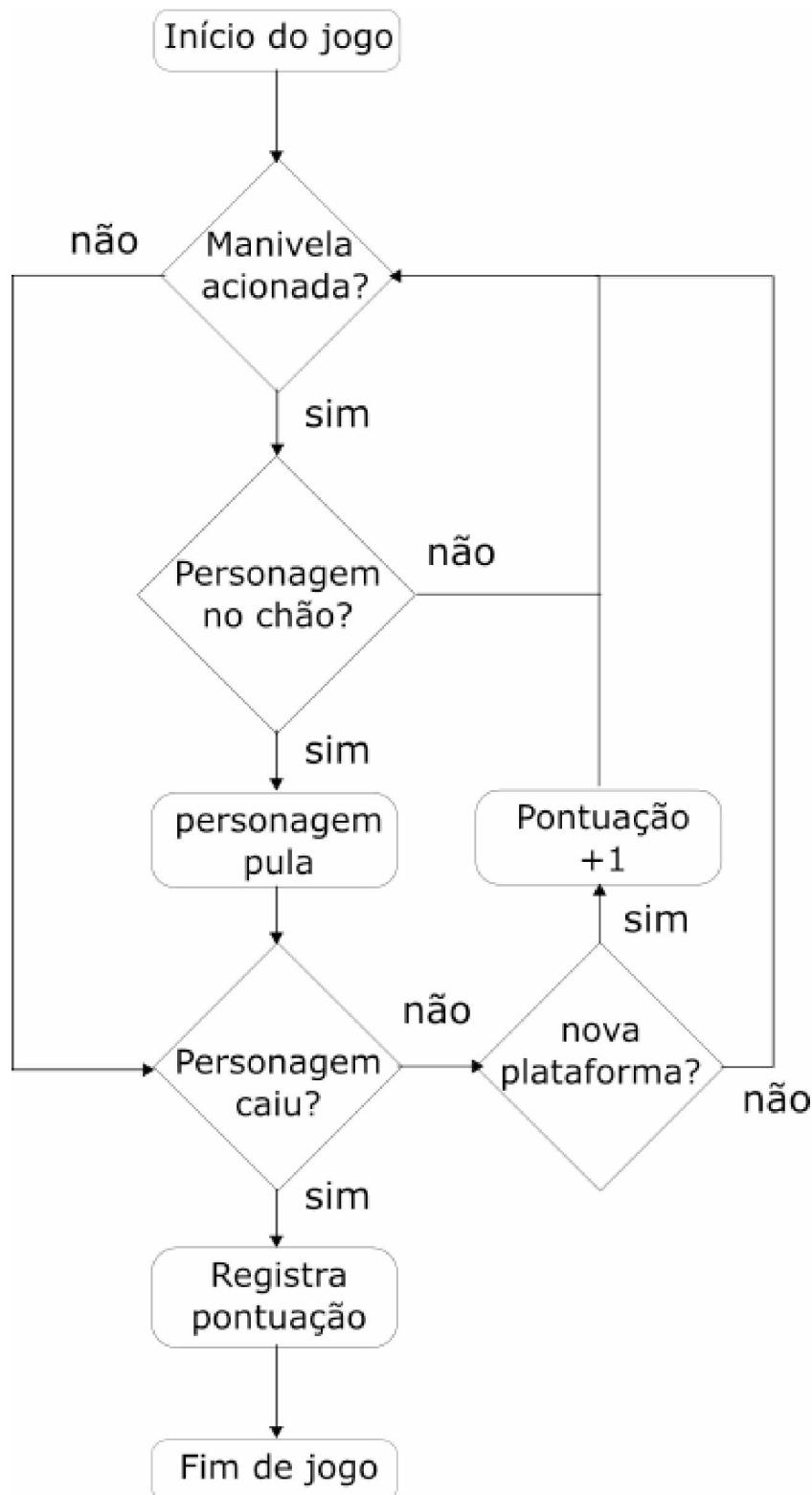


Figura 4.8 – Fluxograma básico da lógica de funcionamento do jogo.

CAPÍTULO V

PROCEDIMENTOS EXPERIMENTAIS

Neste trabalho foram desenvolvidos três mecanismos de 4 barras a serem utilizados em exercícios de reabilitação do tornozelo, joelho e quadril. Conforme apresentado no Capítulo III, os resultados obtidos na modelagem são compatíveis com a amplitude do movimento de flexão-extensão de cada articulação.

Visando incrementar as evidências do correto funcionamento da modelagem matemática, bem como apresentar resultados práticos das aplicações dos mecanismos, um protótipo para o mecanismo do joelho foi construído, o qual foi testado em um boneco de madeira antropométrico no Laboratório de Automação e Robótica da Universidade Federal de Uberlândia.

O protótipo também foi utilizado em conjunto com um jogo eletrônico desenvolvido para funcionar com as estruturas aqui propostas, com o objetivo de melhorar a experiência e a motivação do paciente durante a execução dos exercícios no equipamento.

5.1 Construção do protótipo

O protótipo para o mecanismo do joelho foi construído utilizando os materiais e ferramentas disponíveis no Laboratório de Automação e Robótica da Universidade Federal de Uberlândia, sendo eles:

1. Perfil de alumínio retangular $1'' \times \frac{1}{2}''$;
2. Corrente ASA-40 $\frac{1}{2}''$;
3. Parafusos, porcas e arruelas diversas;
4. Polias dentadas para corrente ASA-40 $\frac{1}{2}''$ de 11 dentes (polia condutora e polia tensora);
5. Polia dentada para corrente ASA-40 $\frac{1}{2}''$ de 33 dentes (polia conduzida).

Para efetuar a montagem do protótipo sobre o boneco no laboratório, permitindo realizar os ensaios, utilizou-se ainda os seguintes equipamentos:

1. Suporte, construído utilizando perfil de alumínio extrudado, para fixação do protótipo;
2. Inclinômetro digital, para medição da amplitude angular do mecanismo;
3. Órtese para perna, pela qual se conecta o mecanismo à perna do boneco de madeira;
4. Câmera digital de alta resolução, para filmagem dos movimentos.

As peças e mecanismos foram construídas a partir do detalhamento e simulações 3D feitos no SolidWorks®. A montagem final está apresentada na Fig. 5.1.

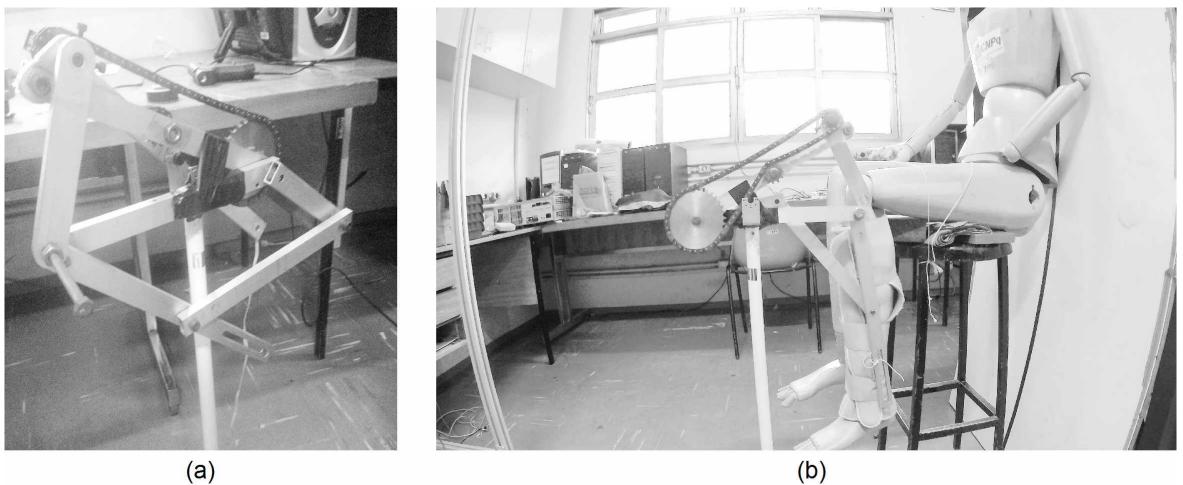


Figura 5.1 – (a) Montagem do protótipo para o joelho realizada no Laboratório de Automação e Robótica da Universidade Federal de Uberlândia, e (b) montagem do protótipo em um boneco de testes.

Para que o movimento possa ser transmitido através do sistema de polias, estas devem estar fixadas aos respectivos parafusos de suas conexões, e estes devem poder girar livremente. Sendo assim, o parafuso conectando as barras a_1 e r_1 deve girar em relação a elas, mas deve estar fixo em relação à polia conduzida e à barra r_2 , para onde o torque deverá ser aplicado.

Destaca-se que na construção do protótipo, foi necessária a inserção de uma polia tensora no centro da barra a_1 , conforme Fig. 5.1, para manter a corrente sempre com a tensão adequada e permitir o correto funcionamento do sistema.

Analogamente, o parafuso conectando à manivela de entrada e à barra a_1 deve estar móvel para a_1 e fixo para a polia condutora e a manivela. O posicionamento correto da montagem destas duas articulações são apresentados na Fig.5.2.

A conexão da barra r_2 com o parafuso, acima citado, e das barras r_2 e r_3 é realizada utilizando uma porca convencional e uma autotrvante. O uso deste tipo de porca é necessário para evitar que a movimentação das barras durante a operação do mecanismo gere folgas entre as barras, ou culmine na desconexão das barras durante a operação. O detalhe na conexão destas barras é destacado na Fig. 5.3(a). O mesmo tipo de conexão é feito nas barras r_3 e r_4 , conforme observado na Fig. 5.3(b).

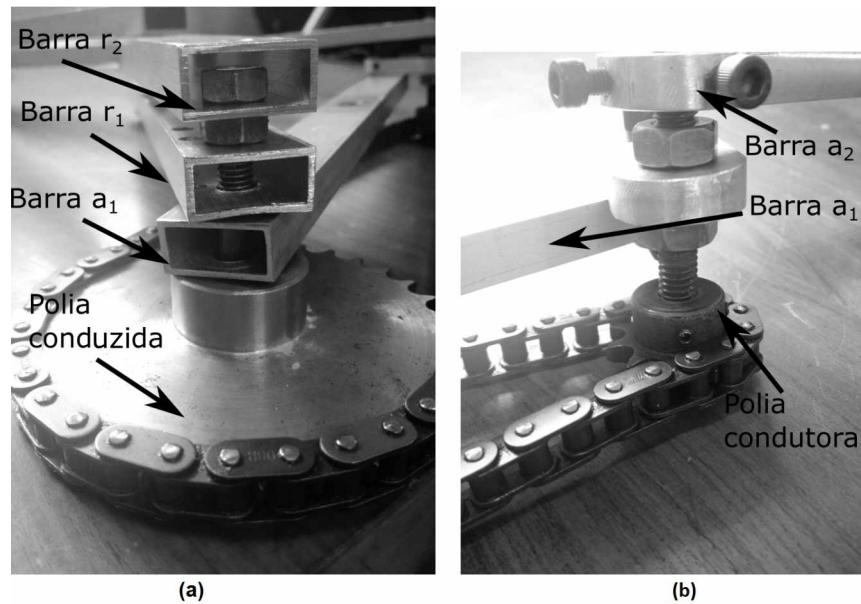


Figura 5.2 – Detalhe na conexão das barras r_1 , r_2 , a_1 e a polia conduzida; e (b) detalhe na conexão da barra a_1 , barra de acionamento a_2 e polia condutora.

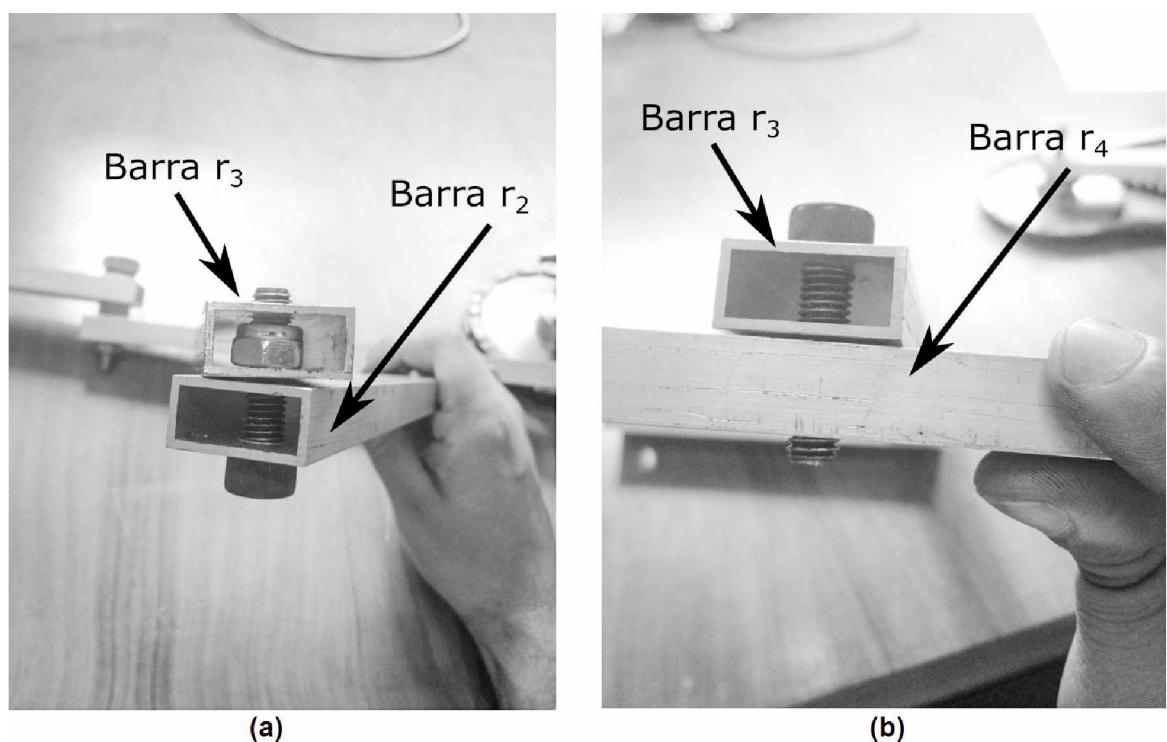


Figura 5.3 – (a) Detalhe na conexão das barras r_2 e r_3 ; e (b) detalhe entre a conexão das barras r_3 e r_4 .

Para compensar o espaçamento gerado entre as barras r_1 e r_4 foi fabricado um pino escalonado para evitar a ocorrência de folgas e/ou desalinhamentos na estrutura. O detalhe da montagem das barras pode ser visto na Fig. 5.4.

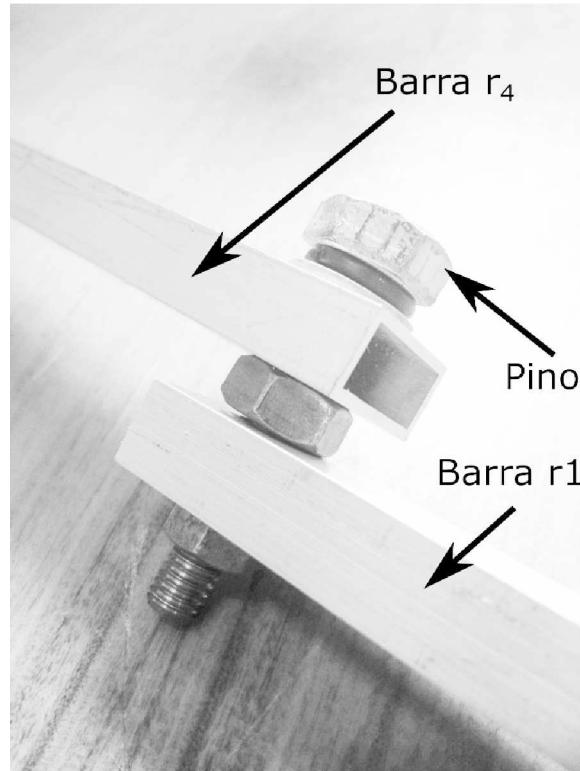


Figura 5.4 – Detalhe na conexão das barras r_1 e r_4 .

5.2 Testes com o boneco de madeira

Após finalizada a construção do protótipo, foram realizados testes com o auxílio de um boneco antropométrico.

Primeiramente, posicionou-se o mecanismo na altura e alinhamento adequados para acioná-lo e movimentar a perna do boneco. A conexão da barra terminal r_4 com a perna foi realizada através de uma órtese, conforme ilustrado na Fig. 5.5.

Realizou-se o teste de movimentação acionando continuamente o mecanismo, manualmente, através da manivela de entrada, enquanto a saída angular obtida foi registrada através da câmera. A sequência deste movimento é apresentada na Fig. 5.6.

Destaca-se que o peso da perna do boneco de madeira utilizado durante os testes é de aproximadamente 6Kg, o que garante que a operação está dentro dos limites previamente estabelecidos na modelagem matemática.

Conforme observado, a saída angular do protótipo realiza o movimento de flexão-extensão no joelho do boneco de madeira de forma natural. Foi possível notar durante a execução dos testes que a redução aplicada através das polias tornou a operação leve e fácil de se realizar.

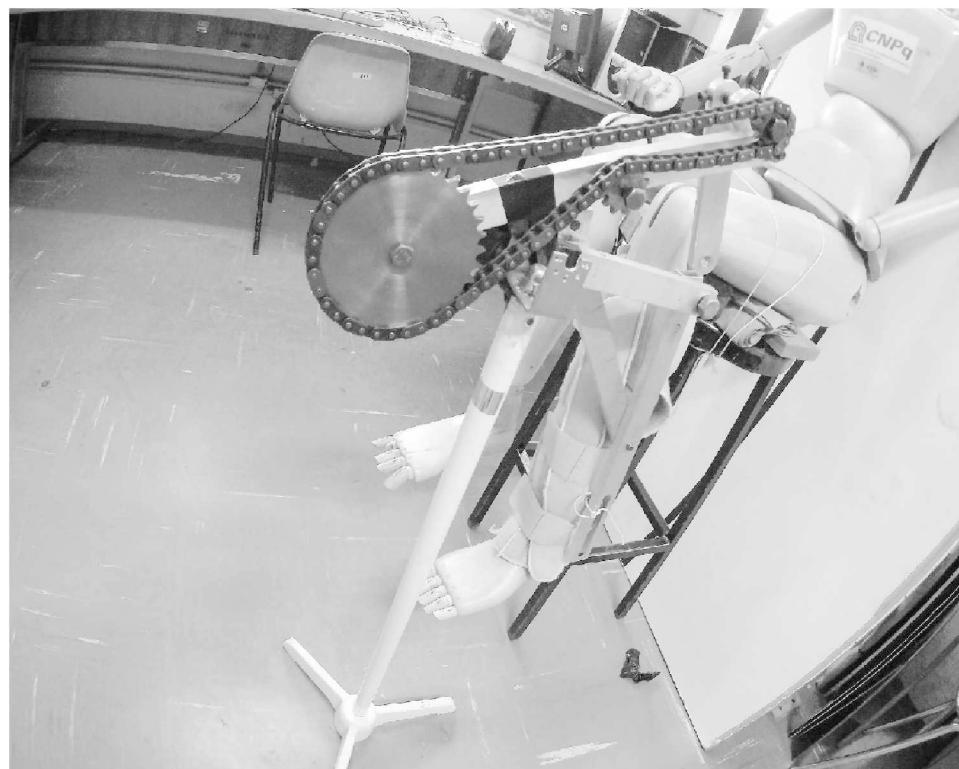


Figura 5.5 – Conexão do protótipo com a perna do boneco de testes através de órtese.

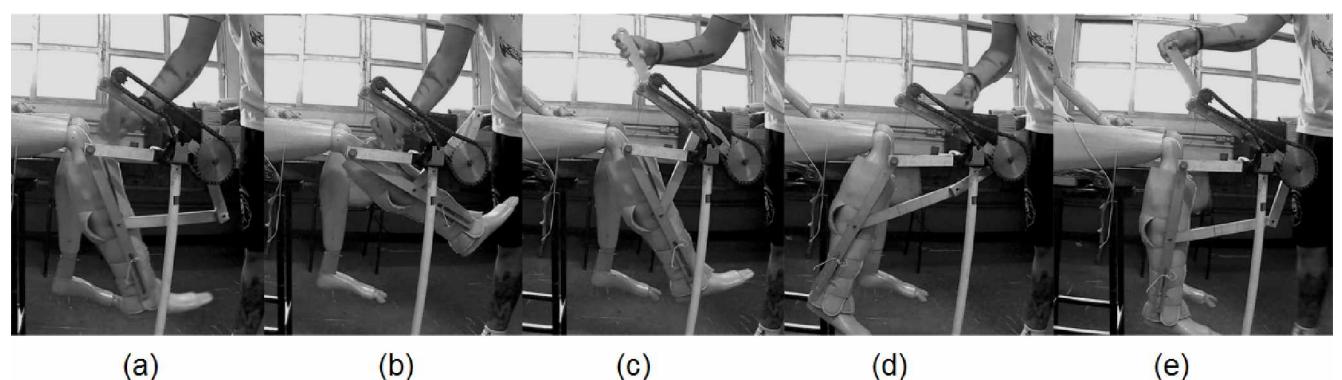


Figura 5.6 – Sequência de movimentos captados da movimentação contínua do protótipo construído: (a,b,c) posição em flexão, (d) posição em extensão, (e) posição anatômica.

O mecanismo foi operado continuamente durante vinte ciclos de movimentação do joelho e manteve a estabilidade do movimento e a fácil operação manual.

Após os testes de movimentação, a amplitude da saída angular do mecanismo foi medida usando o inclinômetro digital posicionado na barra r_4 para averiguar a fidelidade do resultado prático em relação ao modelo matemático. A amplitude de saída é obtida comparando os pontos de máxima extensão e máxima flexão da perna do boneco. Foram realizadas 10 medições de cada ponto e são apresentadas na Tab. 5.1 e analisadas estatisticamente na Tab. 5.2.

Tabela 5.1 – Medições de amplitude de saída do protótipo.

Medição	flexão máx. [°]	Extensão máx. [°]	Diferença [°]	Amp.de saída [°]
1	53.8	17.0	70.8	109.2
2	55.0	18.5	73.5	106.5
3	53.0	17.0	70.0	110.0
4	53.4	15.0	68.4	111.6
5	52.8	17.6	70.4	109.6
6	52.2	18.5	70.7	109.3
7	53.4	17.4	70.8	109.2
8	53.3	18.0	71.3	108.7
9	54.2	18.5	72.5	107.5
10	54.2	19.0	79.2	106.8

Tabela 5.2 – Análise estatística das medições realizadas.

Parâmetro	Valor
média [°]	108.8
desvio padrão [°]	1.6
intervalo de confiança (95%) [°]	107.83-109.85
erro do int. de conf. [%]	10.14-8.46

Na Tabela 5.1 é possível notar que as variações na amplitude de saída do mecanismo são pequenas. Este fato fica mais evidente ao se observar os dados estatísticos das medições na Tab. 5.2, onde, considerando que as medições realizadas seguem uma distribuição normal, para o desvio padrão medido e intervalo de confiança de 95%, obtém-se um erro relativo inferior a 2%.

Esses dados evidenciam uma boa repetibilidade da estrutura, bem como uma boa fidelidade da saída angular experimental com a saída angular obtida no modelo matemático. Dessa forma, fica evidente a capacidade dos mecanismos propostos de realizar os exercícios necessários para uma terapia passiva de movimento contínuo.

5.3 Interface com o protótipo

O jogo desenvolvido deve ser inteiramente operado pelo paciente que estiver utilizando o protótipo construído, de forma que as ações necessárias para que o objetivo do jogo seja

cumprido e estimulem o paciente a realizar o exercício de movimentação.

Para realizar esta interface, foram utilizados os seguintes materiais:

1. Microcontrolador USB atmega Arduino Leonardo;
2. Encoder Rotativo KY-040, com 20 pulsos por volta;
3. Fios e conectores diversos.

O encoder digital foi acoplado diretamente ao eixo da manivela de acionamento do protótipo, de modo que sempre que o paciente a girar, ele irá contar pulsos e enviá-los ao microcontrolador. Uma imagem do encoder utilizado, e a sua montagem na estrutura estão ilustradas na Fig. 5.8.

Seus terminais são conectados ao microcontrolador para que possam ser interfaceadas adequadamente. O esquema de ligação com o Arduino está representado na Fig. 5.7.

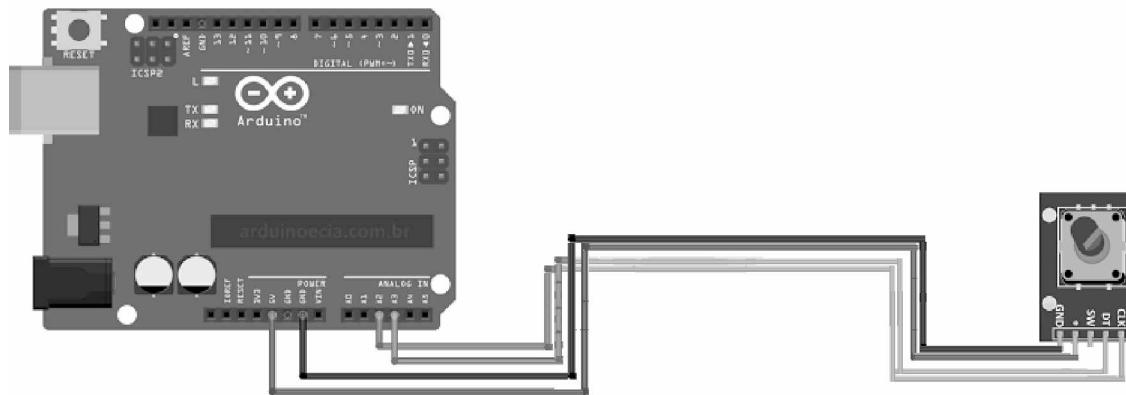


Figura 5.7 – Esquema de conexão utilizado para conectar o encoder ao Arduino (adaptado de: <http://www.arduinoecia.com.br/>).

Uma vez que o jogo desenvolvido aceita apenas entradas feitas por mouse ou teclado, o microcontrolador executa uma rotina para emitir uma entrada de teclado ao computador que, neste caso, é o sinal da barra de espaço, em função da emissão de três pulsos e no sentido horário. A programação necessária para este funcionamento está descrita no Apêndice A.4.

Convencionou-se para o jogo que o sinal para o personagem pular será emitido sempre que três pulsos forem contados na direção horária ao encoder. Dessa forma, garante-se que o paciente jogue apenas com o avanço do mecanismo e deixa aberta a possibilidade de usar o recuo do mecanismo para compensar possíveis limitações da articulação trabalhada.

5.4 Teste de integração jogo-estrutura

O teste de integração com a estrutura foi realizado operando a estrutura diretamente com o jogo ativo, onde o autor jogou durante 20 minutos contínuos. Foram testados aspectos

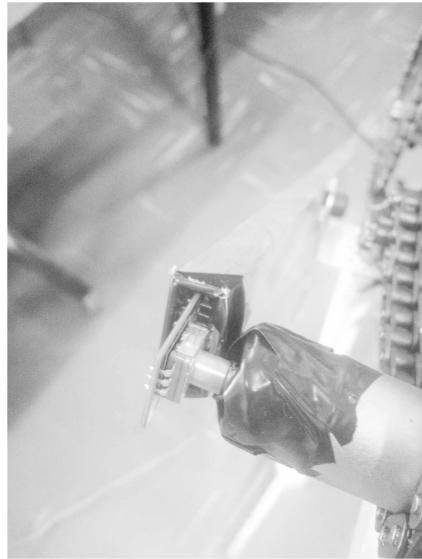


Figura 5.8 – Detalhamento do encoder associado ao protótipo.

como a jogabilidade, facilidade em operar os comandos do jogo, dificuldade e progressão. Uma imagem feita durante a realização do teste é apresentada na Fig. 5.9.

Observou-se que a dificuldade do jogo, apesar de simples, é suficiente para gerar um desafio ao jogador, característica que ajuda a manter o interesse. Este tipo de aspecto já foi explorado em jogos populares para Smartphones como *Flappy Bird* e *Swing Chopters* (disponível em: <http://www.dotgears.com/>).

O teste também demonstrou que a possibilidade de retornar a manivela auxilia a operação dos comandos e a manter o ritmo do jogo e torna a jogabilidade mais dinâmica. A progressão do jogo fica evidenciada com o registro da pontuação máxima do jogador, a qual cresceu conforme o jogador se familiarizou com a mecânica do jogo.



Figura 5.9 – Imagens de um jogador realizando um teste de integração jogo-estrutura.

A importância do correto ajuste nas conexões entre os elementos do mecanismo foi notada durante o teste, pois as folgas presentes na estrutura no momento do teste causaram travamentos e dificuldades na manipulação da manivela de entrada do mecanismo. Após ajustes na tensão dos parafusos, estes problemas foram solucionados.

Nos testes realizados ficou evidenciado a capacidade do mecanismo de realizar os movimentos de flexão-extensão do joelho de forma confiável, validando o modelo matemático

proposto nesta dissertação. O teste de integração também demonstrou que a estrutura interagiu de forma satisfatória com o jogo criado, o que torna os mecanismos propostos possíveis para reabilitação do membro inferior.

CONCLUSÕES

Neste trabalho foram propostas três estruturas não-motorizadas e auto-operadas, baseadas no mecanismo de 4 barras, para reabilitação de cada uma das articulações do membro inferior humano.

Foram apresentados diversos aspectos positivos ligados ao uso de estruturas robóticas no processo de reabilitação de pacientes vítimas de AVC, lesão medular, e outras doenças que provocam deficiências motoras. Também destacou-se as vantagens em se utilizar estruturas auto-operadas pelos pacientes. Uma revisão das estruturas já aplicadas com esta finalidade possibilitou verificar quais os melhores aspectos para serem explorados nos mecanismos desenvolvidos.

A escolha de uma modelagem matemática baseada no equilíbrio estático do mecanismo de quatro barras, combinada com a otimização por meio de um algoritmo genético do tipo evolução diferencial, possibilitou obter estruturas com saídas angulares satisfatoriamente próximas às amplitudes naturais do movimento de flexão-extensão das três articulações do membro inferior.

A análise de tensões realizada na modelagem matemática demonstrou que os mecanismos não sofrem tensões elevadas em suas barras, o que possibilita a escolha de materiais leves, como o alumínio, bem como o uso de perfis de tubos retangulares para os mecanismos do joelho e tornozelo, o que reduz custos de material e favorece a acessibilidade das estruturas.

Através da construção do protótipo para o joelho, foi possível evidenciar a fidelidade do modelo experimental ao esperado no modelo matemático. Os dados coletados apontam uma boa repetibilidade do movimento de saída do mecanismo e baixo erro. Os testes também mostraram que a estrutura pode ser facilmente manuseada por indivíduos saudáveis, sem a necessidade de treinamentos prévios.

O jogo eletrônico foi elaborado e integrado à estrutura, apresentando boa jogabilidade, explorando o aspecto de um jogo desafiador ao usuário, e gerando o sentimento de superação de dificuldades. Devido à simplicidade, o jogo também não requer nenhum conhecimento prévio do jogador para ser operado corretamente.

Com base nos fatos aqui expostos, os mecanismos propostos nesta dissertação podem ser aplicados para terapia passiva de movimento contínuo. Os estudos futuros a serem realizados nesta estrutura devem buscar melhorar o posicionamento das estruturas, bem como os demais aspectos como a especificação de suporte mais adequado, melhorar a forma de se conectar a barra para saída angular à ótese, e encontrar formas de eliminar as folgas recorrentes no protótipo.

Referências Bibliográficas

AKDOGAN, E., ADLI, M.A., **The design and control of a therapeutic exercise robot for lower limb rehabilitation:** Physiotherobot. Mechatronics Journal, 2011, Elsevier, [www.elsevier.com/ locate/mechatronics](http://www.elsevier.com/locate/mechatronics) (acessado em 12/03/2015).

ALTUZARRA, O., PINTO, C., AVILÉS, R., HERNANDÉZ, A., **A Practical Procedure to Analyze Singular Configurations in Closed Kinematic Chains**, IEEE Transactions on Robotics, v. 20, 2004, p. 929-940.

ALTUZARRA, O., SALGADO, O., PETUYA, V., HERNANDÉZ, A., **Point-based Jacobian formulation for computational kinematics of manipulators**, Mechanism and Machine Theory, v. 41, n. 12, 2006, p. 1407-1423.

BATENI, H., MAKI, B.E., **Assistive devices for balance and mobility:** benefits, demands, and adverse consequences. Arch. Phys. Med. Rehabil., v. 86, n. 34), 2005, p. 45.

BOIAN, R.F., BOUZIT, M., BURDEA, G.C., DEUTSCH J.E., Dual **Stewart Platform Mobility Simulator**. Proceedings of the 26th Annual International Conference of the IEEE EMBS, 2004.

BRODELL, R. J., SONI, A. H., **Design of the crank-rocker mechanism with unit time ratio**. Journal of Mechanisms, v.5, 1970, p. 1-4.

BOURI, M., STAUFFER, Y., SCMITT C. et al., **The walktrainer:** a robotic system for walking rehabilitation, Proceedings of the IEEE International Conference on Robotic sand Biomimetics, China, 2006, p. 1616-1621.

CAMARGOS, A.C.R., LACERDA T.T.B., BARROS, T.V., SILA, G.C., PARREIRAS, J.T., VIDAL, T.H.J., **Relação entre independência funcional e qualidade de vida na paralisia cerebral**. Fisioter. Mov., Curitiba, 2012, v. 25, n. 1, p. 83-92.

CHEN, S., WANG, Y., LI, S., HUANG, Y., MAO, X., **Lower Limb Rehabilitation Robot**, Reconfigurable Mechanisms and Robots, ReMAR 2009, ASME/IFTOMM International, 22-24 de Junho de 2009, Londres, p. 439-443.

COSTA, V. S. P., MELO, M. R. A. C., GARANHANI, M. L., FUJISAWA, D.S. **Representações sociais da cadeira de rodas para a pessoa com lesão da medula espinhal**, Rev. Latino-Am. Enfermagem, v.18, n.4, 2010, 8f.

CLAUSER, C.E., McCONVILLE, J.T., YOUNG, J.W., **Weight, volume and center of mass of the human body**, relatório técnico AURL-TR-69-70, Wright Patterson Air Force Base, Ohio, 1969.

DANGELO, J.G. E FATTINI, C.A. **Anatomia Humana Sistêmica e Segmentar**. 3^a ed. São Paulo: Atheneu, 2007.

DÍAZ, I., GIL, J.J., SÁNCHEZ, E., **Lower-Limb Robotic Rehabilitation: Literature Review and Challenges**, Journal of Robotics, 2011, ID: 759764.

FARJATIAN, A. B., NABIAN, M., HOLDEN M., MAVROIDIS C., **Development of 2-DOF Ankle Rehabilitation System**. Northeastern Bioengineering Conference (NEBEC2014), 2014, Boston, MA.

FEYS, H.M., DE WEERDT, W.J., SELZ, B.E., et al., **Effect of a therapeutic intervention for the hemiplegic upper limb in the acute phase after stroke**: A single-blind, randomized, controlled multicenter trial. Stroke. v.29, n.4, 1998, p. 785-792.

FRANCESCHI, A., ANTONELLO, M.G., **Elementos de Máquinas**, Colégio Técnico de Santa Maria, Universidade Federal de Santa Maria, Rede e-Tec Brasil, 2014, 152p.

GALLI, M., CIMOLIN, V., DE PANDIS, M.F., LE PERA, D., SOVA, I., ALBERTINI, G., Stocchi F, FRANCESCHINI, M., **Robot-assisted gait training versus treadmill training in patients with Parkinson's disease: a kinematic evaluation with gait profile score.**, Functional Neurology, v. 31, n. 3, 2016, p.163-170.

GALVEZ, J. A., REINKENSMEYER, D. J., **Robotics for gait training after spinal cord injury**. Topics in Spinal Cord Injury Rehabilitation, v. 11, n. 2, 2005.

GIL-GOMÉZ et al., **Effectiveness of a Wii balance board-based system (eBaViR) for balance rehabilitation: a pilot randomized clinical trial in patients with acquired brain injury**. Journal of Neuroengineering and Rehabilitation, 2011, p. 8-30.

GOFFER, A., **Gait-locomotor apparatus** Número da Patente: 7 153 242, 2006.

GOLDBERG, David E., **Genetic Algorithms in Search, Optimization, and Machine Learning**, EUA: Addison-Wesley, 1989.

HARTENBERG, R., DENAVIT, J., **Kinematic Synthesis of Linkages**. McGraw-Hill Companies, 1964.

- HAUPT, R. L., HAUPT, S. E., **Practical Genetic Algorithms**, Wiley-Interscience Publication, New York, USA ,1998.
- HERNÁNDEZ, A., ALTUZARRA, O., AVILÉS, R., PETUYA, V., **Kinematic analysis of mechanisms via a velocity equation based in a geometric matrix**, Mechanism and Machine Theory, v. 38, 2003, p. 1413-1429.
- HESSE, S. e UHLENBROCK, D., A **mechanized gait trainer for restoration of gait**, Journal of Rehabilitation Research and Development, v. 37, n.6, 2000, p. 701-708.
- JEZERNIK, S., SCHÄRER, R., COLOMBO, G., MORARI, M., **Adaptive robotic rehabilitation of locomotion: a clinical study in spinaly injured individuals**, Spinal Cord, v. 41, 2003, p. 657-666.
- JEZERNIK, S.; COLOMBO, G.; MORARI, M., **Automatic gait pattern adaptation algorithms for rehabilitation with a 4-DOF robotic orthosis**. IEEE Transact. Robot. Autom., v.20, 2004, p. 574-582.
- KAPANDJI, A. I, **Fisiologia Articular Membro Inferior**, 5^a ed., Editorial Médica Panamericana, 2000.
- LLOYD-JONES, D., ADAMS R. J., BROWN, T. M. et al., **Heart disease and stroke statistics2010 update: a report from the American heart association**. Circulation, v. 121,n. 7, 2010, p. e46-e215.
- LOBATO, F. S., STEFFEN Jr, V., OLIVEIRA-LOPES, L. C., **Uma abordagem evolutiva baseada em modelos caóticos de busca associado ao algoritmo de evolução diferencial**, 17º POSMEC - Simpósio do Programa de Pós-graduação em Engenharia Mecânica, 2007, Uberlândia, Brasil.
- LOHSE, K., SHIRZAD, N., VERSTER, A., HODGES, N., MACHIEL van der Loos, H.K., **Video Games and Rehabilitation: Using Design Principles to Enhance Engagement in Physical Therapy**, Journal of neurologic physical therapy: JNPT, v. 37, n. 4, 2013, p.166-175.
- LOPES Jr, G. D., **Desenvolvimento de um exoesqueleto para movimentação/reabilitação de paraplégicos** Dissertação de Mestrado em Engenharia Mecânica - Universidade Federal de Uberlândia, 2015.
- LUM, P., REINKENSMEYER, D., MAHONEY R., RYMER, W. Z., BURGA C., **Robotic Devices for Movement Therapy After Stroke: Current Status and Challenges to Clinical Acceptance**. Top Stroke Rehabil, v. 8, n. 4, 2002, p.40-53.
- MELCONIAN, S., **Elementos de máquinas**,9^a ed., Editora Érica, 2009, 383p.
- MICHALEWICZ, Z., **Genetic Algorithm + Data Structures = Evolution Programs**, 2^a Ed., Springer-Verlag, New York, USA, 1994.

MICHALEWICZ, Z., e FOGEL, D. B., **How to solve it: Modern Heuristics**, 1^a Ed., Springer-Verlag, New York, USA, 2000.

MIREALMAN, A., PATRITTI B. L., BONATO, P., DEUTSCH J. E., **Effects of virtual reality training on gait biomechanics of individuals poststroke**. Journal of Gait & Posture, v. 31, 2010, p. 433-437.

NEUMANN, D. A., **Cinesiologia do aparelho musculoesquelético**: Fundamentos para reabilitação, 2^ª edição, Elsevier Health Sciences, 2010, 743p.

NORTON L. R., **Kinematics and Dynamics of Machinery**. 1^a Ed., McGraw-Hill Companies, 2009, NY 10020, EUA.

PEURALA, S. H., AIRAKSINEN, O., HUUSKONEN P., **Effects of intensive therapy using gait trainer or floor walking exercises early after stroke**, Journal of Rehabilitation Medicine, v. 41, n. 3, 2009, p. 166173.

REZENDE, O. P., PIVETTA, C. S., GRECHI, R., CAMPOS, M. L., BRANDÃO, J. G. T., **Análise Dinâmica Computacional de Mecanismos de 4 Barras**, Nonno Simpósio de Mecânica Computacional, 2010, São Jose Del-Rei, Brasil.

RIBERTO, M., MIYAZAKI, M.H., JUCÁ, S.S.H., LOURENÇO, C., BATTISTELLA, L.R., 2007, **Independência funcional em pessoas com lesões encefálicas adquiridas sob reabilitação ambulatorial**. ACTA FISIATR, v.14, n. 2, 2007, p. 87-94.

RODRIGUES, L. A. O., GONÇALVES, R. S., **Estudo e desenvolvimento de jogos aplicados a reabilitação do corpo humano**. Trabalho de conclusão de curso para graduação em Engenharia Mecatrônica Universidade Federal de Uberlândia, 2014.

SCHMITT, C., M'ETRAILLER, P., AL-KHODAIRY A. et al., **The motion maker: a rehabilitation system combining an orthosis with closed-loop electrical muscle stimulation**, Proceedings of the 8th Vienna International Workshop on Functional Electrical Stimulation, Austria, 2004, p. 117-120.

SOARES JR., G. D. L., **Desenvolvimento de um Exoesqueleto para Movimentação/Reabilitação de Paraplégicos**, 2015, 128f. Dissertação de Mestrado, Universidade Federal de Uberlândia, Uberlândia.

STAUFFER, Y., ALLEMAND, Y., BOURI, M., FOURNIER, J., CLAVEL R., METRAILLER, P., BODARD, R., REYNARD, F., **The WalkTrainera new generation of walking reeducation device combining orthoses and muscle stimulation**. IEEE Trans. Neural Syst. Rehabil, v. 17, n. 1, 2009, p. 3845.

STOLLER, O., DE BRUIN, E.D., SCHINDELHOLZ, M., SCHUSTER-AMFT, C, DE BIE, R.A. HUNT, K.J., **Efficacy of Feedback-Controlled Robotics-Assisted Treadmill Exercise to**

- Improve Cardiovascular Fitness Early After Stroke: A Randomized Controlled Pilot Trial.** Journal of Neurologic Physical Therapy, v. 39, n. 3, 2015, p. 156-165.
- SUI, P., YAO, L., LIN, Z., YAN, H., DAI, J. S., Analysis and synthesis of ankle motion and rehabilitation robots.** Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO 09), 2009, p. 2533-2538.
- TSUGE, B.Y., McCARTHY, J. M., An Adjustable Single Degree-of-Freedom System to Guide Natural Walking Movement for Rehabilitation** Journal of Medical Devices, AMSE, v. 10, 2016.
- UICKER Jr, J. J., PENNOCK, G. R., SHIGLEY, J. E., Theory of Machines and Mechanisms,** d. Oxford University Press, 3^a Ed., 2003, 734p.
- WATERMAN, B. R., OWENS, B. D., DAVEY, S., ZACCHILLI, M. A., BELMONT, P. J., The Epidemiology of Ankle Sprains in the United States.** The Journal of Bone & Joint Surgery, v. 92, n.13, 2010, p. 2279-2284.
- WERNER, C., von FRANKENBERG, S., TERING, T., KONRAD, M., HESSE S., Treadmill training with partial body weight support and an electromechanical gait trainer for restoration of gait in subacute stroke patients: a randomized crossover study,** Stroke, v.33(12), 2002, p.2895-2901.
- WERNIG, A., MULLIER, S., NANASSY, A., CAGOL, E., Laufband therapy based on "rules of spinal locomotion" is effective in spinal cord injured persons.** European Journal of Neuroscience, v. 7, n. 4, 1995, p. 823-829.
- WEST, G. R., Powered gait orthosis and method of utilizing same,** Número da Patente: 6 689 075, 2004.

APÊNDICE A.1

Algoritmo da modelagem matemática dos mecanismos

```
##### Modelagem matemática dos mecanismos de 4 barras #####
clear all
close all
clc
format shortg

%Obs: Os comprimentos de cada elemento dos mecanismos aqui modelados foram
%obtidos através do algoritmo de evolução diferencial, vide anexos.
```

Mecanismo: Tornozelo

```
r = [273; 104; 235; 164]; %[mm]
```

Definição das características do mecanismo

```
%Módulo de Elasticidade - Alumínio
E= 69e9;

%Perfil: tubo retangular (protótipo - boneco)
esp=2; %espessura da barra
a= (1e-3*25.4)*(1e-3*12.7)-(1e-3*(25.4-esp))*(1e-3*(12.7-esp));
I= (1e-3*12.7)*(1e-3*25.4)^3/12-(1e-3*(12.7-esp)*(1e-3*(25.4-esp))^3)/12;

% Número de pontos da análise
ptos = 360;

% Período Angular
theta_2= linspace(0,2*pi,ptos);

% Características Geométricas (vide figura 2.25)
S = sqrt(r(1)^2+r(2)^2-2*r(1)*r(2).*cos(theta_2));
beta = acos((r(1).^2+r(2)^2)./(2*r(1).*S));
phi = acos((r(3).^2+r(4)^2)./(2*r(3).*S));
lambda = acos((r(4).^2+r(3)^2)./(2*r(4).*S));
gama = acos((r(3).^2-S.^2+r(4)^2)./(2*r(3)*r(4)));

% Coordenadas dependentes

%avanço(0 a 180°)
theta_3 = phi(1:end/2)-beta(1:end/2);
theta_4 = pi-lambda(1:end/2)-beta(1:end/2);
%retorno(180 a 360°)
theta_3 = [theta_3 phi(end/2+1:end)+beta(end/2+1:end)];
theta_4 = [theta_4 pi-lambda(end/2+1:end)+beta(end/2+1:end)];

% conferindo a saída angular
figure
th3= radtodeg(theta_3);
th4= radtodeg(theta_4);
plot(1:ptos,th3,'k',1:ptos,th4,'k','LineWidth',2)
grid on
title('saída angular do mecanismo')
xlabel('Posição de \theta_2 [°]')
ylabel('Posição angular [°]')
%legend('\theta_3','\theta_4 (principal)')
```

Posicionamento do sistema de coordenadas

```
% Posição do CM de cada barra

O_1 = r(1)/2;
O_2 = r(2)/2;
O_3 = r(3)/2;
O_4 = r(4)/2;

% Orientações à partir do CM (para calculo dos momentos resultantes)

%barra 2:
r2_x = r(2).*cos(theta_2);
r2_y = r(2).*sin(theta_2);

%barra 3:
r3_x = r(3).*cos(theta_3);
r3_y = r(3).*sin(theta_3);

%barra 4:
r4_x = r(4).*cos(theta_4);
r4_y = r(4).*sin(theta_4);
```

Animação do mecanismo

```
%posição absoluta das articulações fixos:
n1 = [0 0]; %nó 1;
n4 = [r(1) 0]; %nó 4;
figure
%set(gcf,'NextPlot','replaceChildren');
Vid(ptos) = struct('cdata',[],'colormap',[]);
for i=1:ptos
    %posição absoluta das articulações móveis:
    n2 = [r2_x(i) r2_y(i)]; %nó 2;
    n3 = [n2(1)+r3_x(i) n2(2)+r3_y(i)]; %nó 3;

    plot([0 r(1)], [0 0], 'k',[n1(1) n2(1)], [n1(2) n2(2)], 'r-o', [n2(1) n3(1)], [n2(2) n3(2)], 'b-o',...
        [n3(1) n4(1)], [n3(2) n4(2)], 'g-o', 'LineWidth',2)
    grid on
    axis([-200 500 -200 200])
    title('Animação do mov. (tornozelo)', 'FontWeight', 'bold', 'FontSize', 16)
    Vid(i)=getframe;
end
```

Análise Estática

```
%Carregamentos:
%peso do individuo: 158 kg (Obesidade 46,3 IMC)
%altura referencial: 1,80m
%relação da massa: membro inferior = 1,5
%massa do carregamento: 2,25 kg
%fator de ponderação: 1,5
%direção da gravidade: 0°

re_x=2,25*9,81*1,5;
re_y=0;

re=0,8*r(4);
re_x=0,8*r(4)_x;
re_y=0,8*r(4)_y;

%zeros(9,360);

%calculo da matriz de esforços

for i=1:ptos

    A= [-1 0 1 0 0 0 0 0 0;
        0 -1 0 1 0 0 0 0 0;
        0 0 -1e-3*r2_x(i) 1e-3*r2_x(i) 0 0 0 0 1;
        0 0 1 0 1 0 0 0 0;
        0 0 0 1 0 1 0 0 0;
        0 0 0 0 -1e-3*r3_y(i) 1e-3*r3_x(i) 0 0 0;
        0 0 0 0 1 0 -1 0 0;
        0 0 0 0 0 1 0 -1 0;
        0 0 0 0 -1e-3*r4_x(i) 1e-3*r4_x(i) 0 0 0];

    B = [0 0 0 0 0 -re_x -re_y -1e-3*re_x(i)*re_y+1e-3*re_y(i)*re_x];

    F(:,i) = A\B;
end
```

Metodologia para singularidades de Altuzarra et al. (2004)

```
sing_I=0; %singularidades de entrada
inputs1;%theta_2(x1 e y1)
sing_O=0;%singularidades de saída
outputs1;%theta_4(x2 e y2)
ND= [];

for i=1:ptos
    %matriz geométrica G:
    G=[cos(theta_2(i))^2*cos(theta_3(i))*sin(theta_2(i))*cos(theta_3(i))*sin(theta_3(i)) -cos(theta_2(i))^2*cos(theta_3(i))*sin(theta_3(i)) 0 0 0 0;
    cos(theta_2(i))*sin(theta_2(i))*cos(theta_3(i))*sin(theta_3(i)) sin(theta_2(i))^2+sin(theta_2(i))^2*cos(theta_2(i))*sin(theta_2(i)) -sin(theta_2(i))^2 0 0 0 0;
    -cos(theta_3(i))*cos(theta_3(i))*sin(theta_3(i))*cos(theta_4(i)) 2*cos(theta_3(i))*cos(theta_4(i))*sin(theta_4(i))+cos(theta_3(i))*sin(theta_3(i)) 0 0 0 0;
    -cos(theta_3(i))*sin(theta_3(i)) -sin(theta_3(i))*cos(theta_4(i))*sin(theta_4(i))+cos(theta_3(i))*sin(theta_3(i)) sin(theta_4(i))^2+sin(theta_3(i))^2 0 0 0 0;
    0 0 0 0 1 0 0 0;
    0 0 0 0 1 0 0 0;
    0 0 0 0 0 1 0 0;
    0 0 0 0 0 0 1];
    %Problema de autovalores:
    [V,D]= eig(G);
    L=0;
```

```

%Considerando os termos beta unitários;
%Localizando os autovetores com autovalor nulo
for j=1:length(D)
    if(D(j,j)<1e-12)
        L=L+1;
        D(j,j)=0;
        Vb(:,L)=V(:,j);
    else
        NZD(i,j-1)= D(j,j);
    end
end

%Análise de entradas:

if(rank(Vb(1:2,1))<inputs)
    sing_I=sing_I+1;
end

%Análise de saídas:

%1 e 2:
if(rank(Vb(3:4,1))<outputs)
    sing_O=sing_O+1;
end

end

figure
subplot(2,1,1)
plot(1:ptos,NZD(:,1),'k-')
grid on
title('Análise de mobilidade instantânea')
xlabel('ângulo \theta_2[°]')
ylabel('1º autovalor (não nulo)')
axis([0 400 0 1])
subplot(2,1,2)
plot(1:ptos,NZD(:,7),'k-')
grid on
xlabel('ângulo \theta_2[°]')
ylabel('7º autovalor (não nulo)')
axis([0 400 0 3])

%análise final de singularidades
flag=0;
disp("ANÁLISE DE MOBILIDADE ")
disp(" ")
disp("Presença de singularidades:")
if(sing_I>0)
    disp('Presença de redundância de entrada!')
    flag=1;
end

if(sing_O>0)
    disp('Presença de redundância de saída!')
    flag=1;
end

if(min(min(abs(NZD))<1e-12))
    disp('Presença de incremento instantâneo de mobilidade!')
    flag=1;
end

if(flag==0)
    disp('Nenhuma singularidade detectada!')
end

```

Cálculo das tensões

```
%cálculo das tensões
sigma_2= (F(1,:).*cos(theta_2)+F(2,:).*sin(theta_2))/a;
sigma_3= (F(3,:).*cos(theta_3)+F(4,:).*sin(theta_3))/a;
sigma_4= (F(5,:).*cos(gama)+F(7,:).*sin(gama)+Fe_y.*sin(theta_4)+Fe_x.*cos(theta_4))/a;
sigma_1= (F(7,:).*cos(theta_4)+F(8,:).*sin(theta_4))/a;

%cálculo dos momentos
sigma_M2=F(9,:)*1e-6*25.4/(2*I);
sigma_M3= 0;
sigma_M4=(Fe_x.*sin(theta_4)+Fe_y.*cos(theta_4))*re*1e-6*25.4/(2*I);
sigma_M1= 0;

%Total:
sigma1 = (sigma_1+ sigma_M1)*1e-6;
sigma2 = (sigma_2+ sigma_M2)*1e-6;
sigma3 = (sigma_3+ sigma_M3)*1e-6;
sigma4 = (sigma_4+ sigma_M4)*1e-6;
```

plotagens - tensões

```
%Barra 1
figure
plot(1:ptos,2.*sigma1,'k','LineWidth',2)
title('Tensão \sigma_{r_1}', 'FontWeight', 'bold', 'FontSize',16)
xlabel('ângulo \theta_{r_1}[^{\circ}]')
ylabel('Tensão [MPa]')
grid on

%Barra 2
figure
plot(1:ptos,2.*sigma2,'k','LineWidth',2)
title('Tensão \sigma_{r_2}', 'FontWeight', 'bold', 'FontSize',16)
xlabel('ângulo \theta_{r_2}[^{\circ}]')
ylabel('Tensão [MPa]')
grid on

%Barra 3
figure
plot(1:ptos,2.*sigma3,'k','LineWidth',2)
title('Tensão \sigma_{r_3}', 'FontWeight', 'bold', 'FontSize',16)
xlabel('ângulo \theta_{r_3}[^{\circ}]')
ylabel('Tensão [MPa]')
grid on

%Barra 4
figure
plot(1:ptos,2.*sigma4,'k','LineWidth',2)
title('Tensão \sigma_{r_4}', 'FontWeight', 'bold', 'FontSize',16)
xlabel('ângulo \theta_{r_4}[^{\circ}]')
ylabel('Tensão [MPa]')
grid on

%Torque
figure
plot(1:ptos,F(9,:),'k','LineWidth',2)
title('Torque', 'FontWeight', 'bold', 'FontSize',16)
xlabel('ângulo \theta_{r_4}[^{\circ}]')
ylabel('Torque [Nm]')
grid on
```

Resultados

```
disp(' ')
disp('ANÁLISE ESTÁTICA')
disp(' ')

disp(['saída angular: ' num2str(max(th4)-min(th4)) '°'])
disp(' ')

disp('Maiores tensões calculadas: ')
disp(['Barra 1: ' num2str(max(abs(sigma1))) ' [MPa]' '(segurança: ' num2str(2*max(abs(sigma1))) ' [MPa])'])
disp(['Barra 2: ' num2str(max(abs(sigma2))) ' [MPa]' '(segurança: ' num2str(2*max(abs(sigma2))) ' [MPa])'])
disp(['Barra 3: ' num2str(max(abs(sigma3))) ' [MPa]' '(segurança: ' num2str(2*max(abs(sigma3))) ' [MPa])'])
disp(['Barra 4: ' num2str(max(abs(sigma4))) ' [MPa]' '(segurança: ' num2str(2*max(abs(sigma4))) ' [MPa])'])
disp(' ')

Tmax=max(abs(F(9,:)));
disp(['Maior torque de entrada calculado: ' num2str(Tmax) ' [Nm]'])
disp('Torque máximo do braço humano (Masculino): 725 [Kgf]/ 71 [Nm]')
disp('Torque máximo do braço humano (Feminino): 336 [Kgf]/ 33 [Nm]')
disp('Torques ponderados (fator 6): 56[Nm] (M)/ 5.5 [Nm]')
disp(['Redução sugerida para torque ponderado: ' num2str(max(1,Tmax/5.5))'])
```

APÊNDICE A.2

Funções para otimizador por evolução diferencial

A.2.1 Mecanismos

```
function Ft = barr_eval3(r,y)

%saída angular desejada:
y=120;
h7=0;

%Módulo de Elasticidade - Alumínio
E= 69e9;

%Perfil: tubo retangular

a= (1e-3*50.8)*(1e-3*25.4);
I= (1e-3*25.4)*(1e-3*50.8)^3/12;
```

Definição das características do mecanismo principal

```
% Número de pontos da análise
ptos = 360;

% Período Angular
theta_2= linspace(0,2*pi,ptos);

% Características Geométricas (vide figura XXXX)

S = sqrt(r(1)^2+r(2)^2-2*r(1)*r(2).*cos(theta_2));
beta = acos((r(1).^2+r(2).^2-r(3).^2)./(2*r(1).*r(2)));
phi = acos((r(3).^2+r(4).^2-r(1).^2)./(2*r(3).*r(4)));
lambda = acos((r(4).^2+r(1).^2-r(2).^2)./(2*r(4).*r(1)));
gamma = acos((r(3).^2+r(4).^2-r(2).^2)./(2*r(3).*r(4)));

% Coordenadas dependentes

%avanço(0 a 180°)
theta_3 = phi(1:end/2)-beta(1:end/2);
theta_4 = pi-lambda(1:end/2)-beta(1:end/2);
%retorno(180 a 360°)
theta_3 = [theta_3 phi(end/2+1:end)+beta(end/2+1:end)];
theta_4 = [theta_4 pi-lambda(end/2+1:end)+beta(end/2+1:end)];
```

Posicionamento do sistema de coordenadas

```
% Posição do CM de cada barra

Rcm2 = r(2)/2;
Rcm3 = r(3)/2;
Rcm4 = r(4)/2;

% Orientações à partir do CM (para calculo dos momentos resultantes)

%barra 2:
R12_x = -Rcm2.*cos(theta_2);
R12_y = -Rcm2.*sin(theta_2);
R32_x = r(2).*cos(theta_2) -Rcm2.*cos(theta_2);
R32_y = r(2).*sin(theta_2) -Rcm2.*sin(theta_2);

%barra 3:
R23_x = -Rcm3.*cos(theta_3);
R23_y = -Rcm3.*sin(theta_3);
R43_x = r(3).*cos(theta_3) -Rcm3.*cos(theta_3);
R43_y = r(3).*sin(theta_3) -Rcm3.*sin(theta_3);

%barra 4:
R14_x = -Rcm4.*cos(theta_4);
R14_y = -Rcm4.*sin(theta_4);
R34_x = r(4).*cos(theta_4) -Rcm4.*cos(theta_4);
R34_y = r(4).*sin(theta_4) -Rcm4.*sin(theta_4);
```

Análise Estática

```
%Carregamentos:
%direção da gravidade:
dir_g=degtorad(-50);

%massa do pé:
m= 90*0.1610;

Fp_x=m*9.81*cos(dir_g);
Fp_y=-m*9.81*sin(dir_g);

Rp_x=R14_x;
Rp_y=R14_y;

F=zeros(9,360);

%Calculo da matriz de esforços

for i=1:ptos

A= [1 0 1 0 0 0 0 0 0;
     0 1 0 1 0 0 0 0 0;
     -1e-3*R12_y(i) 1e-3*R12_x(i) -1e-3*R32_y(i) -1e-3*R32_x(i) 0 0 0 0 1;
     0 0 -1 0 1 0 0 0 0;
     0 0 0 -1 0 1 0 0 0;
     0 0 1e-3*R23_y(i) -1e-3*R23_x(i) -1e-3*R43_y(i) 1e-3*R43_x(i) 0 0 0;
     0 0 0 0 -1 0 1 0 0;
     0 0 0 0 -1 0 1 0;
     0 0 0 0 1e-3*R34_y(i) -1e-3*R34_x(i) -1e-3*R14_y(i) 1e-3*R14_x(i) 0];
B = [0 0 0 0 Fp_x Fp_y 1e-3*Rp_x(i)*Fp_y 1e-3*Rp_y(i)*Fp_x]';

F(:,i) =A\B;
end
```

Análise das tensões - tração/compressão

```
%cálculo das tensões

sigma_2= (F(1,:).*cos(theta_2)+F(2,:).*sin(theta_2))/a;
sigma_3= (F(3,:).*cos(theta_3)+F(4,:).*sin(theta_3))/a;
sigma_4= (F(6,:).*cos(theta_4)+F(7,:).*sin(theta_4))/a;

%cálculo dos momentos

sigma_M2=((F(1,:).*sin(theta_2)+F(2,:).*cos(theta_2))*Rcm2+F(9,:))*1e-6*25.4/(2*I);
sigma_M3=(F(3,:).*sin(theta_3)+F(4,:).*cos(theta_3))*Rcm3*1e-6*25.4/(2*I);
sigma_M4=(F(5,:).*sin(theta_4)+F(6,:).*cos(theta_4))*Rcm4*1e-6*25.4/(2*I);

%Total:

sigma2 = (sigma_2+ sigma_M2)*1e-6;
sigma3 = (sigma_3+ sigma_M3)*1e-6;
sigma4 = (sigma_4+ sigma_M4)*1e-6;
```

Análise de singularidades

```
sing_I=0; %singularidades de entrada
inputs=2; %x2 e y2
sing_O=0; %singularidades de saída
outputs=2;%x3 e y3
for i=1:ptos
    %matriz geométrica G:
    G= [cos(theta_3(i))^2 cos(theta_3(i))*sin(theta_3(i)) -cos(theta_3(i))^2 -cos(theta_3(i))*sin(theta_3(i));
        cos(theta_3(i))*sin(theta_3(i)) sin(theta_3(i))^2 -cos(theta_3(i))*sin(theta_3(i)) -sin(theta_3(i))^2;
        -cos(theta_3(i))^2 -cos(theta_3(i))*sin(theta_3(i)) cos(theta_3(i))^2 cos(theta_3(i))*sin(theta_3(i));
        -cos(theta_3(i))*sin(theta_3(i)) -sin(theta_3(i))^2 cos(theta_3(i))*sin(theta_3(i)) sin(theta_3(i))^2];

    %Problema de autovalores:
    if (max(max(Inf==G))||max(max(isnan(G))))
        h7=1e5;
        break;
    end
    [V,D]= eig(G);
    L=0;

    %Considerando os termos beta unitários;
    %Localizando os autovetores com autovalor nulo
    for j=1:length(D)
        if(D(j,j)<=1e-12)
            if(D(j,j)<=1e-12)
                L=L+1;
                D(j,j)=0;
                Vb(:,L)=V(:,j);
            end
        end
        if L<3
            h7=1e5;
            break;
        end
    %Análise de entradas:

    %combinando os autovetores
    %1 e 2:
    if(rank(Vb(1:2,1:2))<inputs)
        h7=1e5;
        break;
    end
```

```

%restrições de igualdade:
h1= abs(max(imag(gama)));
h2= abs((max(theta_4)-min(theta_4))/degtorad(y)-1);
h3= abs((max(gama)-min(gama))/degtorad(120)-1);
h4= abs(max(abs(2*R32_y+2*R43_y+2*R14_y)));
h5= abs(max(abs(2*R32_x+2*R43_x+2*R14_x))/r(1)-1);

%restrições de desigualdade:
g1 = -min(gama)/degtorad(30)+1;

%função objetivo:
FO= max([ max(abs(sigma2)) max(abs(sigma3)) max(abs(sigma4)) ]);

%fator Rp:
Rp=1e2;

%função pseudo-objetivo:
Ft = FO + Rp*(max(1e-10,h1)^2+1e3*max(1e-10,h2)^2+max(1e-10,h3)^2+...
max(1e-10,h4)^2+max(1e-10,h5)^2+max(1e-10,h7)^2+max(1e-10,g1)^2);

```

```
end
```

A.2.2 Polia

```

function F = polia_eval(r,y)
    %função objetivo:
    FO= round(pi*(r(1)+r(2))+2*sqrt(y(1)^2-(r(2)-r(1))^2));

    %restrições de igualdade:
    h1=r(2)/r(1)-2;
    h2=mod(FO,y(2));
    h3=imag(FO);

    %fator Rp:
    Rp=1e3;

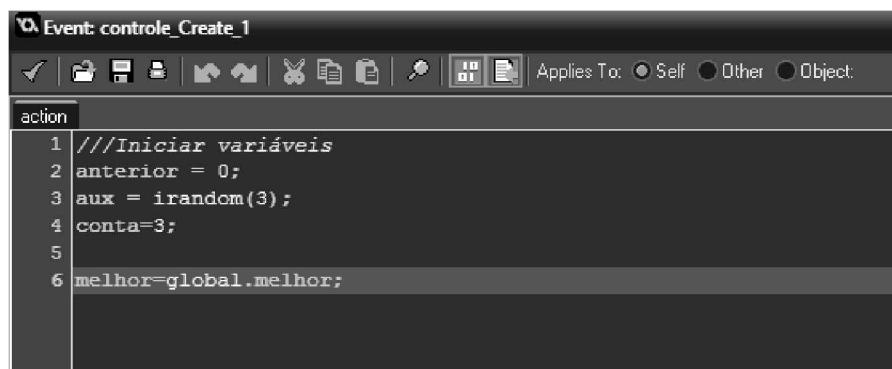
    %função pseudo-objetivo:
    F = FO + Rp*(max(1e-10,h1)^2+max(1e-10,h2)^2+max(1e-10,h3)^2);
end

```

APÊNDICE A.3

Scripts aplicados no jogo

A.3.1 Controle



The screenshot shows a game engine's script editor interface. The title bar says 'Event: controle_Create_1'. The toolbar includes icons for file operations (New, Open, Save, etc.) and a 'Script' icon. Below the toolbar, there are buttons for 'Applies To: Self', 'Other', and 'Object'. The main area is a code editor with a table structure. The first row of the table is labeled 'action'. The second row contains the following C# code:

```
1 //Iniciar variáveis
2 anterior = 0;
3 aux = irandom(3);
4 conta=3;
5
6 melhor=global.melhor;
```

Event: controle_Create_2

Applies To: Self Other Object

```

action
1 //Gerar plataformas
2 //BOT
3 if (aux == 0)
4 {
5     instance_create(720,240,plat_bot);
6     anterior = 0;
7 }
8
9 //MID2
10 if (aux == 1)
11 {
12     instance_create(720,176,plat_mid2);
13     anterior = 1;
14 }
15
16 //MID3
17 if (aux == 2)
18 {
19     if(anterior == 0)
20     {
21         aux2= irandom(1)
22         if(aux2==0)
23         {
24             instance_create(720,240,plat_bot);
25             anterior = 0;
26         }
27         else
28         {
29             instance_create(720,176,plat_mid2);
30             anterior = 1;
31         }
32     }
33     else
34     {
35         instance_create(720,112,plat_mid3);
36         anterior = 2;
37     }
38 }
39
40 //TOP
41 if (aux == 3)
42 {
43     if(anterior == 0)
44     {
45         aux2= irandom(1)
46         if(aux2==0)
47         {
48             instance_create(720,240,plat_bot);
49             anterior = 0;
50         }
51         else
52         {
53             instance_create(720,176,plat_mid2);
54             anterior = 1;

```

```
55     }
56 }
57 else
58 {
59     if(anterior == 1)
60     {
61         aux2= irandom(1)
62         if(aux2==0)
63         {
64             instance_create(720,176,plat_mid2);
65             anterior = 1;
66         }
67         else
68         {
69             instance_create(720,112,plat_mid3);
70             anterior = 2;
71         }
72     }
73     else
74     {
75         instance_create(720,48,plat_top);
76         anterior = 3;
77     }
78 }
79
80 }
81
82 //Novo Roll Up
83 aux = irandom(3);
```

Event: controle_Alarm 0_1

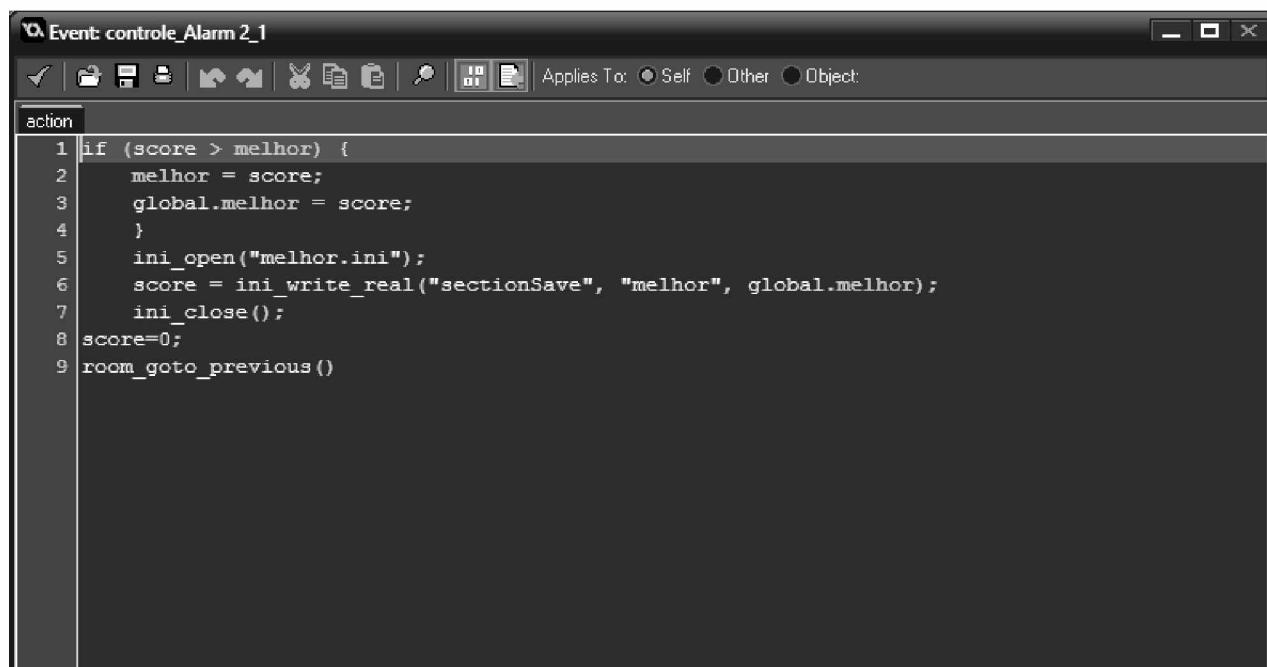
action

```
1 //Gerar plataformas
2 //BOT
3 if (aux == 0)
4 {
5     instance_create(720,240,plat_bot);
6     instance_create(720,240,gandula);
7     anterior = 0;
8 }
9
10 //MID2
11 if (aux == 1)
12 {
13     instance_create(720,176,plat_mid2);
14     instance_create(720,176,gandula);
15     anterior = 1;
16 }
17
18 //MID3
19 if (aux == 2)
20 {
21     if(anterior == 0)
22     {
23         aux2= irandom(1)
24         if(aux2==0)
25         {
26             instance_create(720,240,plat_bot);
27             instance_create(720,240,gandula);
28             anterior = 0;
29         }
30         else
31         {
32             instance_create(720,176,plat_mid2);
33             instance_create(720,176,gandula);
34             anterior = 1;
35         }
36     }
37     else
38     {
39         instance_create(720,112,plat_mid3);
40         instance_create(720,112,gandula);
41         anterior = 2;
42     }
43 }
44
45 //TOP
46 if (aux == 3)
47 {
48     if(anterior == 0)
49     {
50         aux2= irandom(1)
51         if(aux2==0)
52         {
```

```

53         instance_create(720,240,plat_bot);
54         instance_create(720,240,gandula);
55         anterior = 0;
56     }
57     else
58     {
59         instance_create(720,176,plat_mid2);
60         instance_create(720,176,gandula);
61         anterior = 1;
62     }
63 }
64 else
65 {
66     if(anterior == 1)
67     {
68         aux2= irandom(1)
69         if(aux2==0)
70         {
71             instance_create(720,176,plat_mid2);
72             instance_create(720,176,gandula);
73             anterior = 1;
74         }
75         else
76         {
77             instance_create(720,112,plat_mid3);
78             instance_create(720,112,gandula);
79             anterior = 2;
80         }
81     }
82     else
83     {
84         instance_create(720,48,plat_top);
85         instance_create(720,48,gandula);
86         anterior = 3;
87     }
88 }
89
90 }
91
92 //Novo Roll Up
93 aux = irandom(3);

```



Event: controle_Draw_1

Applies To: Self Other Object:

```

action
1 //Desenha
2
3 if (conta == 3) {
4 draw_set_font(vintage);
5 draw_set_colour(c_red);
6 draw_text(350,20,"3");
7 draw_set_font(vintage2);
8 draw_set_colour(c_black);
9 draw_text(30,10,"Barra = pulo");
10 }
11
12 if (conta == 2) {
13 draw_set_font(vintage);
14 draw_set_colour(c_red);
15 draw_text(350,20,"2");
16 draw_set_font(vintage2);
17 draw_set_colour(c_black);
18 draw_text(30,10,"Barra = pulo");
19 }
20
21 if (conta == 1) {
22 draw_set_font(vintage);
23 draw_set_colour(c_red);
24 draw_text(350,20,"1");
25 draw_set_font(vintage2);
26 draw_set_colour(c_black);
27 draw_text(30,10,"Barra = pulo");
28 }
29
30 if (conta == 0) {
31 draw_set_font(vintage);
32 draw_set_colour(c_red);
33 draw_text(295,20,"Vai!");
34 }
35
36 if (conta == -1) {
37     draw_flush();
38 }
39
40 if (conta < -1 && score != 0) {
41     draw_set_font(vintage);
42     draw_set_colour(c_blue);
43     draw_text(600,20,score);
44
45     if (melhor > 0) {
46         draw_set_font(vintage3);
47         draw_set_colour(c_blue);
48         draw_text(600,0,"Melhor: " + string(melhor));
49     }
50 }
51 }

```

A.3.2 Controle da sala inicial

Event: controle_start_Draw_1

```

action
1 ///Desenha
2
3 draw_set_font(vintage)
4 draw_set_color(c_red)
5 draw_text(200,20,"THE HOMIN")
6
7 draw_set_font(vintage2)
8 if (cor==0){
9 draw_set_color(c_black)
10 }
11 else
12 {
13 draw_set_color(c_red)
14 }
15 draw_text(150,200,"Pressione qualquer tecla")

```

Event: controle_start_Create_1

```

action
1 ///Iniciar variáveis
2 anterior = 0;
3 aux = irandom(3);
4 conta=3;
5 cor=0;
6 global.melhor =0;
7
8 if (file_exists("melhor.ini"))
9 {
10   ini_open("melhor.ini")
11   global.melhor = ini_read_real("sectionSave","melhor",0)
12   ini_close();
13 }
14

```

A.3.3 Personagem principal

Event: homin_1_Create_1

```

action
1 ///iniciar variáveis
2 grav = 0.2;
3 hsp = 0;
4 vsp = 0;
5 jumpaspeed = 7;
6
7

```

Event: homin_1_Step_1

Applies To: Self Other Object

action
1 ///acao
2 //Analizando entradas
3
4 key_jump = keyboard_check_pressed(vk_space)
5
6 //reacoes
7
8 if (vsp < 10) vsp += grav;
9
10 if (place_meeting(x,y+1,plat_bot))
11 {
12 if(!place_meeting(x+1,y,plat_bot)){
13 vsp = key_jump * -jumpspeed
14 }
15 }
16 }
17
18 if (place_meeting(x,y+1,plat_mid2))
19 {
20 if(!place_meeting(x+1,y,plat_mid2)){
21 vsp = key_jump * -jumpspeed
22 }
23 }
24
25 if (place_meeting(x,y+1,plat_mid3))
26 {
27 if(!place_meeting(x+1,y,plat_mid3)){
28 vsp = key_jump * -jumpspeed
29 }
30 }
31
32 if (place_meeting(x,y+1,plat_top))
33 {
34 if(!place_meeting(x+1,y,plat_top)){
35 vsp = key_jump * -jumpspeed
36 }
37 }
38
39 //Colisao horizontal
40 if (place_meeting(x+hsp,y,plat_bot))
41 {
42 while(!place_meeting(x+sign(hsp),y,plat_bot))
43 {
44 x += sign(hsp);
45 }
46 hsp = -1.5;
47 }
48
49 if (place_meeting(x+hsp,y,plat_mid2))
50 {
51 while(!place_meeting(x+sign(hsp),y,plat_mid2))
52 {
53 x += sign(hsp);

```
53     x += sign(hsp);
54 }
55 hsp = -1.5;
56 }
57
58 if (place_meeting(x+hsp, y, plat_mid3))
59 {
60     while(!place_meeting(x+sign(hsp), y, plat_mid3))
61     {
62         x += sign(hsp);
63     }
64     hsp = -1.5;
65 }
66
67 if (place_meeting(x+hsp, y, plat_top))
68 {
69     while(!place_meeting(x+sign(hsp), y, plat_top))
70     {
71         x += sign(hsp);
72     }
73     hsp = -1.5;
74 }
75 x += hsp;
76
77 //Colisao vertical
78 if (place_meeting(x, y+vsp, plat_bot))
79 {
80     while(!place_meeting(x, y+sign(vsp), plat_bot))
81     {
82         y += sign(vsp);
83     }
84     if(!place_meeting(x, y, plat_bot))
85     {
86         vsp = 0;
87     }
88 }
89
90 if (place_meeting(x, y+vsp, plat_mid2))
91 {
92     while(!place_meeting(x, y+sign(vsp), plat_mid2))
93     {
94         y += sign(vsp);
95     }
96     if(!place_meeting(x, y, plat_mid2))
97     {
98         vsp = 0;
99     }
100 }
101
102 if (place_meeting(x, y+vsp, plat_mid3))
```

```
103 if (place_meeting(x,y+vsp,plat_mid3))
104 {
105     while(!place_meeting(x,y+sign(vsp),plat_mid3))
106     {
107         y += sign(vsp);
108     }
109     if(!place_meeting(x,y,plat_mid3))
110     {
111         vsp = 0;
112     }
113 }
114
115 if (place_meeting(x,y+vsp,plat_top))
116 {
117     while(!place_meeting(x,y+sign(vsp),plat_top))
118     {
119         y += sign(vsp);
120     }
121     if(!place_meeting(x,y,plat_top))
122     {
123         vsp = 0;
124     }
125 }
126 y += vsp;
127
```

APÊNDICE A.4

Interface do encoder usando Arduino

```
|  
//Carrega a biblioteca do encoder  
#include <RotaryEncoder.h>  
#include <Keyboard.h>  
  
//Pinos de ligacao do encoder  
RotaryEncoder encoder(A2, A3);  
  
//Variável para controle da rotação  
int valor = 2;  
//Registrador da posição  
int newPos = 0;  
  
void setup()  
{  
    Serial.begin(9600);  
    Serial.println("Gire o encoder....");  
    Keyboard.begin();  
}  
  
void loop()  
{  
    //Le as informacoes do encoder  
    static int pos = 0;  
    encoder.tick();  
    int newPos = encoder.getPosition();  
    if(newPos < 0) {  
        encoder.setPosition(0);  
        newPos = 0;  
    }  
    if (pos != newPos) {  
        Serial.print(newPos);  
        Serial.println();  
        pos = newPos;  
    }  
  
    if(pos == valor) {  
        Keyboard.write(32);  
        Serial.print("pulo!");  
        Serial.println();  
        encoder.setPosition(0);  
        newPos = 0;  
    }  
}
```

ANEXO

Algoritmo para evolução diferencial (LOBATO, STEFFEN Jr e OLIVEIRA-LOPES (2007)

```
function [bestmem,bestval,nfeval] = de(fname,VTR,D,XVmin,XVmax,y,NP,itermax,F,CR,strategy,refresh);
% minimization of a user-supplied function with respect to x(1:D),
% using the differential evolution (DE) algorithm of Rainer Storn
% (http://www.icsi.berkeley.edu/~storn/code.html)
%
% Special thanks go to Ken Price (kprice@solano.community.net) and
% Arnold Neumaier (http://solon.cma.univie.ac.at/~neum/) for their
% valuable contributions to improve the code.
%
% Strategies with exponential crossover, further input variable
% tests, and arbitrary function name implemented by Jim Van Zandt
% <jrv@vanzandt.mv.com>, 12/97.
%
% Output arguments:
% -----
%
% bestmem      parameter vector with best solution
% bestval      best objective function value
% nfeval       number of function evaluations
%
% Input arguments:
% -----
%
% fname        string naming a function f(x,y) to minimize
% VTR          "Value To Reach". devec3 will stop its minimization
%              if either the maximum number of iterations "itermax"
%              is reached or the best parameter vector "bestmem"
%              has found a value f(bestmem,y) <= VTR.
% D            number of parameters of the objective function
% XVmin        vector of lower bounds XVmin(1) ... XVmin(D)
%             of initial population
%             *** note: these are not bound constraints!! ***
% XVmax        vector of upper bounds XVmax(1) ... XVmax(D)
%             of initial population
% y            problem data vector (must remain fixed during the
%             minimization)
% NP           number of population members
% itermax      maximum number of iterations (generations)
% F            DE-stepsize F from interval [0, 2]
% CR          crossover probability constant from interval [0, 1]
% strategy     1 --> DE/best/1/exp      6 --> DE/best/1/bin
%              2 --> DE/rand/1/exp      7 --> DE/rand/1/bin
%              3 --> DE/rand-to-best/1/exp 8 --> DE/rand-to-best/1/bin
%              4 --> DE/best/2/exp      9 --> DE/best/2/bin
%              5 --> DE/rand/2/exp      else DE/rand/2/bin
%              Experiments suggest that /bin likes to have a slightly
%              larger CR than /exp.
% refresh       intermediate output will be produced after "refresh"
%              iterations. No intermediate output will be produced
%              if refresh is < 1
%
% The first four arguments are essential (though they have
% default values, too). In particular, the algorithm seems to
% work well only if [XVmin,XVmax] covers the region where the
% global minimum is expected. DE is also somewhat sensitive to
% the choice of the stepsize F. A good initial guess is to
% choose F from interval [0.5, 1], e.g. 0.8. CR, the crossover
% probability constant from interval [0, 1] helps to maintain
% the diversity of the population and is rather uncritical. The
% number of population members NP is also not very critical. A
% good initial guess is 10*D. Depending on the difficulty of the
% problem NP can be lower than 10*D or must be higher than 10*D
% to achieve convergence.
% If the parameters are correlated, high values of CR work better.
% The reverse is true for no correlation.
%
```

```

% default values in case of missing input arguments:
%     VTR = 1.e-6;
%     D = 2;
%     XVmin = [-2 -2];
%     XVmax = [2 2];
%     y=[];
%     NP = 10*D;
%     itermax = 200;
%     F = 0.8;
%     CR = 0.5;
%     strategy = 7;
%     refresh = 10;
%
% Cost function:      function result = f(x,y);
%                      has to be defined by the user and is minimized
%                      w.r. to x(1:D).
%
% Example to find the minimum of the Rosenbrock saddle:
% -----
% Define f.m as:
%         function result = f(x,y);
%             result = 100*(x(2)-x(1)^2)^2+(1-x(1))^2;
%         end
% Then type:
%
%     VTR = 1.e-6;
%     D = 2;
%     XVmin = [-2 -2];
%     XVmax = [2 2];
%     [bestmem,bestval,nfeval] = devec3("f",VTR,D,XVmin,XVmax);
%
% The same example with a more complete argument list is handled in
% run1.m
%
% About devec3.m
% -----
% Differential Evolution for MATLAB
% Copyright (C) 1996, 1997 R. Storn
% International Computer Science Institute (ICSI)
% 1947 Center Street, Suite 600
% Berkeley, CA 94704
% E-mail: storn@icsi.berkeley.edu
% WWW: http://http.icsi.berkeley.edu/~storn
%
% devec is a vectorized variant of DE which, however, has a
% property which differs from the original version of DE:
% 1) The random selection of vectors is performed by shuffling the
%    population array. Hence a certain vector can't be chosen twice
%    in the same term of the perturbation expression.
%
% Due to the vectorized expressions devec3 executes fairly fast
% in MATLAB's interpreter environment.
%
% This program is free software; you can redistribute it and/or modify
% it under the terms of the GNU General Public License as published by
% the Free Software Foundation; either version 1, or (at your option)
% any later version.
%
% This program is distributed in the hope that it will be useful,
% but WITHOUT ANY WARRANTY; without even the implied warranty of
% MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
% GNU General Public License for more details. A copy of the GNU
% General Public License can be obtained from the
% Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, USA.

```

```

%----Check input variables-----
err=[];
if nargin<1, error('de 1st argument must be function name'); else
  if exist(fname)<1; err(1,length(err)+1)=1; end; end;
if nargin<2, VTR = 1.e-6; else
  if length(VTR)~=1; err(1,length(err)+1)=2; end; end;
if nargin<3, D = 2; else
  if length(D)~=1; err(1,length(err)+1)=3; end; end;
if nargin<4, XVmin = [-2 -2];else
  if length(XVmin)~=D; err(1,length(err)+1)=4; end; end;
if nargin<5, XVmax = [2 2]; else
  if length(XVmax)~=D; err(1,length(err)+1)=5; end; end;
if nargin<6, y=[]; end;
if nargin<7, NP = 10*D; else
  if length(NP)~=1; err(1,length(err)+1)=7; end; end;
if nargin<8, itermax = 250; else
  if length(itermax)~=1; err(1,length(err)+1)=8; end; end;
if nargin<9, F = 0.8; else
  if length(F)~=1; err(1,length(err)+1)=9; end; end;
if nargin<10, CR = 0.5; else
  if length(CR)~=1; err(1,length(err)+1)=10; end; end;
if nargin<11, strategy = 7; else
  if length(strategy)~=1; err(1,length(err)+1)=11; end; end;
if nargin<12, refresh = 10; else
  if length(refresh)~=1; err(1,length(err)+1)=12; end; end;
if length(err)>0
  fprintf(stdout,'error in parameter %d\n', err);
  usage('de (string,scalar,scalar,vector,any,integer,integer,scalar,scalar,integer,integer)');
end

if (NP < 5)
  NP=5;
  fprintf(1,' NP increased to minimal value 5\n');
end
if ((CR < 0) | (CR > 1))
  CR=0.5;
  fprintf(1,'CR should be from interval [0,1]; set to default value 0.5\n');
end
if (itermax <= 0)
  itermax = 200;
  fprintf(1,'itermax should be > 0; set to default value 200\n');
end
refresh = floor(refresh);

%----Initialize population and some arrays-----
pop = zeros(NP,D); %initialize pop to gain speed

%---pop is a matrix of size NPxD. It will be initialized-----
%---with random values between the min and max values of the-----
%---parameters-----

for i=1:NP
  pop(i,:) = XVmin + rand(1,D).*(XVmax - XVmin);
end

popold  = zeros(size(pop)); % toggle population
val      = zeros(1,NP);      % create and reset the "cost array"
bestmem  = zeros(1,D);      % best population member ever
bestmemit = zeros(1,D);      % best population member in iteration
nfeval   = 0;                % number of function evaluations

```

```

%-----Evaluate the best member after initialization-----

ibest = 1; % start with first population member
val(1) = feval(fname,pop(ibest,:),y);
bestval = val(1); % best objective function value so far
nfeval = nfeval + 1;
for i=2:NP % check the remaining members
    val(i) = feval(fname,pop(i,:),y);
    nfeval = nfeval + 1;
    if (val(i) < bestval) % if member is better
        ibest = i; % save its location
        bestval = val(i);
    end
end
bestmemit = pop(ibest,:); % best member of current iteration
bestvalit = bestval; % best value of current iteration

bestmem = bestmemit; % best member ever

%-----DE-Minimization-----
%-----popold is the population which has to compete. It is-----
%-----static through one iteration. pop is the newly-----
%-----emerging population.-----

pm1 = zeros(NP,D); % initialize population matrix 1
pm2 = zeros(NP,D); % initialize population matrix 2
pm3 = zeros(NP,D); % initialize population matrix 3
pm4 = zeros(NP,D); % initialize population matrix 4
pm5 = zeros(NP,D); % initialize population matrix 5
bm = zeros(NP,D); % initialize bestmember matrix
ui = zeros(NP,D); % intermediate population of perturbed vectors
mui = zeros(NP,D); % mask for intermediate population
mpo = zeros(NP,D); % mask for old population
rot = (0:1:NP-1); % rotating index array (size NP)
rotd= (0:1:D-1); % rotating index array (size D)
rt = zeros(NP); % another rotating index array
rtd = zeros(D); % rotating index array for exponential crossover
a1 = zeros(NP); % index array
a2 = zeros(NP); % index array
a3 = zeros(NP); % index array
a4 = zeros(NP); % index array
a5 = zeros(NP); % index array
ind = zeros(4);

iter = 1;
while (iter <= itermax) & (bestval > VTR)
    popold = pop; % save the old population

    ind = randperm(4); % index pointer array

    a1 = randperm(NP); % shuffle locations of vectors
    rt = rem(rot+ind(1),NP); % rotate indices by ind(1) positions
    a2 = a1(rt+1); % rotate vector locations
    rt = rem(rot+ind(2),NP);
    a3 = a2(rt+1);
    rt = rem(rot+ind(3),NP);
    a4 = a3(rt+1);
    rt = rem(rot+ind(4),NP);
    a5 = a4(rt+1);

    pm1 = popold(a1,:); % shuffled population 1
    pm2 = popold(a2,:); % shuffled population 2
    pm3 = popold(a3,:); % shuffled population 3
    pm4 = popold(a4,:); % shuffled population 4
    pm5 = popold(a5,:); % shuffled population 5

```

```

for i=1:NP                      % population filled with the best member
    bm(i,:) = bestmemit;          % of the last iteration
end

mui = rand(NP,D) < CR;          % all random numbers < CR are 1, 0 otherwise

if (strategy > 5)
    st = strategy-5;            % binomial crossover
else
    st = strategy;              % exponential crossover
    mui=sort(mui');             % transpose, collect 1's in each column
    for i=1:NP
        n=floor(rand*D);
        if n > 0
            rtd = rem(rotD+n,D);
            mui(:,i) = mui(rtd+1,i); %rotate column i by n
        end
    end
    mui = mui';                  % transpose back
end
mpo = mui < 0.5;                % inverse mask to mui

if (st == 1)                      % DE/best/1
    ui = bm + F*(pm1 - pm2);      % differential variation
    ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 2)                  % DE/rand/1
    ui = pm3 + F*(pm1 - pm2);      % differential variation
    ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 3)                  % DE/rand-to-best/1
    ui = popold + F*(bm-popold) + F*(pm1 - pm2);
    ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 4)                  % DE/best/2
    ui = bm + F*(pm1 - pm2 + pm3 - pm4); % differential variation
    ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 5)                  % DE/rand/2
    ui = pm5 + F*(pm1 - pm2 + pm3 - pm4); % differential variation
    ui = popold.*mpo + ui.*mui;    % crossover
end

for ii=1:NP
    for jj=1:D
        if ui(ii,jj)<XVmin(jj);
            ui(ii,jj)=XVmin(jj);
        end
        if ui(ii,jj)>XVmax(jj);
            ui(ii,jj)=XVmax(jj);
        end
    end
end

```

```

tempfun = [];
%----Select which vectors are allowed to enter the new population-----
for i=1:NP
    tempval = feval(fname,ui(i,:),y); % check cost of competitor
    tempfun(i) = tempval;
    nfeval = nfeval + 1;
    if (tempval <= val(i)) % if competitor is better than value in "cost array"
        pop(i,:) = ui(i,:); % replace old vector with new one (for new iteration)
        val(i) = tempval; % save value in "cost array"

        %---we update bestval only in case of success to save time-----
        if (tempval < bestval) % if competitor better than the best one ever
            bestval = tempval; % new best value
            bestmem = ui(i,:); % new best parameter vector ever
        end
    end
end %---end for imember=1:NP

bestmemit = bestmem; % freeze the best member of this iteration for the coming
% iteration. This is needed for some of the strategies.

%---Output section-----
%tempfun
%fprintf(1,'\\n');
%population=ui(:,1:D)
fprintf(1,'\\n');
average=1/NP*(sum(tempfun));
worst=max(tempfun);
grafico(iter,:)=[iter bestval average worst];
fprintf(1, ' =====\n');
fprintf(1, ' =====\n');
if (refresh > 0)
    fprintf(1, ' Iteration: %d, Best Value: %.11f, Average: %.11f, Worst: %.11f\\n',iter,bestval,average,worst);
    for n=1:D
        fprintf(1, ' x(%d) = %f',n,bestmem(n));
    end
    fprintf(1,'\\n');
end

iter = iter + 1;
end %---end while ((iter < itermax) ...
fprintf(1, ' =====\\n');

```