

ANTÔNIO BENTO FILHO

**MODELAGEM, CONTROLE DE ANDADURA E TRANSPOSIÇÃO DE
OBSTÁCULOS DE UM ROBÔ QUADRÚPEDE COM QUATRO GRAUS DE
LIBERDADE EM CADA PERNA**

Tese de Doutorado apresentada ao Programa de Pós Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do título de Doutor em Engenharia Elétrica , área de concentração Robótica.

Orientador: Prof. Dr. Paulo Faria Santos Amaral

Co-Orientador: Prof. Dr. Benedito Geraldo Miglio Pinto

Vitória

2007

Livros Grátis

<http://www.livrosgratis.com.br>

Milhares de livros grátis para download.

Dados Internacionais de Catalogação-na-Publicação (CIP)

(Biblioteca Central da Universidade Federal do Espírito Santo, ES, Brasil)

Bento Filho, Antônio, 1952-

B478m Modelagem, controle de andadura e transposição de obstáculos de um robô quadrúpede com quatro graus de liberdade em cada perna / Antônio Bento Filho. – 2007.

161 f. : il.

Orientador: Paulo Faria Santos Amaral.

Co-Orientador: Benedito Geraldo Miglio Pinto.

Tese (doutorado) – Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Robôs móveis. 2. Robôs. 3. Desvio de obstáculos. 4. Análise de trajetória. 5. Movimentos mecânicos. 6. Geometria cinemática. I. Amaral, Paulo Faria Santos. II. Pinto, Benedito Geraldo Miglio. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU:621.3

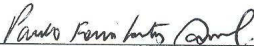
ANTÔNIO BENTO FILHO

**MODELAGEM, CONTROLE DE ANDADURA E TRANSPOSIÇÃO
DE OBSTÁCULOS DE UM ROBÔ QUADRÚPEDE COM QUATRO
GRAUS DE LIBERDADE EM CADA PERNA**


Tese submetida ao Programa de Pós-Graduação em Engenharia Elétrica da
Universidade Federal do Espírito Santo como requisito parcial para a obtenção do grau
de Doutor em Engenharia Elétrica – Automação.

Aprovada em 17/08/2007.


COMISSÃO EXAMINADORA



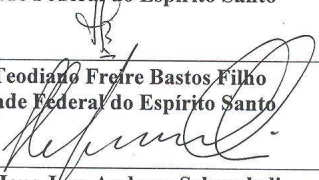
Prof. Dr. Paulo Faria Santos Amaral
Universidade Federal do Espírito Santo
Orientador



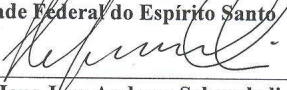
Prof. Dr. Benedito Geraldo Miglio Pinto
Universidade Federal do Espírito Santo
Co-orientador



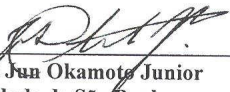
Prof.ª Dr.ª Raquel Frizera Vassallo
Universidade Federal do Espírito Santo



Prof. Dr. Teodiano Freire Bastos Filho
Universidade Federal do Espírito Santo



Prof. Dr. Hans Jörg Andreas Schneebeli
Universidade Federal do Espírito Santo



Prof. Dr. Jun Okamoto Junior
Universidade de São Paulo

AGRADECIMENTOS

Agradeço a Deus, pois “Que posso eu oferecer ao Deus Eterno por tudo o que ele me tem dado?” (Salmo 116, vs. 12). A Ele devo a vida, a saúde, a vontade de vencer e a companhia nos momentos difíceis.

À minha esposa Jaqueline, grande incentivadora e de quem jamais ouvi uma palavra sequer de desânimo durante o longo período em que estive completamente envolvido com este projeto.

Aos meus filhos, que sempre me compreenderam, apoiaram e incentivaram mesmo quando estive ausente para realizar tarefas do projeto.

Ao meu amigo e orientador Paulo, agradeço principalmente, pela orientação segura, pela paciência com que sempre me ouviu e pela grande experiência que me permitiu obter com a realização deste projeto.

Ao meu amigo e co-orientador Benedito, agradeço porque tudo começou longo tempo atrás com o seu incentivo para o meu mestrado, quando ainda era seu aluno de graduação.

Ao meu amigo Luis Eduardo Martins de Lima, companheiro na realização do projeto, agradeço pela dedicação e apoio; ao meu amigo Rafael Lacerda Alves, agradeço pela valiosa colaboração e pelas sugestões sempre oportunas.

A todos os funcionários da Automatica Tecnologia S/A, agradeço pela dedicação e disponibilidade sempre que foram solicitados para me ajudar a realizar tarefas relativas ao projeto.

Aos meus colegas professores, agradeço pela colaboração direta ou indireta na realização deste projeto.

À empresa Automatica Tecnologia S/A, agradeço por ter viabilizado o projeto, proporcionando todos os recursos materiais e humanos para a construção do robô Guará e para os experimentos que foram realizados.

Sumário

<i>Capítulo 1 Preâmbulo</i>	<i>12</i>
1.1 Introdução.....	12
1.2 Definição do problema	15
1.3 Características do problema.....	16
1.4 Contribuições deste trabalho.....	18
1.5 Roteiro desta tese	19
<i>Capítulo 2 Andadura</i>	<i>20</i>
2.1 Introdução.....	20
2.2 Biomecânica da andadura	21
2.3 Características e parâmetros da andadura do Guará	22
2.4 Movimento de deslocamento lateral.....	27
2.5 Geração da andadura para locomoção do robô.....	30
2.6 Trajetória das patas para locomoção em curva.....	33
2.7 Discussão	35
<i>Capítulo 3 Cinemática.....</i>	<i>38</i>
3.1 Introdução.....	38
3.2 Orientação da plataforma do robô	39
3.3 Plano de apoio das patas	47
3.4 Posição da projeção do CG no plano de apoio	47
3.5 Distância da projeção vertical do CG aos lados do polígono de apoio	48
3.7 Geometria da trajetória em curva	49
3.8 Discussão	51
<i>Capítulo 4 Controle de andadura e trajetória do robô.....</i>	<i>53</i>
4.1 Introdução.....	53
4.2 Arquitetura de controle.....	53
4.3 Diagrama de estratégia do ciclo principal de controle.....	57
4.3.1 Lê registros de chaves das patas.....	60
4.3.2 Processa colisão	61
4.3.3 Executando andadura	66
4.3.3.1 Preparando o robô	67
4.3.3.2 Corrigindo postura	71
4.3.3.3 Equalizando patas	73
4.3.3.4 Andando em reta.....	75
4.3.3.4.1 Trajetórias da pata durante os movimentos de empuxo e vôo	76
4.3.3.4.2 Trajetórias das patas na andadura	81
4.3.3.4.3 Empuxo, vôo e sincronismo de trajetória das patas	83
4.3.3.5 Andando em curva	84
4.3.3.6 Envia dados	89

4.3.3.7	Lê os registros de dados dos potenciômetros	92
4.3.3.8	Atualiza variáveis e controles	92
4.4	Discussão	95
<i>Capítulo 5 Estratégia de transposição de obstáculos.....</i>		<i>97</i>
5.1	Introdução.....	97
5.2	Percepção e transposição de obstáculos.....	98
5.2.1	Canal	100
5.2.2	Degrau descendente.....	103
5.2.3	Rampa	106
5.2.4	Degrau ascendente	109
5.2.5	Degrau estreito.....	112
5.2.6	Desviar	115
5.3	Discussão	117
<i>Capítulo 6 Resultados.....</i>		<i>119</i>
6.1	Introdução.....	119
6.2	Robô quadrúpede Guará	120
6.3	Geração de trajetória e controle de andadura reta	123
6.3.1	Processos e grupos de movimento de andadura reta.....	126
6.4	Andadura em curva	128
6.4.1	Processos e grupos de movimento de andadura curva.....	132
6.5	Processos e grupos de movimento de subida de um degrau no piso.....	133
6.6	Discussão.....	139
<i>Capítulo 7 Conclusões e Sugestões</i>		<i>142</i>
<i>ANEXOS</i>		<i>146</i>
A1.	Dados técnicos do Guará.....	146
A2.	Programa de controle do Nível Supervisor do Guará.....	148
A3.	Tabela de combinações do ciclo principal de controle.....	153
A4.	Cinemática Direta da perna do robô.....	156
A5.	Cinemática inversa da perna do robô	158
<i>Referências</i>		<i>160</i>

Lista de Figuras

Figura 2.3-1: Robô Guará.	24
Figura 2.3-2: Andadura regular simétrica em onda periódica de ciclo completo.	26
Figura 2.3-3: Tela de definição da andadura.	28
Figura 2.4-1: Geração do movimento lateral em sincronismo com a andadura.	31
Figura 2.5-1: Interface para definição e configuração de andaduras.	33
Figura 2.6-1: Tela de Interface para andar com destaque para os controles de andadura em curva.	34
Figura 2.6-2: Geometria de curva com dirigibilidade nas quatro patas.	35
Figura 3.2-1: Foto do Guará deslocado para a esquerda, preparando para andar em linha reta.	40
Figura 3.2-2: Referenciais utilizados e variáveis do robô.	40
Figura 3.2-3: Sistemas de referência das juntas, conforme convenção de Denavit-Hatemberg.	41
Figura 3.2-4: Configurações de apoio com 3 patas no solo.	43
Figura 3.7-1: Parâmetros geométricos para definição da trajetória curva.	50
Figura 4.2-1: Arquitetura de controle do Guará.	55
Figura 4.2-2: Diagrama simplificado de elementos de controle e acionamento de uma perna.	56
Figura 4.3-1: Diagrama de estratégia do ciclo principal de controle do Guará.	58
Figura 4.3-2: Diagrama de estratégia de leitura de registros de chaves das patas.	61
Figura 4.3-3: Diagrama de estratégia para processamento de colisão.	65
Figura 4.3-4: Diagrama de estratégia para processamento de andadura.	68
Figura 4.3-5: Diagrama de estratégia para preparar o robô para andar.	70
Figura 4.3-6: Diagrama de estratégia de correção da postura do robô.	72
Figura 4.3-7: Diagrama de estratégia para equalização das patas do robô.	74
Figura 4.3-8: Diagrama de estratégia de andar em linha reta	77
Figura 4.3-9: Trajetória da pata com vôo em ciclóide; empuxo da direita para a esquerda.	79
Figura 4.3-10: Trajetória da pata com vôo em poligonal.	80
Figura 4.3-11: Coordenadas geradas para as juntas da pata 0 no espaço operacional.	82
Figura 4.3-12: Coordenadas geradas para pata 0 no espaço de juntas.	83
Figura 4.3-13: Coordenadas geradas para as patas no espaço operacional, referencial da perna.	84
Figura 4.3-14: Esquema para interface de dirigibilidade do Guará.	85
Figura 4.3-15: Diagrama de estratégia da andadura em curva.	87
Figura 4.3-16: Coordenadas laterais e verticais geradas para as patas 0 a 3 na andadura em curva.	88
Figura 4.3-17: Coordenadas de empuxo geradas para as patas 0 e 2 na andadura em curva.	88
Figura 4.3-18: Coordenadas de juntas geradas para as patas 0 a 3 na andadura em curva.	89
Figura 4.3-19: Diagrama de estratégia de enviar dados para o robô.	91
Figura 4.3-20: Diagrama de estratégia de ler os registros de dados dos potenciômetros.	93
Figura 4.3-21: Diagrama de estratégia de atualizar variáveis e controles.	94
Figura 5.2-1: Toque das patas durante o vôo ou movimentando para superar obstáculo.	99
Figura 5.2-2: Diagrama de estratégia para cruzar um canal.	102
Figura 5.2-3: Obstáculo tipo canal; (a) canal largo; (b) canal estreito.	100
Figura 5.2-4: Obstáculo tipo descida de degrau.	103

Figura 5.2-5: Diagrama de estratégia de degrau descendente.....	105
Figura 5.2-6: Obstáculo tipo rampa; descendente (a); ascendente (b).	106
Figura 5.2-7: Diagrama de estratégia de rampa ascendente ou descendente.	108
Figura 5.2-8: Obstáculo tipo degrau ascendente.....	109
Figura 5.2-9: Diagrama de estratégia de subir um degrau.....	111
Figura 5.2-10: Obstáculo tipo degrau estreito.	112
Figura 5.2-11: Diagrama de estratégia de degrau estreito.	114
Figura 5.2-12: Diagrama de estratégia de desviar.	116
Figura 6.2-1: Fotografia do robô quadrúpede Guará.	121
Figura 6.2-2: Detalhes das chaves de pata, do hardware a bordo e da perna do Guará.....	122
Figura 6.3-1: Coordenadas geradas e recebidas do robô, junta 3, perna 0.	123
Figura 6.3-2: Coordenadas de juntas da pata 0 enviadas e recebidas do robô.	125
Figura 6.3-3: Coordenadas geradas e recebidas do robô, junta 3 da pata 0 no espaço operacional.	125
Figura 6.3-4: Andadura mostrando o vôo das patas na primeira passada da andadura.	127
Figura 6.3-5: Andadura mostrando o pouso das patas após a primeira passada da andadura.	127
Figura 6.3-6: Andadura mostrando as patas de um mesmo lado no instante de maior proximidade.....	128
Figura 6.4-1: Coordenadas geradas e recebidas do robô na andadura em curva, junta 3 da pata 0.....	130
Figura 6.4-2: Coordenadas geradas em curva, junta 3 das pernas 0 e 2 no espaço operacional.	130
Figura 6.4-3 Coordenadas geradas em curva, junta 3 das pernas 2 e 3 no espaço operacional.	131
Figura 6.4-4: Seqüência de andadura em curva, raio médio de 1,40 m.....	133
Figura 6.5-1: Andadura com colisão mostrando o vôo das patas 3 e 2 na primeira passada.	135
Figura 6.5-2: Equalização das patas 3, 1 e 0 preparando para subir degrau.	135
Figura 6.5-3: Seqüência de movimentos para apoiar as patas 2 e 0 no degrau em cima.	136
Figura 6.5-4: Equalização após das patas apoio das dianteiras no piso em cima do degrau.....	136
Figura 6.5-5: Andadura mostrando a colisão da pata 3 e a equalização das patas dianteiras.	137
Figura 6.5-6: Apoiando as patas 3 e 1 no piso em cima do degrau.....	138
 Figura A# 1: Tela inicial, de escolha de andadura e entrada de parâmetros.....	 148
Figura A# 2: Tela de interface para andar.....	149
Figura A# 3: Tela de acesso direto ao comando dos graus de liberdade e diagnóstico.	150
Figura A# 4: Tela de configuração de parâmetros do robô.	151
Figura A# 5: Tela de definição de uma nova andadura, mostrando a andadura 8.	152
Figura A# 6: Diagrama cinemático da perna com o grau de liberdade de rolagem no quadril.....	156

Lista de Tabelas

Tabela 2.3-1: Ângulos de fase da andadura do Guará.....	27
Tabela 2.4-1: Contadores de sincronismo para movimento lateral da plataforma.	29
Tabela 2.4-2: Valores de deslocamento lateral em função do incremento de avanço do robô.	29
Tabela 4.3-1: Variáveis para direcionar o processamento do ciclo operacional do robô.	59
Tabela 4.3-2: Variáveis de processamento de Colisão.....	63
Tabela 4.3-3: Estados das chaves das patas.....	63
Tabela 4.3-4: Combinações de chaves que ativam as variáveis funcionais de processamento de colisão..	64
Tabela 4.3-5: Variáveis funcionais do processo Executando Andadura	67
Tabela 4.3-6: Variáveis funcionais do processo preparando robô.	69
Tabela 4.3-7: Variáveis associadas aos processos da andadura.....	76
Tabela 4.3-8: Variáveis associadas ao processo de enviar dados	90
Tabela 5.2-1: Variáveis associadas aos processos para cruzar um canal.....	101
Tabela 5.2-2: Variáveis associadas aos processos de colisão com degrau descendente.....	104
Tabela 5.2-3: Variáveis associadas aos processos de rampa.	107
Tabela 5.2-4: Variáveis associadas aos processos de colisão com degrau ascendente.....	110
Tabela 5.2-5: Variáveis associados aos processos de transpor um degrau estreito.....	113
Tabela 5.2-6: Variáveis associadas aos processos para desviar.....	115

RESUMO

O presente trabalho apresenta uma abordagem para o problema de modelagem, controle de andadura e transposição de obstáculos pré-definidos, de um robô quadrúpede com quatro graus de liberdade em cada perna e dispondo de chaves de contato nas patas.

A transposição de obstáculos ocorre em ambientes semi-estruturados, utilizando apenas as informações obtidas das chaves de contato instaladas nas patas do robô. Após perceber o obstáculo, as informações são utilizadas para gerenciar uma sequência autônoma de movimentos complexos e integrados para movimentar o robô de tal forma a transpor o obstáculo ou desviar-se dele, preservando o equilíbrio e estabilidade.

Para a execução deste trabalho, foi desenvolvido e construído um robô quadrúpede com quatro graus de liberdade por perna, dotado de uma estratégia de navegação e equilíbrio baseada em padrões de movimento observados nos mamíferos.

Um algoritmo incremental para execução de andaduras retas e curvas permite a dirigibilidade do robô em movimento e preserva o seu equilíbrio e estabilidade.

O software de controle, desenvolvido usando Programação Orientada a Objetos (*POO*), é um ambiente integrado para acionamento, simulação, ajuste de parâmetros e diagnóstico de funcionamento do robô.

São apresentados os resultados da geração de andadura reta, em curva e a subida de um degrau no piso.

ABSTRACT

These work deals with modeling, gait control and obstacles transposing of a quadruped robot with four degrees of freedom in each leg, and having contact keys in the feet.

The obstacles transposing is done using only the information of the contact sensors installed in the robot feet. After contact, the obstacle information is used to manage a complex and integrated movement sequence to move the robot in order to transpose, or to turn aside the obstacle, preserving the robot balance and stability.

For this purpose, it was developed and constructed a quadruped robot with 4 degrees of freedom per leg, which navigation and balance scheme is based on the movements of mammals.

An incremental algorithm for execution of both straight and curved paths allows the robot to turn while walking and preserves its balance and stability.

The control software developed in OOP is an integrated environment for drive, simulation, adjustment of parameters and diagnosis of robot functioning.

Results includes gait generation for straight and curved paths and to go one step up in the floor.

Capítulo 1

Preâmbulo

1.1 Introdução

A integração de robôs móveis aos ambientes sociais, a interação com pessoas, mobiliário e outros robôs móveis tem sido um dentre vários desafios da robótica móvel.

Robôs capazes de realizar tarefas de limpeza em shoppings, escritórios e residências; de recolher e distribuir objetos em escritórios e fábricas; de distribuir medicamentos e recolher objetos em hospitais; aplicações envolvendo perigo tais como manipulação de substâncias tóxicas; etc. são algumas das muitas aplicações vislumbradas pelos pesquisadores atualmente. Todas estas aplicações pressupõem a habilidade do robô em perceber e evitar colisões com pessoas, mobiliário e outros robôs móveis, além de não se desviar de seu plano ou tarefa.

Construir um time de robôs humanóides que possa vencer uma partida de futebol contra a seleção humana vencedora da Copa do Mundo da FIFA no ano de 2050, é o objetivo final da “RoboCup”, www.robocup.org. Em BURKHARD (BURKHARD et al., 2002), os autores consideram este objetivo da perspectiva de quão perto estamos dela hoje e o que resta ser feito para atingi-lo.

Para operar com segurança em ambientes públicos, muitos sistemas bem sucedidos de robôs móveis confiam nos módulos de prevenção de colisões, baseados em sensores, para controlar o robô. O paradigma predominante destes é estritamente sentir - reagir: a leitura do sensor é analisada continuamente para determinar o movimento livre de colisão.

A navegação autônoma representa um alto nível de dificuldade uma vez que requer a detecção e o desvio de obstáculos, simultaneamente, guiando o robô na direção do seu objetivo. Em geral pressupõe a existência de obstáculos fixos ou móveis, conhecidos e desconhecidos.

Diversas estratégias para detecção de obstáculos e planejamento de trajetórias com robôs móveis, estão descritas em KROGH (1984), KHATIB (1985), ELFES (1985), BROOKS (1986), JARVIS (JARVIS; BYRNE, 1986), ARKIN (1993), BORENSTEIN (BORENSTEIN; KOREN, 1989), BORENSTEIN (BORENSTEIN; KOREN, 1991), ZELINSKY (1992), CHEN (CHEN; QUINN, 1997), FOX (FOX et al., 1998), dentre outros.

Para os robôs móveis a pernas, além das habilidades de navegação exigidas, a preservação do equilíbrio do robô é fundamental para que o mesmo possa realizar a tarefa pretendida.

A dificuldade no desenvolvimento de veículos móveis a pernas é, principalmente, de coordenação do grande número de graus de liberdade que devem ser acionados com relativa precisão a um simples movimento do robô partindo de um ponto A para um ponto B. A coordenação precisa de tais movimentos, de forma que as forças dinâmicas resultantes deles mantenham o robô em equilíbrio, é um grande desafio, se for considerada a possibilidade de colisões com obstáculos ou a existência de irregularidades no piso que podem destruir o balanço e o robô cair por falta de pontos adicionais de apoio ou por colisão da perna com alguma elevação. Tal desafio é mapear, em intervalos de tempo muito pequenos, a simples tarefa de mover-se de A para B, em ângulos e trajetórias de juntas, que tipicamente, devem satisfazer a diversas restrições, evitando ainda obstáculos fixos e móveis (BERKEMEYER, 1998).

Estas restrições são superadas modelando, embora às vezes estaticamente, o balanço dinâmico de forças resultantes do movimento do robô. Em MCMAHON (1984) é apresentado um estudo de avaliação energética do passo e corrida do homem e, segundo o autor, embora não exista uma maneira única de descrever o movimento das pernas durante o passo, existem seis movimentos característicos de um passo humano normal, dentre os quais, um de rolagem e um de guinada do quadril, que permitem variar o comprimento da perna durante o passo e achatar a trajetória vertical do CG. Para realizar tais movimentos as juntas de fixação das pernas aos quadris devem ter graus de liberdade de rolagem, para deslocamento lateral do CG do robô e rolagem do quadril, e de guinada para permitir o movimento de guinada do quadril, além do grau de liberdade de mergulho para o movimento pendular da perna.

Diversos trabalhos de detecção e transposição de obstáculos, bem como, de desenvolvimento de técnicas de andadura que permitam aos robôs a pernas caminhar em ambientes com obstáculos e terrenos irregulares, têm sido desenvolvidos pelos principais grupos de pesquisa dentre os quais MCGHEE (MCGHEE; ISWANDHI, 1979) , LEE (LEE; SONG, 1991), LEPINAY (LEPINAY; ZAPATA; JOUVENCEL, 1993), BAI (BAI; LOW, 2000) e outros.

IGARASHI (IGARASHI; MASAYOSHI, 2002) apresentam uma abordagem do problema de locomoção de um robô quadrúpede em ambientes complexos. Devido a alta mobilidade necessária para adaptar a locomoção do robô para andar nesses ambientes e transpor obstáculos, padrões periódicos de andadura não são adequados sendo necessária a implementação de padrões livres de andadura, construída para permitir a variação de postura durante o movimento do robô entre os obstáculos, abaixando o corpo e estreitando a excursão das pernas, que no caso do robô “JROB-1” utilizado nos experimentos, são do tipo “Scara”.

Com uma estratégia de sensoriameto local semelhante à proposta no presente trabalho HIROSHI (HIROSHI; MASAYOSHI, 2003) apresentam uma metodologia para detecção de obstáculos em robôs a pernas usando sensores de proximidade, para ampliar a capacidade de locomoção em ambientes complexos. Embora a informação ao redor do robô seja importante para o planejamento de sua trajetória, para os obstáculos locais, como obstáculos baixos e curtos e obstáculos aéreos, é necessário um reconhecimento que permita a sua transposição. Os obstáculos baixos são detectados por sensores de distância instalados nas patas dianteiras, e o robô pode passar através deles ou subir com essa abordagem. Os obstáculos aéreos são ultrapassados, aproveitando a habilidade do robô a pernas poder baixar a estrutura e continuar a caminhar.

O presente trabalho trata de uma estratégia de detecção de obstáculos, inserida no contexto das estratégias de navegação autônoma para robôs móveis, conforme enfatiza HIROSHI (HIROSHI; MASAYOSHI, 2003), porém desenvolvida com características mais amplas com movimentos complexos autônomos que conferem ao robô maior habilidade e autonomia de navegação em ambiente semi-estruturado e com obstáculos de forma conhecida.

O esquema proposto habilita o robô para a transposição de obstáculos de forma conhecida utilizando apenas informações de sensores de contato instalado nas patas para perceber o obstáculo e gerenciar uma seqüência autônoma de movimentos complexos integrados para movimentar o robô de tal forma a transpor o obstáculo, ou desviar-se dele, preservando o equilíbrio e estabilidade.

Os movimentos principais do robô são precedidos de movimentos secundários axiais e transversais da plataforma, planejados para permitir um posicionamento que sempre maximize a margem de estabilidade do robô. Os movimentos das pernas para avanço e recuo, transversal e axial das patas são sempre realizados isoladamente e após o posicionamento da plataforma na posição mais afastada possível, longitudinal e transversalmente, em relação à pata que vai ser suspensa.

A estratégia de execução de curvas é através de segmentos de reta no espaço de coordenadas. Foi implementado um algoritmo incremental, que permite variar o raio e o ângulo de curva, sincronizar uma seqüência padronizada de movimentos da andadura e preservar o equilíbrio e estabilidade com o robô em movimento.

O algoritmo apresentado por BAI (BAI; LOW; ZIELINSKA, 1999) para geração de uma andadura circular de um robô quadrúpede utiliza uma trajetória em arco de círculo para a trajetória da pata, e, segundo os autores, a metodologia proposta é eficiente. Entretanto, não são apresentados detalhes sobre o custo computacional de execução e nem há um robô mostrando a utilização da metodologia proposta, não sendo possível concluir sobre a estabilidade e dirigibilidade do robô.

A habilidade que o algoritmo proposto e implementado no presente trabalho permite de variar o raio e o ângulo de curva com o robô em movimento, respeitados os limites cinemáticos das juntas, torna o robô dirigível e preserva o seu equilíbrio e estabilidade, sendo assim uma estratégia de dirigibilidade para um robô quadrúpede.

1.2 Definição do problema

O esforço de desenvolvimento da robótica móvel a pernas tem sido realizado em diversas frentes dentre as quais:

- a. Pesquisas da locomoção e biomecânica dos animais e insetos com pernas;
- b. Construção de modelos para representação da locomoção de animais e insetos com pernas;
- c. Construção de modelos matemáticos de equilíbrio para robôs a pernas com o fim de simular os modelos construídos para os animais e insetos com pernas;

- d. Projeto e construção de robôs, pernas e dispositivos para aplicações especiais, pesquisa de locomoção e validação de modelos;
- e. Projeto e construção de robôs móveis a pernas, capazes de exibir ao mesmo tempo reatividade e adaptabilidade ao seu ambiente de operação.

O presente trabalho enquadra-se inteiramente na linha de desenvolvimento descrita no item (e) anterior, tendo compreendido:

- ✓ O estudo de soluções de mecatrônica para a construção de pernas e patas para robôs (BENTO FILHO; et al., 2000);
- ✓ O projeto mecânico e a construção de um robô quadrúpede;
- ✓ A implementação de uma plataforma de software com estruturas de dados, interfaces e modelos, construída com técnicas de programação orientada a objetos (*POO*);
- ✓ A implementação de uma interface para a construção de andaduras;
- ✓ A implementação dos modelos e do software de cinemática de movimento e equilíbrio de um robô quadrúpede;
- ✓ A implementação dos algoritmos de percepção e transposição de obstáculos em ambientes semi-estruturados, utilizando apenas informações obtidas de sensores de contato instalados nas patas do robô.

Os problemas abordados são a modelagem, o controle de andadura, a transposição ou o desvio de obstáculos de um robô quadrúpede em ambientes semi - estruturados, com piso firme e regular, utilizando apenas informações obtidas de sensores de contato instalados nas patas do robô. Os obstáculos considerados são o degrau ascendente, degrau descendente, degrau estreito, canal e rampa.

1.3 Características do problema

A abordagem do problema de modelagem, controle de andadura e transposição de obstáculos de um robô quadrúpede em ambientes semi-estruturados, utilizando informações obtidas de sensores de contato instalados nas patas do robô tem, dentre outras, as seguintes características:

- *Quanto aos sensores:* devem ter resposta rápida e estarem integrados para fornecer as informações que permitirão o reconhecimento do obstáculo, a formulação das estratégias para a sua superação ou desvio e a manutenção da estabilidade para o cumprimento da tarefa do robô;
- *Quanto ao modelo matemático, às andaduras, aos parâmetros do passo e aos algoritmos de transposição de obstáculos:* devem ter precisão e velocidade para definição da trajetória das patas no espaço operacional, na andadura normal e na transposição de obstáculos, e no mapeamento das coordenadas do espaço operacional no espaço de juntas, para acionamento das pernas;
- *Quanto ao software:* deve prover os programas e as estruturas de dados para gerar as informações do sistema de navegação; executar os modelos matemáticos de andadura, cinemática direta e inversa; sincronizar o movimento das patas; verificar a segurança do movimento; realimentar informações aos demais sistemas; e comunicar-se com o hardware de controle a bordo do robô;
- *Quanto à comunicação entre os níveis de controle:* deve ter velocidade compatível com a temporização das tarefas de controle e aquisição de dados, possibilitando a troca de informações entre os níveis de processamento na temporização necessária;
- *Quanto ao hardware de controle:* deve ser instalado a bordo do robô, para a execução dos algoritmos de controle, aquisição de dados e comunicação.

O sistema de controle do Guará foi implementado em três níveis: supervisor, coordenador e atuador, conforme Figura 4.2-1. Uma rede CAN de 1 Mb/s interliga o *Nível Supervisor* e o *Nível Coordenador*. O *Nível Coordenador* possui quatro microcontroladores cada um deles responsável pelo acionamento, aquisição de dados e reações de uma perna. Através de um algoritmo de controle “*Fuzzy*” e um acelerômetro instalado a bordo, não utilizados no presente trabalho, o robô terá capacidade de reação a imprevistos que possam colocar em risco o seu equilíbrio. O algoritmo de controle “*Fuzzy*”; o projeto e a implementação do hardware; a configuração da rede CAN; e os algoritmos de controle, aquisição de dados e comunicação são objeto de outro projeto de doutorado e estão descritos no projeto de tese de LIMA (2006).

A bordo do Guará também será instalado um sistema de identificação de obstáculos baseado em laser, formado por uma câmera “*CCD*” com um sistema micro-processado e um laser que gera linhas para varrer a superfície à frente do robô. As imagens adquiridas são então tratadas

por algoritmos para extração dos perfis do laser. Imagens de profundidade do ambiente são geradas, permitindo assim seu mapeamento 3D. Desta maneira, o robô navega de forma segura e eficiente, desviando-se ou ultrapassando os objetos que porventura possam estar na sua trajetória. Este sistema é objeto de outro projeto de doutorado, e está descrito no projeto de tese de ORLANDI (2007).

1.4 Contribuições deste trabalho

Esta tese descreve a *Modelagem, Controle de Andadura e Transposição de Obstáculos de um Robô Quadrúpede com Quatro Graus de Liberdade em Cada Perna*, um trabalho de pesquisa e desenvolvimento em robótica móvel a pernas, que traz as seguintes contribuições:

Principal:

- *Uma estratégia para transposição de obstáculos* utilizando apenas informações de sensores de contato instalados nas patas para perceber o obstáculo e gerenciar uma seqüência autônoma de movimentos complexos integrados para movimentar o robô de tal forma a transpor o obstáculo ou desviar-se dele, preservando o equilíbrio e estabilidade;

Secundárias:

- *Um algoritmo incremental para execução de curvas com robôs quadrúpedes* através de segmentos de reta no espaço de coordenadas, que permite variar o raio e o ângulo de curva com o robô em movimento, sincroniza uma seqüência padronizada de movimentos da andadura e preserva o equilíbrio e estabilidade do robô;
- *Uma estratégia de navegação e equilíbrio* para um robô quadrúpede envolvendo o acionamento incremental das pernas, de forma a obter um movimento complexo e coordenado, com base em padrões de movimento observados nos mamíferos, e mapeados para padrões cinemáticos de andadura;
- *Um ambiente de software integrado* para acionamento, simulação, ajuste de parâmetros e diagnóstico de funcionamento de um robô quadrúpede, desenvolvido com técnicas de Programação Orientada a Objetos (POO);
- *O desenvolvimento e construção de um robô quadrúpede* com quatro graus de liberdade por perna para servir como plataforma para pesquisa e desenvolvimento na área de robótica móvel a pernas.

1.5 Roteiro desta tese

O Capítulo 1 apresenta o assunto, define o problema abordado e suas características e apresenta as contribuições do trabalho.

No Capítulo 2 é detalhado o modelo de andadura e seus parâmetros, bem como, a sua implementação no algoritmo de controle do robô. Com base na biomecânica da andadura humana, são detalhadas as estratégias para movimentar o robô lateralmente durante a andadura e a trajetória da pata para locomoção em trajetória curva.

No Capítulo 3 são apresentados os modelos cinemáticos utilizados no controle e supervisão do robô. Os modelos são baseados em geometria analítica e análise vetorial e permitem definir, dentre outras variáveis, a orientação da plataforma, a margem de estabilidade e a geometria da trajetória em curva.

O Capítulo 4 descreve o esquema de controle geral e do controle de trajetória, que processa os modelos e gerencia as tarefas de controle e comunicação utilizados para acionar o robô. O controle é em três níveis, sendo o *Nível Coordenador* e o *Nível Atuador*, que rodam a bordo do robô, e o *Nível Supervisor*, que é descrito neste capítulo, e roda em um PC externo ou em um “notebook” a bordo do robô.

O Capítulo 5 apresenta a estratégia implementada para transposição ou o desvio de obstáculos que se colocam na trajetória do robô. São descritos os algoritmos e processos utilizados para a detecção, transposição ou desvio de obstáculos, utilizando-se apenas informações obtidas de sensores de contato instalados nas patas do robô.

No Capítulo 6 são mostrados os resultados dos modelos de geração de trajetória, de execução da andadura e dos movimentos complexos implementados para a movimentação em reta, curva e em obstáculos. Os resultados apresentados são as saídas do controle enviadas e os valores recebidos do robô após a sua execução, que mostram o sincronismo e as características dos movimentos de andadura e obstáculos, validando os modelos implementados, e seqüências de fotografias mostrando o robô andando em reta, em curva e subindo um degrau de acordo com as estratégias implementadas no Capítulo 5.

No Capítulo 7 são apresentadas as conclusões relacionadas às contribuições, as quais foram confirmadas experimentalmente, e as sugestões para trabalhos futuros.

Capítulo 2

Andadura

2.1 Introdução

A expectativa de obter robôs móveis a pernas com as qualidades e habilidades dos animais e insetos tem motivado pesquisas sobre a sua locomoção para permitir a sua representação gráfica, a construção de modelos matemáticos e o desenvolvimento de programas capazes de reproduzi-los nos computadores de controle dos robôs. Parte importante da robótica móvel a pernas tem sido inspirada em animais e insetos, e a representação e modelagem das andaduras desses seres vivos, tem sido utilizada em simuladores e em robôs móveis, em um processo interdependente, para melhor compreender e representar seus mecanismos de locomoção e também para construir robôs mais eficientes e precisos.

Estas pesquisas buscam também, inspirar a produção de veículos e robôs móveis que sejam capazes de cruzar facilmente terrenos íngremes, ultrapassar grandes obstáculos e agilmente manobrar em espaços restritos. Tais aplicações são típicas da agricultura, exploração de planetas, aplicações militares, inspeção em locais perigosos, etc. (BERKEMEYER, 1998).

A estratégia de locomoção implementada para o Guará foi desenvolvida a partir do conhecimento e observação da biomecânica do andar humano e dos mamíferos quadrúpedes. Ambos realizam movimentos complexos ao andar pois dispõe de um grande número de graus de liberdade em sua estrutura.

Neste capítulo é detalhado o modelo de andadura e seus parâmetros, bem como, a sua implementação no algoritmo de controle do robô.

2.2 Biomecânica da andadura

O estudo da mecânica do homem, dos animais e dos insetos terrestres, a modelagem matemática da sua locomoção e a construção de modelos para a descrição do seu comportamento cinemático e dinâmico, é o objeto de interesse da biomecânica.

MACMAHON (1984), revê os princípios de mecânica e energia associados ao passo e à corrida do homem, com base em resultados de plataformas de medição de forças.

Estas plataformas permitem obter as forças verticais e horizontais durante o andar ou a corrida e, com o uso de câmaras de vídeo rápidas, é possível se obter informações de velocidade e aceleração das pernas.

Existem 6 movimentos característicos numa andadura humana normal, que foram definidos em 1953 por Saunders, Innam e Eberhart, de acordo com ALEXANDER.(1984):

São eles:

- *Movimento radial*, onde a perna em apoio permanece rígida, gira em relação ao pé de apoio e o quadril se move em um arco de círculo;
- *Movimento de rotação pélvica* (guinada do quadril), onde o quadril pode girar em relação ao eixo vertical, aumentando o comprimento do passo e achatando a trajetória circular do quadril pelo aumento do comprimento efetivo da perna em apoio;
- *Movimento de inclinação pélvica* (rolagem do quadril), onde o quadril pode inclinar-se em relação à horizontal, achatando ainda mais a trajetória circular do quadril e que força a dobra do joelho da perna em movimento aéreo, para evitar choque do dedão do pé com o solo;
- *Movimento de flexão da perna em apoio*, que tem o efeito de achatar um pouco mais a trajetória circular do quadril;
- *Movimento de flexão da planta do pé em apoio*, para suavizar a transição entre as fases de apoio com dois e com um pé apenas, imediatamente antes do descolamento do dedo do pé da pata em apoio do chão;
- *Movimento de deslocamento lateral do quadril*, que é um movimento senoidal com metade da frequência dos movimentos de subida e descida do quadril, utilizado para transferir o peso do corpo para a perna em apoio.

O movimento de andar humano e de animais alterna períodos com e sem atividade muscular. Durante o andar humano, por exemplo, grande parte do esforço é utilizado na fase com ambas pernas no solo e o movimento de comutação entre as pernas pode ser assimilado como o de um pêndulo livre conforme MCMAHON(1984).

2.3 Características e parâmetros da andadura do Guará

Para descrever a andadura implementada para o Guará, Figura 2.3-1, foi utilizada a nomenclatura apresentada em SONG (SONG; WALDRON, 1989), que se baseou nos trabalhos de MCGHEE (MCGHEE; ISWANDHI, 1979) e seus colaboradores. Nas definições a seguir, os termos perna e pata são utilizados como referência comum às pernas do robô, de acordo com o contexto.

Uma *andadura* do robô é caracterizada pela seqüência, periódica ou não, de movimentos que representam a evolução do estado das patas, passando de um estado em apoio a um estado em voo, quando a pata está sendo transferida para um novo ponto de apoio, e novamente em apoio.

Uma passada ou ciclo de locomoção do robô é um ciclo completo dos movimentos das pernas do robô, tomando-se como referência, o momento em que a pata dianteira esquerda é apoiada.

A fase de transferência da perna é o período no qual a pata está em voo para um novo empuxo; o estado de uma perna em transferência é 0.

A *fase de apoio* da perna é o período no qual a pata correspondente está em empuxo; o estado de uma perna em apoio é 1.

O *tempo total de ciclo* T é o tempo para um ciclo completo de locomoção de uma pata em um passo periódico.

O *fator de carga* β_i de uma pata i é a razão entre o tempo da fase de apoio e o tempo total de ciclo T .

$$\beta_i = \frac{\text{Tempo da fase de apoio da pata "i"}}{\text{Tempo total de ciclo (T)}} \quad (2.3.1)$$

A *fase* ϕ_i de uma pata é a razão entre o tempo decorrido desde o apoio da pata 0 dianteira esquerda até o apoio da pata i e o tempo total de ciclo T .

As andaduras mais utilizados para aplicações de robôs a pernas são as *andaduras simétricas*, onde, para um par de pernas à esquerda e um à direita, a diferença de fase entre elas é exatamente 180° ou $0,5 * T$. O andar e o correr humanos, e o salto do canguru, são exemplos de andaduras simétricas. Nos quadrúpedes, se o valor de ϕ de uma das patas traseiras é conhecido, todos os demais ϕ são determinados.



Figura 2.3-1: Robô Guará.

Uma andadura pode ser ainda classificada como: *singular* se quaisquer 2 ou mais eventos ocorrem simultaneamente em um mesmo ciclo de locomoção; *periódica* se os estados das pernas se repetem em intervalos regulares iguais ao tempo total de ciclo T para todas as pernas; e *regular* quando todas as pernas têm o mesmo fator de carga β .

A seleção da andadura depende de fatores, tais como: condições do terreno, estabilidade, facilidade de controle, suavidade do movimento do corpo, velocidade, mobilidade e potência (SONG; WALDRON, 1989). Em geral, as andaduras adequadas para andar em piso regular sem picos, vales, obstáculos ou regiões de baixa compactação são as periódicas, pelo fato de serem de implementação computacional mais simples.

A *andadura em onda* (*wave gait*) é utilizada por diversos animais e sua margem de estabilidade é ótima, indicando que os animais automaticamente otimizam seu passo para locomoção (SONG; WALDRON, 1989). Esta andadura é *regular* e é *simétrica*, ou seja, $\beta_i = \beta = \text{constante}$ para todas as patas e duas patas correspondentes em lados opostos, estarão 180° , ou $0.5 \cdot T$ fora de fase, conforme mostrado na Tabela 2.3-1.

Com os eventos de apoiar e elevar distribuídos uniformemente ao longo do tempo total de ciclo T , o passo é de *fase uniforme* e deve ser utilizado sempre que o sistema de acionamento do robô for sensível a picos de variação de potência (SONG; WALDRON, 1989).

Considerando os fatores de seleção abordados, a andadura escolhida para o Guará é a *regular simétrica em onda periódica de ciclo completo não singular*. Nessa andadura, os eventos de vôo para novo posicionamento e empuxo são simétricos e estão distribuídos em um ciclo, sendo que os eventos de vôo estão na seqüência das patas 3, 2, 1 e 0, Figura 2.3-1, iniciando com a pata traseira direita, 3, e terminando com a pata dianteira esquerda, 0, conforme mostrado na matriz da equação (2.3.2) e no diagrama da Figura 2.3-2.

Para um tempo fixo de acionamento t_s , é possível variar a velocidade da pata no ar, e conseqüentemente o tempo do percurso em vôo, variando-se o número de estágios da trajetória aérea e mantendo fixo o número de estágios da trajetória em apoio da pata. O ângulo de fase de uma pata i em relação à pata 0 é definido de maneira análoga, em número de acionamentos da pata 0 decorridos até o primeiro acionamento da pata i . Com esta estratégia de acionamento dos graus de liberdade, é possível variar o ângulo de fase e o fator de carga na montagem da matriz da andadura, e também realizar o acionamento diferencial das patas quando o robô estiver em uma trajetória curva em uma andadura qualquer.

$$P_D = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \dots \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (2.3.2)$$

$$\dots \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (4 \times 32)$$

Na andadura implementada para o Guar, Figura 2.3-2, trs ou mais patas sempre estaro apoiadas, o que permite que o rob caminhe em equilbrio estaticamente estvel.

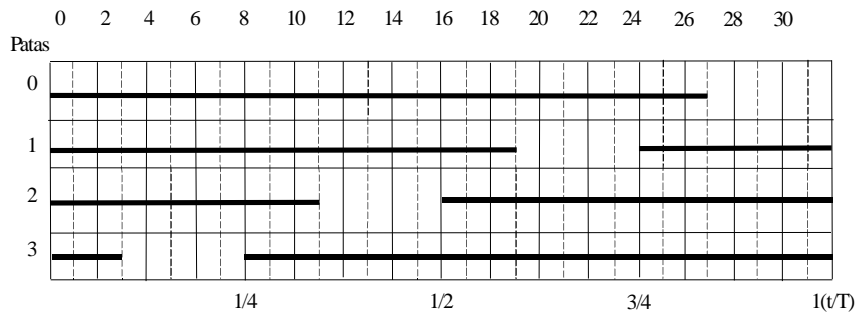


Figura 2.3-2: Andadura regular simtrica em onda perdica de ciclo completo.

O tempo total de cada fase  um mltiplo do nmero de colunas e o fator de carga pode ser expresso conforme mostrado na equao (2.3.3), onde 32  o nmero total de colunas e 27  o nmero de colunas correspondente ao perodo em que a pata est em empuxo.

$$\beta_i = \frac{27}{32} = 0.84375 \quad (2.3.3)$$

Um *estgio*  a menor diviso do tempo total de ciclo T que permite definir com preciso a fase ϕ_i . A andadura da matriz da equao (2.3.2) tem 32 estgios, dos quais 27 em empuxo e 5 em vo.

Conforme mostrado na matriz da equação (2.3.2), os ângulos de fase são:

Tabela 2.3-1: Ângulos de fase da andadura do Guará.

ϕ	<i>Pata 0</i>	<i>Pata 1</i>	<i>Pata 2</i>	<i>Pata 3</i>
t_{apoiar}/T	0	0,75000	0,50000	0,25000
t_{elevar}/T	0.84375	0,59375	0,34375	0,09375

onde t_{apoiar} e t_{elevar} são os tempos de pouso e vôo, respectivamente, e T é o tempo total da andadura.

A Figura 2.4-1 mostra a tela de *Definição da Andadura*, onde são definidos a andadura e os parâmetros que serão utilizados pelo modelo na geração das coordenadas de juntas do robô.

Na caixa *Andadura* é possível definir a *Trajétoria de empuxo* com ou sem movimento lateral; os *Fatores de vôo* para a compressão da altura e do empuxo; a posição da junta 2, *Joelho*, para execução da cinemática inversa; e o *Avanço ou retardo*, que permite deslocar longitudinalmente o percurso empuxo–vôo. Estes parâmetros foram importantes no início do trabalho, pois permitiram avaliar diferentes opções de andadura. Atualmente, com as opções definidas, eles permanecem fixos.

Ainda na caixa *Andadura*, têm-se a *Altura do robô*, que é utilizada para obter o alcance cinemático necessário para o empuxo do robô; a *Altura do passo*, que é a altura máxima da pata no percurso em vôo; o *Empuxo*, cujo valor será sempre menor ou igual ao valor máximo obtido em função da altura do robô; o *Número de pontos* para interpolação da trajetória das patas, que será sempre um múltiplo inteiro do número total de estágios da andadura; o fator de carga β calculado na andadura; e o valor máximo do *Movimento lateral* da plataforma.

2.4 Movimento de deslocamento lateral

O projeto do Guará levou em conta a necessidade de, além de implementar a andadura movimentando a perna em um plano vertical paralelo ao eixo longitudinal do robô, realizar um dos movimentos efetuados pelos mamíferos que é o deslocamento lateral do centro de gravidade, e que resulta no movimento da perna também em um plano vertical perpendicular ao eixo longitudinal do robô (MCMAHON, 1984).

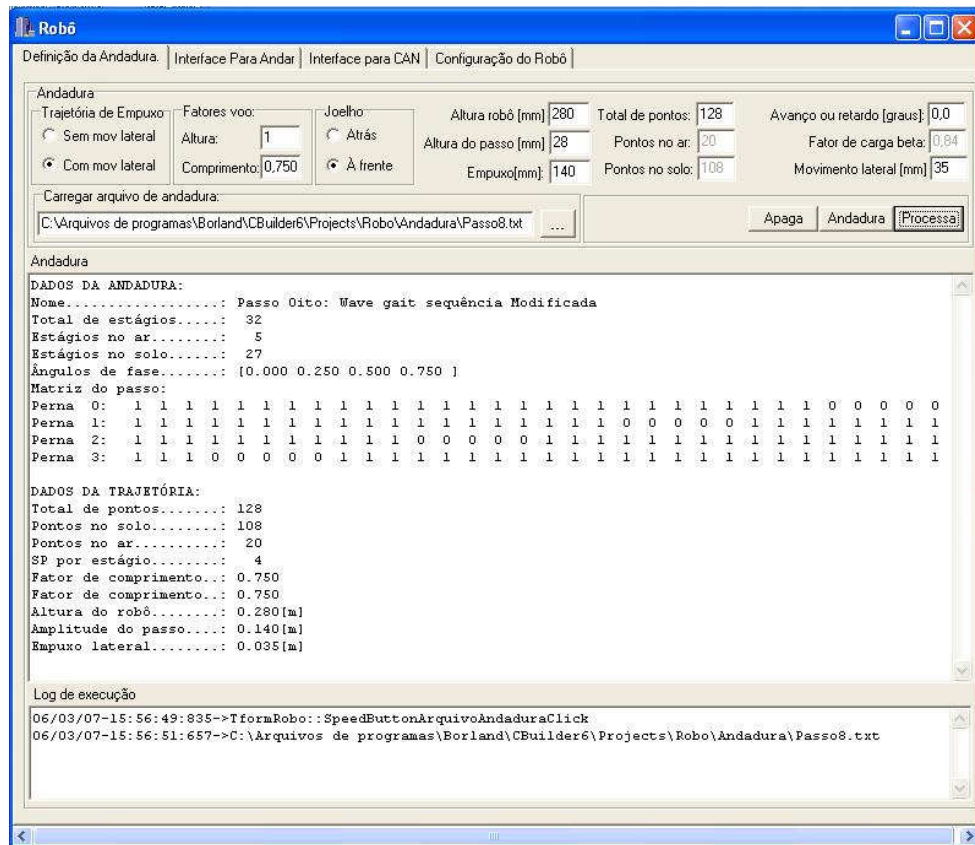


Figura 2.4-1: Tela de definição da andadura.

Nos humanos, que caminham em equilíbrio dinâmico, este movimento tem um período igual ao dobro do período do movimento longitudinal de cada perna, e é realizado durante o movimento longitudinal.

No Guará, que caminha em equilíbrio estaticamente estável, este movimento é efetuado durante o período em que o robô permanece com as quatro patas apoiadas, no início da andadura e entre os eventos de vôo das patas, e posiciona a plataforma à esquerda, para vôo das patas 2 e 3, e à direita, para vôo das patas 0 e 1.

Este movimento resulta no aumento da margem de estabilidade do robô, que é a menor distância da projeção vertical do CG aos lados do polígono de apoio, pois move o CG do robô para dentro do polígono.

O movimento é realizado em sincronismo com a matriz do passo, equação (2.3.2), conforme mostrado na Tabela 2.4-1, sendo: *PASSO*: número de ciclos empuxo-vôo completados; *NPT*: número total de pontos da trajetória; *NP4*: número de colunas da matriz com quatro patas em apoio; *NSP*: número de *setpoints* por coluna da matriz; e *NPV*: número de pontos da trajetória em vôo.

Tabela 2.4-1: Contadores de sincronismo para movimento lateral da plataforma.

Coluna(s)	Movimento	Limites do contador de colunas i	
		Inferior	Superior
0 a 2	Movendo para esquerda	$n_1 = \text{PASSO} * \text{NPT}$	$n_2 = \text{NP4} * \text{NSP}$
3 a 15	Parado na esquerda	n_2	$n_3 = n_2 + (2 * \text{NPV} * \text{NP4}) * \text{NSP}$
16 a 18	Movendo para direita	n_3	$n_4 = n_3 + \text{NP4} * \text{NSP}$
19 a 31	Parado na direita	n_4	$n_5 = (\text{PASSO} + 1) * \text{NPT}$

Os limites são incrementados sempre após completar um ciclo de empuxo - vôo completo porque a geração de coordenadas das patas no espaço operacional é feita incrementalmente e, com base no contador de colunas i e na excursão lateral máxima l , obtém-se o deslocamento lateral, conforme mostrado na Tabela 2.4-2.

Com a pata em apoio, a trajetória lateral, em relação à plataforma, imposta a ela resulta na trajetória lateral da plataforma, já que a pata no estágio de empuxo é a base e o seu ponto de fixação na plataforma é o efetuidor do manipulador que é a perna.

A Figura 2.5-1 mostra a geração do movimento lateral em sincronismo com a andadura, sendo (a) o gráfico do passo e (b) a coordenada de movimento lateral da pata em função do *set point*, e os contadores de sincronismo e de estágio do passo. As variáveis de sincronismo do movimento lateral, que são utilizadas para ligar, desligar e inverter o movimento lateral da plataforma são definidas na Tabela 2.4-1 e na Tabela 2.4-2, sendo l a excursão lateral máxima.

Tabela 2.4-2: Valores de deslocamento lateral em função do incremento de avanço do robô.

Intervalo de i	Deslocamento lateral
$n_1 \leq i < n_2$	$l * (1.0 - 2.0 * (i - n_2 + 1) / (n_3 - n_2))$
$n_2 \leq i < n_3$	l
$n_3 \leq i < n_4$	$l * (2.0 * (i - n_4 + 1) / (n_5 - n_4) - 1)$
$n_4 \leq i < n_5$	$-l$

Durante a geração da trajetória da andadura no espaço operacional, é gerada a coordenada de excursão lateral em função do *setpoint* em execução.

O movimento lateral da plataforma no intervalo de *setpoints* $(n1, n2]$ não é executado na primeira passada porque, antes de iniciar a andadura, o robô é posicionado com a plataforma na excursão lateral máxima à esquerda, definida pelo parâmetro *Movimento lateral* na tela de *Definição da andadura*, mostrada na Figura 2.4-1.

Nos intervalos $(n1, n2]$ e $(n3, n4]$ o movimento lateral é variável proporcionalmente à ordem do *setpoint*, em execução em relação ao comprimento do intervalo, e nos intervalos $(n2, n3]$ e $(n4, n5]$ o deslocamento lateral permanece constante e igual à máxima excursão lateral. Os intervalos são abertos à esquerda porque sempre iniciam com o último valor do movimento lateral do intervalo anterior para garantir uma melhor continuidade do movimento.

A coordenada lateral é adicionada ao modelo de andadura e os valores dos ângulos de giro da junta θ são determinados pela cinemática inversa do robô.

Esta estratégia permite qualquer configuração de movimento lateral em sincronismo com a andadura.

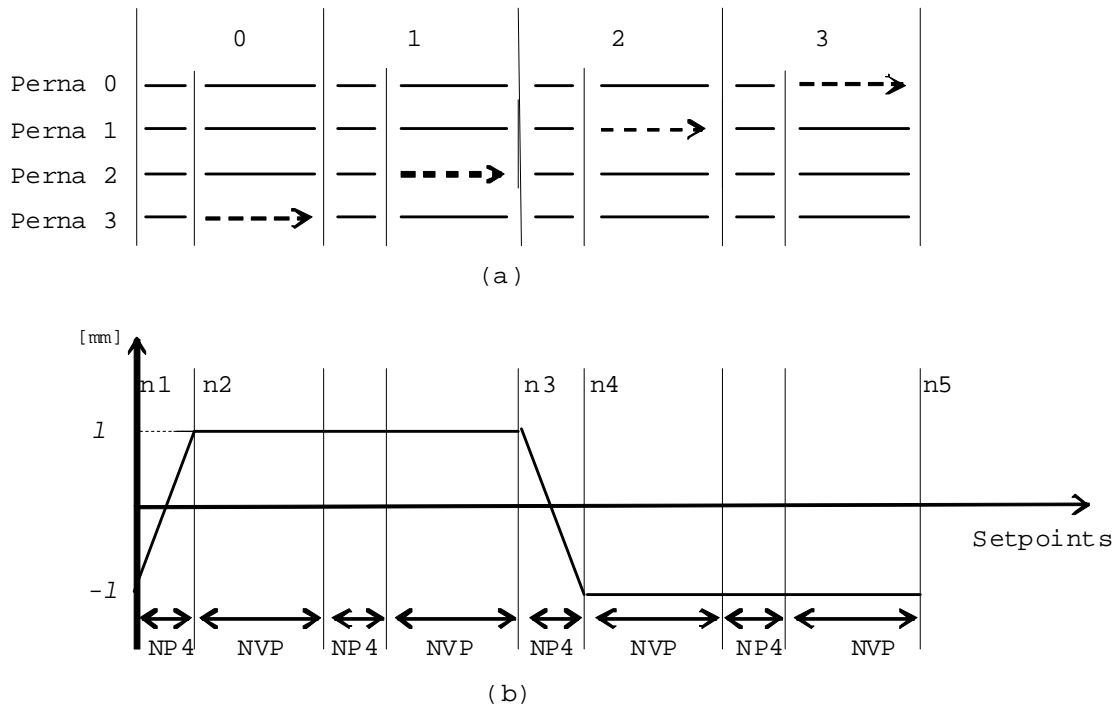
No Guará, optou-se por movimentar a plataforma paralelamente ao eixo, situando o vôo de duas patas do lado oposto em relação à trajetória plana da Figura 2.5-1(b).

É possível implementar uma trajetória com vôo de uma pata de cada vez de um lado, resultando em um movimento de guinada da plataforma, porém, com maior consumo de energia já que, nesse caso, será necessário o dobro de movimentos laterais a cada andadura, dos quais resultaria, ainda, uma menor estabilidade para o robô.

Este movimento seria cinematicamente semelhante ao executado para a trajetória em curva, mostrada na Figura 2.6-2, com as pernas dianteiras se movimentando contrariamente às traseiras em relação ao eixo transversal do robô, como se o centro de curvatura mudasse de um lado do robô para outro.

2.5 Geração da andadura para locomoção do robô

As andaduras periódicas permitem a sua implementação na forma matricial, conforme mostrado na matriz da equação (2.3.2). A andadura é composta de estágios e, nessa matriz, as linhas são associadas às pernas e as colunas aos estágios da andadura, sendo que o elemento $p_{ij} = 1$ indica que a pata i no estágio j está em empuxo e $p_{ij} = 0$ indica que ela está em vôo.



Legenda;

l : excursão lateral máxima

NP4: pontos com 4 patas apoiadas

NPV: pontos da trajetória em vôo

$n1$, $n2$, $n3$, $n4$ e $n5$ são os controladores de movimento lateral

Figura 2.5-1: Geração do movimento lateral em sincronismo com a andadura.

Na construção da matriz de andadura observa-se que com um fator de carga $\beta = 0,5$ apenas duas patas estarão em apoio simultaneamente, sendo necessário um fator de carga $\beta \geq 0,75$ para que seja possível três patas em apoio simultaneamente, e que, quanto maior o número de estágios com quatro patas no solo maior será o fator de carga β .

O aumento do fator de carga β aumenta o tempo da pata em empuxo e diminui o tempo disponível para vôo, o que, além de exigir maior potência de acionamento, resulta em um movimento ruidoso e descontínuo da pata.

Para permitir um estudo da relação ideal entre o número de estágios em cada fase da andadura, o programa de controle do robô dispõe também de uma biblioteca para o desenvolvimento de matrizes de andaduras, compatíveis com o esquema de geração das coordenadas no espaço operacional da pata.

No sistema de controle do Guará é possível construir novas andaduras através da interface *Nova Andadura* mostrada na Figura 2.5-2, variando-se o número de estágios no solo e no ar e

a posição relativa das seqüências de vôo das patas na matriz do passo, conforme detalhes na referida figura.

Na caixa *Parâmetros* da Figura 2.5-2 são especificados o número de estágios no solo e no ar, que definirão o fator de carga; a primeira pata a voar; e o menor empuxo na partida, caso esta ocorra com as patas na origem do eixo longitudinal do referencial da perna.

O número total de estágios é o número de colunas da planilha *Matriz do passo*, onde são especificadas as fases de apoio, com o valor da célula igual a 1, e vôo, com o valor da célula igual a 0, de cada pata, de acordo com as características desejadas para a andadura. O programa recalcula os ângulos de fase ϕ na aterrissagem e na decolagem, conforme mostrado na Tabela 2.3-1

A caixa *Empuxo* apresenta o valor do empuxo disponível na pata no momento de início do vôo, como uma fração do empuxo total. Se a partida do robô ocorrer com as patas na origem do eixo longitudinal do referencial da perna, ou seja, na metade do percurso de empuxo, o empuxo disponível considerado no processamento da primeira passada será igual 0.5.

Na caixa *Fases do vôo* são listadas, por perna, as colunas correspondentes ao vôo da pata; estes endereços são utilizados para sincronizar o movimento lateral da plataforma com a andadura.

Após analisar diversas combinações, concluiu-se que três estágios com quatro patas no solo, intercalados entre os eventos de vôo, são suficientes para permitir uma andadura com a excursão lateral máxima do robô, distribuindo-a simetricamente em relação ao eixo longitudinal, evitando assim acelerações que poderiam comprometer o equilíbrio do robô.

Se a partir da matriz de andadura for implementado um passo com apenas um *setpoint* por estágio, resultando em três incrementos para realizar a excursão completa, haverá um movimento lateral brusco porque todo o percurso será realizado em apenas três incrementos de extensão lateral.

Com três setpoints por estágio, resultando em 9 incrementos para realizar a excursão completa, resulta em um passo com 96 pontos e com 12 incrementos, resulta em um passo com 128 pontos, proporcionando, em ambos casos, um movimento de excursão lateral mais suave para o robô.

O modelo da andadura em linha reta é implementado em uma biblioteca especializada de funções que geram de maneira incremental as trajetórias desejadas no espaço operacional. Assim, coordenam-se os movimentos laterais, efetuados nos períodos de apoio em quatro patas, com os movimentos de empuxo e vôo das patas. As coordenadas geradas no modelo são estruturadas na forma matricial, ainda no espaço operacional das patas, e posteriormente mapeadas no espaço de juntas.

O modelo de andadura para a trajetória em curva utiliza funções específicas da mesma biblioteca, as quais geram os incrementos tangenciais e radiais no espaço operacional das patas. O grau de liberdade de rolagem da perna que realiza o movimento lateral é que possibilita o deslocamento radial da pata em relação à plataforma percorrendo a trajetória tangente à circunferência de curva. Os incrementos são gerados e estruturados na mesma forma matricial da andadura reta, ainda no espaço operacional das patas.

A execução da curva é obtida com a soma dos vetores da andadura reta e curva no espaço operacional das patas, em tempo de execução da andadura, não interferindo com o sincronismo dos movimentos de empuxo e vôo da pata e nem com o movimento lateral da plataforma, sendo posteriormente mapeadas no espaço de juntas pela cinemática inversa do robô.

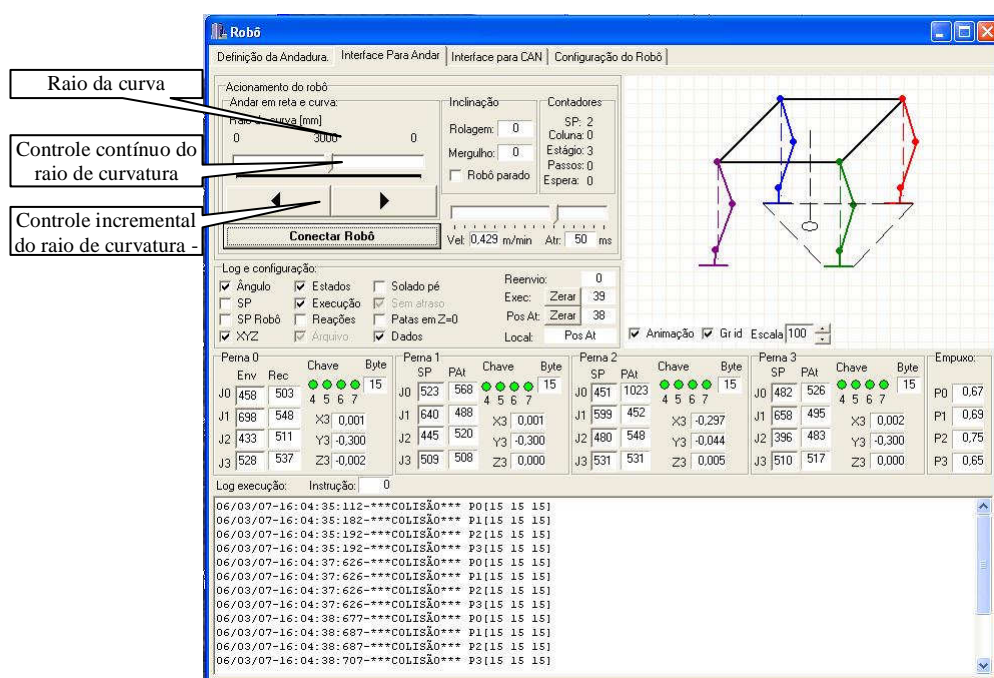


Figura 2.6-1: Tela de Interface para andar com destaque para os controles de andadura em curva.

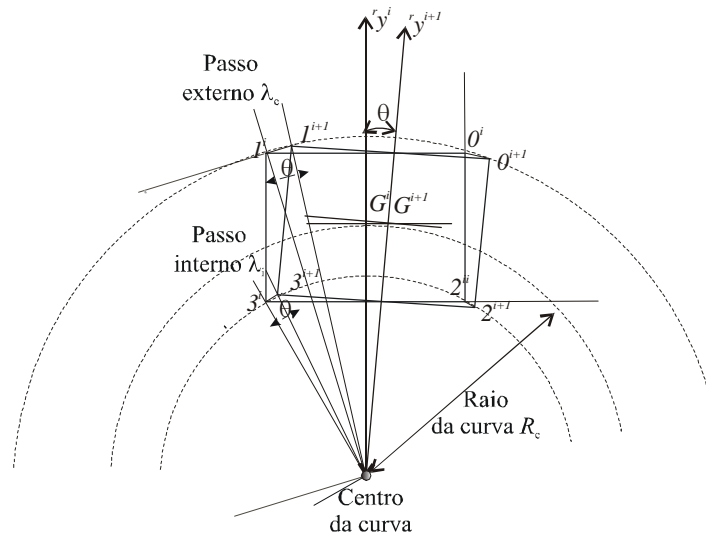


Figura 2.6-2: Geometria de curva com dirigibilidade nas quatro patas.

A trajetória curva é formada por segmentos de reta incrementais, conforme mostrado na Figura 2.6-2, porque, ao final de qualquer método ou curva de interpolação, o sistema de controle do robô irá gerar pequenos segmentos de reta para que a pata percorra uma curva qualquer.

Esta metodologia, desenvolvida para caminhar em curva com dirigibilidade nas quatro patas, utilizando segmentos de reta incrementais, permite o acionamento da perna no espaço operacional com os valores de saída do modelo, eliminando uma camada de processamento, o que, considerando o grande esforço computacional necessário para um robô a patas, representa uma considerável economia desse recurso.

Também é possível variar continuamente o raio de curvatura durante a andadura em curva, o que torna o robô dirigível e, além disso, os valores de empuxo podem ser ajustados e permitem variar o avanço de modo a tornar mais suaves os movimentos do robô na andadura com a guinada da plataforma em curva.

2.7 Discussão

O estudo da biomecânica permitiu o conhecimento de características importantes das andaduras utilizadas pelos mamíferos e pelos humanos, de forma a construir um robô capaz

de reproduzir, ao menos parcialmente, movimentos importantes utilizados pelos mamíferos e humanos.

O Guará foi desenvolvido e construído utilizando as pernas artrópodes do Roque (Ishiara, 1998), porém adicionando um grau de liberdade de rolagem e um de mergulho, habilitando-o a executar movimentos laterais e trajetórias curvas. O grau de liberdade de rolagem, em relação ao de guinada, foi escolhido por facilitar a execução de movimentos importantes utilizados pelos mamíferos e humanos.

Os modelos de mapeamento cinemático das pernas, são mostrados nos Anexos A4 e A5. A cinemática direta utiliza a convenção de Denavit Hattenberg e a cinemática inversa utiliza o mesmo princípio utilizado por BONITZ (1977), para tratar geometricamente a redundância de graus de liberdade da perna.

A andadura, baseada em um padrão tipo onda, executada graficamente sem movimento lateral da plataforma, pode ser controlada pelo empuxo disponível, ou limite cinemático, da pata ou de forma matricial.

Para possibilitar a execução do movimento lateral e a trajetória em curva, é mais interessante que a andadura seja definida de forma matricial, inclusive para incorporar trechos com quatro patas em apoio. O módulo de *Nova andadura* do programa de controle, Figura 2.5-2, permitiu a construção de diversas andaduras e a escolha da distribuição ideal das fases de empuxo e vôo de cada perna, em função da estabilidade do robô e da suavidade dos movimentos das pernas e da plataforma.

A execução do movimento lateral da plataforma permite incrementar a margem de estabilidade, pois desloca o CG do robô na direção do polígono de apoio. Isto se mostrou ainda mais útil porque o movimento é efetuado com as quatro patas no solo e, ao erguer uma pata, o CG já estará deslocado, favorecendo o equilíbrio com as outras três patas em apoio..

O sincronismo de movimento das patas em empuxo, importante para a estabilidade e o consumo de energia dos robôs a pernas, é assegurado pelo modo de geração incremental de coordenadas utilizado pelo modelo cinemático implementado, descrito no Capítulo 3. A cada intervalo, todas as coordenadas do modelo são atualizadas e é feito o sincronismo do movimento lateral.

A andadura em curva utiliza os vetores posição dos pontos de suporte das pernas à plataforma e o modelo cinemático considera que cada posição é resultante de um deslocamento e uma rotação em relação à posição anterior para determiná-la. Esta operação, efetuada

geometricamente, elimina uma camada de cálculo, sendo as componentes transversais e longitudinais, associadas às trajetórias tangenciais das patas em relação à plataforma, adicionada à andadura no espaço operacional da perna.

Capítulo 3

Cinemática

3.7 Introdução

Neste capítulo são apresentados os modelos cinemáticos utilizados no controle e supervisão do robô.

Para se locomover, o robô necessita das coordenadas do ponto de apoio onde irá pisar e da trajetória de vôo para novo posicionamento, para então acionar precisamente as patas, como manipuladores isolados, para as coordenadas de apoio ou de vôo.

O modelo cinemático fornece a orientação da plataforma, a partir das coordenadas de juntas medidas nos potenciômetros das pernas em apoio, de forma a estabelecer as coordenadas de apoio para as quais as patas serão comandadas, a posição do CG em relação ao plano de apoio, a margem de estabilidade e a velocidade do robô (BENTO FILHO; et al., 2001), sendo possível, também, obter a localização aproximada do robô (BENTO FILHO; et al., 2004).

Estes modelos foram desenvolvidos com recursos de geometria analítica, análise e mecânica vetorial; não foram encontradas publicações específicas de modelos utilizados em outros robôs quadrúpedes e a forma como foram desenvolvidos.

O sistema de controle do robô coordena o movimento das patas definindo as posições para as quais as patas devem ser acionadas, a partir da orientação ou rota a ser seguida pelo robô, mapeando também este movimento no espaço de juntas e enviando os valores de *setpoints* para acionamento das pernas.

3.8 Orientação da plataforma do robô

A Figura 3.8-1 mostra uma foto do Guará executando a andadura reta, com a plataforma deslocada para a direita para, voo da pata 1 e a Figura 3.8-2 apresenta um diagrama esquemático com a pata 3 já em voo, com a finalidade de mostrar os sistemas de coordenadas e as variáveis da cinemática do robô.

O robô dispõe de quatro pernas com quatro graus de liberdade cada uma, que funcionam como se fossem manipuladores cuja base é a plataforma do robô, e cujo espaço operacional é definido pela trajetória da extremidade da pata.

Durante a fase de apoio, a trajetória ideal da extremidade da pata em relação à plataforma é uma reta em um plano paralelo à plataforma. Durante a fase de voo para novo posicionamento, a trajetória inicialmente implementada para a pata foi uma cicloide achatada, porém, devido a ocorrência de empuxo reverso e choque nos momentos de decolagem e pouso, respectivamente, foi implementada posteriormente uma trajetória poligonal, conforme mostrado no item 4.3.3.4.1.

O modelo cinemático é construído com o robô apoiado em um plano horizontal.

Para efeito da análise cinemática, a plataforma e as pernas do robô são consideradas um sistema de corpos rígidos, sendo que os pontos de apoio das patas formam sempre um plano que será paralelo à superfície de apoio se nenhuma das pernas estiver apoiada em uma elevação ou depressão no piso.

Na partida, o robô posiciona a plataforma a uma altura h_0 do solo, comandando ângulos iguais para juntas equivalentes de cada perna, de acordo com a cinemática inversa.

Na Figura 3.8-2, L e W são o comprimento e a largura entre pontos de fixação das pernas; $G^r x^r y^r z^r$ é o *referencial robô* instalado a bordo e fixo no centro de gravidade; ${}^0 x^0 y^0 z^0$ é o *referencial da perna 0* fixo no ponto de conexão da perna ao robô; G e G_{xy} são o *centro de gravidade (CG)* do robô e sua projeção no plano de apoio; e as pernas são numeradas a partir da *dianteira esquerda*, número 0, *traseira esquerda*, número 1, *dianteira direita*, número 2 e *traseira direita*, número 3.

A Figura 3.8-3 mostra a configuração dos quatro graus de liberdade da perna e os respectivos sistemas de referência, de acordo com a convenção de Denavit-Hatenberg. A perna foi construída de maneira que os graus de liberdade 0 e 1 tenham a mesma origem, com o comprimento do primeiro elo nulo. Os graus de liberdade são: rolagem e mergulho da junta 0 , e rolagem das juntas 1 , 2 e 3 .

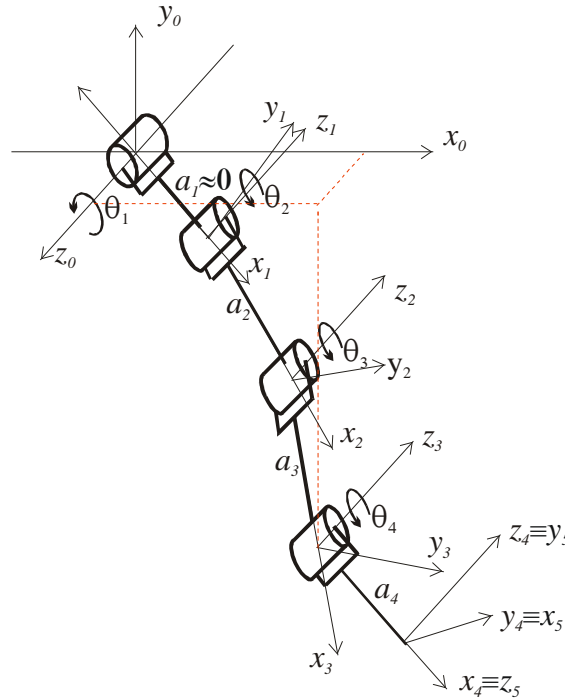


Figura 3.8-3: Sistemas de referência das juntas, conforme convenção de Denavit-Hatenberg.

Na partida do repouso, instante $i = 0$ indicado pelo super-índice 0 à direita, definimos o referencial ${}^0x^{00}y^{00}z^0$, fazendo coincidirem os eixos ${}^0z^0$ do referencial fixo e ${}^r_z^0$ do referencial robô e mantendo-se paralelos, e com mesmo sentido, os eixos ${}^0x^0//{}^fx^0$ e ${}^0y^0//{}^fy^0$ do referencial fixo e do referencial robô, respectivamente, conforme Figura 3.8-2, de maneira semelhante à utilizada por LIN (LIN; SONG, 1999).

Em relação ao referencial robô, as coordenadas dos pontos de fixação das pernas 0 , 1 , 2 e 3 , respectivamente, são dadas na equação (3.1), onde o sub-índice e o super-índice esquerdo identificam o número da junta e o referencial, respectivamente, e o sub-índice direito identifica a perna, conforme Figura 3.8-2.

$$\begin{aligned}
({}^r_0x_0, {}^r_0y_0, {}^r_0z_0) &= \left(\frac{L}{2}, \frac{w}{2}, 0 \right) \\
({}^r_0x_1, {}^r_0y_1, {}^r_0z_1) &= \left(-\frac{L}{2}, \frac{w}{2}, 0 \right) \\
({}^r_0x_2, {}^r_0y_2, {}^r_0z_2) &= \left(\frac{L}{2}, -\frac{w}{2}, 0 \right) \\
({}^r_0x_3, {}^r_0y_3, {}^r_0z_3) &= \left(-\frac{L}{2}, -\frac{w}{2}, 0 \right)
\end{aligned} \tag{3.1 a, b, c, d}$$

Em relação ao referencial da perna, as coordenadas dos pontos de apoio das patas em relação à plataforma são definidas por:

$$\begin{aligned}
{}^0p_0 &= ({}^0_4x_0, {}^0_4y_0, {}^0_4z_0) \\
{}^1p_1 &= ({}^1_4x_1, {}^1_4y_1, {}^1_4z_1) \\
{}^2p_2 &= ({}^2_4x_2, {}^2_4y_2, {}^2_4z_2) \\
{}^3p_3 &= ({}^3_4x_3, {}^3_4y_3, {}^3_4z_3)
\end{aligned} \tag{3.2 a, b, c, d}$$

onde o super-índice e o sub-índice de ${}^r p_m$ são o referencial e a identificação da pata, respectivamente.

Em relação ao referencial robô, temos:

$$\begin{aligned}
{}^r p_0 &= \left[{}^r_4x_0 + \frac{L}{2} \quad {}^r_4y_0 + \frac{W}{2} \quad {}^r_4z_0 \right]^t \\
{}^r p_1 &= \left[{}^r_4x_1 - \frac{L}{2} \quad {}^r_4y_1 + \frac{W}{2} \quad {}^r_4z_1 \right]^t \\
{}^r p_2 &= \left[{}^r_4x_2 + \frac{L}{2} \quad {}^r_4y_2 - \frac{W}{2} \quad {}^r_4z_2 \right]^t \\
{}^r p_3 &= \left[{}^r_4x_3 - \frac{L}{2} \quad {}^r_4y_3 - \frac{W}{2} \quad {}^r_4z_3 \right]^t
\end{aligned} \tag{3.3 a, b, c, d}$$

onde, conforme Figura 3.8-2, W é a distância entre os pontos de fixação de duas patas em lados distintos e L a distância entre os pontos de fixação de duas patas em um mesmo lado, respectivamente, e:

$$\begin{Bmatrix} {}^r_4x_i \\ {}^r_4y_i \\ {}^r_4z_i \end{Bmatrix} = [{}^{ir}R] \begin{Bmatrix} {}^i_4x_i \\ {}^i_4y_i \\ {}^i_4z_i \end{Bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} {}^i_4x_i \\ {}^i_4y_i \\ {}^i_4z_i \end{Bmatrix} \quad (3.4)$$

onde o sub-índice e o super-índice esquerdo em r_4x_i identificam o número da junta e o referencial robô, e o sub-índice direito identifica a pata; o sub-índice e o super-índice esquerdo em i_4x_i identificam o número da junta e o referencial da perna, e o sub-índice direito identifica a pata; e $[{}^{ir}R]$ é a matriz constante que dá a rotação do referencial da perna em relação ao referencial do robô, sendo que suas colunas são as coordenadas do referencial robô em relação ao referencial da perna, conforme Figura 3.8-2.

Existem 4 combinações possíveis de 3 patas no solo: 012 , 013 , 023 e 123 , mostradas na Figura 3.8-4(a, b, c e d) a seguir: (a) patas 0, 1 e 2; (b) patas 0, 1 e 3; (c) patas 0, 2 e 3 e (d) patas 1, 2 e 3. Apesar de dispostos de forma retangular os pontos de apoio podem ocupar qualquer coordenada no espaço de alcance das pernas no plano de apoio.

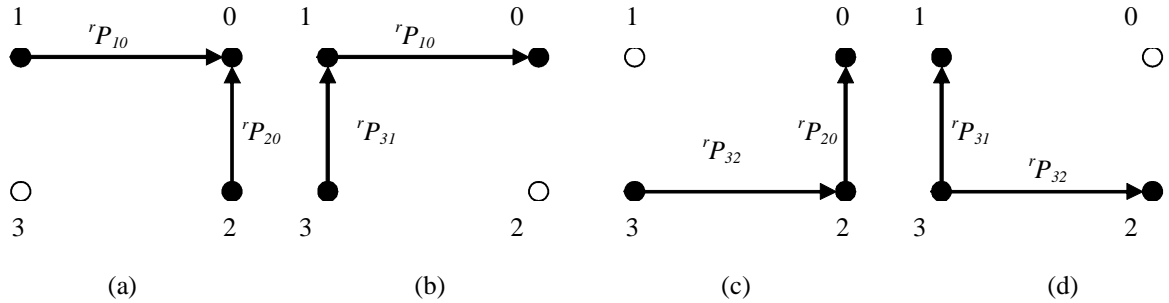


Figura 3.8-4: Configurações de apoio com 3 patas no solo.

Define-se os vetores posição de um ponto de apoio em relação a outro, orientados de acordo com o referencial robô, de acordo com as configurações mostradas na Figura 3.8-2, conforme a seguir:

$$\begin{aligned} {}^rP_{10} &= {}^rP_0 - {}^rP_1 \\ {}^rP_{20} &= {}^rP_0 - {}^rP_2 \\ {}^rP_{31} &= {}^rP_1 - {}^rP_3 \\ {}^rP_{32} &= {}^rP_2 - {}^rP_3 \end{aligned} \quad (3.5 \text{ a, b, c, d})$$

onde o super-índice esquerdo e o sub-índice direito do vetor ${}^r p_{mn}$ identificam o referencial e o sentido do vetor posição, respectivamente.

Os vetores posição unem os pontos de apoio e definem o plano de apoio das patas. Efetuando o produto vetorial dos vetores posição nas configurações da Figura 3.8-4, na ordem dos vetores paralelos para os perpendiculares ao eixo longitudinal da plataforma, de acordo com a regra da mão direita, obtemos o vetor normal ao plano de apoio, em cada configuração de apoio, na forma:

$$\begin{aligned} {}^{012}p &= {}^r p_{10} \times {}^r p_{20} = [S({}^r p_{10})] [{}^r p_{20}] \\ {}^{013}p &= {}^r p_{10} \times {}^r p_{31} = [S({}^r p_{10})] [{}^r p_{31}] \\ {}^{023}p &= {}^r p_{32} \times {}^r p_{20} = [S({}^r p_{32})] [{}^r p_{20}] \\ {}^{123}p &= {}^r p_{32} \times {}^r p_{31} = [S({}^r p_{32})] [{}^r p_{31}] \end{aligned} \quad (3.6 \text{ a, b, c, d})$$

onde

$$[S({}^r p_{ij})] = \begin{bmatrix} 0 & -{}^r p_{ij}^x & {}^r p_{ij}^y \\ {}^r p_{ij}^x & 0 & -{}^r p_{ij}^z \\ -{}^r p_{ij}^y & {}^r p_{ij}^z & 0 \end{bmatrix} \quad (3.7)$$

Os vetores unitários são então:

$${}^{mnp}n = \frac{{}^{mnp}p}{\sqrt{{}^{mnp}p \cdot {}^{mnp}p}} = \begin{bmatrix} {}^{mnp}n^x & {}^{mnp}n^y & {}^{mnp}n^z \end{bmatrix}^t \quad (3.8)$$

onde $mnp = 012, 013, 023$ ou 123 , ${}^{mnp}n$ é o vetor unitário na direção normal ao plano de apoio, ${}^{mnp}n^x$, ${}^{mnp}n^y$ e ${}^{mnp}n^z$ são os seus co-senos diretores e o super-índice à direita em ${}^{mnp}n^x$ identifica a componente x , y ou z do vetor ${}^{mnp}n$.

Define-se os vetores unitários na direção ${}^{mnp}x$ a partir dos vetores posição que unem os pontos de apoio de 2 patas de um mesmo lado do robô em cada configuração, conforme a seguir:

$$\begin{aligned}
{}_{012}^r x &= \frac{{}_{10}^r p}{\sqrt{{}_{10}^r p \cdot {}_{10}^r p}} = \begin{bmatrix} {}_{012}^r x^x & {}_{012}^r x^y & {}_{012}^r x^z \end{bmatrix}^t \\
{}_{013}^r x^r &= {}_{012}^r x; \\
{}_{023}^r x &= \frac{{}_{32}^r p}{\sqrt{{}_{32}^r p \cdot {}_{32}^r p}} = \begin{bmatrix} {}_{023}^r x^x & {}_{023}^r x^y & {}_{023}^r x^z \end{bmatrix}^t \\
{}_{123}^r x &= {}_{023}^r x
\end{aligned} \tag{3.9 a, b, c, d}$$

O vetor unitário ${}_{mnp}^r y$, que completa o referencial, é obtido pelo produto vetorial do vetor unitário normal ${}_{mnp}^r n$ de (3.8) pelo vetor unitário ${}_{mnp}^r x$ na direção x , obtido de (3.9), conforme a seguir:

$$\begin{aligned}
{}_{mnp}^r y &= {}_{mnp}^r n \times {}_{mnp}^r x = \begin{bmatrix} -{}_{mnp}^r n^x {}_{mnp}^r x^y + {}_{mnp}^r n^y {}_{mnp}^r x^x \\ {}_{mnp}^r n^x {}_{mnp}^r x^z - {}_{mnp}^r n^z {}_{mnp}^r x^x \\ -{}_{mnp}^r n^y {}_{mnp}^r x^z + {}_{mnp}^r n^z {}_{mnp}^r x^y \end{bmatrix} = \begin{bmatrix} {}_{mnp}^r y^x & {}_{mnp}^r y^y & {}_{mnp}^r y^z \end{bmatrix}^t \\
{}_{023}^r y^r &= {}_{012}^r y \\
{}_{123}^r y &= {}_{013}^r y
\end{aligned} \tag{3.10 a, b, c}$$

A matriz de rotação do plano de apoio em relação ao referencial robô, cujas colunas são as componentes dos vetores unitários nas direções x , y e z do referencial robô, é dada por:

$$\begin{bmatrix} {}_{i-1}^r R^i \end{bmatrix}_{mnp} = \begin{bmatrix} {}_{mnp}^r x^x & {}_{mnp}^r y^x & {}_{mnp}^r n^x \\ {}_{mnp}^r x^y & {}_{mnp}^r y^y & {}_{mnp}^r n^y \\ {}_{mnp}^r x^z & {}_{mnp}^r y^z & {}_{mnp}^r n^z \end{bmatrix} \tag{3.11}$$

onde o sub-índice à esquerda e o super-índice à direita e de $[{}_{i-1}^r R^i]_{mnp}$ identificam a matriz de rotação do referencial robô entre dois acionamentos consecutivos $i-1$ e i .

A matriz de rotação (3.11) foi obtida a partir das coordenadas dos pontos de apoio das patas em relação ao referencial robô.

Conforme definido anteriormente, na partida do repouso, definimos o referencial ${}^0x^0y^0z^0$, fazendo coincidirem os eixos ${}^0z^0$ do referencial fixo e ${}^r z^0$ do referencial robô e mantendo-se paralelos, e com mesmo sentido, os eixos ${}^0x^0//{}^r x^0$ e ${}^0y^0//{}^r y^0$ do referencial fixo e do referencial robô, respectivamente, conforme Figura 3.8-2. Então, entre os instantes 0 da partida, com o robô ainda em repouso, e 1 , imediatamente após o primeiro acionamento de movimento das patas, a matriz (3.11) representa também a rotação da plataforma em relação ao referencial fixo.

Após o segundo acionamento, a matriz de rotação obtida será a rotação do referencial robô em relação a ele próprio na posição anterior, após o primeiro acionamento. Assim, sucessivamente, após o acionamento $i-1$ a matriz de rotação obtida será a rotação do referencial robô em relação a ele próprio, ocorrida entre os acionamentos $i-1$ e i .

A matriz de rotação da plataforma em relação ao referencial fixo após o acionamento i será dada por:

$$\begin{bmatrix} R_{mnp}^i \end{bmatrix} = \begin{bmatrix} R_{mnp}^0 \end{bmatrix} \begin{bmatrix} R_{mnp}^1 \end{bmatrix} \begin{bmatrix} R_{mnp}^2 \end{bmatrix} \dots \begin{bmatrix} R_{mnp}^i \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}_{mnp} \quad (3.12)$$

Onde a matriz $[R_{mnp}^0]$ é a matriz de rotação da plataforma em relação à horizontal no instante 0. Esta matriz é matriz identidade de ordem 3, se o robô está apoiado em um plano horizontal no instante 0.

A matriz de rotação da plataforma em função dos ângulos de rolagem ϕ , de mergulho θ e de guinada ψ , é dada por (SPONG; VIDYASAGAR, 1992):

$$\begin{aligned} \begin{bmatrix} R^i \end{bmatrix} &= \begin{bmatrix} R_{z,\phi} R_{y,\theta} R_{x,\psi} \end{bmatrix} \\ &= \begin{bmatrix} c^a \phi & c^a \theta & -s^a \phi & c^a \psi + c^a \phi s^a \theta s^a \psi & s^a \phi s^a \psi + c^a \phi s^a \theta c^a \psi \\ s^a \phi c^a \theta & c^a \phi c^a \psi + s^a \phi s^a \theta s^a \psi & -c^a \phi s^a \psi + s^a \phi s^a \theta c^a \psi \\ -s^a \theta & c^a \theta s^a \psi & c^a \theta c^a \psi \end{bmatrix} \end{aligned} \quad (3.13)$$

Igualando-se os termos de (3.12) e (3.13), sabendo que os termos da diagonal de (3.12) não são nulos, poderemos obter os ângulos de rolagem, mergulho e guinada em relação ao plano de apoio como:

$$\begin{aligned} {}^a_{mnp} \psi &= \operatorname{atan} \left(\frac{{}_{mnp} r_{32}}{{}_{mnp} r_{33}} \right) \\ {}^a_{mnp} \theta &= -\operatorname{asen}({}_{mnp} r_{31}) \\ {}^a_{mnp} \phi &= \operatorname{asen} \left(\frac{{}_{mnp} r_{21}}{{}_{mnp} r_{11}} \right) \end{aligned} \quad (3.14 \text{ a, b, c})$$

onde o super-índice a à esquerda significa que os ângulos são em relação ao plano de apoio.

3.3 Plano de apoio das patas

Para obtermos a equação do plano de apoio das patas, basta efetuar o produto escalar de qualquer vetor do plano com a sua normal e igualá-lo a zero (Efimov, 1987). Toma-se o vetor que une o ponto de apoio de uma das patas e um ponto qualquer no plano de apoio, conforme a seguir:

$${}^r_{mnp}n \cdot {}^r_{mnp}p = {}^r_{mnp}n^x({}^r_x - {}^r_{mnp}x^x) + {}^r_{mnp}n^y({}^r_y - {}^r_{mnp}y^x) + {}^r_{mnp}n^z({}^r_z - {}^r_{mnp}x^z) = 0 \quad (3.15)$$

onde ${}^r_{mnp}p$, é o vetor posição do ponto, $({}^r_{mnp}x, {}^r_{mnp}y, {}^r_{mnp}z)$ e $({}^r_x, {}^r_y, {}^r_z)$ são um ponto de apoio e um ponto qualquer no plano de apoio, respectivamente, e ${}^r_{mnp}n$ é a normal ao plano de apoio.

Reduzindo-se (3.15) à forma geral da equação do plano, tem-se:

$${}^x_{mnp}n^x {}^r_x + {}^x_{mnp}n^y {}^r_y + {}^x_{mnp}n^z {}^r_z - ({}^x_{mnp}n^x {}^r_{mnp}x^x + {}^x_{mnp}n^y {}^r_{mnp}x^y + {}^x_{mnp}n^z {}^r_{mnp}x^z) = 0 \quad (3.16)$$

onde:

$${}^r_{mnp}p = ({}^x_{mnp}n^x {}^r_{mnp}x^x + {}^x_{mnp}n^y {}^r_{mnp}x^y + {}^x_{mnp}n^z {}^r_{mnp}x^z) \quad (3.17)$$

O valor numérico de ${}^r_{mnp}p$ é obtido substituindo-se em (3.17) as coordenadas de um dos pontos do plano de apoio.

3.4 Posição da projeção do CG no plano de apoio

A projeção do CG no plano de apoio é definida pelo ponto de interseção de uma reta normal ao plano de apoio que contenha o CG do robô.

As equações canônicas de uma reta paralela à normal que contém o CG, em cada uma das configurações de apoio das patas, tendo em conta que o CG é a origem do referencial robô, são dadas por (Efimov, 1987):

$$\frac{{}^r_x}{{}^r_{mnp}n^x} = \frac{{}^r_y}{{}^r_{mnp}n^y} = \frac{{}^r_z}{{}^r_{mnp}n^z} = {}^r_{mnp}t \quad (3.18)$$

Tomando-se os valores de ${}^r x$, ${}^r y$ e ${}^r z$ de cada uma das equações (3.18) e substituindo na equação do plano referente à configuração, dada de forma geral em (3.16), obteremos os valores de ${}_{mnp}{}^r t$ conforme a seguir:

$${}_{mnp}{}^r t = \frac{\left({}_{mnp}{}^x n^x {}_{mnp}{}^r x^x + {}_{mnp}{}^x n^y {}_{mnp}{}^r x^y + {}_{mnp}{}^x n^z {}_{mnp}{}^r x^z \right)}{{}_{mnp}{}^r n^x {}_{mnp}{}^r n^x + {}_{mnp}{}^r n^y {}_{mnp}{}^r n^y + {}_{mnp}{}^r n^z {}_{mnp}{}^r n^z} = {}_{mnp}{}^r p \quad (3.19)$$

Voltando com os valores de ${}_{mnp}{}^r t$ de (3.19) em (3.18), obteremos as seguintes coordenadas para a projeção do CG no plano de apoio:

$$G_{xy} = \begin{bmatrix} {}_{mnp}{}^r x & {}_{mnp}{}^r y & {}_{mnp}{}^r z \end{bmatrix}_{G_{xy}} = {}_{mnp}{}^r t \begin{bmatrix} {}_{mnp}{}^r n^x & {}_{mnp}{}^r n^y & {}_{mnp}{}^r n^z \end{bmatrix} \quad (3.20)$$

3.5 Distância da projeção vertical do CG aos lados do polígono de apoio

Os vetores unitários na direção dos vetores posição, dados por (3.1), são obtidos conforme a seguir:

$${}_{mn}{}^r u = \frac{{}_{mn}{}^r p}{\sqrt{{}_{mn}{}^r p \cdot {}_{mn}{}^r p}} = \begin{bmatrix} {}_{mn}{}^r u^x & {}_{mn}{}^r u^y & {}_{mn}{}^r u^z \end{bmatrix}^t \quad (3.21)$$

onde ${}_{mn}{}^r u$ é o vetor unitário e ${}_{mn}{}^r u^x$, ${}_{mn}{}^r u^y$ e ${}_{mn}{}^r u^z$ são os cossenos diretores.

Uma rotação de 90° de qualquer vetor em torno da normal n ao plano de apoio é definida conforme a seguir (SPONG; VIDYASAGAR, 1992):

$${}_{mn}{}^r R_{n,\theta} = \begin{bmatrix} \left({}_{mn}{}^r n^x \right)^2 & {}_{mn}{}^r n^x {}_{mn}{}^r n^y - {}_{mn}{}^r n^z & {}_{mn}{}^r n^x {}_{mn}{}^r n^z + {}_{mn}{}^r n^y \\ {}_{mn}{}^r n^x {}_{mn}{}^r n^y + {}_{mn}{}^r n^z & \left({}_{mn}{}^r n^y \right)^2 & {}_{mn}{}^r n^y {}_{mn}{}^r n^z - {}_{mn}{}^r n^x \\ {}_{mn}{}^r n^x {}_{mn}{}^r n^z + {}_{mn}{}^r n^y & {}_{mn}{}^r n^y {}_{mn}{}^r n^z + {}_{mn}{}^r n^x & \left({}_{mn}{}^r n^z \right)^2 \end{bmatrix} \quad (3.22)$$

Os vetores diretores das normais aos vetores posição, são então obtidos por:

$${}_{mn}{}^r d = \begin{bmatrix} {}_{mn}{}^r R_{n,\theta} \end{bmatrix} {}_{mn}{}^r u \quad (3.23)$$

onde ${}_{mn}{}^r d$ é o vetor diretor nas direções perpendiculares aos vetores posição definidos em (3.1).

As equações canônicas das retas normais aos vetores posição e que passam pela projeção do CG, em cada uma das configurações de apoio das patas, são então as seguintes (Efimov, 1987):

$$\frac{{}^r x - {}^{mnp}x_G}{{}_{mn}d^x} = \frac{{}^r y - {}^{mnp}y_G}{{}_{mn}d^y} = \frac{{}^r z - {}^{mnp}z_G}{{}_{mn}d^z} = {}_{mn}^r t \quad (3.24)$$

onde ${}_{mn}^r d^x$, ${}_{mn}^r d^y$ e ${}_{mn}^r d^z$ são os co-senos diretores dos vetores em (3.23). Individualmente, temos:

$$\begin{aligned} {}_{mn}^r x &= {}^{mnp}x_G - {}_{mn}^r t_p {}_{mn}^r d_p^x \\ {}_{mn}^r y &= {}^{mnp}y_G - {}_{mn}^r t_p {}_{mn}^r d_p^y \\ {}_{mn}^r z &= {}^{mnp}z_G - {}_{mn}^r t_p {}_{mn}^r d_p^z \end{aligned} \quad (3.25)$$

Tomando-se os valores de ${}_{mn}^r x$, ${}_{mn}^r y$ e ${}_{mn}^r z$ de cada uma das equações (3.25) e substituindo na equação do plano referente à configuração, dada de forma geral em (3.16), obteremos os valores de ${}_{mn}^r t$ conforme a seguir:

$${}_{mn}^r t_p = \frac{{}^{mnp}x_G {}_{mn}^r n^x + {}^{mnp}y_G {}_{mn}^r n^y + {}^{mnp}z_G {}_{mn}^r n^z - {}^{mnp}p}{{}_{mn}^r d_p^x {}_{mn}^r n^x + {}_{mn}^r d_p^y {}_{mn}^r n^y + {}_{mn}^r d_p^z {}_{mn}^r n^z} \quad (3.26)$$

As coordenadas dos pontos de interseção das normais com os respectivos vetores posição são obtidas de (3.25) com o valor de ${}_{mn}^r t_p$ obtido de (3.19) e as distâncias entre estes pontos e a projeção do CG no plano de apoio são:

$${}_{mn}^r d = \sqrt{\left({}_{mn}^r x - {}^{mnp}x_G\right)^2 + \left({}_{mn}^r y - {}^{mnp}y_G\right)^2 + \left({}_{mn}^r z - {}^{mnp}z_G\right)^2} \quad (3.27)$$

3.7 Geometria da trajetória em curva

A Figura 3.7-1 mostra a geometria do robô realizando um incremento de arco θ em uma curva de raio R_c sendo: R_e e R_i os raios de curvatura externo e interno à curva; θ o incremento de arco para um acionamento; λ_e e λ_i as amplitudes do acionamento das pernas

externa e interna; W e L são o comprimento e a largura do robô; e os números identificam as patas nos instantes i e $i+1$.

As patas são comandadas para as posições $0'1'2'3'$ que correspondem a um arco de θ graus.

O percurso das patas externas e internas deve ser conforme a seguir:

$$\begin{aligned}\lambda_e &= \theta R_e = \theta \left[\left(R_c + W/2 \right)^2 + \left(L/2 \right)^2 \right]^{1/2} \\ \lambda_i &= \theta R_i = \theta \left[\left(R_c - W/2 \right)^2 + \left(L/2 \right)^2 \right]^{1/2}\end{aligned}\quad (3.28)$$

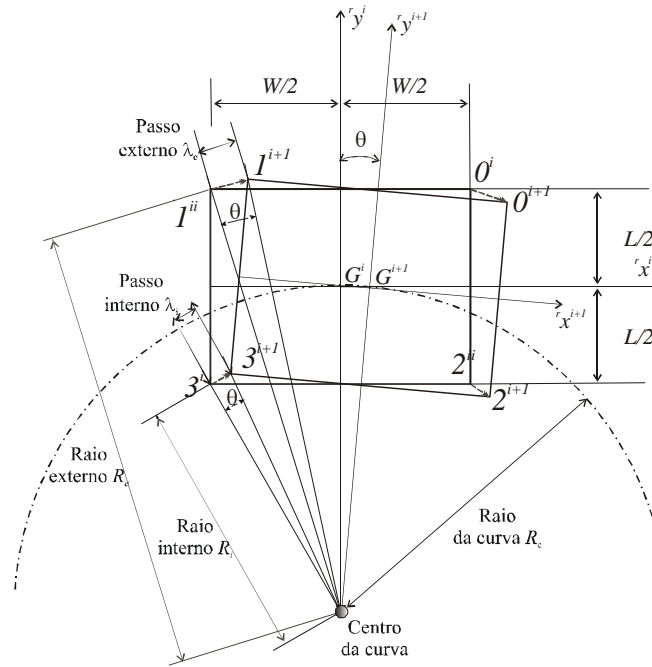


Figura 3.7-1: Parâmetros geométricos para definição da trajetória curva.

A amplitude do passo é sempre determinada para o percurso do centróide do robô. Para realizar uma curva de raio R_c à direita, conforme mostrado na Figura 3.7-1, o robô corrige o comprimento do percurso apoiado das patas externas 0 e 1 de acordo com a equação (3.31).

$$\begin{aligned}
{}^r \begin{bmatrix} x \\ y \\ z \end{bmatrix}^i &= \begin{bmatrix} {}^r x^{i-1} \\ {}^r y^{i-1} \\ {}^r z^{i-1} \end{bmatrix}^{i-1} + \begin{bmatrix} \lambda_e \cos \theta \\ \lambda_e \sin \theta \\ 0 \end{bmatrix} \\
{}^r \begin{bmatrix} x \\ y \\ z \end{bmatrix}^i &= \begin{bmatrix} {}^r x^{i-1} \\ {}^r y^{i-1} \\ {}^r z^{i-1} \end{bmatrix}^{i-1} + \begin{bmatrix} \lambda_e \cos \theta \\ -\lambda_e \sin \theta \\ 0 \end{bmatrix}
\end{aligned} \tag{3.31a, b}$$

e para comando das patas internas, 0 e 1 :

$$\begin{aligned}
{}^r \begin{bmatrix} x \\ y \\ z \end{bmatrix}^i &= \begin{bmatrix} {}^r x^{i-1} \\ {}^r y^{i-1} \\ {}^r z^{i-1} \end{bmatrix}^{i-1} + \begin{bmatrix} \lambda_i \cos \theta \\ \lambda_i \sin \theta \\ 0 \end{bmatrix} \\
{}^r \begin{bmatrix} x \\ y \\ z \end{bmatrix}^i &= \begin{bmatrix} {}^r x^{i-1} \\ {}^r y^{i-1} \\ {}^r z^{i-1} \end{bmatrix}^{i-1} + \begin{bmatrix} \lambda_i \cos \theta \\ -\lambda_i \sin \theta \\ 0 \end{bmatrix}
\end{aligned} \tag{3.32 a, b}$$

onde ${}^r [x^i \ y^i \ z^i]^t$ e ${}^r [x^{i-1} \ y^{i-1} \ z^{i-1}]^t$ são as coordenadas dos pontos de fixação das pernas à plataforma entre dois acionamentos $i-1$ e i .

3.8 Discussão

A cinemática do robô permite o conhecimento das relações e das restrições que existem entre as variáveis, as quais são impostas para preservar a integridade do sistema cinemático aberto definido pelo robô. O cálculo das variáveis com base na análise vetorial e na geometria analítica, apesar de aparentemente limitado, ajuda a simplificar a cinemática por permitir a divisão do problema em uma estrutura mais simples.

Baseados em um sistema de referência principal de bordo, incorpora-se ao modelo as dimensões físicas do robô na forma dos vetores-posição dos referenciais das pernas. Estas recebem os comprimentos dos *links*, e seus sistemas de referências de juntas é definido de acordo com a convenção de Denavit-Hatemberg.

Os sistemas de referência então ficam posicionados e é possível montar as matrizes de relacionamento entre eles, permitindo a formulação de elementos do modelo em qualquer um deles de acordo com a necessidade.

Utilizando-se as configurações de apoio define-se um grupo de vetores posição linearmente independentes a partir dos quais é possível obter, em função das coordenadas de apoio, a matriz de rotação do plano de apoio em relação à plataforma no momento considerado e obter a matriz da rotação acumulada desde a partida, pós-multiplicando-a pela matriz de rotação a cada *setpoint*. Diversas outras informações são disponíveis a partir dos vetores posição tais como: plano de apoio das patas; posição da projeção do CG no plano de apoio; e distância da projeção do CG aos lados do polígono de apoio.

A cada *setpoint* também são montadas as matrizes da cinemática direta do robô, Anexo A4, que são utilizadas para obter as coordenadas de posição no espaço operacional das pernas, as quais são utilizadas pelo simulador gráfico, e pelo sistema de controle durante o acionamento direto da pata em transposição de obstáculos.

A andadura em curva, concebida com dirigibilidade nas quatro patas, apesar de consistir em um problema de rotação de vetores-posição, foi modelada geometricamente de forma bem simples, considerando a particularidade de que, apesar de haverem quatro referenciais de pernas eles estão conectados rigidamente entre si pela plataforma e seus movimentos são restritos fisicamente. Isto permite modelar geometricamente o movimento de guinada aliviando considerável esforço computacional para atualização e rotação dos vetores posição. A tarefa fica ainda mais leve do ponto de vista do processamento, porque os incrementos de ângulo para a realização da curva são convertidos em incrementos de reta e adicionados à andadura no espaço operacional da perna, antes do mapeamento para as coordenadas de juntas.

Capítulo 4

Controle de andadura e trajetória do robô

4.1 Introdução

Neste capítulo são apresentados os esquemas de controle geral e do controle de trajetória utilizados pelo robô.

Além de caminhar em linha *reta* e *curva* em um piso plano e firme, o robô é capaz de superar os obstáculos *Degrau ascendente*, *Degrau descendente*, *Degrau estreito*, *Canal* e *Rampa*, ou se *Desviar* deles.

4.2 Arquitetura de controle

A arquitetura de controle implementada para o Guará utiliza um *Nível Supervisor*, que roda em um PC fora do robô ou em um *notebook* embarcado no robô, um *Nível Coordenador* e um *Nível Atuador*, que rodam em uma rede de microcontroladores a bordo do robô.

O *Nível Coordenador* possui quatro microcontroladores INTEL 80C196KB, cada um deles responsável pelo acionamento, aquisição de dados e reações de uma perna, e é conectado ao *Nível Supervisor* por um barramento CAN de 1 Mb/s, conforme mostrado na Figura 4.2-1.

Através de um algoritmo de controle “Fuzzy”, que roda no *Nível Coordenador*, e um acelerômetro instalado a bordo, não utilizados no presente trabalho, o robô terá capacidade de reação a imprevistos que possam colocar em risco o seu equilíbrio.

O *Nível Atuador* possui dois microcontroladores PIC16F877, sendo um deles responsável pela execução do algoritmo PID das juntas 0 e 1 e o outro das juntas 2 e 3, e também pela aquisição do estado das chaves de contato da pata.

O algoritmo de controle “Fuzzy”; o projeto e a implementação do hardware; a configuração da rede CAN; e os algoritmos de controle PID, de aquisição de dados e de comunicação são objeto de outro projeto de doutorado e estão descritos no projeto de tese de LIMA (2006).

A Figura 4.2-2 mostra o diagrama simplificado de elementos de controle e acionamento e uma foto com um detalhe construtivo de uma pata. Cada junta é acionada por um motorreductor, em cuja saída é acoplado um potenciômetro para adquirir a posição angular do eixo. A saída do motorreductor aciona uma engrenagem acoplada à junta e, a redução dessa conexão, mostrada no Anexo A1, permite amplificar mecanicamente o sinal de saída do potenciômetro em relação ao ângulo descrito pela junta, de um valor igual ao inverso da relação de transmissão, melhorando assim a resolução da posição angular da junta.

Ainda na Figura 4.2-2 é mostrado o detalhe construtivo de uma pata com destaque para as chaves de contato, as quais estão conectadas às portas digitais do controlador das juntas 2 e 3. As chaves *Dedão* e *Calcanhar* estão posicionadas para detectar colisão com objetos verticais na frente e atrás da pata, e as duas chaves *Sola à frente* e as duas *Sola atrás* são ligadas em paralelo e são posicionadas para detectar colisão com o piso. As chaves *Sola à frente* e *Sola atrás* sinalizam também a decolagem e o pouso da pata para novo posicionamento.

Ocorrendo uma mudança de estado em qualquer das chaves, é gerada uma interrupção no respectivo controlador do *Nível Atuador* da perna e o *Nível Coordenador* coloca os quatro bits no registrador de colisão da perna na CAN, os quais ficam disponíveis para leitura pelo *Nível Supervisor*.

O programa desenvolvido para o *Nível Supervisor* inicia na tela de informações do tipo e dos parâmetros da andadura a ser utilizada, mostrada pela aba *Definição da Andadura*, Figura 2.4-1. Conforme descrito no item 2.3, esta interface permite carregar uma andadura existente e iniciar sua execução, ou modificá-la acessando a tela *Nova Andadura*, mostrada na Figura 2.5-2, através do botão *Andadura*.

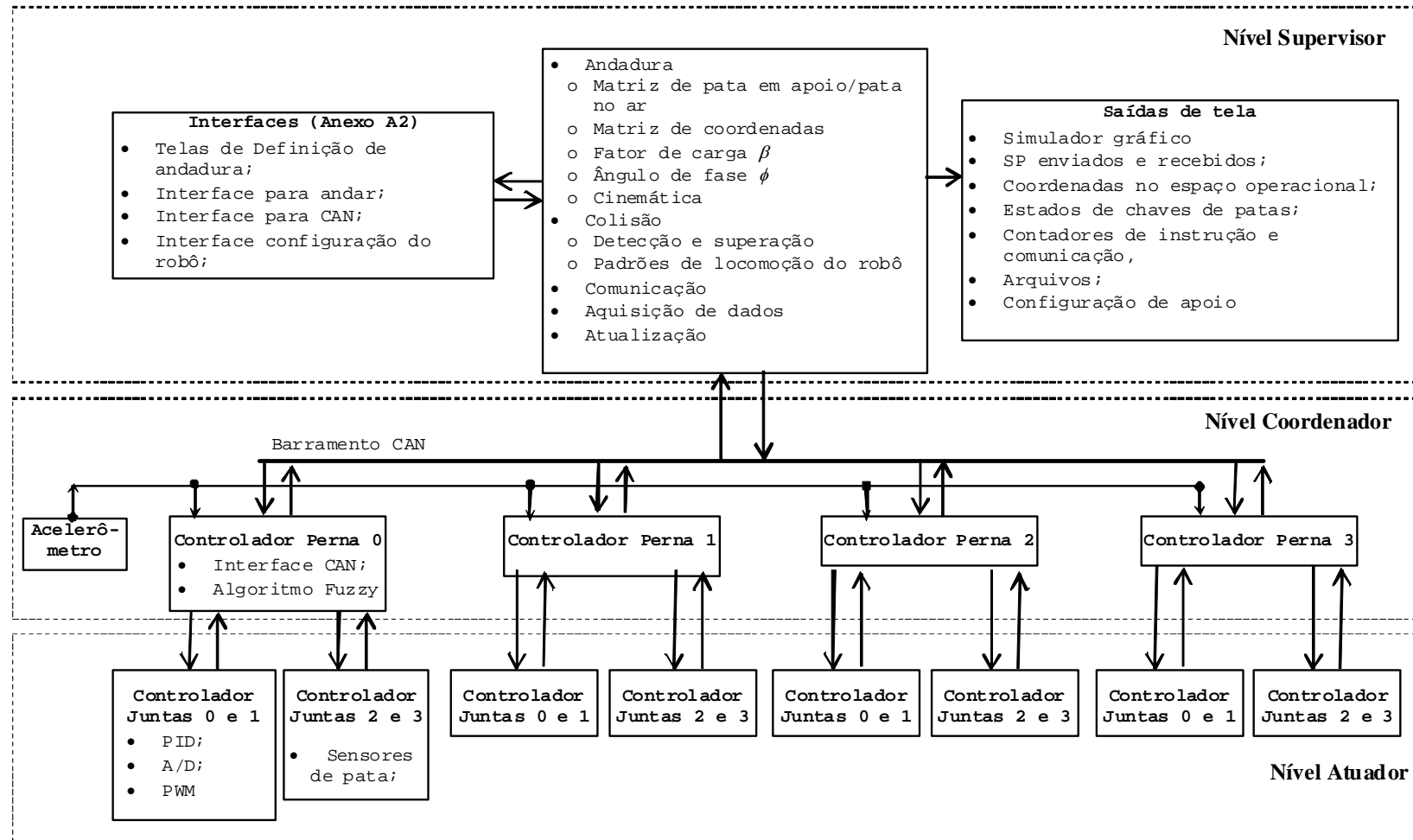


Figura 4.2-1: Arquitetura de controle do Guará.

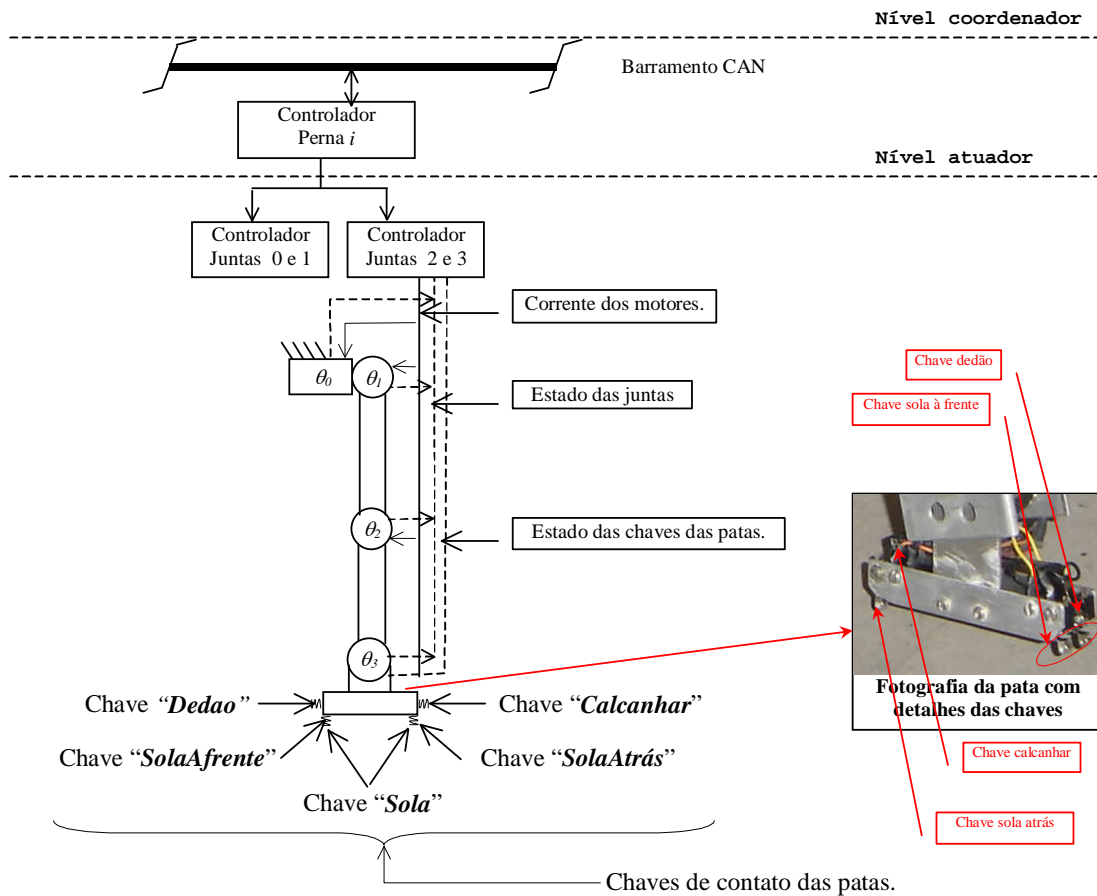


Figura 4.2-2: Diagrama simplificado de elementos de controle e acionamento de uma perna.

As informações para iniciar a execução da andadura são a altura do robô, que determina o empuxo disponível, a altura do passo, que é a altura máxima da trajetória em vôo da pata, o deslocamento lateral máximo e o empuxo máximo da andadura.

A transmissão dos parâmetros para o controlador do *Nível Coordenador* é feita através da tela mostrada na aba *Interface para andar*, Figura 2.6-1.

Ao iniciar esta tela, o programa envia comandos ao *Nível Coordenador* e mostra na tela o valor lido dos potenciômetros de cada junta, as coordenadas atuais das patas em relação ao referencial da perna e o estado atual das chaves das patas do robô, mostradas na Figura 4.2-2.

Acionando-se o botão *Conectar Robô*, é aberta a interface padrão para leitura do arquivo de ganhos e, após carregado, inicia-se a andadura do robô.

Nos itens a seguir serão descritos os diagramas de estratégia que efetuam o controle, a aquisição de dados, controlam a sequência dos processos de andadura e de colisão e fazem a atualização das interfaces gráficas do robô.

4.3 Diagrama de estratégia do ciclo principal de controle

A Figura 4.3-1 mostra o diagrama de estratégia do ciclo principal de controle do Guará e a Tabela 4.3-1 apresenta as variáveis que são ativadas pelo sistema para direcionar o processamento.

O ciclo de controle foi implementado de maneira que todos os processos são executados em uma sequência padrão no nível do ciclo principal de processamento e numa sequência variável, nos níveis secundários.

O diagrama de estratégia do ciclo principal de controle do Guará é o núcleo do processamento do robô e está estruturado em uma sequência de processos que, com base em eventos de chaves, modelos e limites, comutam as variáveis que executam outros processos e grupos de movimento para realizar uma tarefa em curso.

O processo *Iniciar andadura* é iniciado pressionando o botão *Conectar robô* na tela *Interface Para Andar*, Figura 2.6-1, que ativa a variável de processamento ***ExecutandoPasso***.

A *Barra de controle de velocidade* permite variar a velocidade da andadura com o robô em movimento e os *Controles de direção* são um botão direcional e uma barra de rolagem. Estes controles podem ser acionados pelo teclado ou pelo *mouse* e permitem variar o raio de curva e a velocidade com o robô em movimento.

O diagrama de estratégia do ciclo principal de controle do Guará é iniciado no processo *Lê os registros de chaves das patas*, o qual faz a leitura, na CAN, dos *bits* de novo dado recebido dos registros de mensagens de colisão nas patas.

Caso algum registro de mensagem de colisão da CAN estiver com o *bit* de novo dado recebido ligado, o processamento é desviado para *Processa colisão*, o qual pode direcionar para *Degrau ascendente*, *Degrau descendente*, *Canal*, *Rampa*, ou *Desviar*, de acordo com o padrão de chaves recebido ou com a variável de processamento ativa, a qual determina o processo em curso.

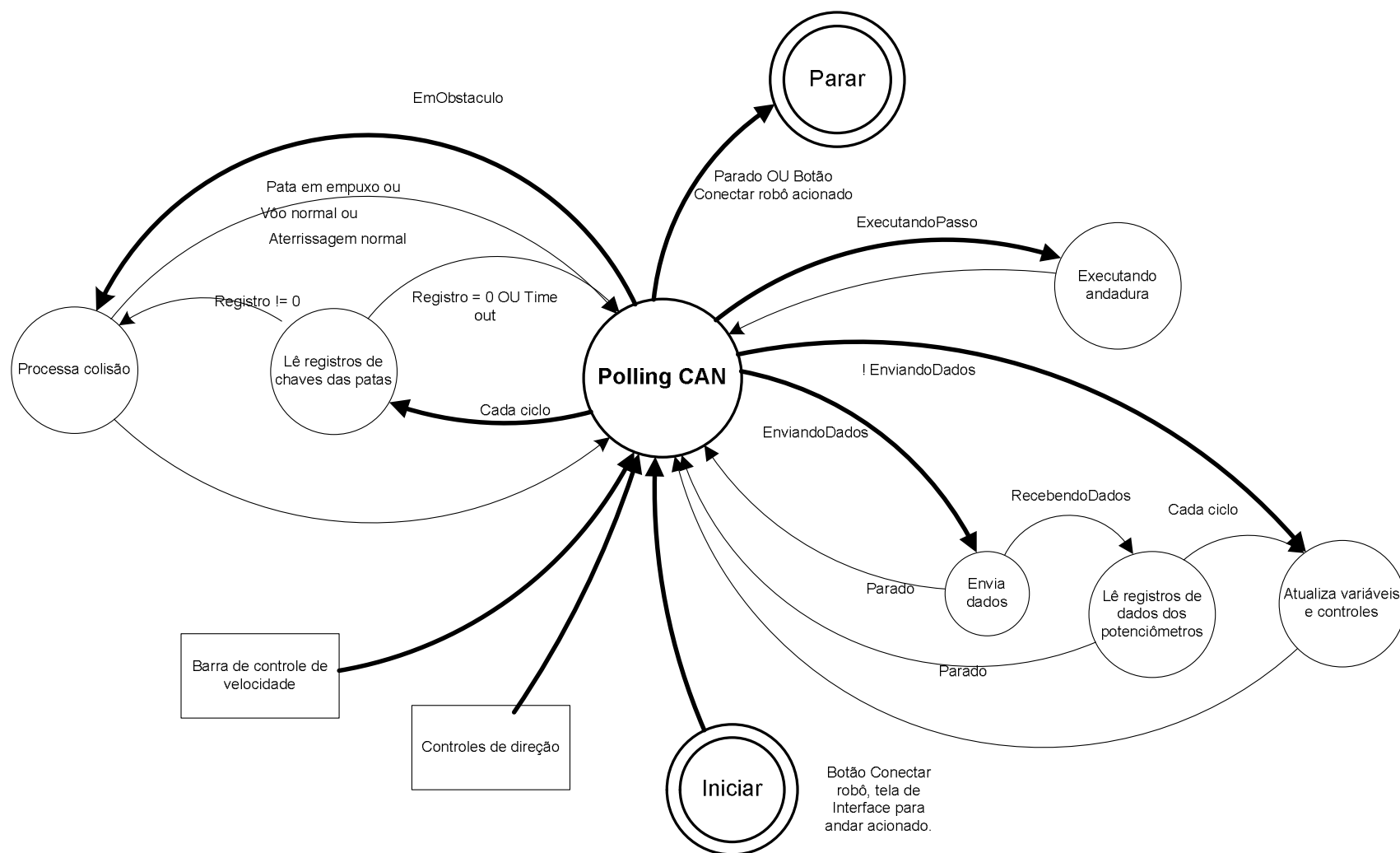


Figura 4.3-1: Diagrama de estratégia do ciclo principal de controle do Guará.

Tabela 4.3-1: Variáveis para direcionar o processamento do ciclo operacional do robô.

Nome	Descrição
<i>Variáveis de processamento do ciclo principal</i>	
ExecutandoPasso	Ativado pelo botão conectar robô sinaliza execução da andadura
EmObstaculo	Sinaliza para processar colisão com obstáculo
EnviandoDados	Sinaliza para enviar dados através dos registros de comunicação
RecebendoDados	Sinaliza para ler registros de posição dos potenciômetros
Parado	Comutado pelo botão conectar robô ou por <i>time out</i> de comunicação com a CAN; sinaliza parada do processamento
<i>Variáveis de processamento de Colisão</i>	
DegrauAscendente	Sinaliza colisão com degrau ascendente
DegrauDescendente	Sinaliza colisão com degrau descendente
DegrauEstreito	Sinaliza colisão com degrau estreito
TranspondoCanal	Sinaliza transposição de canal estreito
Rampa	Sinaliza subida ou descida de rampa
Desvio	Sinaliza desvio de obstáculo

O processo seguinte é *Executando andadura*, o qual direciona o processamento de acordo com a variável funcional ativa, dentro do processamento em curso.

Em seguida, são executados os processos *Envia dados*, se a variável de processamento ***EnviandoDados*** foi ativada, *Lê registros de dados dos potenciômetros* e *Atualiza variáveis e controles*.

Os processos *Envia dados* e *Lê registros de dados dos potenciômetros* podem terminar normalmente ou por um evento de *Time out*.

Se o evento de *Time out* ocorrer em uma etapa não crítica isto será apenas sinalizado na interface, sendo que o processo continuará normalmente. Caso contrário é ativada a variável de processamento **Parado** que pára o robô.

O processo *Atualiza variáveis e controles* é executado mesmo que a variável que ativa o processo *Envia dados*, não tenha sido ativada por um dos processos anteriores.

O ciclo principal de controle do Guará é parado com um acionamento do botão *Conectar robô* na tela *Interface Para Andar*, Figura 2.6-1, a qual ativa a variável de processamento **Parado** ou quando esta variável tiver sido ativada por um evento de *Time out* no processo *Envia dados* ou no processo *Lê registros de dados dos potenciômetros*.

4.3.1 Lê registros de chaves das patas

A Figura 4.3-2 mostra o diagrama de estratégia de leitura de registros de chaves das patas, e as combinações possíveis para as quatro chaves das patas são mostradas na Tabela 4.3-3.

Neste trabalho utilizam-se apenas chaves de contato instaladas nas patas para a interação do robô com o ambiente.

Durante o andar normal e na transposição de obstáculos, as mudanças de estado dessas chaves podem ativar, conforme mostrado na Tabela 4.3-4, variáveis do processamento do ciclo *Processa Colisão*, que direcionam o ciclo entre os processos que encapsulam os movimentos mais complexos, os quais gerenciam os grupos de movimentos mais simples que os compõem.

Quando ocorre uma mudança de estado em uma chave qualquer, o *Nível Coordenador* pára o robô, liga o bit de novo dado recebido do respectivo registro de comunicação da CAN. Por outro lado, o *Nível Supervisor* desvia o processamento para leitura do registro de dados correspondente.

A seqüência de leitura é da pata 0 para a 3 e caso todos os registros de dados estejam no estado 0, o processo encerra e volta ao processamento principal sem alterar nenhuma variável do processamento.

O registro de dados cujo *bit* de novo dado recebido estiver em 1 é lido e desligado para 0 em *loop*, até que permaneça nulo, para garantir que a leitura foi feita após completada a transmissão do dado pela CAN.

O processo é então encerrado e o controle do processamento passa para *Processa colisão*, caso tenha ocorrido uma das combinações mostradas na Tabela 4.3-4.

Caso mais de um *bit* de novo dado é ligado, ele será tratado no próximo ciclo de processamento do ciclo principal de controle do Guará.

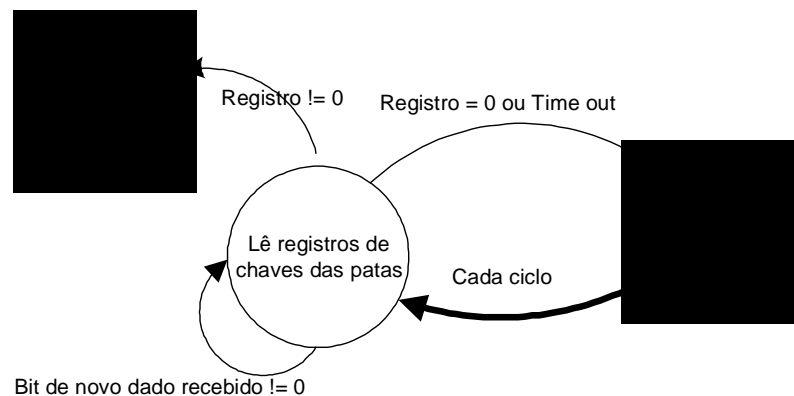


Figura 4.3-2: Diagrama de estratégia de leitura de registros de chaves das patas.

As chaves de pata estão localizadas conforme mostrado na Figura 4.2-2 sendo que o bit 3 corresponde à chave *Calcanhar*, o bit 2 corresponde à chave *SolaAtras*, o bit 1 corresponde à chave *SolaAfrente*, e o bit 0 corresponde à chave *Dedão*.

Na Tabela 4.3-3 a chave foi acionada quando o respectivo bit está em 0. Algumas combinações não são tratadas porque resultariam em colisões com obstáculos que não estão presentes no ambiente considerado.

4.3.2 Processa colisão

A Figura 4.3-3 mostra o diagrama de estratégia de processamento de colisão. As variáveis funcionais que são ativadas pelo sistema para direcionar o processamento são mostradas na Tabela 4.3-2, os estados das chaves associadas aos processos são mostradas na Tabela 4.3-3 e as combinações que ativam as variáveis funcionais dentro do processo são mostradas na Tabela 4.3-4.

Este processo gera a sequência de movimentos necessários para a transposição ou desvio de obstáculos. Ele identifica o obstáculo e ativa variáveis e processos com o objetivo de superá-lo ou se desviar do mesmo.

Quando ocorre uma colisão da pata com um obstáculo, configurando um evento não previsto na andadura, a variável associada ao processamento ***EmObstaculo*** é ativada, direcionando o processamento do ciclo principal para este processo.

As variáveis ***EmEmpuxo***, ***EmVoo*** e ***MomentoCrítico*** são ativadas pelo programa de controle da andadura do *Módulo Supervisor*, quando a pata correspondente estiver em apoio, em vôo ou no momento de aterrissar após o vôo para novo posicionamento, respectivamente.

Em *Processa colisão*, se o evento de chave ocorreu em uma pata ***EmEmpuxo***, ***Vôo normal*** ou ***Aterrissagem normal***, conforme tabela do Anexo A3, a variável de processamento ***ExecutandoPasso*** permanece ativa e segue para executar a andadura no processo *Executando andadura*.

Um evento de chave qualquer em uma pata ***EmEmpuxo*** não é considerado no modelo proposto porque a pata não poderia colidir com algum objeto parado estando apoiada.

Um evento de chave ***NenhumaChave*** em uma pata em empuxo, no momento previsto para decolagem, ou após um tempo predefinido, significa que a pata iniciou o ***Vôo normal*** para novo posicionamento.

Um evento de chave tipo ***Sola***, no momento em que a variável ***MomentoCrítico*** está ativa, ou um tempo predefinido antes, significa uma ***Aterrissagem normal*** da pata.

Um evento de chave ***SolaAfrente*** ou ***SolaAtras*** também é considerado uma ***Aterrissagem normal*** por mais um ciclo de processamento, para compensar os atrasos na mudança de estado das chaves que ocorrem na decolagem e no pouso da pata devido ao balanço em movimento e às deformações elásticas que ocorrem na estrutura e nas pernas do robô.

Tabela 4.3-2: Variáveis de processamento de Colisão

DegrauAscendente	Sinaliza colisão com degrau ascendente
DegrauDescendente	Sinaliza colisão com degrau descendente
DegrauEstreito	Sinaliza colisão com degrau estreito
TranspondoCanal	Sinaliza transposição de canal estreito
Rampa	Sinaliza subida ou descida de rampa
Desvio	Sinaliza desvio de obstáculo

Tabela 4.3-3: Estados das chaves das patas.

Nome da combinação	Bit				Valor (Hex)
	3 <i>Calcanhar</i>	2 <i>SolaAtras</i>	1 <i>SolaAfrente</i>	0 <i>Dedão</i>	
01	0	0	0	1	01
Não tratada	0	0	1	0	02
03	0	0	1	1	03
Não tratada	0	1	0	0	04
05	0	1	0	1	05
Não tratada	0	1	1	0	06
Calcanhar	0	1	1	1	07
SolaEDedao	1	0	0	0	08
Sola	1	0	0	1	09
SolaAtrasEDedao	1	0	1	0	0A
SolaAfrente	1	1	0	1	0D
SolaAfrenteEDedao	1	1	0	0	0C
SolaAtras	1	0	1	1	0B
Dedão	1	1	1	0	0E
NenhumaChave	1	1	1	1	0F

Tabela 4.3-4: Combinações de chaves que ativam as variáveis funcionais de processamento de colisão.

<i>Processo</i>	<i>Combinação</i>	<i>Variável ativada</i>
DegrauAscendente	(EmVoo && ChaveDedao) && !(TranspondoCanal)	SubindoDegrau
DegrauDescendente	(MomentoCritico && NenhumaChave) && !(TranspondoCanal)	DescendoDegrau
DegrauEstreito	(NoPisoEmCima && NenhumaChave)	DegrauEstreito
Canal	(MomentoCritico && NenhumaChave)	CruzandoCanal
Rampa	(MomentoCritico && NenhumaChave) (MomentoCritico && Dedao)	Rampa
Desviar	É ativado em caso de fracasso em superar obstáculo	Desvio

Em *Processa colisão*, se a variável de processamento ativa é ***ExecutandoPasso*** e o evento de chave é identificado como um dos padrões de colisão previstos, a variável de processamento ***EmObstaculo*** é ativada e o variável de operação da pata é comutada para ***SubindoDegrau***, ***DescendoDegrau***, ***CruzandoCanal***, ***DescendoRampa***, ***SubindoRampa***, ***DegrauEstreito*** ou ***Desviando***, conforme combinação de chaves e variáveis mostrada na Tabela 4.3-4.

As variáveis de operação são ativadas conforme a seguir:

- ***SubindoDegrau*** é ativada quando uma pata em vôo colide com um obstáculo vertical e ocorre uma mudança de estado na chave ***Dedão***;
- ***CruzandoCanal*** é ativada quando não ocorre uma mudança de estado na chave ***SolaDoPe***, quando a variável ***MomentoCritico*** está ativada, indicando que é o momento de apoiar a pata;
- ***DescendoDegrau*** é ativada quando a pata não apóia no momento esperado e o controlador já confirmou não de tratar de um obstáculo tipo ***Canal***;

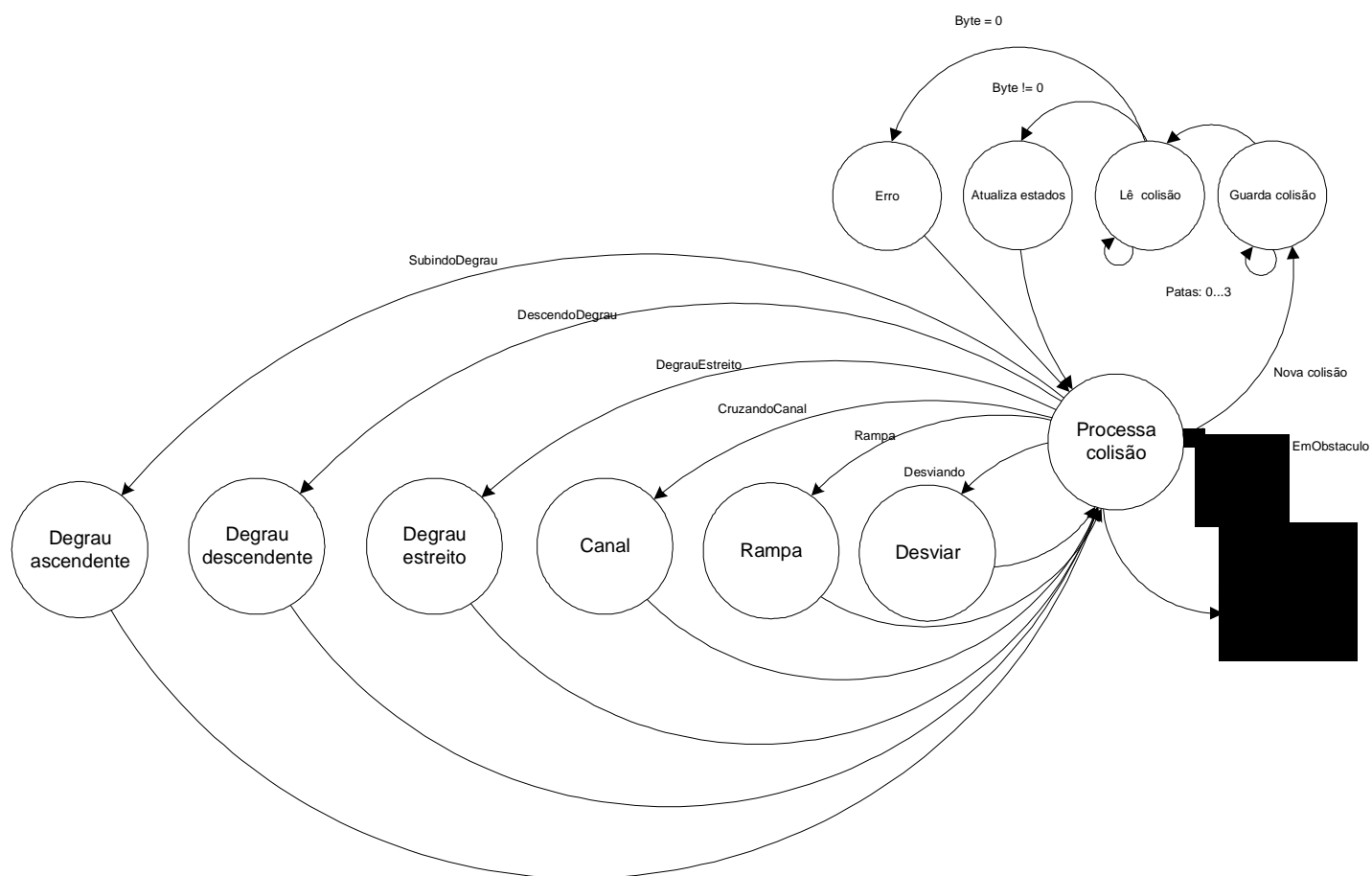


Figura 4.3-3: Diagrama de estratégia para processamento de colisão.

- **Rampa** é ativada quando a pata não apoiou no momento esperado e ocorreu uma mudança de estado na chave **SolaAtras**, que ativa a variável **SubindoRampa**, ou quando ocorre uma mudança de estado na chave **SolaAfrente**, com uma pata em vôo, antes do momento previsto para apoiar, que ativa a variável **DescendoRampa**;
- **DegrauEstreito** é ativada quando, após colidir com um degrau, apoiar as duas patas dianteiras no degrau em cima e começar a andar, o modelo sinalizar o momento de apoiar e não ocorrer uma mudança de estado na chave **Sola**;
- **Desvio** é ativada em caso de fracasso na transposição de quaisquer dos obstáculos definidos.

Se a variável de processamento ativa for **EmObstaculo**, mesmo que nenhuma das chaves das patas mude de estado, o processamento também será desviado para *Processa colisão*, continuando a processar uma colisão já ocorrida.

A partir do momento em que a variável de processamento **EmObstaculo** for ativada, ela permanecerá ativa até que o processamento de colisão com obstáculo seja finalizado com a transposição do obstáculo ou com o desvio dele.

Os processos de colisão *Degrau ascendente*, *Degrau descendente*, *Canal*, *Rampa*, *Degrau estreito* e *Desviar* são descritos no Capítulo 5.

4.3.3 Executando andadura

A Figura 4.3-4 mostra o diagrama de estratégia de processamento de andadura. As variáveis funcionais que são ativadas pelo sistema para direcionar o processamento são mostradas na Tabela 4.3-5.

De acordo com as variáveis ativas em cada fase, este processo é executado enquanto estiver ativa a variável de processamento do ciclo principal de controle **ExecutandoPasso**.

As variáveis funcionais de processamento da andadura são ativadas em uma seqüência predefinida para a preparação do robô para iniciar a andadura, para a retomada da andadura após um processo de colisão e após partida em uma posição qualquer. A seqüência de processamento é escolhida em função das variáveis mostradas na Tabela 4.3-5 as quais vão sendo ativadas à medida que as tarefas nos respectivos processos vão sendo concluídas.

Tabela 4.3-5: Variáveis funcionais do processo Executando Andadura

Andando	Sinaliza para processar modelo para execução da andadura
PreparandoParaAndar	Sinaliza para verificar e corrigir o posicionamento para começar a andar
Equalizando	Sinaliza para ajuste das patas em relação a uma posição longitudinal
Postura	Sinaliza para verificar e restaurar a postura do robô

4.3.3.1 Preparando o robô

A variável funcional de processamento *PreparandoParaAndar* é acionada no início da andadura ou durante a transposição de obstáculo, após as pernas dianteiras já terem sido processadas.

A Figura 4.3-5 mostra o diagrama de estratégia do processo *Preparando o robô* e a Tabela 4.3-6 mostra as variáveis funcionais associadas aos processos.

As variáveis de operação são ativadas conforme a seguir:

- **BaixandoParaAndar:** é ativada no início da andadura se as pernas estão esticadas e as patas estão na origem, a plataforma do robô é baixada e posicionada à esquerda na máxima excursão lateral para o vôo das patas 2 e 3. Os eventos limitadores desses movimentos são a coordenada vertical e a máxima excursão lateral da plataforma especificadas na definição da andadura.
- **Corrigindo postura:** é ativada no início da andadura se as pernas estão em uma posição qualquer desconhecida;
- **Andando:** é ativada ao final do processo *Preparando robô*.

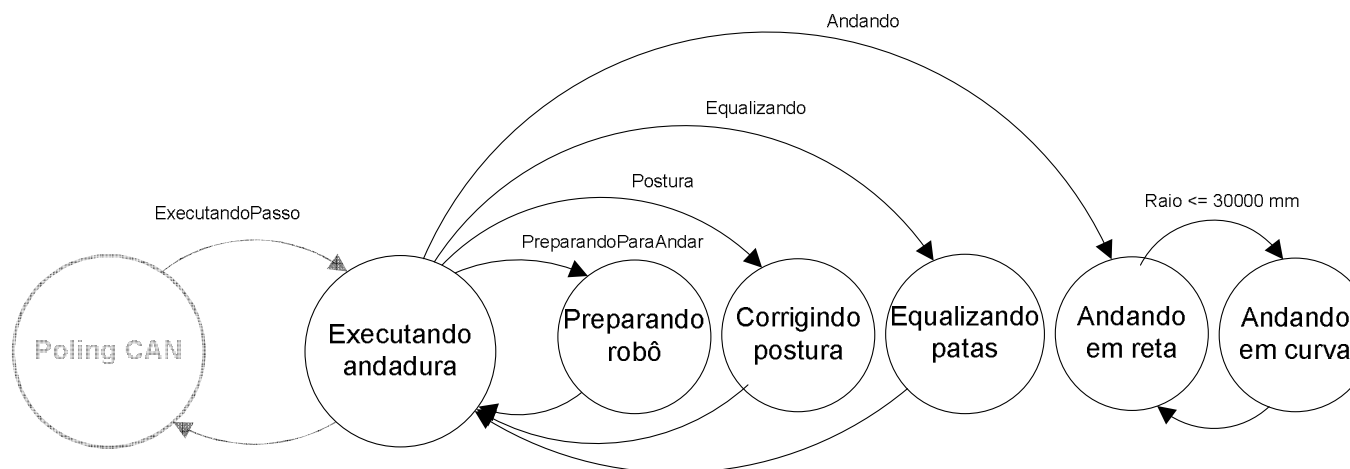


Figura 4.3-4: Diagrama de estratégia para processamento de andadura.

As variáveis de operação ativam movimentos complexos que são compostos de movimentos mais simples, ativados por variáveis de movimento, conforme a seguir:

- **Baixa plataforma:** move a plataforma até o limite de altura especificado na definição da andadura;
- **Move a plataforma à esquerda:** move a plataforma até o limite e de deslocamento lateral especificado na definição da andadura;

O processamento dos movimentos é simultâneo e, quando o limite do primeiro grupo de movimentos é atingido, o respectivo incremento de movimento é anulado e o processamento continua no outro grupo de movimentos até que o seu limite seja atingido.

Tabela 4.3-6: Variáveis funcionais do processo preparando robô.

<i>Nome</i>	<i>Descrição da ação</i>
Andando	Sinaliza processamento de andadura
PreparandoParaAndar	Sinaliza preparação do robô para andar
BaixandoParaAndar	Sinaliza baixar plataforma na preparação do robô para andar
Postura	Sinaliza para verificar e restaurar a postura do robô

O processo *Preparando robô* é acionado também durante a transposição de obstáculos, após as pernas dianteiras já terem sido processadas.

No caso de processamento de colisão com *Degrau ascendente*, após equalizar as patas dianteiras em relação à posição da pata 2, a plataforma é posicionada na posição extrema à esquerda para iniciar a andadura que será executada até a colisão de uma das patas traseiras com o obstáculo. Neste ponto, o processo equaliza as patas dianteiras e coloca a plataforma na posição extrema à esquerda para apoiar a pata 3 em cima do degrau.

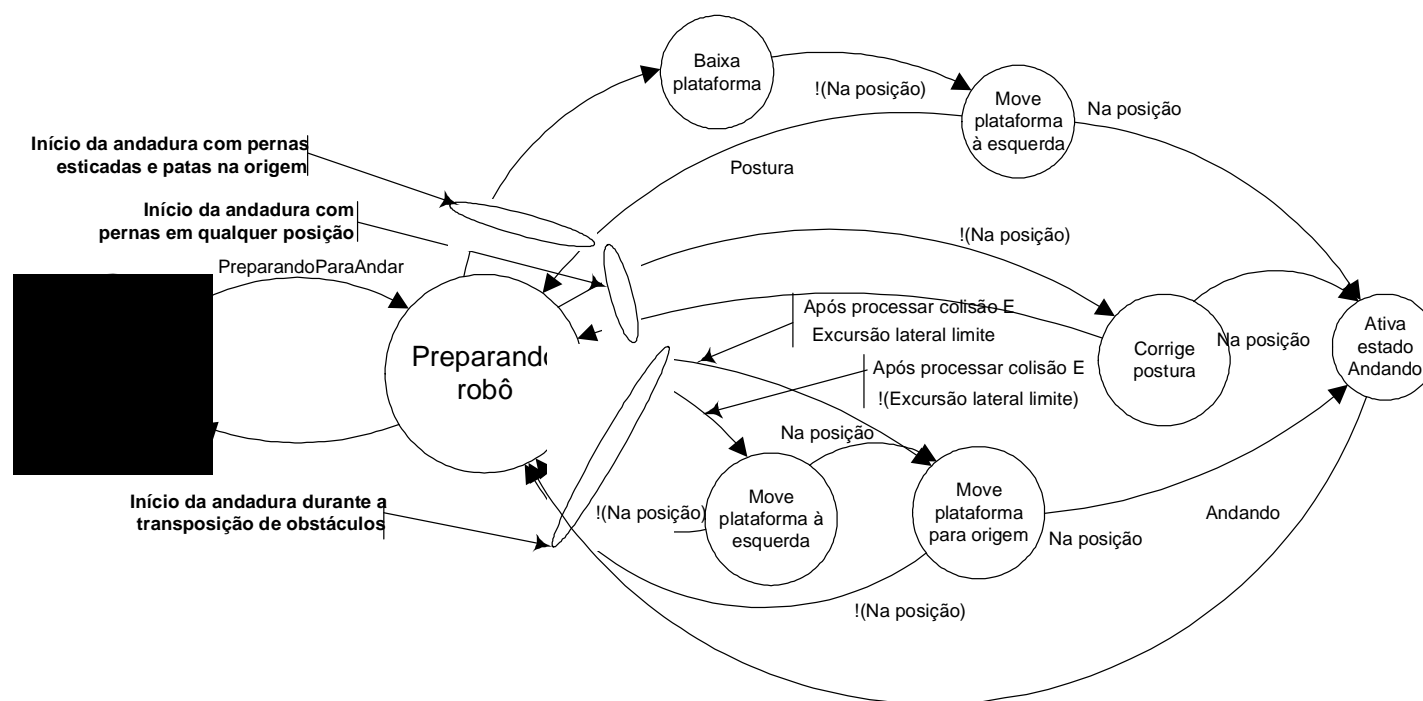


Figura 4.3-5: Diagrama de estratégia para preparar o robô para andar.

4.3.3.2 Corrigindo postura

A variável funcional de processamento **Postura** é acionada se no início da andadura as patas do robô estão em uma posição desconhecida. A Figura 4.3-6 mostra o diagrama de estratégia do processo *Corrigindo postura*.

As variáveis de operação são ativadas conforme a seguir:

- **Equalizando:** é ativada no início da andadura com as quatro patas apoiadas, em uma posição qualquer e desconhecida;
- **Movendo:** é ativada no início da andadura com as quatro patas apoiadas, em uma posição conhecida, e posiciona a plataforma à esquerda na máxima excursão;
- **Baixando:** é ativada também no início da andadura com as quatro patas apoiadas, em uma posição conhecida, e posiciona a plataforma na altura limite para a andadura.

Os eventos limitadores desses movimentos são a coordenada vertical e a máxima excursão lateral da plataforma especificadas na definição da andadura

As variáveis de operação ativam as variáveis de movimento das patas e da plataforma, conforme a seguir:

- **Baixa:** move a pata verticalmente até o contato com o piso quando o estado das chaves da pata muda para **Sola** e é ativada a variável no **NoPiso**. Concluído esse movimento, as quatro patas estarão apoiadas
- **Move plataforma:** move-a até o limite de deslocamento lateral especificado na definição da andadura. O processo consiste em deslocar a plataforma do robô para a posição extrema oposta lateral e longitudinalmente em relação à pata que será movimentada durante o processo de equalização.

O processamento dos movimentos laterais e longitudinais da plataforma é simultâneo para as quatro patas. Quando o limite de um dos movimentos é atingido o processamento continua no outro movimento até completar o posicionamento da plataforma.

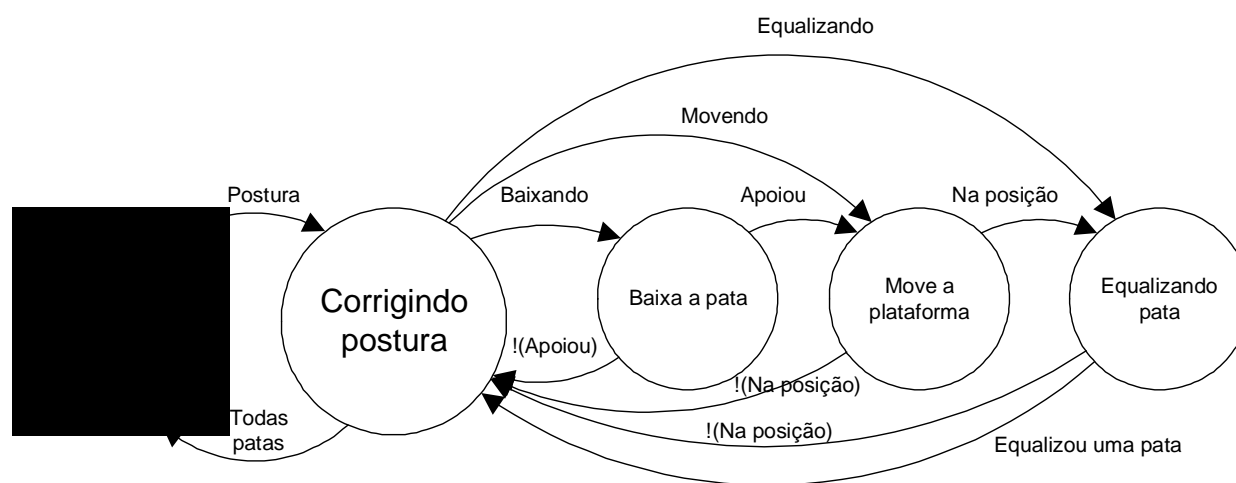


Figura 4.3-6: Diagrama de estratégia de correção da postura do robô.

4.3.3.3 Equalizando patas

A variável funcional ***Equalizando*** é ativada pelo processo *Corrigindo postura* para ativar o processo *Equalizando patas*, para ajuste da coordenada Z3 longitudinal de uma ou de todas as patas.

O processo *Equalizando patas* é executado sempre após a plataforma estar posicionada no extremo oposto lateral e longitudinalmente em relação à pata a ser equalizada. A Figura 4.3-7 mostra o diagrama de estratégia do processo *Equalizando patas*.

Para deslocar uma pata apoiada com o robô parado, são necessários os movimentos de erguer a pata, deslocá-la para frente ou para trás e baixá-la na posição desejada.

As variáveis de movimento das patas são ativadas conforme a seguir:

- ***Erguendo:*** é ativada no início da equalização e o evento limitador é a altura limite cinemático da pata;
- ***Movendo:*** é ativada com a pata na altura limite cinemático e o evento limitador é a excursão longitudinal da pata. Durante este movimento a altura da pata pode variar caso uma ou mais juntas em movimento atinjam o limite cinemático;
- ***Baixando:*** é ativada com a pata na posição limite da excursão longitudinal e o evento limitador é a mudança de estado das chaves da sola da pata.

As variáveis de operação ativam os movimentos *Ergue a pata*, *Move a pata* e *Baixa a pata*, utilizados para mover verticalmente a pata para a altura limite ou para o contato com o piso e horizontalmente para frente ou para trás. Não há movimento lateral das patas durante a equalização.

A seqüência de equalização é sempre na ordem da pata 3 para a 0. A plataforma é movimentada sucessivamente para a posição extrema dianteira esquerda para equalizar a pata 3; para a posição extrema traseira esquerda para equalizar a pata 2; para a posição extrema traseira direita para equalizar a pata 0; e para a posição extrema dianteira direita para equalizar a pata 1.

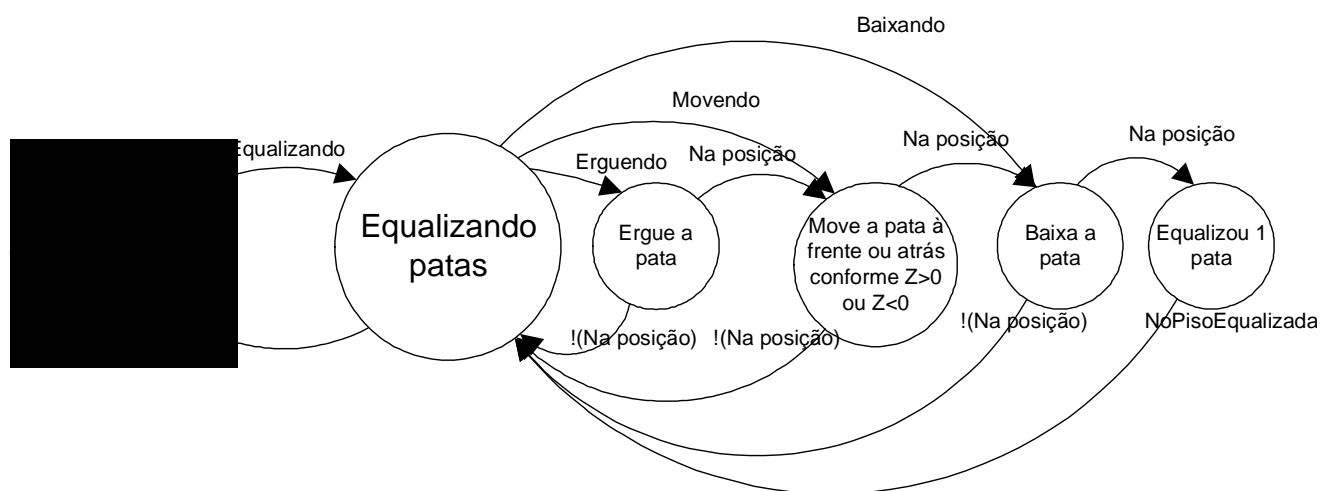


Figura 4.3-7: Diagrama de estratégia para equalização das patas do robô.

No processamento de colisão a coordenada de equalização é a da pata que colidiu e no caso geral pode ser especificada uma coordenada longitudinal qualquer. Em ambos os casos existe uma tolerância definida na tela *Interface para CAN*, mostrada no Anexo A2.

4.3.3.4 Andando em reta

O processo *Andando em reta* é acionado pelo processo *Executando andadura* com a variável *Andando* ativada.

A Figura 4.3-8 mostra o diagrama de estratégia do processo *Andando em reta* e a Tabela 4.3-7 mostra as variáveis funcionais associados aos processos e grupos de movimento.

O modelo matemático da andadura é implementado em uma biblioteca de funções que geram as trajetórias desejadas no espaço operacional, de acordo com a andadura, e uma biblioteca de procedimentos responsável pela seqüência de movimentos das patas. Esta biblioteca de procedimentos é responsável também pela coordenação dos movimentos laterais, efetuados nos períodos de apoio em quatro patas, com os movimentos de empuxo e vôo das patas, com a atualização dos endereços de dados e dos contadores.

As trajetórias das patas em relação à plataforma, são calculadas no espaço operacional, sendo que a pata em empuxo descreve uma trajetória reta e em vôo uma trajetória composta de segmentos de reta formando um polígono em forma de losango.

As variáveis de operação da pata são *EmEmpuxo*, *EmVoo* ou *MomentoCrítico*. As variáveis são ativadas pela matriz do passo e, durante a andadura, confirmadas pelas chaves de pata. Andando, o robô balança e se deforma elasticamente, provocando defasagem entre a ativação das variáveis funcionais determinadas pela matriz do passo, e a mudança de estado na chave *Sola*. Na Figura 4.3-8 a seqüência de processos de andadura *EmEmpuxo* é ativada pelo elemento $m_{ij} = 1$ da matriz do passo e *EmVoo* é ativado pelo elemento $m_{ij} = 0$.

Ao final da trajetória em vôo m_{ij} muda de 0 para 1 e é ativada a variável *MomentoCrítico* a qual sinaliza o momento de pouso da pata quando o *Nível Supervisor* deverá receber um sinal de chave *Sola* no registro de colisão da CAN, referente àquela perna.

Ocorrendo uma mudança de estado da chave da pata em vôo sem que variável *MomentoCrítico* esteja ativa, ocorre o desvio do processamento para *Processa colisão*, o qual verifica se é uma ocorrência crítica de colisão com obstáculo, caso em que o controle passa

para o referido processo, ou se é uma ocorrência que pode ser descartada no ciclo de processamento em curso.

Tabela 4.3-7: Variáveis associadas aos processos da andadura.

<i>Variável</i>	<i>Descrição da ação</i>
<i>Variáveis de processamento</i>	
Andando	Sinaliza processamento de andadura
Reta	Sinaliza cálculo do modelo de trajetória reta
Curva	Sinaliza cálculo do modelo de trajetória curva
<i>Variáveis de operação da pata</i>	
EmEmpuxo	Sinaliza pata apoiada em empuxo
EmVoo	Sinaliza pata em voo para novo posicionamento
MomentoCritico	Sinaliza momento previsto para aterrissagem da pata em voo

Ao iniciar a andadura o primeiro ciclo completo empuxo, voo para novo posicionamento e aterrissagem de cada pata devem ser realizados com base no empuxo disponível na última pata prevista para decolar para nova posição. O empuxo total do robô até que a última pata decole para um novo posicionamento deve ser menor que o empuxo disponível na pata de forma a evitar que o alcance cinemático da perna seja ultrapassado.

4.3.3.4.1 Trajetórias da pata durante os movimentos de empuxo e voo

O robô de quatro patas é uma plataforma com quatro manipuladores a bordo. Os movimentos de cada perna durante o voo ocorrem em um plano perpendicular ao eixo de rotação das juntas 1, 2 e 3. A junta 0 permite o movimento de rolagem desse plano, o que possibilita a movimentação lateral e a implementação de trajetórias curvas no robô. Durante o voo, a pata é o efetuator do manipulador, e durante o empuxo ela é seu o ponto de fixação, sendo o efetuator o ponto de fixação da perna à plataforma.

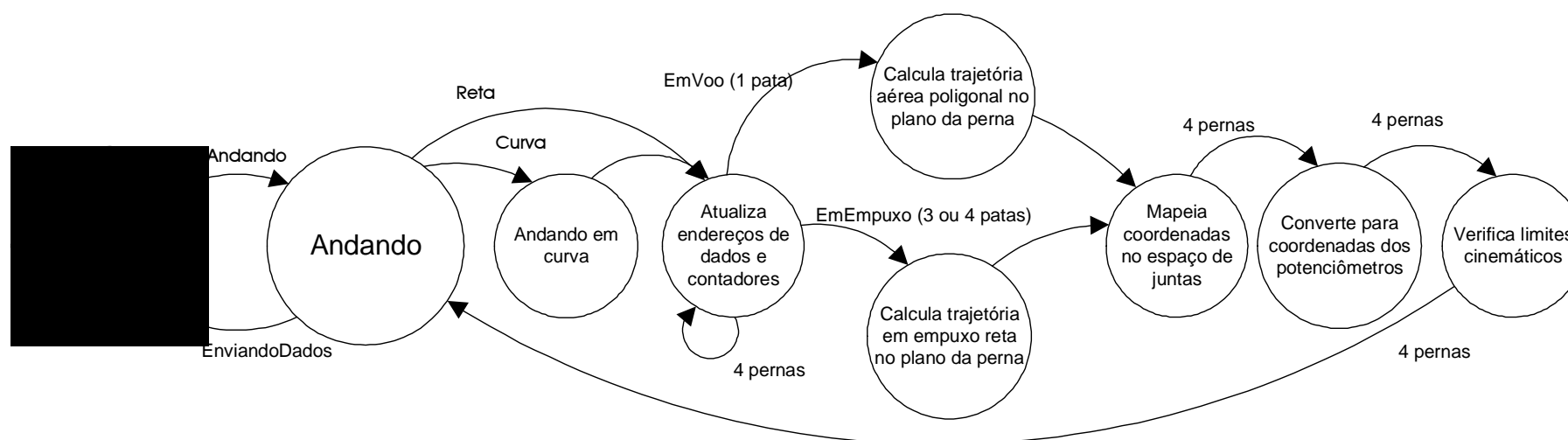


Figura 4.3-8: Diagrama de estratégia de andar em linha reta

Durante o empuxo, a trajetória da pata no espaço operacional, em relação à plataforma, é uma reta definida em relação ao referencial robô, de forma a minimizar os movimentos verticais do robô. Se as retas descritas pelas patas em apoio forem paralelas, ele caminhará sempre em um plano paralelo ao piso, naquela configuração de apoio.

A Figura 4.3-9 mostra a trajetória da pata em um ciclo completo empuxo – vôo com trajetória em ciclóide. O percurso em vôo inicialmente utilizado para a pata foi o de uma ciclóide, devido à sua característica de velocidade nula nos pontos de decolagem e aterrissagem. Entretanto, a ciclóide apresentou como inconveniente um deslocamento muito pequeno no início e no final da trajetória, ocasionando empuxo reverso no robô, já que, devido à flexibilidade de suas hastes, a pata permanece no solo até 6 *setpoints* após ter sido iniciado o vôo e toca no solo sempre 2 a 5 *setpoints* antes do momento previsto pelo modelo. Nesses períodos ela está iniciando ou terminando o período de vôo e está sendo movida em sentido contrário ao empuxo em curso nas outras patas.

A Figura 4.3-10 mostra a trajetória da pata em um ciclo completo empuxo – vôo com trajetória com segmentos de reta formando uma poligonal. A linha reta inferior corresponde ao trecho com pata em empuxo e a poligonal à pata em vôo; a andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo; $Y3$ é a coordenada vertical da junta 3 traçada no plano ZY; e $Z3$ é a coordenada longitudinal da junta 3 traçada no plano ZX. O sistema de referência é o da perna, conforme mostrado na Figura 3.8-3 e, para diminuir o tamanho da figura, o eixo $Z3$ está posicionado no menor valor da escala de $Y3$.

O robô partiu de uma posição de repouso com as pernas esticadas, e o empuxo do robô é obtido baixando-se a plataforma, já que as pernas não variam o comprimento, como no andar humano.

O percurso mostrado na Figura 4.3-10, considerado adequado para o vôo da pata, foi obtido com segmentos de reta formando uma poligonal, onde a pata inicia o vôo ainda em empuxo, na mesma velocidade de deslocamento do robô, e passa a avançar apenas após completamente fora de contato com o piso. Na aterrissagem, o último trecho é efetuado com a pata já em empuxo, o que assegura o contato da pata com o piso na mesma velocidade, e sentido contrário de deslocamento do robô, evitando assim um empuxo reverso e permitindo uma aterrissagem bem mais suave.

Conforme pode-se observar na Figura 4.3-10 esta trajetória é composta de cinco trechos correspondentes aos cinco estágios de vôo de cada pata na andadura implementada, conforme a matriz da equação (2.3.2), sendo que, em relação à Figura 4.3-10, o robô se desloca da esquerda para a direita.

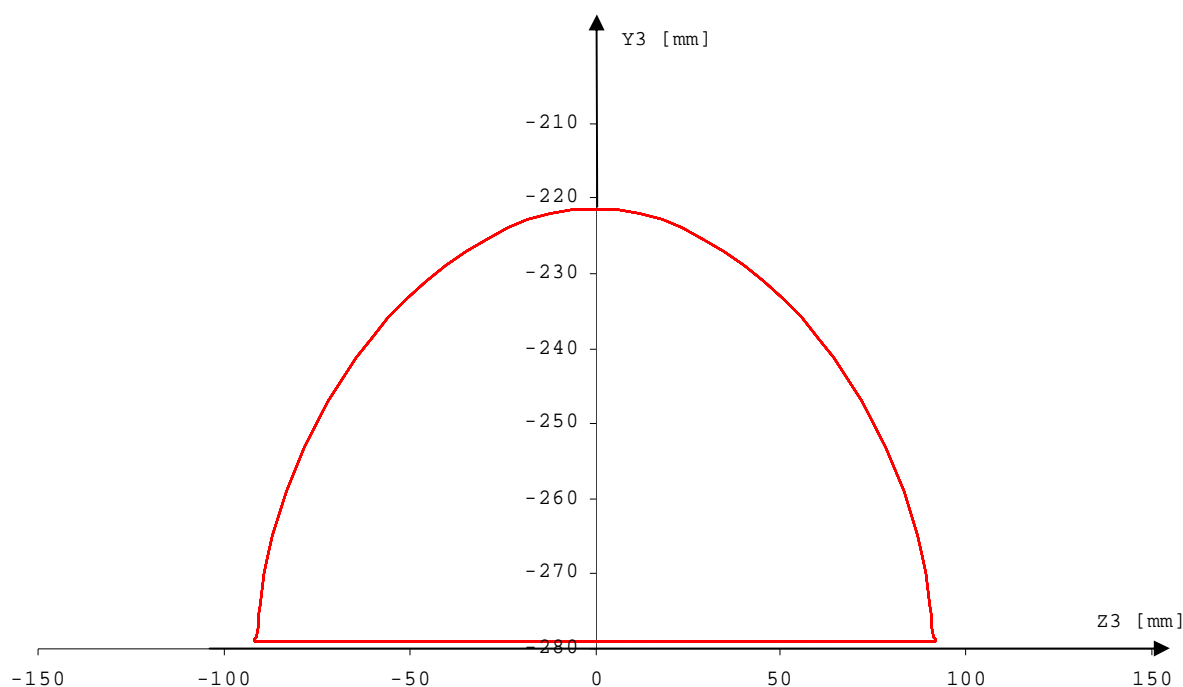


Figura 4.3-9: Trajetória da pata com vôo em ciclóide; empuxo da direita para a esquerda.

No primeiro trecho da esquerda para a direita, a pata inicia o vôo, sendo erguida ainda em empuxo, o que evita o empuxo reverso que ocorre porque a pata ainda não se elevou do piso, devido à deformação elástica das pernas e da estrutura do robô. A componente horizontal de velocidade é igual à velocidade do robô. No segundo trecho, a pata continua sendo erguida, mas a componente horizontal de velocidade de vôo é maior para compensar os trechos em que a pata é movida ainda em tração ao decolar e ao aterrissar; no terceiro trecho há apenas movimento de avanço rápido em vôo; no quarto trecho é iniciado o movimento para pouso, ainda com velocidade de avanço rápido em vôo; e no quinto trecho, o movimento para apoiar a pata é feito baixando-a com a componente horizontal de velocidade igual à velocidade do robô, proporcionando um pouso sem empuxo reverso.

Este esquema mostrou-se bastante útil para amenizar os choques nas decolagens e aterrissagens, já que as deformações das juntas e correias de acionamento resultam em uma baixa precisão de posicionamento, com conseqüentes atrasos na decolagem, e colisões na aproximação da pata com o piso, ao final do vôo para novo posicionamento.

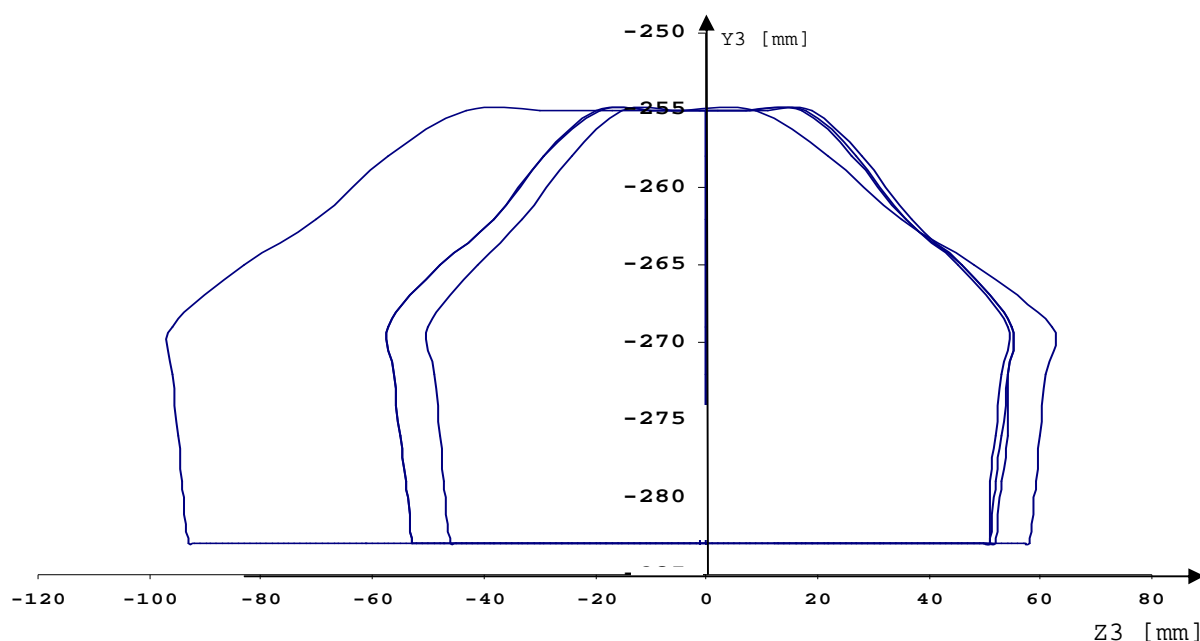


Figura 4.3-10: Trajetória da pata com vôo em poligonal.

A geração das trajetórias das patas é feita no espaço operacional em relação ao referencial da perna, sendo o movimento conjunto das patas sincronizado pela matriz da andadura.

Na Figura 4.3-10 está mostrada a trajetória da pata 0, que é a última a voar. Na primeira passada, a pata realiza um percurso menor em empuxo, mostrado na linha interna, com percurso de empuxo estável a partir do terceiro vôo.

A matriz de andadura sinaliza os períodos de empuxo e vôo, e os momentos de decolagem e aterrissagem da pata. A partir da segunda passada, para assegurar que estes movimentos não excederão o limite cinemático das pernas, o empuxo total é sempre menor que o empuxo disponível para as pernas. Além disso, existe um filtro no modelo que impede que seja acionada uma pata além do limite cinemático das juntas.

4.3.3.4.2 Trajetórias das patas na andadura

A Figura 4.3-11 mostra as projeções da trajetória gerada no *Nível Supervisor* para a pata 0 em relação ao sistema de referência da perna. A abscissa comum são os *setpoints* enviados, e a ordenada está em metros; X_3 é a coordenada lateral da junta 3; Y_3 é a coordenada vertical da junta 3; e Z_3 é a coordenada longitudinal da junta 3. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

A linha X_3 mostra o movimento lateral que ocorre sempre com as patas em empuxo. O trecho plano corresponde ao movimento de avanço do robô com três patas em apoio e uma pata em vôo para novo posicionamento. Durante este percurso as duas patas de um mesmo lado voam para novo posicionamento. O trecho inclinado corresponde ao movimento de excursão lateral do robô em empuxo com as quatro patas apoiadas.

A linha Z_3 mostra movimento de empuxo e vôo da pata em função do tempo. O trecho de menor inclinação, com a coordenada variando do extremo positivo ao extremo negativo do gráfico, corresponde ao empuxo, e o trecho de maior inclinação, com a coordenada variando do extremo negativo ao extremo positivo do gráfico, corresponde ao percurso em vôo. A trajetória é mais íngreme porque o percurso em vôo tem uma trajetória com um comprimento maior e deve ser percorrido em um tempo muito menor, igual a $(1-\beta)*T$, com a velocidade média da pata muito maior.

A linha Y_3 mostra o movimento vertical da pata, que permanece a maior parte do tempo em um valor vertical fixo em empuxo, apresentando picos nos momentos do vôo da pata. Estes movimentos estão sincronizados com os trechos de maior inclinação, mostrados na linha Z_3 , correspondente ao percurso em vôo.

A Figura 4.3-12 mostra os ângulos de juntas gerado pelo *Nível Supervisor*, após o mapeamento das coordenadas geradas no espaço operacional para o espaço de juntas, e que são enviados para o robô. A abscissa comum são os *setpoints* enviados e a ordenada é em [radianos*1000]; θ_0 junta 0; θ_1 junta 1; θ_2 junta 2; θ_3 junta 3. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

O grau de liberdade θ_0 da junta 0 descreve uma trajetória com um primeiro segmento inclinado a partir da origem, correspondente ao movimento de baixar o robô, movendo a plataforma para a esquerda, para a decolagem da pata 3, que é a primeira a voar, de acordo com a matriz da equação (2.3.2).

Os movimentos seguintes alternam segmentos planos, onde as patas de um mesmo lado voam para novo posicionamento, e segmentos inclinados correspondentes à excursão lateral da plataforma de um lado para outro.

A pequena ondulação no final da linha de θ_0 corresponde ao vôo da pata 0, onde o movimento exige maior excursão da junta para manter a trajetória vertical do movimento no espaço operacional da perna.

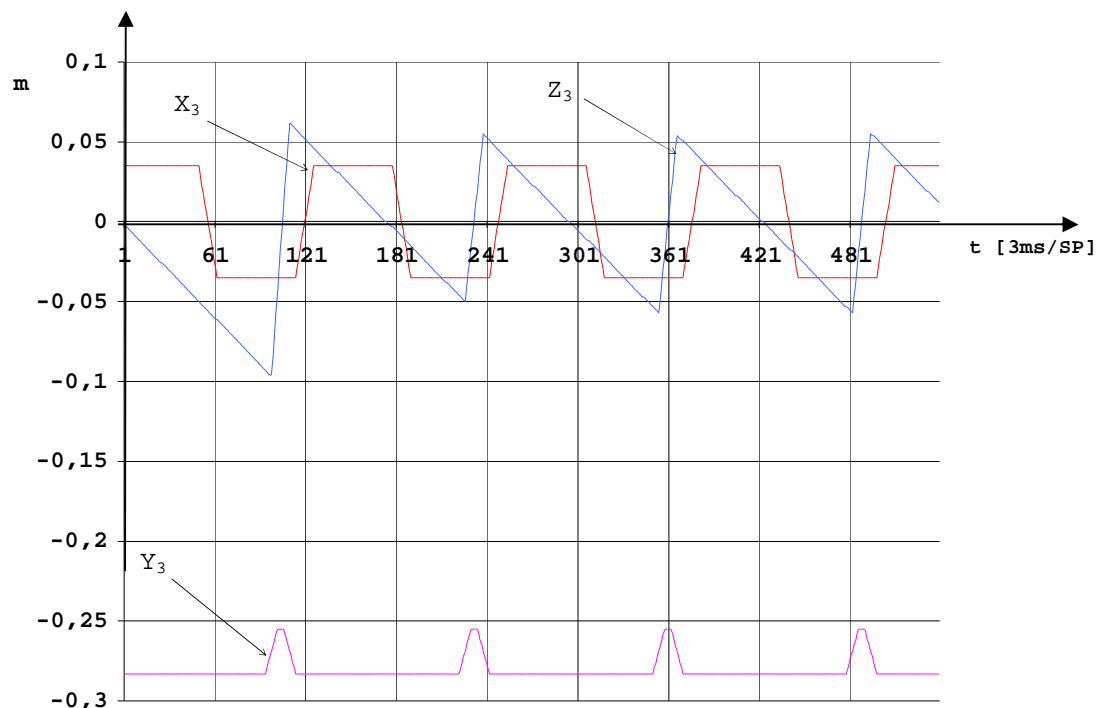


Figura 4.3-11: Coordenadas geradas para as juntas da pata 0 no espaço operacional.

Nas linhas correspondentes aos graus de liberdade θ_1 e θ_2 , o primeiro pico observado na parte superior da linha refere-se ao vôo da pata, conforme pode ser observado pelo sincronismo com os picos da linha de Y_3 , mostrados na Figura 4.3-11, correspondentes ao período de vôo da pata. Há um outro ponto de descontinuidade no meio das trajetória em empuxo, correspondente ao limite da excursão lateral da plataforma, o qual é feito mantendo-se a altura constante.

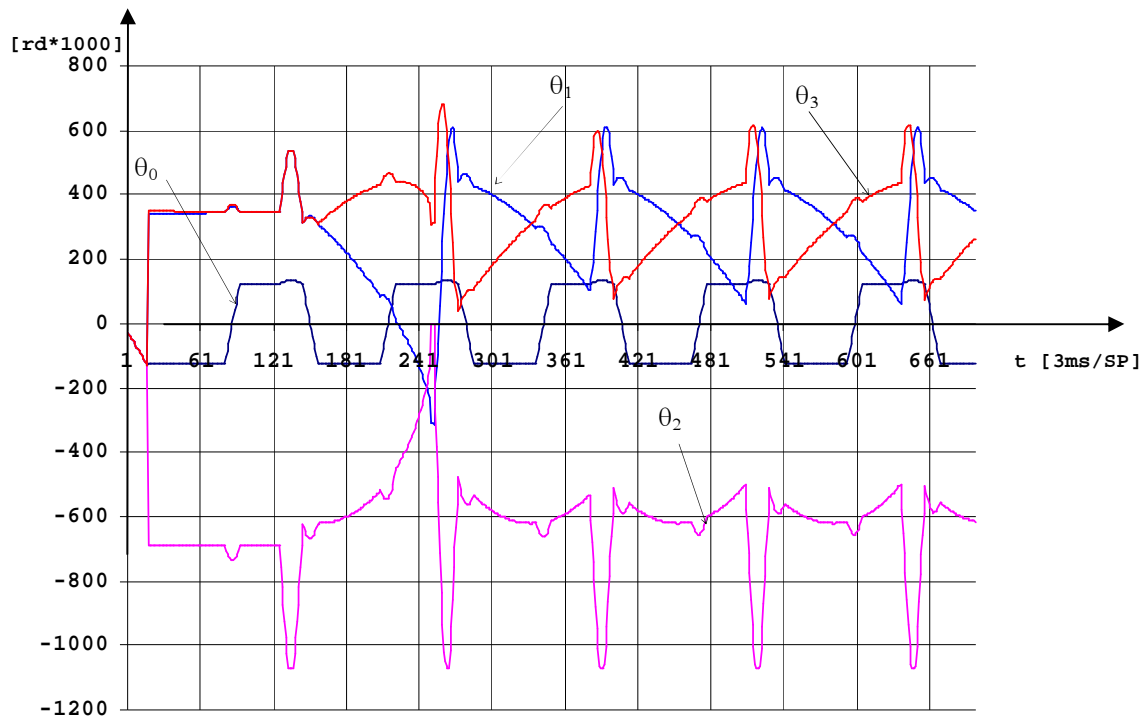


Figura 4.3-12: Coordenadas geradas para pata 0 no espaço de juntas.

4.3.3.4.3 Empuxo, vôo e sincronismo de trajetória das patas

A Figura 4.3-13 mostra as trajetórias sincronizadas das patas, durante o movimento. X_0 , Y_0 , Z_0 , X_1 , Y_1 , Z_1 , ..., X_3 , Y_3 e Z_3 são as coordenadas cartesianas da junta 3 das pernas 0, 1, 2 e 3, respectivamente. A abscissa comum são os *setpoints* enviados, e a ordenada está em milímetros. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

Pode-se observar na Figura 4.3-13 que o movimento lateral é o mesmo para todas as patas, já que não existe movimento de guinada da plataforma. O movimento lateral das patas é diferente quando o robô realiza uma trajetória em curva, onde as patas em empuxo descrevem uma trajetória reta inclinada em relação ao eixo longitudinal do robô.

Os trechos de pata em vôo podem ser observados pelas elevações das linhas Y . Durante o trecho de vôo de uma das patas, todas as outras patas estão em empuxo, conforme podemos observar nas linhas Z logo acima no gráfico.

Comparativamente ao percurso em empuxo, as linhas Z apresentam uma inclinação mais acentuada, devido ao curto espaço de tempo disponível para mover a pata em vôo, da posição

de limite cinemático para a posição de empuxo máximo. Este efeito provoca um aumento no consumo de energia e torna irregulares os movimentos da perna. A alternativa seria aumentar o número de pontos da trajetória da pata, o que resultaria em menores incrementos de *setpoints*, porém, diminuindo a velocidade final máxima do robô.

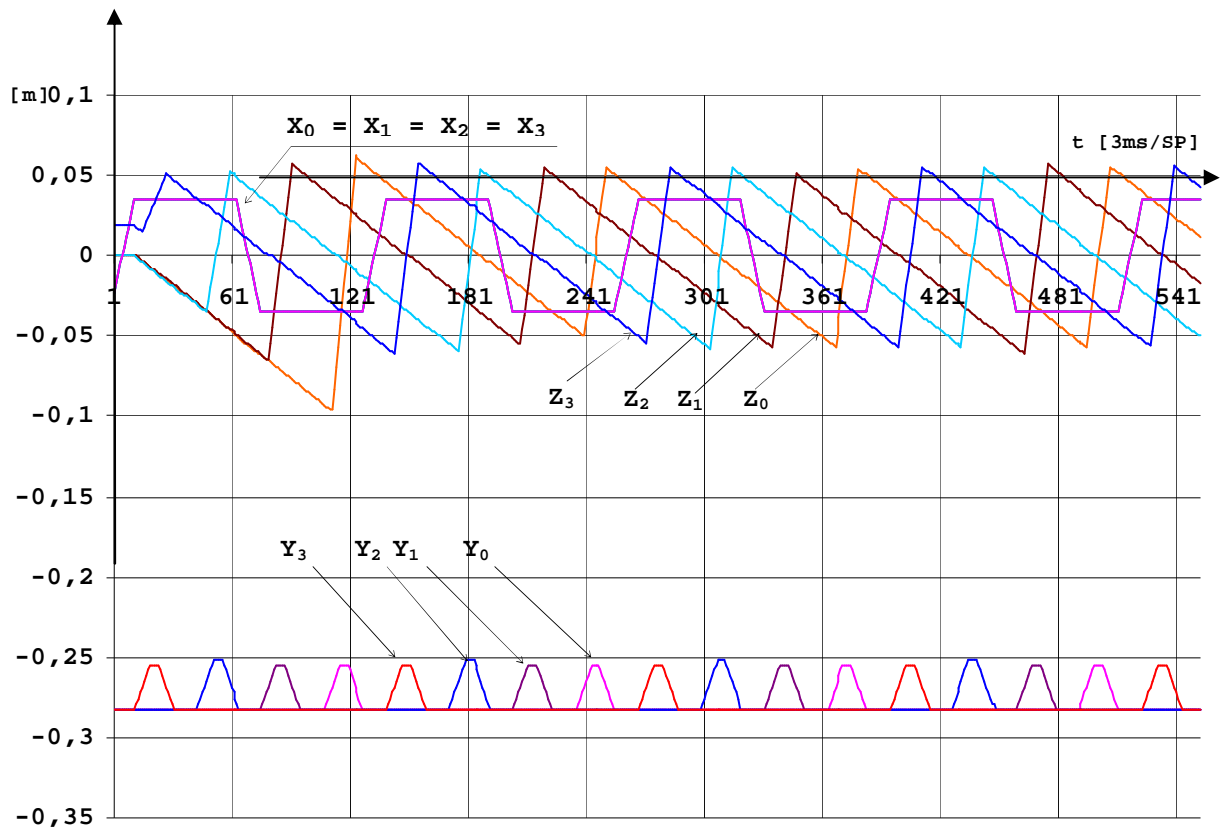


Figura 4.3-13: Coordenadas geradas para as patas no espaço operacional, referencial da perna.

4.3.3.5 Andando em curva

Para obter características de dirigibilidade reais ao acionar o controle de *Raio de curvatura*, foi montada uma função composta não linear, em função da saída linear do controle, de forma a variar o raio de curvatura mais rapidamente.

Na transição da trajetória reta para a curva, o raio muda de um valor infinito para um valor finito pequeno, considerada a ordem de grandeza do robô. Para permitir uma rápida redução do raio nessa situação, é utilizada uma função composta quadrática, e é definido um raio

máximo de curva, no caso de 30 m, acima do qual a trajetória é uma reta, e um raio mínimo de 1,2 m, devido aos limites cinemáticos da junta 0.

A função composta possibilita um raio máximo na origem, correspondente à posição central do controle, e um raio mínimo não nulo nas posições extremas à direita e à esquerda, conforme mostrado na Figura 4.3-14.

Acionando-se o *Controle de direção* da tela *Interface para andar*, mostrado na Figura 2.6-1, provoca-se uma saída ascendente ou descendente, conforme Figura 4.3-14(a), a qual é modulada em uma saída parabólica em (b). Esta saída é subtraída da reta constante (c) e resulta na função de direção (d).

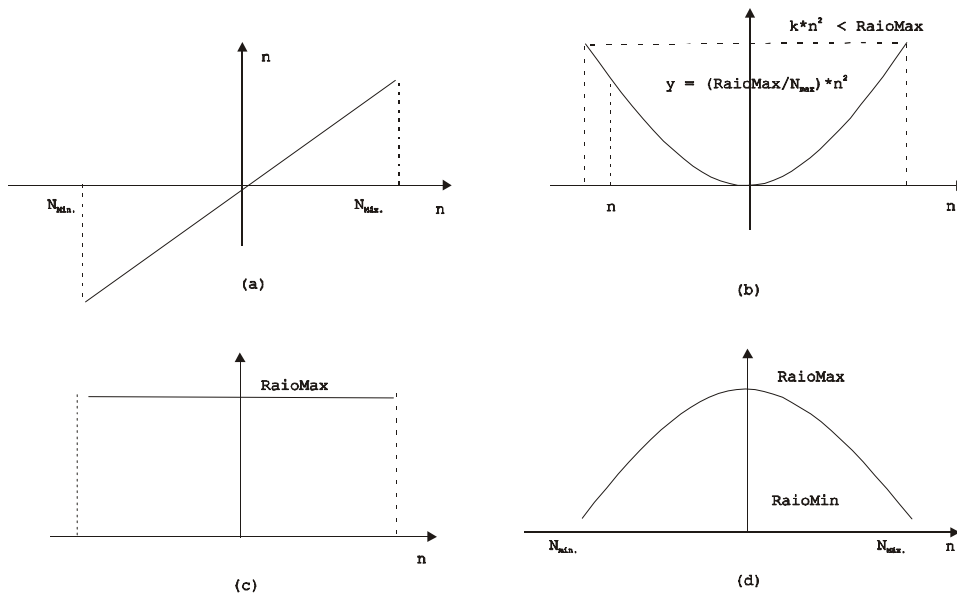


Figura 4.3-14: Esquema para interface de dirigibilidade do Guará.

Além disso, o incremento do *Controle de direção* é maior nas faixas de raios maiores, e menor nas menores, conforme mostrado no Anexo A1. Além de acionar o *Controle de direção*, é possível movimentar continuamente o controle deslizante *Raio de curvatura* com o *mouse* ou o teclado.

A Figura 4.3-15 mostra o diagrama de estratégia do processo *Andando em curva*, o qual é executado a partir do momento em que o controle de *Raio de curvatura*, mostrado na Figura 2.6-1, é acionado, iniciando assim um valor de raio de curvatura menor que infinito.

O processo *Lê e atualiza controles de direção na tela* lê a saída do *Controle de direção* ou do controles deslizante *Raio de curvatura*, calcula o raio de curvatura e atualiza a tela. Em seguida, são calculados os vetores posição e a matriz de rotação, conforme descrito no item 3.7. O processo seguinte calcula os incrementos resultantes nas direções radial e tangencial da pata, em relação à plataforma, os quais resultarão na trajetória curva do robô, e monta o vetor de coordenadas que será adicionado ao vetor de coordenadas da trajetória reta no espaço operacional. O processo é concluído e o mapeamento das coordenadas no espaço de juntas é feito após a soma dos vetor de coordenadas das trajetórias reta e curva.

A Figura 4.3-16 mostra as projeções da trajetória gerada no *Nível Supervisor* para as patas, em relação ao sistema de referência da perna, com o robô realizando uma curva com raio aproximado de 1,4 m e centro de curvatura à sua direita. A abscissa comum são os *setpoints* enviados e a ordenada está em metros; X_{30} , X_{31} , X_{32} e X_{33} são as coordenadas laterais da junta 3 das pernas 0, 1, 2 e 3; Y_{30} , Y_{31} , Y_{32} e Y_{33} são as coordenadas verticais da junta 3. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

As linhas X_{30} , X_{31} , X_{32} e X_{33} mostram o movimento lateral que ocorre sempre com as patas em empuxo. Pode-se observar a inclinação das linhas no período de empuxo, correspondente ao deslocamento lateral, em contrapartida às linhas planas em empuxo da trajetória reta, mostradas na Figura 4.3-11.

Conforme mostrado, as discontinuidades nas linhas X_{30} , X_{31} , X_{32} e X_{33} , mostradas na Figura 4.3-16, são devidas à excursão adicional da pata que ocorre durante o vôo, a qual é gerada no espaço operacional, no momento em que a pata está inclinada, alterando o ângulo da junta 0.

Observa-se também que as linhas das patas dianteiras X_{30} e X_{32} são inclinadas simetricamente em relação às linhas das patas traseiras X_{31} e X_{33} , conforme previsto no modelo de curva da Figura 2.6-2.

Na Figura 4.3-17 as linhas Z_{30} e Z_{32} mostram movimento de empuxo e vôo das patas em função do tempo. Pode-se observar que o percurso da pata 2 é maior que o da pata 0, cuja trajetória curva tem o maior raio, na maior amplitude vertical da linha correspondente.

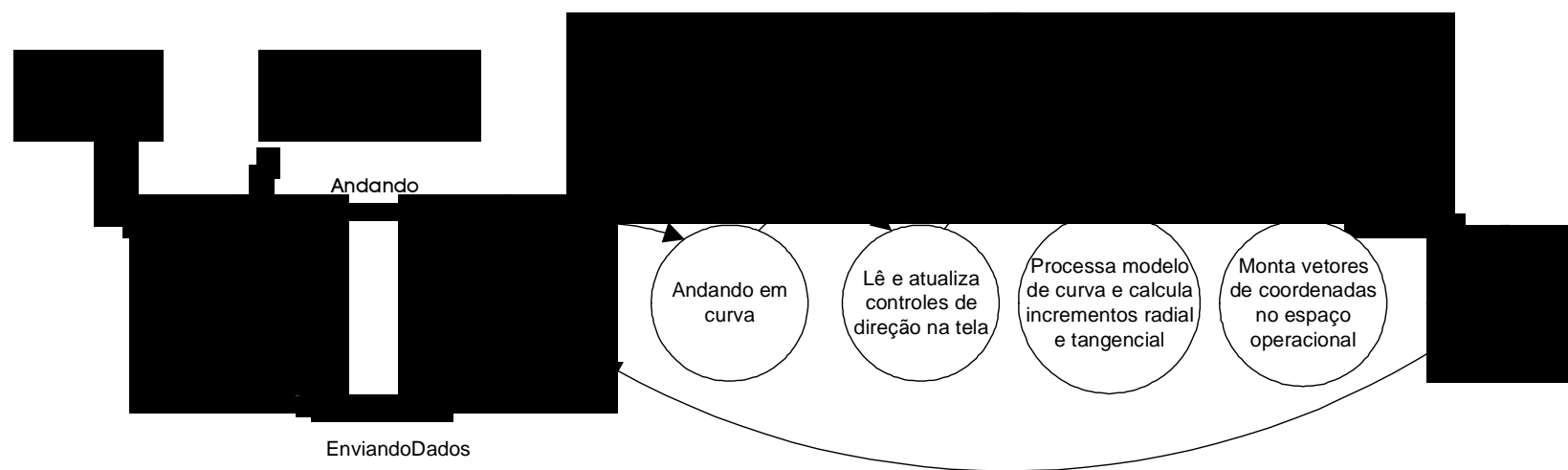


Figura 4.3-15: Diagrama de estratégia da andadura em curva.

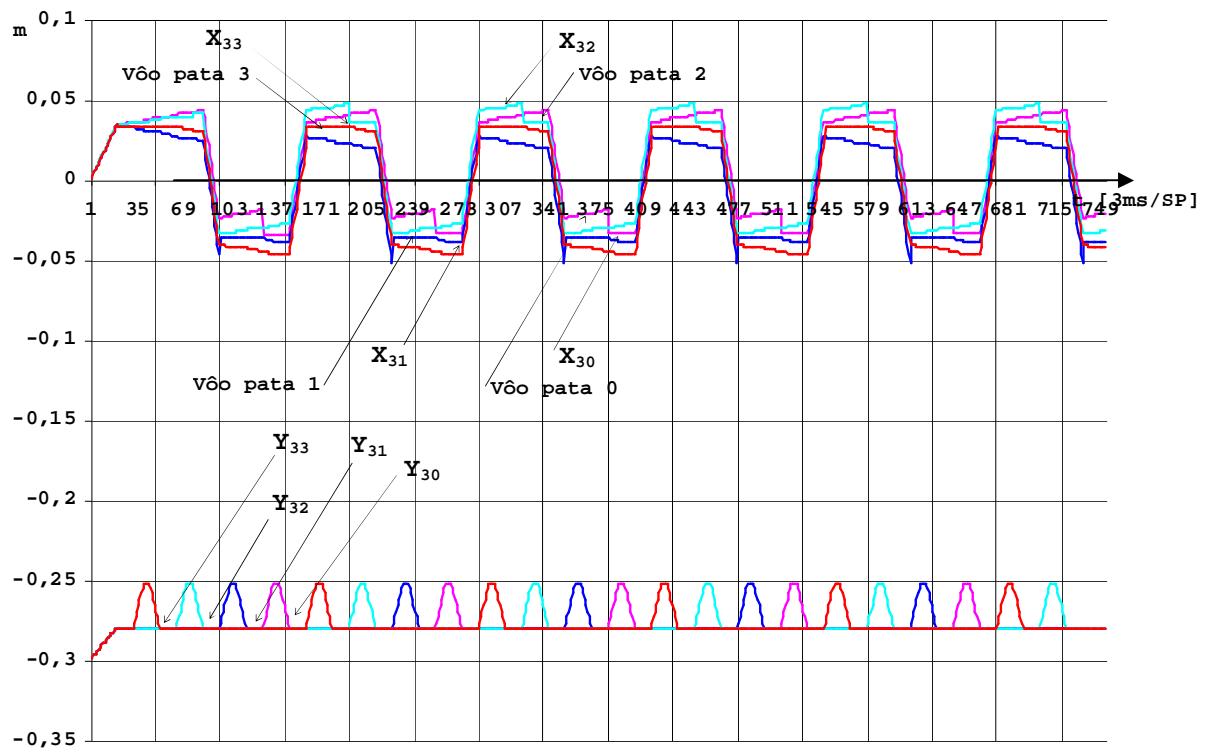


Figura 4.3-16: Coordenadas laterais e verticais geradas para as patas 0 a 3 na andadura em curva.

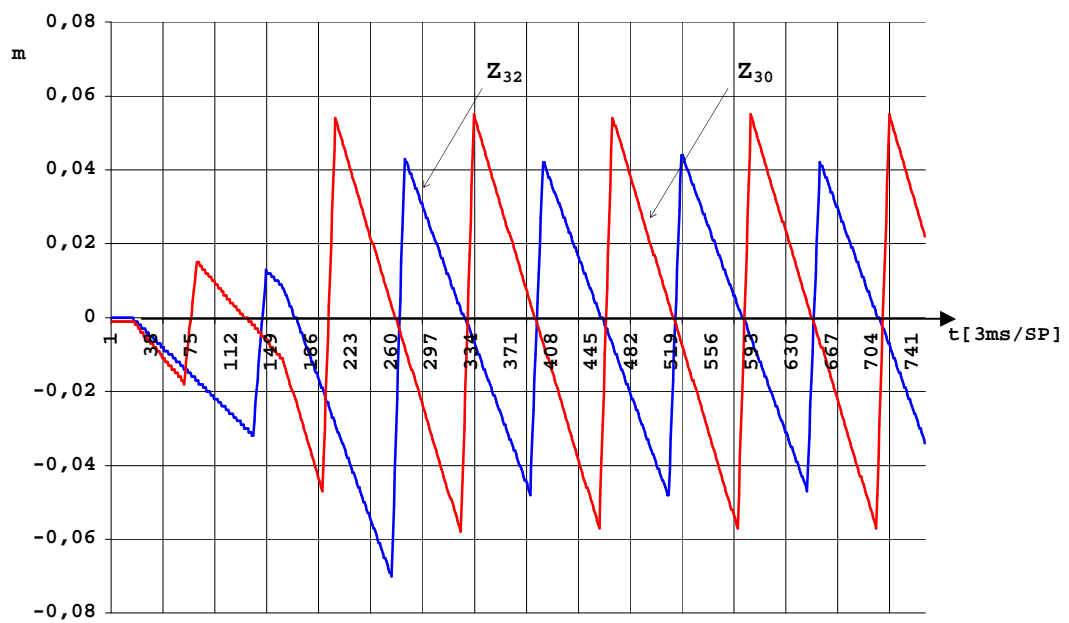


Figura 4.3-17: Coordenadas de empuxo geradas para as patas 0 e 2 na andadura em curva.

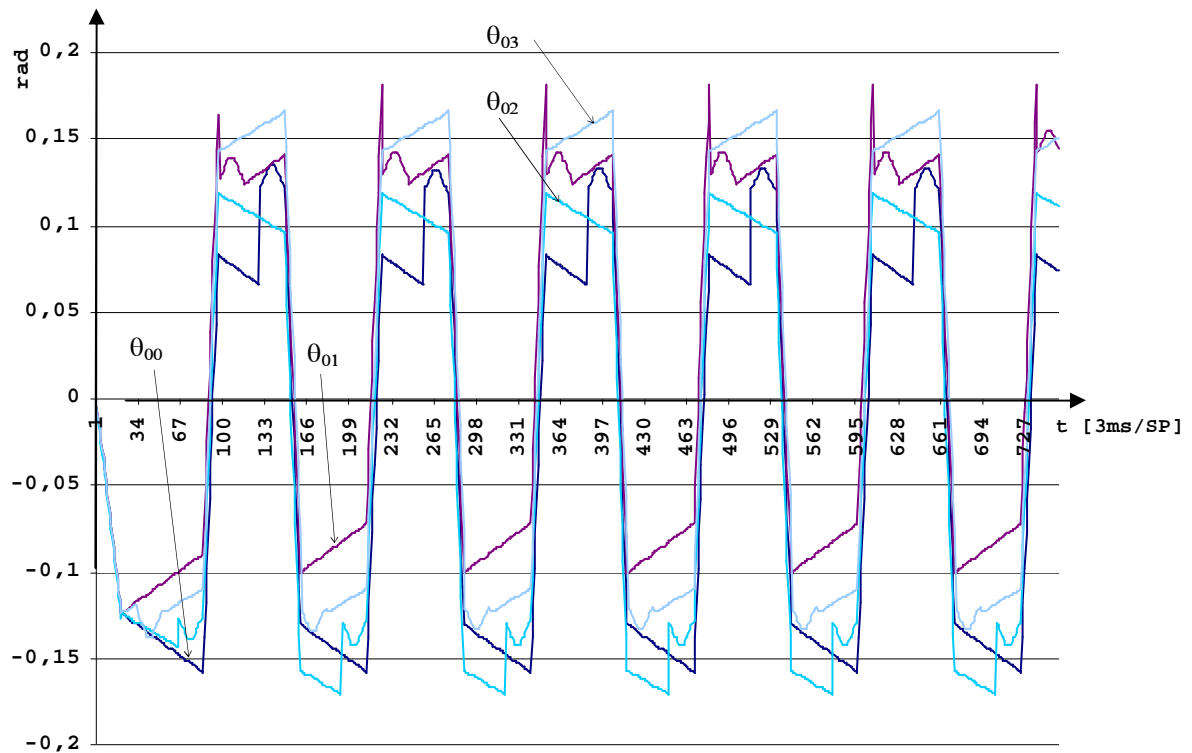


Figura 4.3-18: Coordenadas de juntas geradas para as patas 0 a 3 na andadura em curva.

A Figura 4.3-18 mostra os ângulos θ_{00} , θ_{01} , θ_{02} , e θ_{03} , das juntas 0 das pernas na andadura em curva. Pode-se observar que os ângulos de juntas de um mesmo lado são simétricos, porque a pata 0 dianteira move-se para fora e a pata traseira 1 move-se para dentro.

As descontinuidades nas linhas θ_{00} , θ_{01} , θ_{02} , e θ_{03} ocorrem nos períodos de vôo da pata, onde uma excursão maior da junta 0 ocorre quando a pata é erguida do piso, mantendo a coordenada lateral.

4.3.3.6 Envia dados

O processo *Envia dados* é acionado pelo processo principal no final do processamento de trajetória das patas, quando for ativada a variável funcional **EnviandoDados**.

A Figura 4.3-19 mostra o diagrama de estratégia do processo *Envia dados*, e a Tabela 4.3-8 mostra as variáveis funcionais associadas aos processos e grupos de movimento.

O processo consiste em transmitir os valores de *setpoints* para as quatro juntas de cada perna, na sequência da perna 0 para a 3, utilizando um registro de envio com identificadores de cada perna e confirmando a recepção da transmissão.

Terminada com sucesso a transmissão, é enviada uma mensagem solicitando ao *Nível Coordenador* a execução dos *setpoints* enviados para as quatro pernas.

Cada perna tem um registro de mensagem para receber, no *Nível Supervisor*, a confirmação de recepção do dado enviado e da solicitação de execução do *setpoint*. Após cada transmissão, o bit de novo dado recebido do registro de confirmação de cada pata é lido e desligado ciclicamente até que permaneça neste estado, confirmando a conclusão da transmissão pela CAN.

Devido às características de movimentação de alguns processos, durante o processamento de colisão algumas etapas não ativam a variável funcional ***EnviandoDados*** porque elas apenas comutam as variáveis de processamento, operação de patas e movimentação, com base em eventos e limites de movimento da etapa anterior, não havendo novo processamento de coordenadas de trajetória da pata. Nesses casos, as coordenadas das patas no espaço operacional são atualizadas com os valores da etapa anterior.

Tabela 4.3-8: Variáveis associadas ao processo de enviar dados

<i>Variável</i>	<i>Descrição da ação</i>
<i>Variável de processamento</i>	
EnviandoDados	Sinaliza para enviar dados através dos registros de comunicação
RecebendoDados	Sinaliza para ler registros de posição dos potenciômetros
Parado	Comutado por time out de comunicação com a CAN, sinaliza parada do processamento

Este processo é o único ponto de comunicação para envio de dados ao robô durante a andadura e o processamento de colisão. É executado a cada ciclo, independentemente do que foi executado anteriormente, exceto quando os processos anteriores não ativaram a variável funcional associada ao envio de dados para o robô.

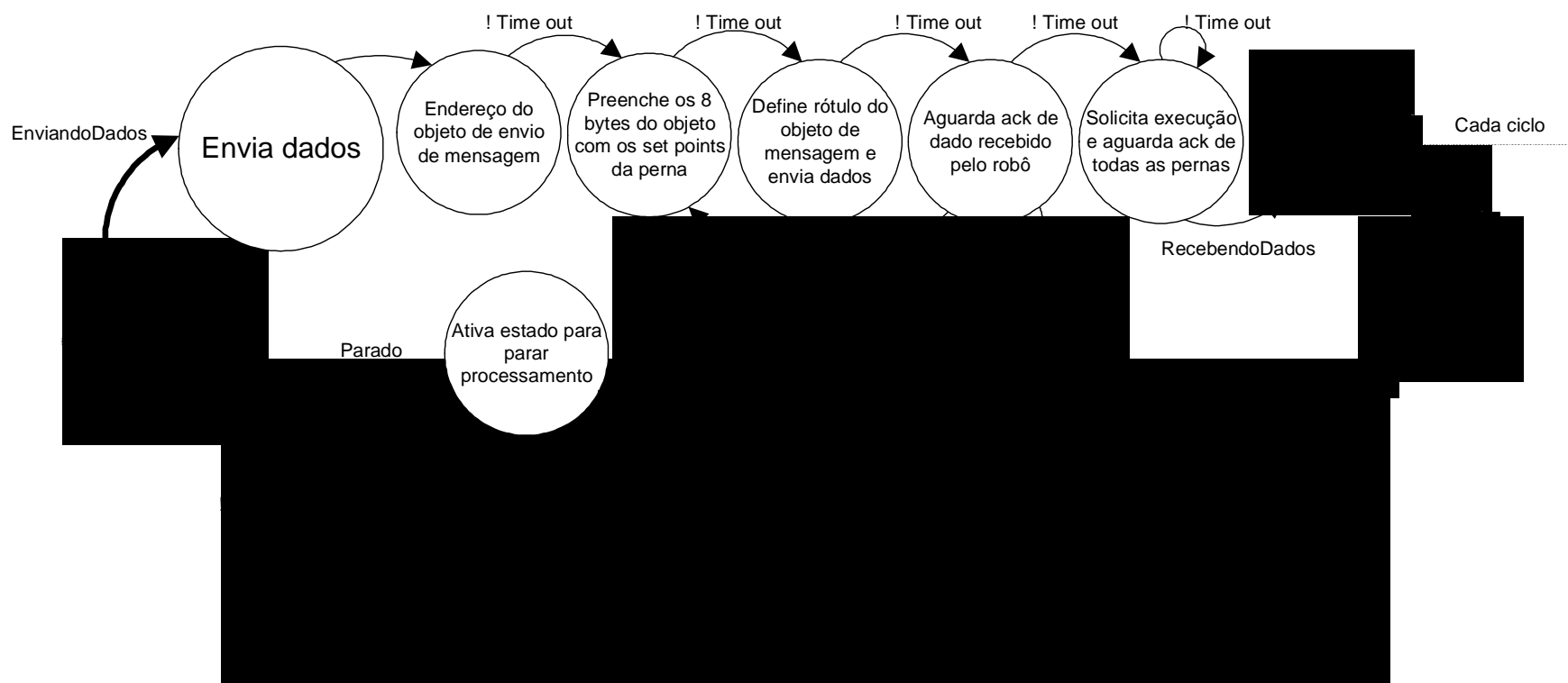


Figura 4.3-19: Diagrama de estratégia de enviar dados para o robô.

4.3.3.7 Lê os registros de dados dos potenciômetros

O processo *Lê os registros de dados dos potenciômetros* é acionado pelo processo principal no final do processamento de trajetória das patas, quando a variável de processamento *RecebendoDados* foi acionado após *Envia dados*. A Figura 4.3-20 mostra o diagrama de estratégia do processo *Lê os registros de dados dos potenciômetros*.

Cada perna tem um registro de mensagens de oito *bytes*, sendo dois *bytes* para cada junta, para receber os valores de ângulo de cada junta após a execução do *setpoint*. O processo consiste em ler estes registros, na sequência da perna 0 para a 3, em *loop*, até completar a leitura de dados das quatro pernas. A leitura de cada registro de mensagens desliga o respectivo registro de novo dado recebido e é repetida até que ele permaneça neste estado, de forma a confirmar a conclusão da transmissão do dado pela CAN.

4.3.3.8 Atualiza variáveis e controles

O processo *Atualiza variáveis e controles* é acionado por *Lê os registros de dados dos potenciômetros* quando houver envio de dados ao robô ou pelo processo principal no final do processamento de trajetória das patas.

A Figura 4.3-21 mostra o diagrama de estratégia do processo *Atualiza variáveis e controles*.

O processo mapeia as coordenadas de juntas recebidas dos potenciômetros das juntas do robô no espaço de ângulos de juntas, executa a cinemática direta de cada perna com os respectivos valores de ângulos de juntas para obter as coordenadas cartesianas das patas no espaço operacional e no referencial da respectiva perna, e atualiza as tela *Interface para Andar* e *Interface para CAN* do programa do *Nível Supervisor*.

Também são atualizados o estado das chaves de contato das patas, os valores de *setpoints* enviados e recebidos e, internamente, as matrizes de coordenadas no espaço operacional e no espaço de juntas, as quais são utilizadas pelos processos *Executando andadura* e *Processa Colisão*.

Este processo é executado a cada ciclo, independentemente do que foi executado anteriormente, pois os valores de coordenadas no espaço operacional realimentam os modelos de andadura e colisão no *Nível Supervisor*.

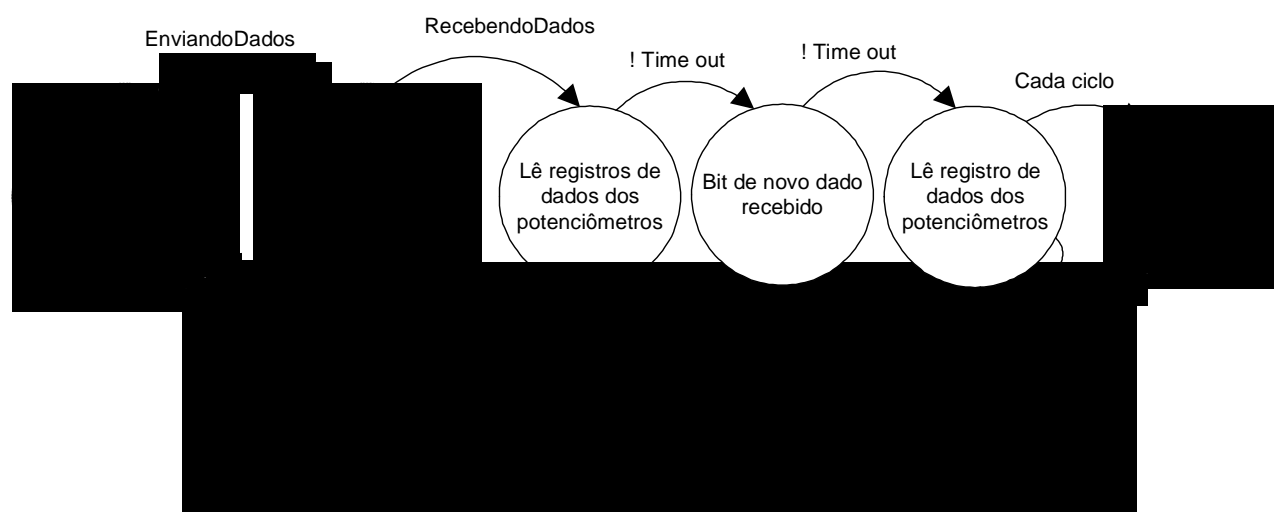


Figura 4.3-20: Diagrama de estratégia de ler os registros de dados dos potenciômetros.

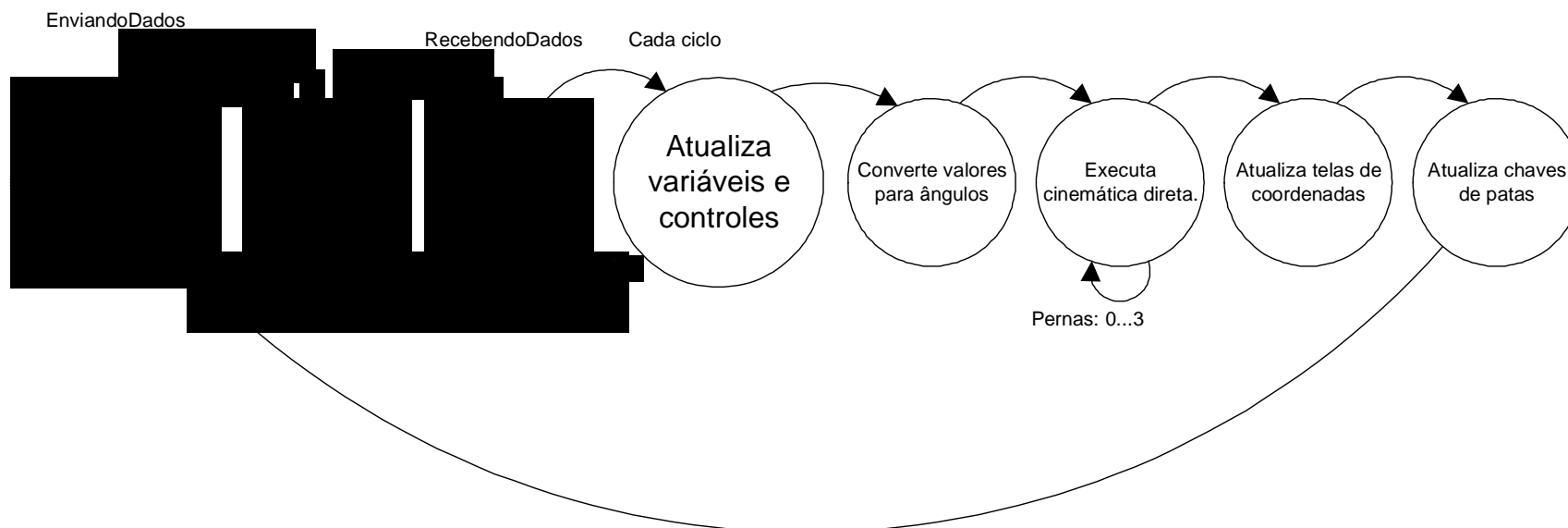


Figura 4.3-21: Diagrama de estratégia de atualizar variáveis e controles.

4.4 Discussão

O ciclo principal de processamento do robô para a execução de cada tarefa envolve não somente as informações de modelos de andadura, modelos cinemáticos, considerações de equilíbrio, estratégias de cálculo, obtenção das variáveis de controle e sincronismo no tempo de execução da tarefa, mas também a construção de estruturas de dados e de processos que possibilitem desempenho compatível em termos de tempo.

Uma grande capacidade de processamento é exigida pelos robôs quadrúpedes e pode ser satisfeita distribuindo-a em camadas de hardware com conexão de banda larga entre elas. A alternativa adotada para o Guará foi a construção de uma arquitetura de três camadas, sendo um *Nível Supervisor* e um *Nível Coordenador* conectados por uma rede CAN de 1 Mbs e um *Nível Atuador*, que permite que todo o processamento, objeto do presente trabalho, seja executado em um PC de mesa ou um *notebook* a bordo do robô, liberando os microcontroladores dessa tarefa. A rede facilita o acesso aos elementos de controle, ao acionamento dos motores e à aquisição de dados para o modelo comunicar-se com o robô.

Um núcleo central implementa toda a estratégia de processamento e, um grupo de eventos e processos o direciona de acordo com a necessidade da tarefa sendo desempenhada pelo robô. Ao final de cada ciclo do núcleo central, são atualizadas todas as coordenadas das patas no espaço operacional, a partir dos valores lidos através da CAN nos potenciômetros de cada junta e da cinemática direta. Também são atualizados os estados das chaves das patas, as interfaces gráficas e o simulador gráfico.

Diversas fases do processamento utilizam-se de grupos de movimentos para impor restrições de posicionamento ao robô com a finalidade de assegurar a continuidade entre os processos e também que as tarefas sejam executadas com a melhor margem de estabilidade possível. Esta estratégia visa reduzir o número de graus de liberdade do espaço de execução das manobras comuns ou especiais.

O robô sempre parte de uma posição conhecida quando seu estado é comutado por um evento aleatório, ao final de um processo de colisão e na partida do repouso, até que as patas se encontrem em fase. O empuxo é reduzido na primeira passada para evitar que as patas atinjam o limite cinemático antes de voar para novo posicionamento. A posição das patas na partida é

previamente analisada e, se necessário, elas são equalizadas na origem antes de iniciar a andadura.

A geração das coordenadas das patas é feita pelo modelo de forma incremental e é sincronizada pela matriz da andadura. O controle dos limites cinemáticos é feito na geração das coordenadas no espaço cartesiano e, através de um filtro, no momento de enviar os *setpoints* para o robô. Os resultados obtidos mostram uma boa precisão na execução dos *setpoints*, que é confirmado pela suavidade do movimento do robô.

A estrutura de dados utilizada para o sistema de coordenadas é uma biblioteca de matrizes, desenvolvida para aplicação no presente trabalho. Uma classe encapsula as estruturas de dados e as funções de soma, multiplicação, produto interno, produto vetorial e operações de rotação de vetores, que facilita a cinemática direta e inversa, além das operações com os vetores posição na andadura em curva.

Capítulo 5

Estratégia de transposição de obstáculos

5.1 Introdução

Neste capítulo é apresentada a estratégia para transposição ou o desvio de obstáculos que se colocam na trajetória do robô.

O desvio de obstáculos em tempo real é uma das habilidades críticas para a aplicação de sistemas de robôs móveis e sua integração com o ambiente humano.

Todos os robôs móveis realizam algum tipo de detecção de obstáculo, desde algoritmos primitivos que detectam o obstáculo e param o robô a uma determinada distância do mesmo, a algoritmos sofisticados que habilitam o robô a desviar-se dele. Estes últimos são muito mais complexos, uma vez que envolvem não somente a detecção, mas também alguma medida quantitativa das dimensões, que, uma vez determinadas, permitem ao algoritmo desviar do obstáculo e retomar o movimento em direção ao seu objetivo. A navegação autônoma representa um alto nível de dificuldade, uma vez que requer a detecção e o desvio de obstáculos, simultaneamente, guiando o robô na direção do seu objetivo. Em geral pressupõe a existência de obstáculos conhecidos e desconhecidos e a necessidade de algoritmos de planejamento global de trajetória, de forma a guiar o robô entre obstáculos, e também algoritmos locais para detecção e desvio dos obstáculos (BORENSTEIN; KOREN, 1989)

Este trabalho concentra-se na elaboração de algoritmos locais para detecção, transposição ou desvio dos obstáculos com tipos predefinidos, utilizando apenas informações de sensores de contato nas patas.

5.2 Percepção e transposição de obstáculos

O robô percebe um obstáculo quando há um toque com a pata, ou ainda, quando a pata não atinge o solo na coordenada vertical prevista.

No primeiro caso, o *Nível Coordenador* pára o robô, envia o estado das chaves de contato para o *Nível Supervisor*, o qual executa procedimentos de identificação do obstáculo e inicia o procedimento específico até a sua superação, ou desvio, quando então retorna à condição operacional anterior ao obstáculo.

No segundo caso, o *Nível Supervisor* executa procedimentos para identificar um obstáculo tipo *Canal* ou *Degrau descendente*.

O Guará utiliza uma seqüência de processos e grupos de movimentos que são escolhidos em função do estado de chaves de contato, dos limites cinemáticos das juntas, de valores limites de distâncias e de variáveis funcionais.

Todas as trajetórias das patas são geradas no espaço operacional e as patas realizam movimentos básicos em uma linha vertical ou em um plano paralelo à plataforma.

Os movimentos para deslocar uma pata em um plano, em relação à plataforma, são efetuados sempre com a pata no ar. Se a pata a ser deslocada estiver apoiada, ela é elevada do solo até a mudança de estado das chaves da sola, indicando que não há mais contato com o solo, antes de ser deslocada.

Os movimentos para mover a plataforma vertical ou horizontalmente, são sempre efetuados com as quatro patas em apoio.

Todos os movimentos complexos do robô são construídos a partir dos movimentos básicos das patas e da plataforma.

Um grupo de movimentos é uma seqüência de movimentos básicos, separados por eventos de chaves ou limites de coordenadas, para efetuar movimentos complexos tais como baixar uma pata após uma colisão ou apoiá-la no piso acima do degrau. Além de movimentos básicos, um grupo de movimentos mais complexo pode conter outros grupos de movimento.

Os processos de colisão são combinações de grupos de movimentos, em vários níveis, os quais resultam em movimentos complexos que habilitam o robô a detectar um obstáculo, identificá-lo e implementar a estratégia completa para superá-lo ou, em caso de fracasso, desviar-se do mesmo.

Assim sendo, cada grupo de movimentos realiza um ou mais ciclos, ou outros grupos de movimentos, cada um compreendendo a geração da trajetória da pata em relação à plataforma, no espaço operacional da perna, o mapeamento das coordenadas no espaço de juntas e a conversão das coordenadas de juntas expressas em ângulos para coordenadas dos potenciômetros.

Na sequência, é feito o envio dos *setpoints* para o controlador do *Nível Coordenador* o qual, após executar os *setpoints*, devolve as coordenadas executadas dos potenciômetros, para o *Nível Supervisor* o qual, ainda no mesmo ciclo, efetua a conversão para coordenadas angulares, a cinemática direta para mapeamento das coordenadas de juntas recebidas nas coordenadas do espaço operacional e a atualização das variáveis de controle do robô.

A detecção do obstáculo é feita primeiramente pelas patas dianteiras 0 e 2 do robô. A Figura 5.2-1 mostra as posições das chaves de contato e como elas podem ser acionadas isoladamente, sendo: (a) chave calcanhar; (b) chave sola do pé atrás; (c) chave sola do pé à frente, (d) chave dedão e (e) nenhuma chave.

Existem situações onde mais de uma chave é acionada; ao baixar a pata durante a andadura, ocorre o toque simultâneo das chaves sola à frente e sola atrás, e ao tocar o degrau, já no fim da trajetória aérea, pode ocorrer o acionamento simultâneo das chaves da sola e dedão.

Também podem ocorrer isoladamente as chaves sola à frente ou atrás, no movimento em andadura ou colisão. Nesses casos, o controlador continua o movimento por um ou mais *setpoints* na expectativa de que ocorra o acionamento da chave oposta. Caso não ocorra, um processo de transposição de obstáculo é ativado pelo controlador.

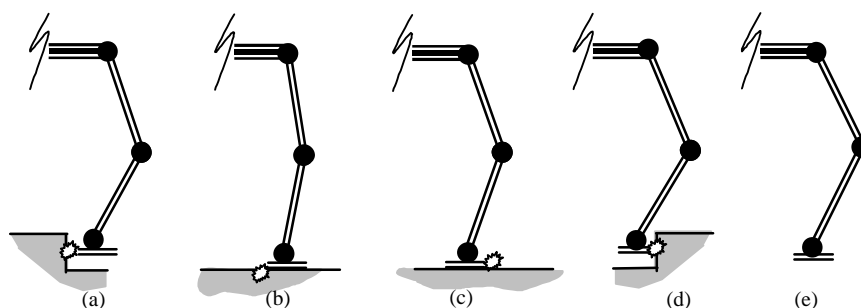


Figura 5.2-1: Toque das patas durante o voo ou movimentando para superar obstáculo.

Com o robô andando normalmente, a variável de processamento do ciclo principal é *ExecutandoPasso* e a variável de processamento da andadura é *Andando*, sendo executados pelas pernas, em relação à plataforma, os grupos de movimentos inerentes ao andar normal onde cada pata executa um movimento cíclico de empuxo e vôo, para novo posicionamento, e ocorrem os eventos decolar ao final do empuxo e um evento aterrissar ao final do vôo.

5.2.1 Canal

O processo *Canal* é acionado por *Processa colisão* quando não acontecer uma mudança de estado na chave da pata em um intervalo predefinido de *setpoints* após a ativação da variável de *MomentoCritico*, conforme mostrado na Figura 5.2-2, ou se já está ativada a variável de processamento *TranspondoCanal*.

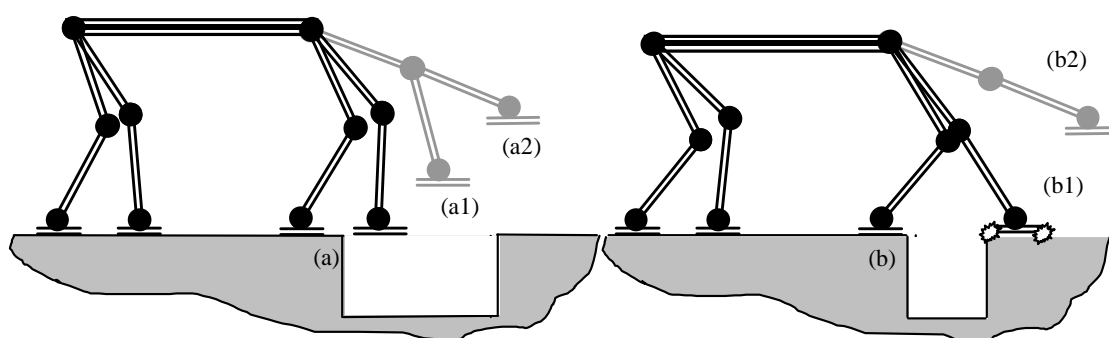


Figura 5.2-2: Obstáculo tipo canal; (a) canal largo; (b) canal estreito.

A Figura 5.2-3 mostra o diagrama de estratégia do processo *Canal*, e a Tabela 5.2-1 mostra as variáveis associadas aos grupos de movimento para cruzar um canal.

Durante a superação de um obstáculo tipo canal, composto de um trecho curto de piso baixo, em relação ao piso no qual o robô está andando, primeiro é feita uma tentativa de apoiar a pata na outra borda do canal.

A pata que estava em vôo e não aterrissou é recuada até tocar o calcanhar na parede e adquirir a sua coordenada. Em seguida, a pata é afastada e elevada até a posição limite, deslocada para trás de um valor igual ao comprimento da sola do pé de forma a ser apoiada no piso na posição mais próxima possível da borda.

Nesse ponto as demais patas são equalizadas em relação à posição de apoio das patas, e a plataforma do robô é deslocada para a posição de excursão máxima à esquerda para movimentação da pata 2.

A pata 2 é erguida, deslocada para a frente e abaixada na tentativa de obter apoio na borda oposta do canal, conforme mostrado na Figura 5.2-2(a), cujos detalhes (a1) e (a2) mostram o movimento de elevação da pata. Nesta figura pode-se deduzir que não há alcance para apoiar a pata na extremidade oposta do canal.

Se o canal é estreito como na Figura 5.2-2(b), pode-se deduzir que há alcance para apoiar a pata na borda oposta do canal, conforme detalhes (b2) e (b1) que mostram a pata na altura máxima e apoiada, respectivamente.

Em caso de sucesso na tentativa de apoiar a pata 2 do outro lado, o canal é considerado estreito, conforme Figura 5.2-2(b), a pata é apoiada e o processo é então reiniciado, movendo a plataforma lateralmente, de forma a movimentar a pata oposta à frente e apoiá-la no lado oposto do canal. Para movimentar uma pata à esquerda, a plataforma é deslocada à direita e vice versa.

Tabela 5.2-1: Variáveis associadas aos processos para cruzar um canal.

<i>Variável</i>	<i>Descrição da ação</i>
<i>Variáveis de operação do robô</i>	
TranspondoCanal	Sinaliza processamento de colisão com canal
Andando	Sinaliza processamento de andadura
DegrauDescendente	Sinaliza processamento para descer degrau
<i>Variáveis de movimento da pata</i>	
Baixando	Baixa a pata até uma posição pré determinada
Erguendo	Ergue a pata até uma posição pré determinada
MovendoAtras	Move para trás até uma posição pré determinada
MovendoAfrente	Move para frente até uma posição pré determinada

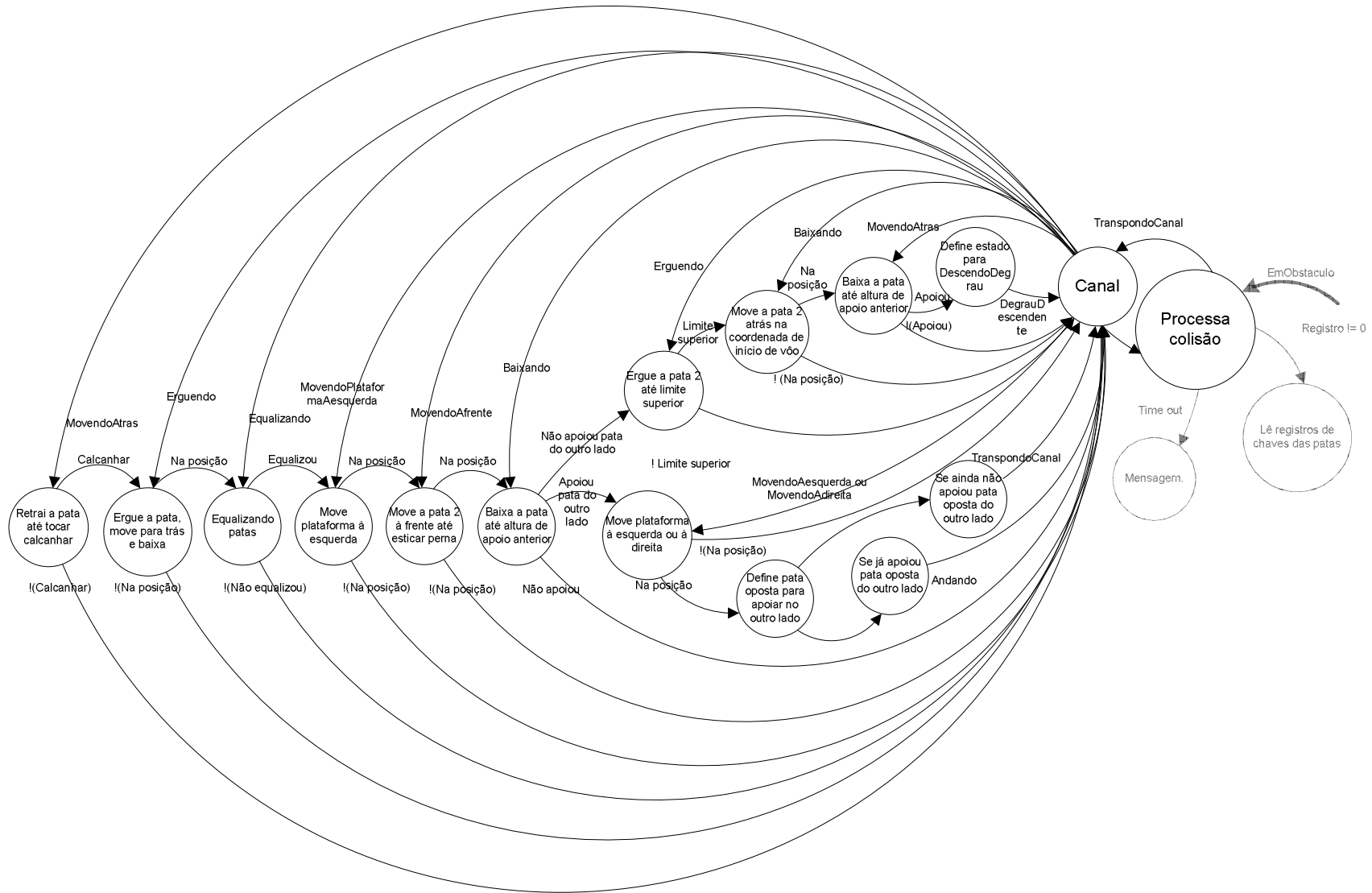


Figura 5.2-3: Diagrama de estratégia para cruzar um canal.

Com as patas dianteiras apoiadas no lado oposto do canal, o processo ativa a variável de processo *ExecutandoPasso* que retorna o controle para a variável de operação *Andando* e o mesmo processo, desde a detecção do obstáculo até o apoio na borda oposta do canal, mostrado na Figura 5.2-3 é executado para as patas traseiras 1 e 3.

Em caso de fracasso na tentativa de apoiar a pata 2 do outro lado, significa que pode ser um canal largo, conforme Figura 5.2-2(a), ou um degrau descendente, conforme Figura 5.2-4. A pata 2 é retornada à posição inicial e é ativada a variável *DescendoDegrau*, que direciona o processamento para *Degrau descendente*.

5.2.2 Degrau descendente

O processo *Degrau descendente* é acionado por *Processa colisão* quando não acontecer uma mudança de chave da pata em um intervalo predefinido de *setpoints*, após a ocorrência de *MomentoCrítico*, e não se tratar de obstáculo tipo *Canal*, conforme mostrado na Figura 5.2-4, ou se já está ativada a variável *DegrauDescendente*.

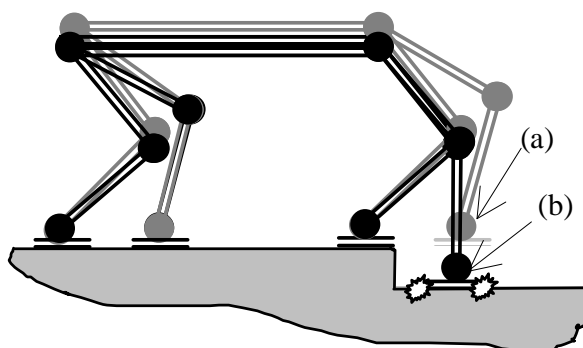


Figura 5.2-4: Obstáculo tipo descida de degrau.

A Figura 5.2-5 mostra o diagrama de estratégia do processo *Degrau descendente*, e a Tabela 5.2-2 mostra as variáveis associadas aos grupos de movimento para descer um degrau.

O procedimento para *Degrau descendente* é efetuado depois de verificar que não se trata de um *Canal* e é iniciado após o recuo da pata e apoio na posição mais próxima da borda, da equalização das patas e do posicionamento da plataforma na excursão máxima à esquerda para início de movimentação da pata 2.

A pata 2 é erguida, deslocada para frente, conforme Figura 5.2-4, detalhe (a), e abaixada até sua posição limite inferior. Se a pata não apoiar no piso embaixo, conforme Figura 5.2-4, detalhe (b), a plataforma é abaixada. Se a pata não apoiar até o limite de altura da plataforma, a pata é retornada à posição anterior ao vôo e é ativada a variável associada ao processo **Desvia**.

Se ao tentar baixar a pata ocorrer o acionamento de uma chave **SolaAtras**, trata-se de uma rampa descendente e variável associada é ativada para desviar o processo para **Rampa**.

Em caso de sucesso, se a pata apoiar embaixo, a plataforma é posicionada na excursão lateral máxima do outro lado, é ativada a variável associada **DegrauDescendente** e o processo é então reiniciado para apoio da pata oposta no piso embaixo.

Tabela 5.2-2: Variáveis associadas aos processos de colisão com degrau descendente.

Variáveis	Descrição da ação
<i>Variáveis de operação do robô</i>	
DegrauDescendente	Sinaliza processamento de descida de degrau
Andando	Sinaliza processamento de andadura
Rampa	Sinaliza processamento de rampa
Desvia	Sinaliza desvio do obstáculo
<i>Variáveis de movimento da pata</i>	
Baixando	Baixa a pata até uma posição pré determinada
Erguendo	Ergue a pata até uma posição pré determinada
MovendoAtras	Move para trás até uma posição pré determinada
MovendoAfrente	Move para frente até uma posição pré determinada
<i>Variáveis de movimento da plataforma</i>	
BaixandoPlataforma	Baixa a plataforma até uma posição pré determinada
MovendoPlataforma	Deslocando plataforma na horizontal

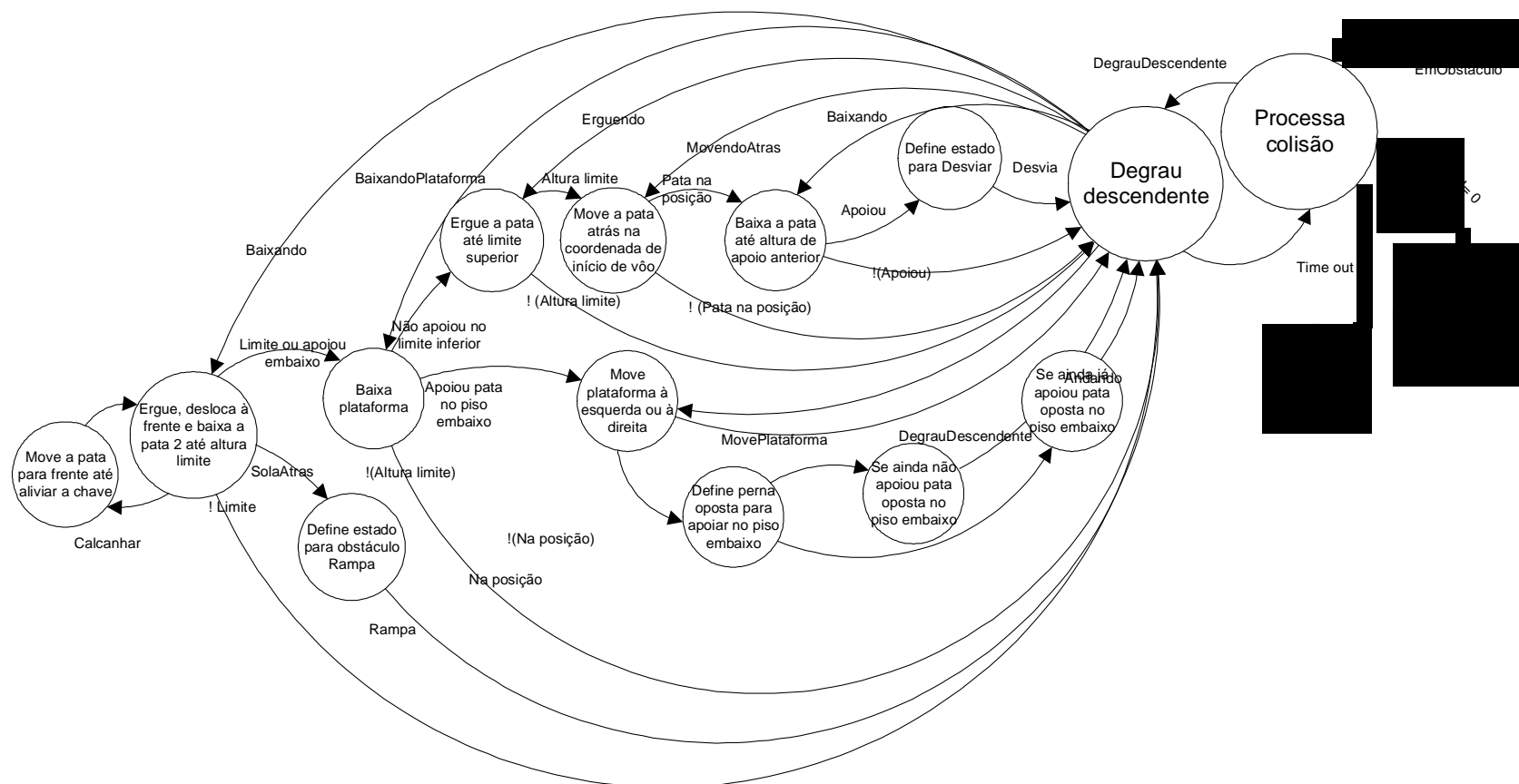


Figura 5.2-5: Diagrama de estratégia de degrau descendente.

A pata é girada no sentido anti horário e baixada simultaneamente, conforme mostrado na Figura 5.2-6(a) no detalhe (a2), para obter um evento de chave ***SolaAtras*** e adquirir o valor do ângulo de inclinação da rampa.

A plataforma é posicionada na excursão lateral máxima do outro lado, a variável ***Rampa*** é mantido ativa e o processo é então reiniciado para apoio da pata oposta na rampa.

Tabela 5.2-3: Variáveis associadas aos processos de rampa.

<i>Variáveis</i>	<i>Descrição da ação</i>
<i>Variáveis de operação do robô</i>	
Rampa	Sinaliza processamento de rampa
RampaAscendente	Sinaliza processamento de rampa ascendente
RampaDescendente	Sinaliza processamento de rampa descendente
Andando	Sinaliza processamento de andadura
<i>Variáveis de movimento da pata</i>	
GirandoParaApoiar	Girando para tentar apoiar no plano
Baixando	Baixa a pata até uma posição pré determinada
<i>Variáveis de movimento da plataforma</i>	
MovendoPlataforma	Deslocando na horizontal para esquerda

Ocorrendo ***SolaAtras***, conforme mostrado na Figura 5.2-6(b) no detalhe (b1), se a mudança ocorrer dentro de um intervalo predefinido de *setpoints* após a ocorrência, de ***MomentCritico***, o controle continua acionando a pata mais um *setpoint*, só transferindo o processamento para ***Rampa*** no modo descendente, mostrado na Figura 5.2-6(b), se não ocorrer um evento de chave ***Sola***. Se ***SolaAtras*** ocorrer fora do intervalo, o processo ***Rampa*** no modo descendente é ativado imediatamente.

A pata é girada no sentido horário e baixada simultaneamente, conforme mostrado na Figura 5.2-6(b) no detalhe (b2), de forma a obter um evento de chave ***SolaAfrente*** e adquirir o valor do ângulo de inclinação da rampa.

Assim, a plataforma é posicionada na excursão lateral máxima do outro lado, a variável funcional ***Rampa*** é mantida ativa e o processo é então reiniciado para apoio da pata oposta na rampa.

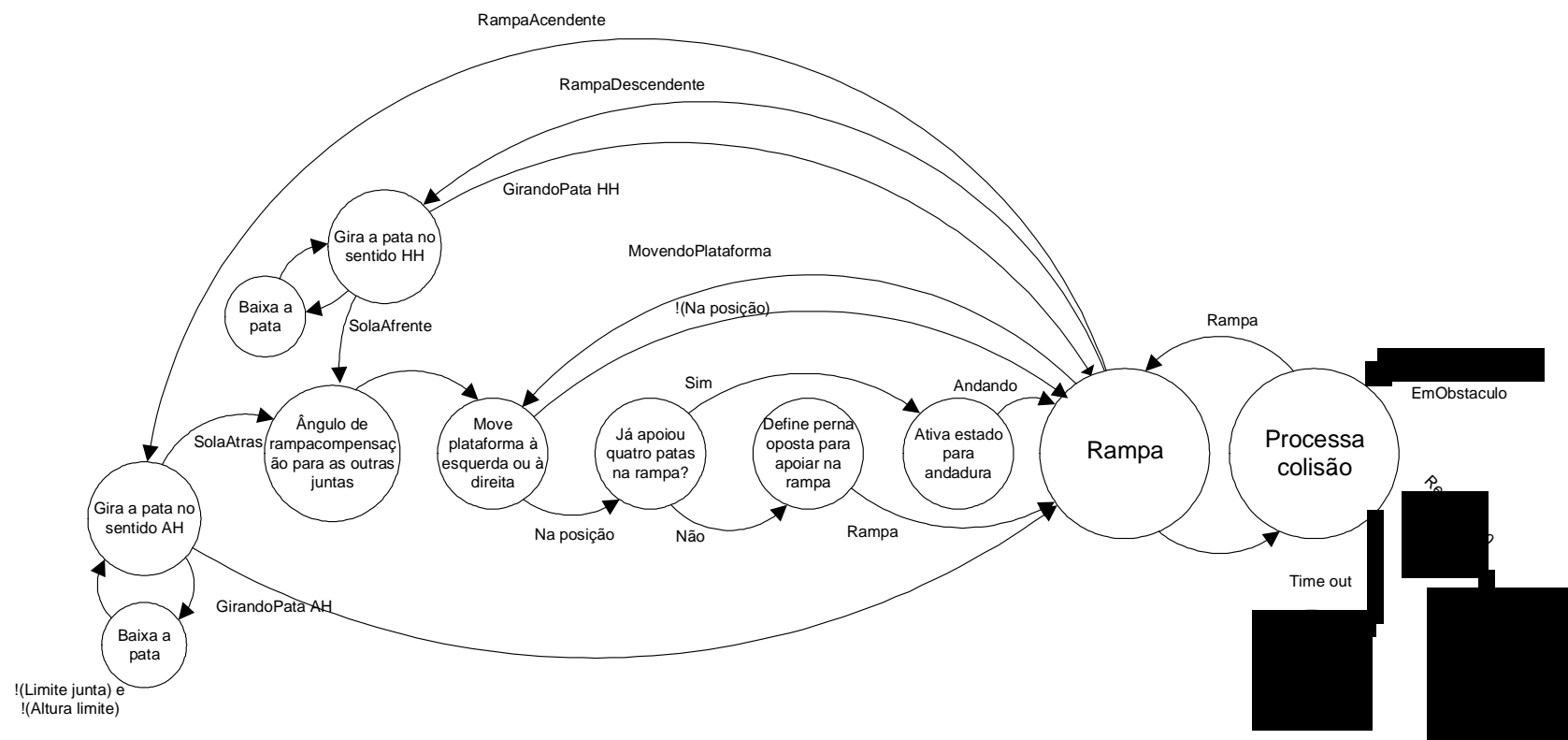


Figura 5.2-7: Diagrama de estratégia de rampa ascendente ou descendente.

Com as patas dianteiras apoiadas na rampa, o processo continua até que as quatro patas estejam na rampa ou que os ângulos da junta 3 das pernas indiquem que o robô está apoiado em um único plano.

O ângulo de rampa de cada pata varia no percurso até que as quatro patas estejam apoiadas na rampa. Durante a transposição da rampa ascendente ou descendente, cada vez que apoiar a pata 2 e adquirir o ângulo de rampa, o modelo corrige o ângulo da junta 3 para todas as patas para aquela passada, de acordo com a coordenada de empuxo da pata. Dessa maneira, é possível manter as patas paralelas à superfície de apoio durante a andadura na rampa, com pequeno erro, já que os incrementos de empuxo são reduzidos.

Com as quatro patas apoiadas na rampa a variável associada ao processo *Andando* é ativada e *Processa colisão* corrige a altura da plataforma ao retornar o controle à andadura.

5.2.4 Degrau ascendente

O processo *Degrau ascendente* é acionado por *Processa colisão* quando a pata em vôo colide com a elevação, conforme mostrado na Figura 5.2-8, ou se já está ativado a variável funcional *Degrau ascendente*.

A Figura 5.2-5 mostra o diagrama de estratégia do processo *Degrau ascendente*, e a Tabela 5.2-4 mostra as variáveis associadas aos grupos de movimento para subir um degrau.

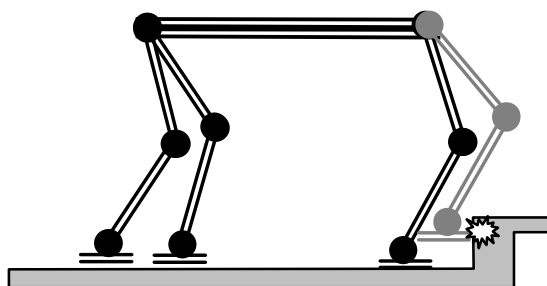


Figura 5.2-8: Obstáculo tipo degrau ascendente

A pata que estava em vôo e colidiu é movida para trás até a liberação da chave *Dedão*, a uma distância predefinida do degrau, e baixada até uma posição de apoio. Em seguida, as patas são equalizadas pelo processo *Equalizando patas*, descrito no item 4.3.3.3, em função da posição

de apoio da pata que colidiu. A plataforma é posicionada na posição extrema à esquerda e a pata à direita vai ser apoiada no piso em cima do degrau.

Em caso de sucesso de apoiar a pata à direita no piso em cima do degrau, a plataforma é movida para a posição extrema à direita e a pata à esquerda é apoiada no piso em cima do degrau.

Tabela 5.2-4: Variáveis associadas aos processos de colisão com degrau ascendente.

<i>Variável</i>	<i>Descrição da ação</i>
<i>Variável de operação do robô</i>	
DegrauAscendente	Sinaliza processamento de subida de degrau
Andando	Sinaliza processamento de andadura
Desvia	Sinaliza processamento de desvio de obstáculo
Equalizando	Sinaliza processamento de equalização de patas
<i>Variável de movimento da pata</i>	
Erguendo	Ergue a pata até um altura pré determinada
Baixando	Baixa a pata até uma posição pré determinada
MovendoAtras	Deslocando na horizontal para trás
MovendoAfrente	Deslocando a pata erguida na horizontal para frente
<i>Variável de movimento da plataforma</i>	
MovendoPlataforma	Deslocamento horizontal da plataforma

Com as patas dianteiras apoiadas no piso em cima do degrau, o processo retorna o controle para a variável ***Andando*** e o mesmo processo, desde a detecção do obstáculo até o apoio no piso mais elevado, é executado para as patas traseiras *1* e *3*.

Com as quatro patas apoiadas no piso em cima do degrau, a variável associada ao processo ***Andando*** é ativada e *Processa colisão* corrige a altura da plataforma ao retornar o controle à andadura.

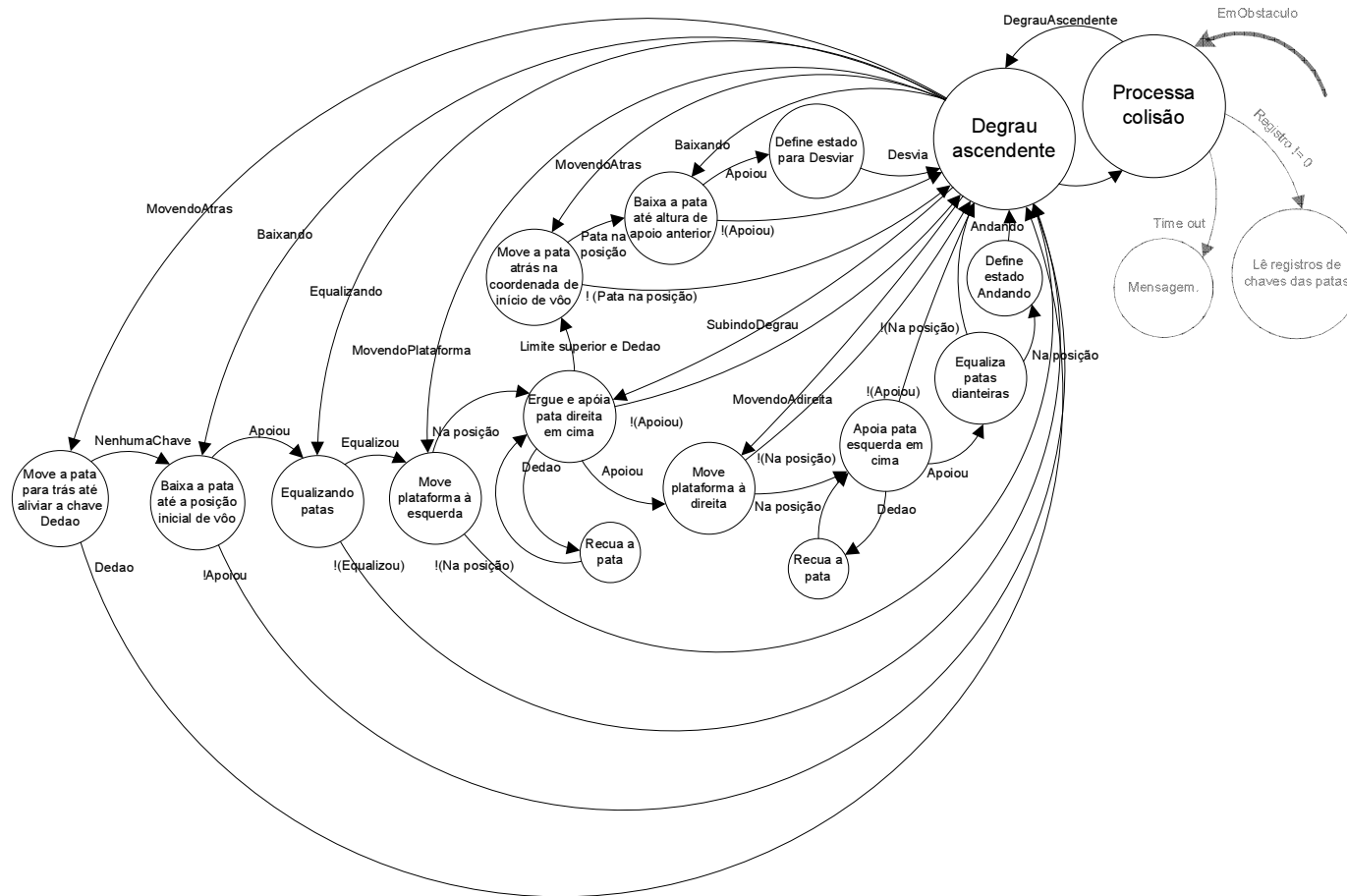


Figura 5.2-9: Diagrama de estratégia de subir um degrau.

5.2.5 Degrau estreito

O processo *Degrau estreito* é acionado por *Processo colisão* se o robô perceber um *Degrau descendente* durante a transposição de um *Degrau ascendente*, situação em que ao iniciar a andadura com as duas patas apoiadas no degrau em cima, não haja o toque com a superfície ao nível esperado, conforme mostrado na Figura 5.2-10, ou se já estiver ativada a variável associada ao processo *DegrauEstreito*.

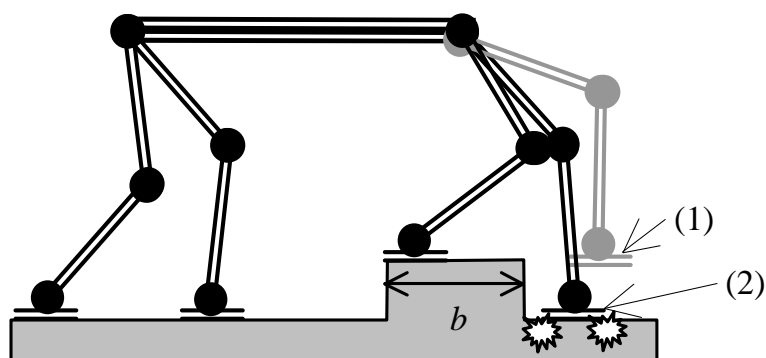


Figura 5.2-10: Obstáculo tipo degrau estreito.

O degrau estreito deverá ter largura b mínima igual ao comprimento da pata e largura máxima $b = \phi_{ij} * E + W$, sendo ϕ_{ij} o ângulo de fase entre duas patas do mesmo lado, E o empuxo total e W a distância entre os pontos de fixação das pernas, conforme mostrado na Figura 3.8-2..

A Figura 5.2-11 mostra o diagrama de estratégia do processo *Degrau estreito*, e a Tabela 5.2-5 mostra as variáveis associadas aos grupos de movimento para descer um degrau.

O *Degrau estreito* é detectado quando, após apoiar as patas dianteiras no piso em cima do degrau no processo *Degrau ascendente*, a variável associada ao processo *Andando* é ativada, o processamento passa para execução da andadura, e a pata dianteira 2 voa, mas não apóia no momento previsto pela variável *MomentoCrítico*, ativada ao final do vôo pela matriz da andadura.

Os primeiros movimentos são com a plataforma na extremidade lateral esquerda e um trecho em empuxo com as quatro patas apoiadas, seguido do vôo da pata 3, mais um trecho em

empuxo com as quatro patas apoiadas, e o vôo da pata 2 que não apóia no final da trajetória, conforme mostrado na Figura 5.2-10 no detalhe (1).

Tabela 5.2-5: Variáveis associados aos processos de transpor um degrau estreito.

Variável	Descrição da ação
<i>Variáveis de operação do robô</i>	
DegrauEstreito	Sinaliza processamento de degrau estreito
Andando	Sinaliza processamento de andadura
Desvia	Sinaliza processamento de desvio de obstáculo
<i>Variáveis de movimento da pata</i>	
MovendoAtras	Deslocando na horizontal a pata erguida para trás
Baixando	Baixa a pata até uma posição pré determinada
<i>Variáveis de movimento da plataforma</i>	
MovendoPlataforma	Deslocamento horizontal da plataforma

A pata é baixada até obter apoio, detalhe (2), e, em caso de sucesso, a pata à direita está apoiada no piso embaixo, a plataforma é movida para a posição extrema à direita e a pata à esquerda vai ser apoiada no piso embaixo. Durante o processo de baixar, se ocorrer colisão com a chave *Calcanhar*, a pata é movida à frente simultaneamente.

Com as patas dianteiras apoiadas no piso embaixo do degrau, o processo ativa a variável associada ao processo *Andando* e o mesmo processo, desde a detecção do obstáculo até o apoio no piso do outro lado do obstáculo, é executado para as patas traseiras 1 e 3.

Com as quatro patas apoiadas na rampa a variável associada ao processo *Andando* é ativada e *Processa colisão* corrige a altura da plataforma ao retornar o controle à andadura.

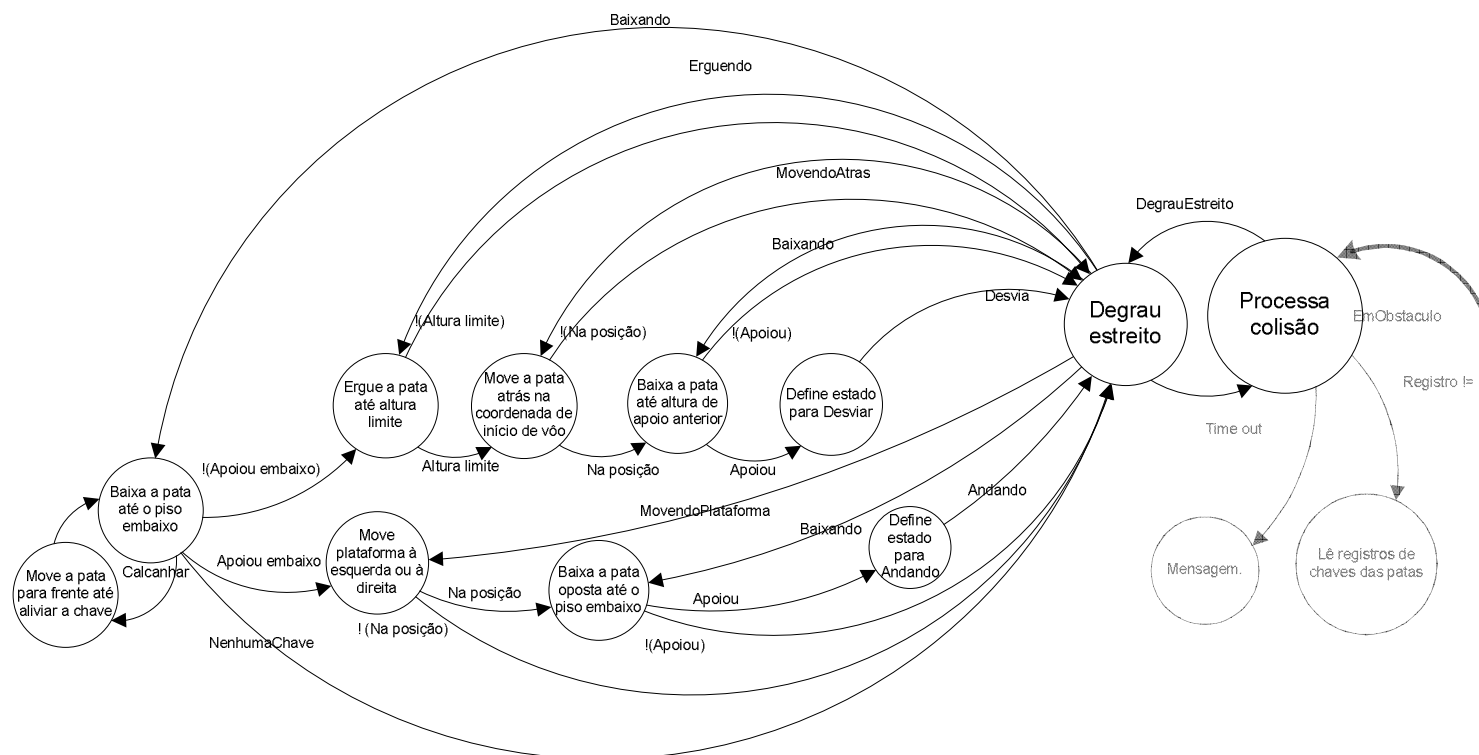


Figura 5.2-11: Diagrama de estratégia de degrau estreito.

5.2.6 Desviar

O processo *Desviar* é acionado por *Processa colisão* se algum processo de transpor obstáculo não for bem sucedido e ativar a variável associada.

A Figura 5.2-12 mostra o diagrama de estratégia do processo *Desviar*, e a Tabela 6.2.6 mostra as variáveis associados aos grupos de movimento para desviar de obstáculo.

A estratégia para desviar consiste em andar lateralmente com o robô à esquerda de uma distância pré estabelecida maior que sua largura, utilizando o grau de liberdade de rolagem da junta 0 para o empuxo lateral do robô.

A sequência padrão é feita com a plataforma baixada e as patas na origem; ela é movida à frente e as patas traseiras são apoiadas à esquerda; em seguida é movida para trás e as patas dianteiras são apoiadas à esquerda; e após movidas as quatro patas, a plataforma é movida à esquerda até a origem. A sequência é repetida até que a distância lateral desejada seja atingida.

Caso não seja possível obter uma trajetória alternativa com a andadura lateral, a direção dos graus de liberdade de rolagem da junta 0 das pernas do Guará permitem implementar um movimento de guinada da plataforma e obter uma saída em qualquer ângulo em relação ao obstáculo. É possível ainda girar o robô parado no ponto e retomar a andadura numa direção qualquer.

Tabela 5.2-6: Variáveis associadas aos processos para desviar.

Variável	Descrição da ação
<i>Variáveis de operação do robô</i>	
Desvia	Sinaliza processamento de desvio de obstáculo
Andando	Sinaliza processamento de andadura
<i>Variáveis de movimento da pata</i>	
Erguendo	Ergue a pata até uma posição pré-determinada
Movendo	Deslocando a pata erguida no plano
Baixando	Baixa a pata até uma posição pré determinada
<i>Variáveis de movimento da plataforma</i>	
MovendoPlataforma	Deslocamento horizontal da plataforma

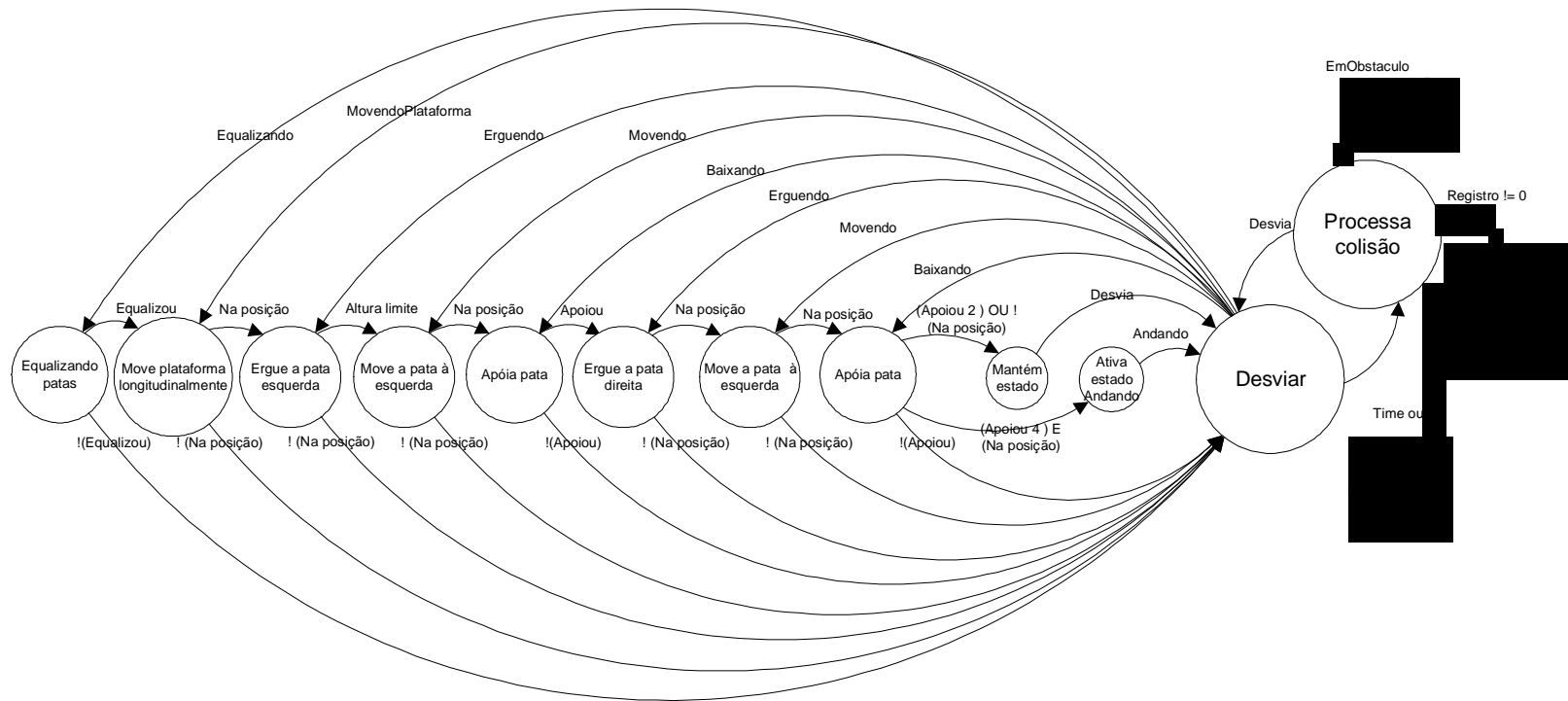


Figura 5.2-12: Diagrama de estratégia de desviar.

5.3 Discussão

Na andadura em um piso firme e plano, a cadeia de eventos é bastante previsível e permite a implementação de esquemas padronizados para andadura normal e de esquemas particularmente adaptados para as assimetrias, porque as pernas são acionadas em conjunto e o espaço de alternativas de seqüências fora do padrão tem dimensão reduzida.

Na transposição de obstáculos ocorre o contrário porque as pernas são acionadas isoladamente e o espaço de alternativas de seqüências fora do padrão tem dimensão significativa, o que exige sofisticados mecanismos de decisão para assegurar a execução de uma seqüência capaz de cumprir uma tarefa, mantendo a estabilidade do robô.

Considerando o sistema de 16 chaves de contato das patas, dirigir o robô com informações fornecidas por elas apresenta um quadro total de 2^{16} alternativas a cada *setpoint* os quais, descontados os cinematicamente incompatíveis, ainda representariam um número significativo de alternativas com um número também significativo de seqüências de saída diferentes.

Uma das maneiras de construir um formalismo para esta abordagem, é utilizar o conceito de espaço estratificado. GOLDWINE (GOLDWINE; BURDICK, 2002) apresenta um algoritmo para planejamento de sistemas robóticos cujas leis físicas impõe uma estrutura com um espaço de configuração estratificado (*stratified space*). A estratificação aparece naturalmente nas operações de robôs a pernas. Estas operações são caracterizadas pelo sistema fazer contato com o ambiente, ao apoiar a pata, e perder contato com o ambiente, ao elevar a pata em vôo para novo posicionamento. O espaço de configuração desses sistemas é estratificado em sub-espacos que correspondem a diferentes sub-espacos de contato. As equações de movimento dependem do estado do contato e são descontínuas durante a mudança de estado.

ROWE (1999) aborda o problema de um espaço de ações de grande dimensão no desenvolvimento de um sistema adaptativo autônomo de escavação para uma escavadeira hidráulica, com a lança semelhante a um robô Puma. Ela trabalha estacionada sobre o barranco, realizando movimentos de escavação, elevação e descarga na caçamba do caminhão. Devido a complexidade de modelagem da escavadeira, foi decidido não construir

um modelo dinâmico, mas sim separar as tarefas em espaços de menor dimensão. Isto pôde ser feito devido à natureza sequencial existente nas ações.

A estratégia desenvolvida no presente trabalho para a superação dos obstáculos baseia-se no paradigma da redução da dimensão do espaço de alternativas de ações, utilizando processos e grupos de movimentos padronizados para cumprir uma tarefa e também impondo ao robô posicionamentos conhecidos antes de realizar uma tarefa. Esta escolha é possível porque muitas ações podem ser agrupadas em seqüências de processos e grupos de movimentos padronizados.

Existem variáveis associadas à plataforma do robô, que ativam processos, e grupos de movimentos que alteram a margem de estabilidade, de acordo com a manobra necessária; e existem variáveis associadas às pernas que ativam seqüências de processos e grupos de movimentos que colocam a pata em uma determinada posição.

Para aumentar a capacidade e autonomia de navegação do Guará, encontra-se em desenvolvimento um sistema de identificação de obstáculos baseado em laser, objeto do projeto de doutorado de ORLANDI (2007), que fará um mapeamento 3D do ambiente à frente do robô, permitindo uma navegação local, e ultrapassando ou desviando-se dos objetos que porventura possam estar na trajetória do Guará. Este sistema, porém, não faz parte do presente trabalho.

Capítulo 6

Resultados

6.1 Introdução

Neste capítulo são apresentados e comentados os resultados dos modelos que dão suporte às operações do Guará e os resultados experimentais dos testes de andadura reta, curva e de colisão com degrau.

Os resultados gerados pelos modelos de andadura, comparados com os obtidos experimentalmente do robô, mostram boa precisão de acionamento e permitem observar as características de sincronismo e movimento na andadura reta e na curva. As seqüências fotográficas apresentadas para as andaduras reta e curva mostram detalhes de sincronismo e detalhes de movimento das pernas do robô.

A andadura em curva é mostrada em detalhes, permitindo observar as simetrias dos movimentos das patas dianteiras e traseiras e também as externas e as internas à curva. Os movimentos das patas obedecem às linhas de tangência construídas para a andadura com dirigibilidade nas quatro patas e mostram-se compatíveis com o movimento de avanço e guinada do robô.

A transposição de obstáculos é apresentada através de seqüências de fotografias, com destaque para os processos e grupos de movimentos executados nas diversas etapas do processo.

No trabalho foram apresentadas as estratégias para transpor os obstáculos *Degrau ascendente*, *Degrau descendente*, *Degrau estreito*, *Canal*, *Rampa* e para *Desviar* de obstáculos.

Na transposição do *Degrau ascendente* as principais estratégias de ação propostas são executadas e, por isso, para validar a estratégia de transposição de obstáculos proposta no

presente trabalho, são apresentadas as seqüências de movimento para a subida de um degrau no piso.

O movimento é composto de uma fase de colisão e subida das patas dianteiras; uma fase de andadura com duas patas dianteiras no piso em cima e as traseiras embaixo; colisão e subida das patas traseiras; e a andadura no piso em cima.

6.2 Robô quadrúpede Guará

A Figura 6.2-1 apresenta uma foto do robô quadrúpede Guará realizando uma curva à esquerda com raio aproximado de 1,4 m.

As pernas são fixadas em um chassi construído com perfis de alumínio de seção retangular e quadrada. As placas de microcontroladores e barramento CAN, uma para cada perna, com os circuitos de potência dos motores, são instalados na face superior da respectiva perna.

Para cada perna há um chassi interno e um externo em chapa de alumínio. O chassi externo fixa a perna no robô e o motorreductor que aciona o grau de liberdade de rolagem. O chassi interno efetua o grau de liberdade de rolagem em relação ao chassi externo e nele estão instalados os motorredutores CC de acionamento dos graus de liberdade de mergulho da perna. Os *links* das pernas e as patas são feitos em chapa de alumínio dobrada, e o acionamento dos graus de liberdade do joelho e do tornozelo é feito através de correias dentadas.

A realimentação dos ângulos de juntas é feita por potenciômetros instalados nos motorredutores das juntas, antes da redução que existe entre a saída dos motorredutores e a engrenagem de acionamento das juntas.

O microcomputador do *Nível Supervisor* é conectado aos microcontroladores Intel 80C196KB do *Nível Coordenador* através de um barramento CAN. A alimentação elétrica dos motores e do sistema de controle é fornecida por fontes e transmitida ao robô através de cabos. O acionamento dos motorredutores é feito por circuitos individuais de ponte H.

O robô se movimenta em equilíbrio estaticamente estável e é considerado um sistema mecânico de corpos rígidos, ou seja, assume-se que a deformação ocorrida nos componentes do robô durante o movimento, tem influência desprezível no seu equilíbrio estático.

Um acelerômetro instalado no CG do robô mede a inclinação e a aceleração do robô em duas direções, e fornece informações de mergulho e guinada do robô a um software de reação que roda no *Nível Coordenador* (LIMA, 2006). Este sistema não faz parte do presente trabalho.



Figura 6.2-1: Fotografia do robô quadrúpede Guará.

A Figura 6.2-2(a) e (b) mostra os detalhes construtivos das patas com as micro-chaves de contato. Cada pata dispõe de quatro micro-chaves de contato, sendo uma instalada verticalmente na parte de trás para detectar contato com um objeto na vertical; quatro instaladas na parte de baixo, duas atrás e duas à frente conectadas em paralelo, para detectar contato com o piso ou com um objeto na horizontal; e uma na parte da frente para detectar contato com um objeto na vertical.

Durante um passo normal, as chaves instaladas na parte de baixo da pata sinalizam sempre que a pata tocar o solo ou decolar para novo posicionamento. As demais sinalizam apenas quando ocorre o contato com um obstáculo vertical.

Sinalizado o contato da pata com um obstáculo, o robô pára, executa procedimentos de identificação do obstáculo e inicia o procedimento específico até a sua superação, quando então retorna à condição operacional anterior ao obstáculo, ou até a desistência, quando é acionado o processo de desvio.

A Figura 6.2-2(c) mostra as placas de controle e comunicação a bordo do robô, com o cabeamento de rede, e a placa com o acelerômetro. O robô é alimentado e recebe os sinais de controle através do chicote de cabos negro que aparece à direita nas Figura 6.2-2(c, d).

A Figura 6.2-2(d) mostra o detalhe construtivo com a articulação e o acionamento do grau de liberdade de rolagem da perna.



(a)



(b)



(c)



(d)

Figura 6.2-2: Detalhes das chaves de pata, do hardware a bordo e da perna do Guará.

6.3 Geração de trajetória e controle de andadura reta

As estratégias de controle de andadura e coordenação de movimentos foram implementadas em uma plataforma de software com paradigma de orientação a objetos, o que permitiu grande flexibilidade de programação dos processos e grupos de movimentos necessários em cada tarefa.

A Figura 6.3-1 mostra as linhas das coordenadas $Y3$ no espaço operacional, que foram calculadas pelo modelo e enviadas para o robô, e os valores recebidos após a execução. A abscissa comum são os *setpoints* enviados, e a ordenada está em milímetros; $Y3$ é a coordenada enviada pelo modelo; e $Y3R$ é a coordenada calculada pela cinemática direta a partir dos valores dos potenciômetros das juntas recebidos do robô. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

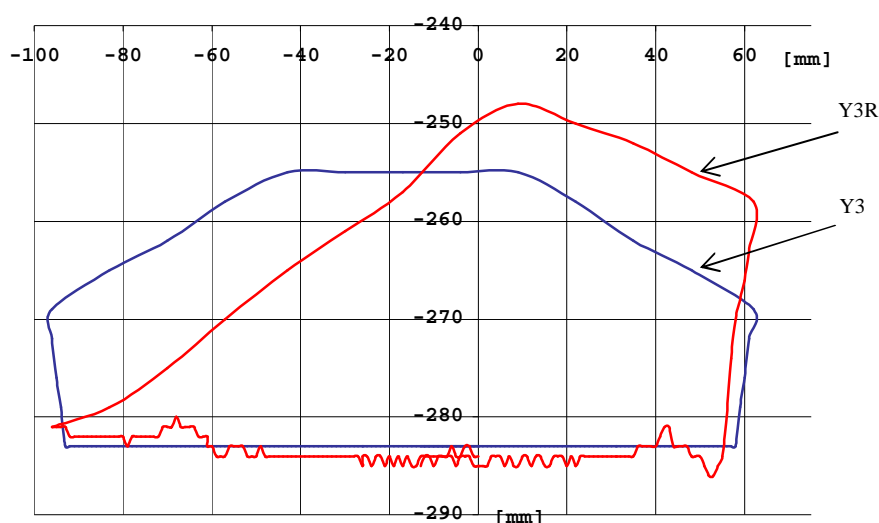


Figura 6.3-1: Coordenadas geradas e recebidas do robô, junta 3, perna 0.

Observa-se um erro muito grande na trajetória aérea da pata devido ao número de pontos utilizado para interpolar a trajetória em vôo, neste caso 20, ser muito menor do que o que é utilizado para interpolar a trajetória em empuxo da pata, neste caso 108 pontos. Para melhorar a resposta do controlador nessa região, utilizando a mesma andadura, seria necessário aumentar o número total de pontos. Isto, porém, não foi feito pois, acarretaria a redução da velocidade final do robô.

Foi verificado também que a trajetória aérea não influi na estabilidade e não impõe cargas dinâmicas significativas ao robô. A resposta mostrou-se plenamente satisfatória para a execução da trajetória da andadura nos testes experimentais.

A trajetória em empuxo, em contrapartida, possui maior número de pontos, 108 neste caso, e apresenta um comportamento bem mais suave durante o período de empuxo e sustentação da plataforma, o que assegura uma andadura suave e estável.

A Figura 6.3-2 mostra as linhas com os valores das coordenadas no espaço de juntas, os quais foram calculados pelo modelo e enviados para o robô, e os valores recebidos após a execução. A abscissa comum são os *setpoints* enviados, e a ordenada está em unidades de conversão A/D, sendo θ_0 , θ_1 , θ_2 e θ_3 , coordenadas dos potenciômetros enviadas pelo modelo; e θ_{R0} , θ_{R1} , θ_{R2} e θ_{R3} , coordenadas dos potenciômetros das juntas recebidas do robô. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

Observa-se um erro médio muito pequeno para a resposta de θ_0 , θ_1 , θ_2 e θ_3 , respectivamente, incluindo o erro devido à descontinuidade que ocorre nos picos que sinalizam a inversão de direção do movimento no final da excursão lateral da plataforma, e nas inversões de movimento na direção longitudinal Z, no início e no final da trajetória aérea, e também devido às folgas que se invertem nesses pontos e são incorporadas ao movimento.

As linhas são praticamente coincidentes, mostrando a precisão do modelo e do acionamento. O movimento do robô resultou suave e estável com a andadura executada.

A Figura 6.3-3 mostra as linhas com os valores das coordenadas no espaço operacional, os quais foram calculados pelo modelo e enviados para o robô. São mostrados também os valores recebidos após a execução, sendo as coordenadas recebidas mapeadas no espaço operacional pela cinemática direta a partir dos valores lidos dos potenciômetros das juntas. A abscissa comum são os *setpoints* enviados e a ordenada está em milímetros; X_0 , Y_0 , e Z_0 são as coordenadas enviadas pelo modelo; X_{R0} , Y_{R0} , e Z_{R0} são as coordenadas recebidas. A andadura foi gerada com 128 pontos, sendo 108 em empuxo e 20 em vôo.

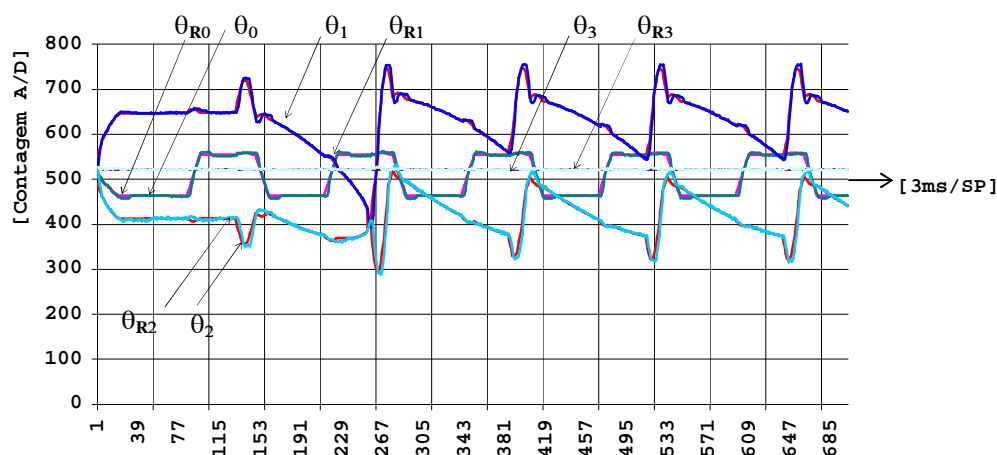


Figura 6.3-2: Coordenadas de juntas da pata 0 enviadas e recebidas do robô.

Pode-se observar, mesmo considerando a precisão de leitura e o mapeamento inverso das coordenadas de potenciômetros para coordenadas de ângulo e destas para coordenadas operacionais pela cinemática direta, um erro médio baixo para a resposta de $X3$, $Y3$ e $Z3$, considerados os fatores mencionados anteriormente.

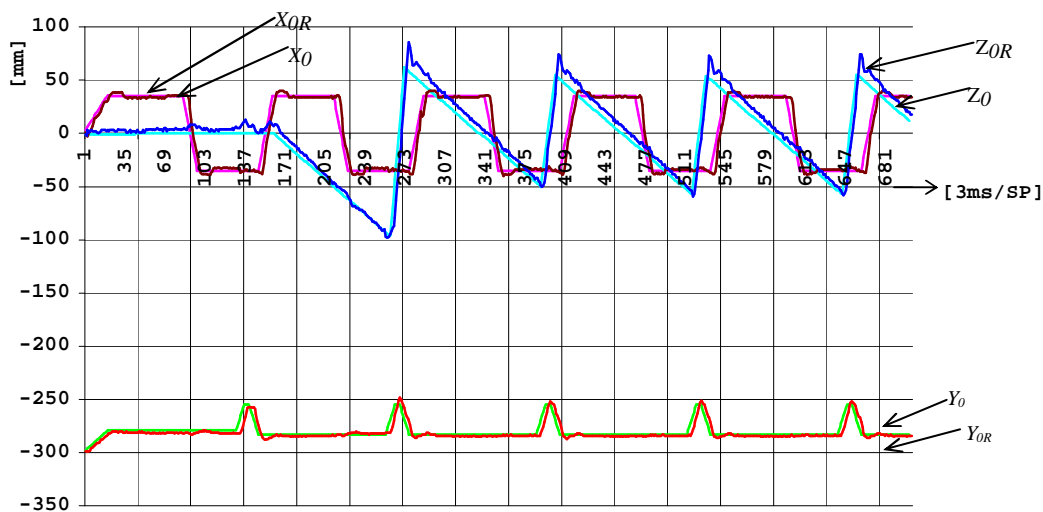


Figura 6.3-3: Coordenadas geradas e recebidas do robô, junta 3 da pata 0 no espaço operacional.

O erro maior observado em $Z3$ ao final do percurso em voo da pata está associado principalmente ao *overshoot* devido à alta velocidade de deslocamento da pata livremente em

vôo e às folgas nas juntas 1, 2 e 3 incorporadas a cada inversão no sentido de rotação das juntas.

6.3.1 Processos e grupos de movimento de andadura reta

Na Figura 6.3-4(a, b, c e d) são mostradas as patas do robô executando uma passada da andadura descrita no item 2.3. Os detalhes mostram as patas próximas ao pouso após o vôo para novo posicionamento. Na Figura 6.3-4 (a) e (b), a plataforma está deslocada na excursão lateral máxima à esquerda para vôo das patas 3 e 2 e na Figura 6.3-4 (c) e (d), à direita para vôo das patas 1 e 0.

Os movimentos de empuxo e vôo são realizados com amplitude reduzida até que seja completada a primeira passada, a partir da qual as patas já estarão ajustadas em seus respectivos ângulos de fase.

Na Figura 6.3-5 são mostradas as patas do robô após a primeira passada da andadura ter sido completada. Pode-se observar em relação ao quadro equivalente da Figura 6.3-4 que as patas têm avanço bem maior em relação às suas origens. Após a primeira passada da andadura, os movimentos de empuxo e vôo já são realizados com amplitude máxima e com as patas ajustadas em seus respectivos ângulos de fase.

A Figura 6.3-6 (a, b) mostra a configuração de apoio com duas patas do mesmo lado, no instante de maior proximidade onde a pata traseira está finalizando sua trajetória de vôo para novo posicionamento e a pata dianteira está na sua excursão máxima de empuxo.

A andadura em reta do Guará baseada em um padrão matricial que define os períodos de empuxo e vôo de cada pata, de acordo com o ângulo de fase e com o movimento lateral sincronizado para vôo de duas patas de um mesmo lado, resultou em um movimento estável e suave.

Os momentos de pouso e vôo são sinalizados pela mudança de estado das chaves de contato das patas e possibilitaram a implementação de procedimentos de sincronismo para ajustar os efeitos do balanço, das deformações elásticas e das folgas, que provocam atraso na decolagem e antecipação no pouso da pata.

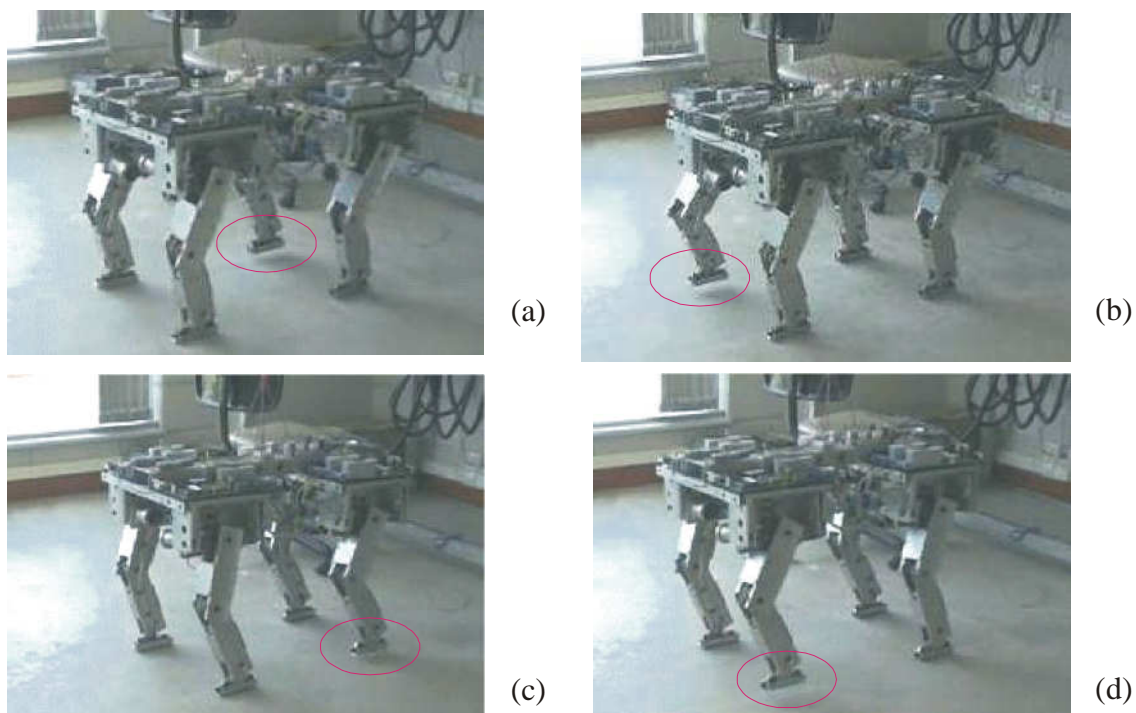


Figura 6.3-4: Andadura mostrando o vôo das patas na primeira passada da andadura.

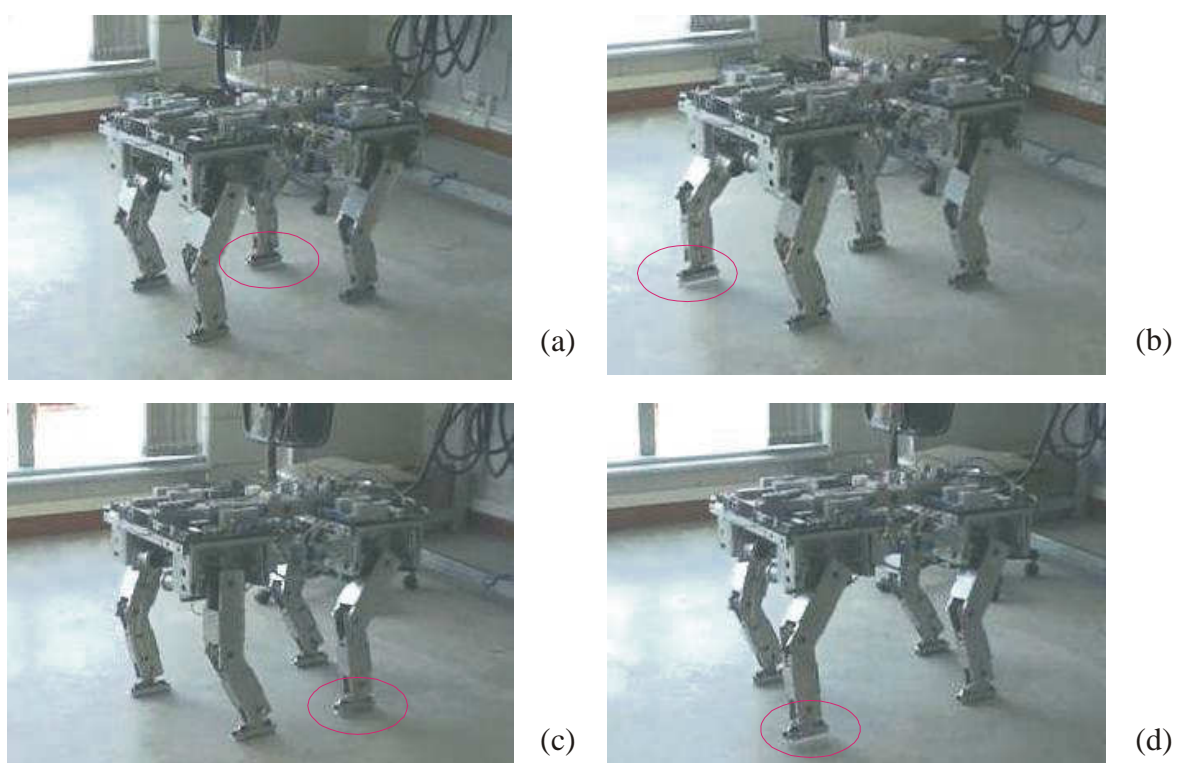


Figura 6.3-5: Andadura mostrando o pouso das patas após a primeira passada da andadura.

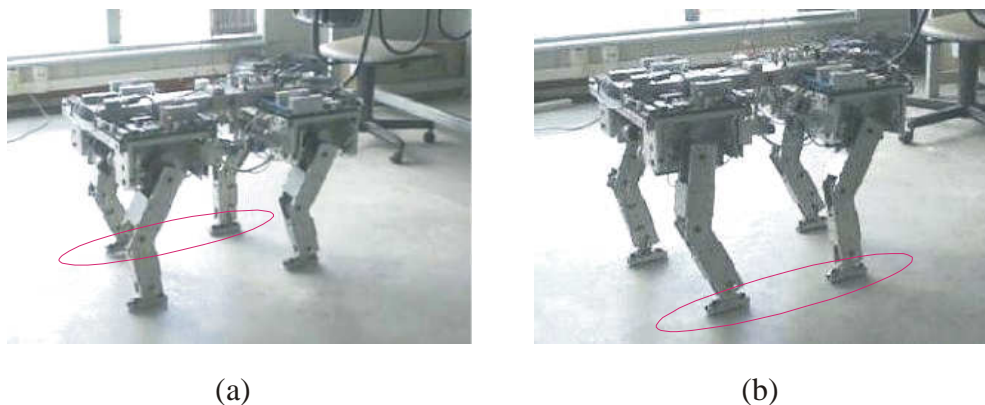


Figura 6.3-6: Andadura mostrando as patas de um mesmo lado no instante de maior proximidade.

6.4 Andadura em curva

Para caminhar com o robô em curva foi desenvolvida uma geometria com dirigibilidade nas quatro patas, conforme mostrado na Figura 2.6-2, com as trajetórias instantâneas das patas tangentes a duas circunferências concêntricas no centro de curvatura, as quais contém os pontos de apoio das patas. A trajetória do robô em curva é executada movendo a pata na direção da tangente à circunferência de curva respectiva, em pequenos segmentos de reta.

A estratégia adotada para o Guará realizar trajetória em curva é bastante simplificada, em comparação com a estratégia proposta por BAI (BAI; LOW; ZIELINSKA, 1999), porque utiliza a mesma sequência fixa da andadura em linha reta, evitando o esforço de processamento dos algoritmos de andadura livre.

A trajetória do robô em curva é gerada em arcos de circunferência, definindo-se o raio da curva com o robô em movimento, através da *Interface para andar* mostrada na Figura 2.6-1. Os demais parâmetros são determinados pelo modelo, conforme descrito no item 3.7 e mostrados nas Figura 2.6-2 e Figura 3.7-1.

O modelo para a trajetória em curva gera os incrementos tangenciais e radiais no espaço operacional das patas, correspondentes à trajetória curva no plano, estruturando-as na mesma forma matricial da andadura reta. O modelo da andadura para a trajetória em linha reta gera as coordenadas de maneira incremental no espaço operacional, estruturando-as na forma

matricial. A soma dos dois vetores resulta no vetor de coordenadas da pata no espaço operacional, em relação ao referencial da perna.

Durante os testes em curvas, foi observado que é necessário um número maior de pontos para decompor a trajetória da pata, para que o robô realize uma curva suave e sem descontinuidades entre dois *setpoints*. Com poucos pontos, os segmentos de reta resultam mais longos, e são observadas deformações nas pernas, com escorregamento posterior na decolagem, e conseqüente descontinuidade no movimento do robô. Os números observados nos testes foram de 96 pontos, com trajetória pouco suave e 128 pontos, com trajetória mais suave e estável para o robô.

Foi observado também que, devido às suas características construtivas, que limitam a excursão das juntas de rolagem das pernas, para realizar curvas com raios menores que 1,2 m com o Guará, é necessário reduzir o empuxo máximo disponível para evitar que a junta 0 das pernas atinja seu limite cinemático. No caso de a trajetória gerada no espaço operacional resultar em ângulos de junta fora do limite cinemático, o movimento ficará limitado ao limite cinemático pelo filtro de envio, o que poderá provocar desequilíbrio do robô.

A Figura 6.4-1 mostra as coordenadas de juntas da pata 0 enviadas e recebidas do robô na andadura em curva, e a trajetória das patas com o Guará realizando uma curva de raio médio aproximado de 1,4 m.

Observa-se um erro médio muito pequeno para a resposta de θ_0 , θ_1 , e θ_2 , pois as linhas de dados enviados e recebidos correspondentes são praticamente coincidentes.

Os picos observados nas linhas de θ_0 , θ_1 e θ_2 , ocorrem no trecho final da trajetória e correspondem aos trechos de vôo da pata. Os picos ocorrem nos momentos de vôo da pata, o que provoca uma variação brusca dos ângulos de juntas.

Na Figura 6.4-2 pode-se observar, a partir do *setpoint* 301, uma das características do movimento em curva onde duas patas dianteiras têm excursão lateral em um mesmo sentido e tração diferencial entre a pata externa 2 e a interna 0. A pata externa à curva tem um deslocamento maior que a interna, o qual é proporcional à razão entre os raios de curvatura externo e interno.

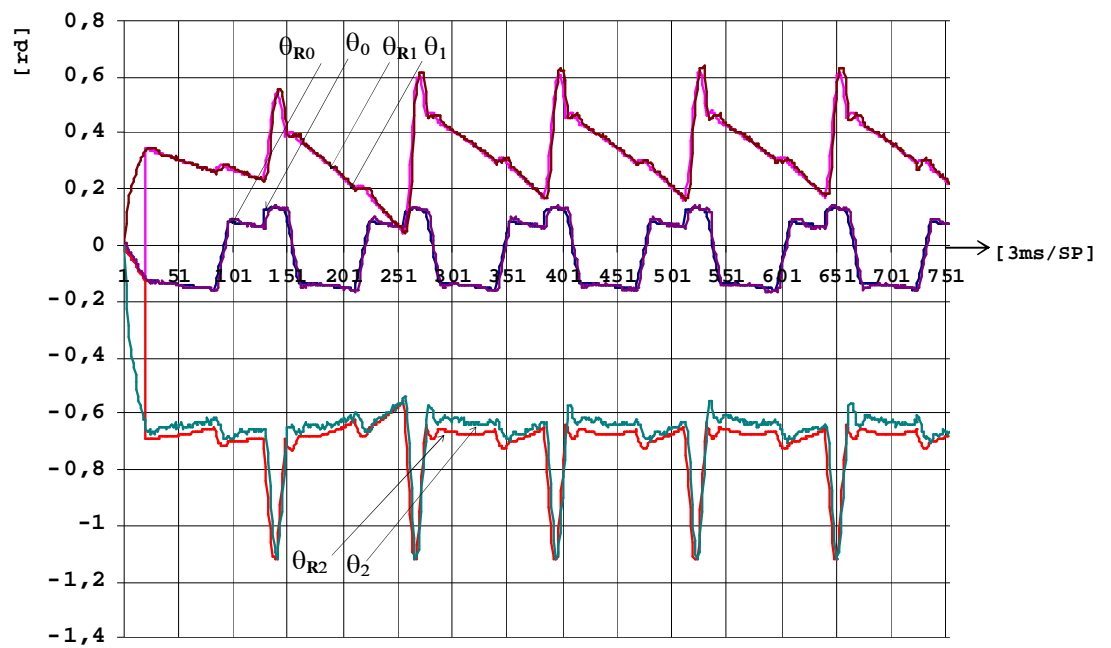


Figura 6.4-1: Coordenadas geradas e recebidas do robô na andadura em curva, junta 3 da pata 0.

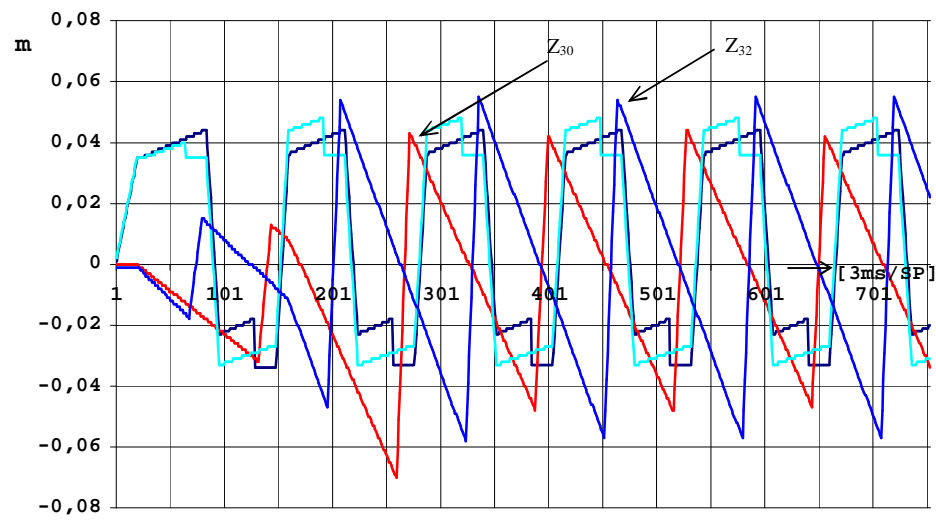


Figura 6.4-2: Coordenadas geradas em curva, junta 3 das pernas 0 e 2 no espaço operacional.

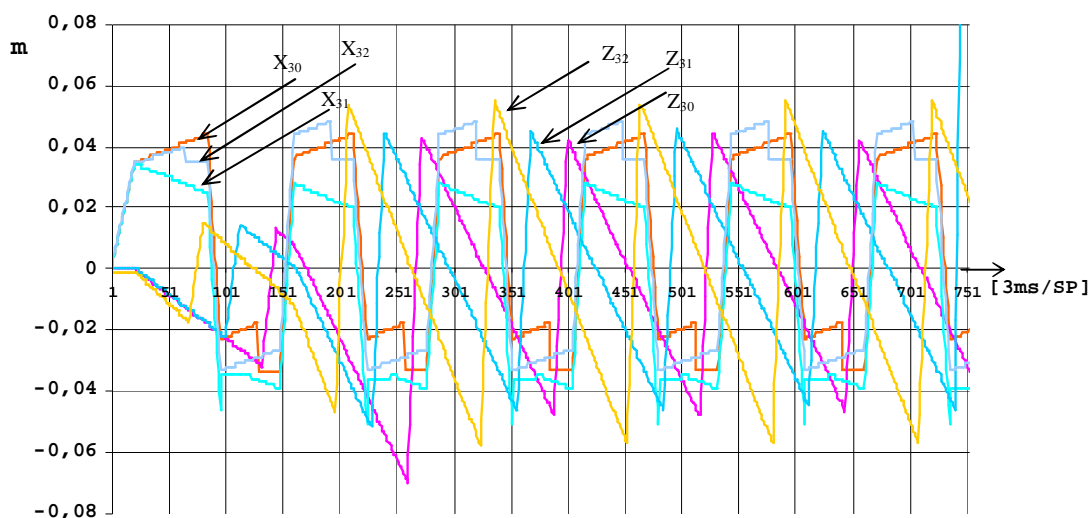


Figura 6.4-3 Coordenadas geradas em curva, junta 3 das pernas 2 e 3 no espaço operacional.

Na Figura 6.4-3, pode-se observar outra das características do movimento em curva, onde as patas 0 e 1, do mesmo lado, uma à frente e outra atrás, têm excursão lateral em um sentidos opostos em relação ao centro da curva, conforme mostrado pelas curvas de X_{30} e X_{31} , e a mesma tração entre a pata dianteira 0 e a traseira 1, conforme mostrado pelas curvas de Z_{30} e Z_{31} . Elas têm o mesmo deslocamento mas, como as trajetórias são tangentes em pontos diferentes da circunferência, elas são cruzadas, uma para dentro e outra para fora da curva.

As patas 0 e 2, ambas à frente mas em lados opostos, têm excursão lateral no mesmo sentido, em relação ao centro da curva, conforme mostrado pelas curvas de X_{30} e X_{32} , e a tração diferencial entre a pata 0, interna à curva, e a pata 2, externa à curva, conforme mostrado pelas curvas de Z_{30} e Z_{32} . Elas têm o deslocamento diferencial proporcional à razão entre os raios interno e externo, e, como as trajetórias são tangentes em diferentes circunferências, elas são em ângulo, ambas para dentro da curva.

Em ambas as figuras, observam-se descontinuidades que ocorrem nas linhas de X_3 as quais, conforme já mencionado, ocorrem nos trechos de vôo da pata correspondente.

6.4.1 Processos e grupos de movimento de andadura curva

Na Figura 6.4-4(a, b, c, d, e, f) são mostrados os movimentos executados pelas patas do robô em uma andadura curva onde o robô parte em uma trajetória reta, realiza uma curva e retorna à trajetória reta.

A curva é à direita e a pata 2 interna à curva é mostrada nos detalhes da Figura 6.4-4(a, b). É possível observar que ela terminará o vôo e iniciará o empuxo deslocada radialmente para dentro do centro de curvatura, em relação ao referencial da perna.

O percurso em empuxo terminará com a pata deslocada da mesma distância para fora da curva, após descrever uma linha poligonal, formada pelos segmentos de reta tangentes à circunferência de curvatura do robô, se o raio de curvatura se mantiver fixo.

Na Figura 6.4-4(c), os detalhes mostram as pernas 0 e 1 e é possível observar a pata 1 traseira posicionada para dentro da curva em relação à dianteira, e a plataforma já bastante guinada em relação ao ponto de partida, Figura 6.4-4(a).

Na Figura 6.4-4(e) os detalhes mostram a pata 0 dianteira deslocada radialmente em relação à pata 1 traseira como resultado da simetria de direção do deslocamento radial da pata dianteira em relação à traseira.

Na sequência Figura 6.4-4(c, d, e, f), pode-se observar que a plataforma vai sendo guinada no sentido horário e que a trajetória das patas dianteiras começa fechada e termina aberta em relação à curva e às patas traseiras.

Os sistemas de referência das pernas são paralelos, sendo que os eixos de rolagem das juntas 0 são dirigidos para a frente do robô, e os ângulos de junta são positivos, de acordo com a regra da mão direita.

Para simplificar a necessidade de acionar cooperativamente as pernas, optou-se por construir a andadura em curva do robô, acionando as quatro patas, e mover a plataforma em pequenos avanços, o que permite que não seja violada a compatibilidade cinemática do movimento, e confere dirigibilidade nas quatro patas ao robô.

O movimento resultante é de avanço e guinada do robô, seguindo de forma incremental a trajetória da circunferência definida pelo vetor posição entre o centro de curvatura e o CG do robô.

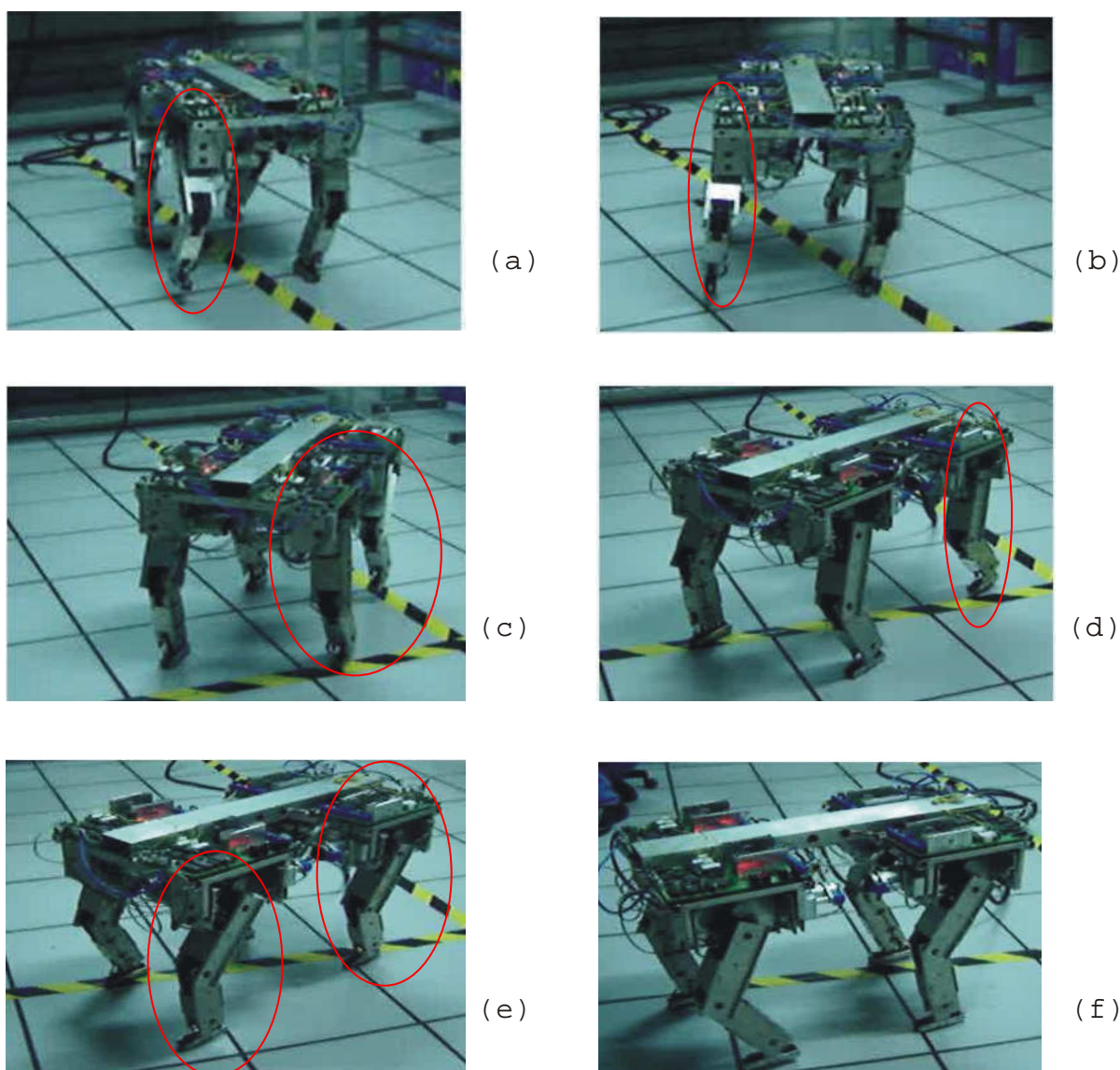


Figura 6.4-4: Sequência de andadura em curva, raio médio de 1,40 m.

6.5 Processos e grupos de movimento de subida de um degrau no piso

No teste experimental para o Guará subir um degrau, uma plataforma metálica é colocada à sua frente. A plataforma tem altura menor do que a máxima altura que a pata pode ser elevada, devido aos limites cinemáticos das juntas das pernas.

O robô está caminhando quando uma pata dianteira colide com a plataforma e é iniciado o processo para mover o robô apoiando-o com as quatro patas no piso em cima da plataforma.

Os resultados aqui apresentados são seqüências de fotografias agrupadas de acordo com o movimento ou o processo que elas representam. Cada etapa é também associada ao diagrama de estratégia para a subida de um degrau no piso, descrita na Figura 5.2-9.

O degrau encontra-se em local indefinido e só será percebido pelo robô no momento do toque da pata, o que provocará uma mudança de estado na chave **Dedão**.

A Figura 6.5-1 mostra as fotos com estágios dos movimentos de andadura do robô, partindo da posição já ajustada para iniciar a andadura, com o vôo da pata 3, Figura 6.5-1(a).

A próxima pata a voar é a pata 2, que inicia o vôo e colide com o degrau, Figura 6.5-1(b).

No momento da colisão é executado o procedimento *Lê registros de chaves das patas* do diagrama de estratégia, Figura 4.3-2, que detecta que o registro de colisão da perna 2 está com o *bit* de novo dado recebido ligado e lê o registro de colisão da perna, identificando o obstáculo. O degrau é reconhecido e as variáveis de processamento **EmObstaculo** e de operação **SubindoDegrau** são ativadas e direcionam o processamento para a seqüência de tarefas para subir o degrau.

A Figura 6.5-1(b) mostra o momento após a pata 2 colidir com o degrau, em um estágio em que ela está se afastando do degrau, e a Figura 6.5-1(c) mostra a pata sendo apoiada, já afastada do degrau. Estes movimentos correspondem aos processos *Move*, que move a pata para trás até aliviar a chave **Dedão**, e *Baixa*, que baixa a pata até a posição inicial de vôo do diagrama de estratégia, Figura 5.2-9.

A variável de operação da pata é **BaixandoAposColisao** e as variáveis de movimento são **MovendoAtras** e **Baixando**, as quais movem a pata desde o toque no degrau até o piso.

Após a Figura 6.5-1(c) a pata 2 é apoiada e o controle passa para o tratamento do obstáculo, que é iniciado com a equalização das patas, conforme mostrado na Figura 6.5-2, e *Equalizando patas* do diagrama de estratégia, Figura 5.2-9.

A plataforma do robô é deslocada para a esquerda e para frente, para a equalização da pata 3, Figura 6.5-2 (a); depois é deslocada para a direita e para trás, para a equalização da pata 0, Figura 6.5-2 (b); e finalmente é deslocada para frente para equalização da pata 1, Figura 6.5-2 (c).

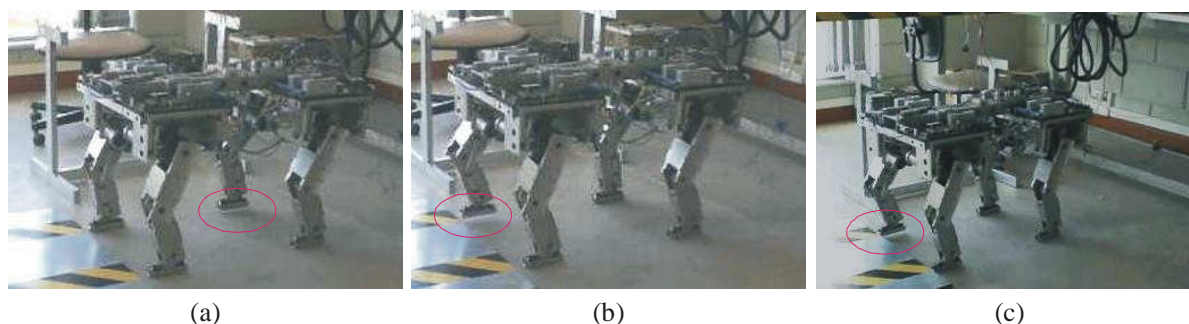


Figura 6.5-1: Andadura com colisão mostrando o vôo das patas 3 e 2 na primeira passada.

A variável de operação da plataforma é *MovendoPlataforma*, mostrada na Tabela 5.2-4, e as variáveis de movimento são *MovendoAfrente*, *MovendoAdireita*, *MovendoAtras* e *MovendoAesquerda*, respectivamente. As variáveis de movimento da plataforma usam funções básicas de movimento de patas da biblioteca de cinemática e encapsulam seus próprios grupos de movimentos. Para a equalização de cada pata, as variáveis de movimento das patas são *Erguendo*, *DeslocandoAfrente* ou *DeslocandoAtras* e *Baixando*. Ao final da equalização a plataforma é deslocada para a extrema excursão à esquerda, conforme mostrado no diagrama de estratégia.

Na Figura 6.5-3(a, b) são mostrados os movimentos para apoiar a pata 2 no piso em cima do degrau. A variável de operação da pata é *SubindoDegrau* e a sequência de variáveis de movimentos da pata é *Erguendo*, *DeslocandoAfrente* e *Baixando*. Os eventos delimitadores são a altura limite e a coordenada de apoio à frente, que foi definida quando a pata colidiu com o degrau.

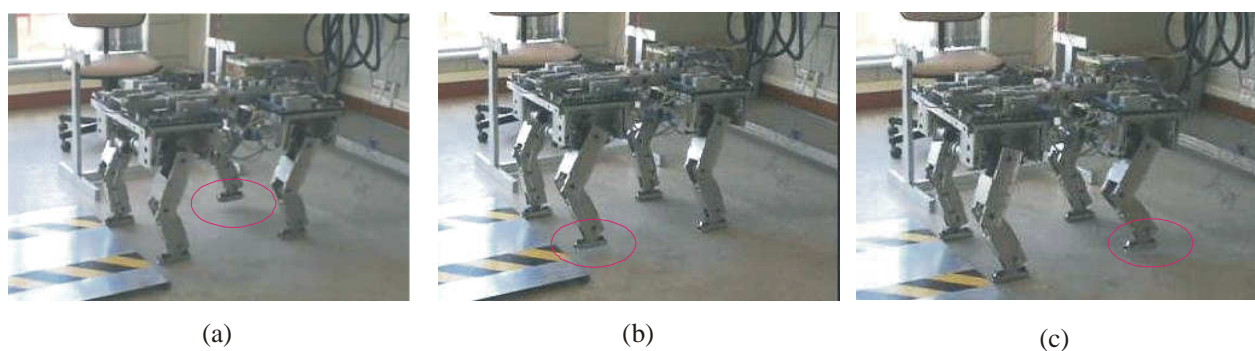


Figura 6.5-2: Equalização das patas 3, 1 e 0 preparando para subir degrau.



(a)



(b)



(c)



(d)

Figura 6.5-3: Sequência de movimentos para apoiar as patas 2 e 0 no degrau em cima.



(a)



(b)



(c)



(d)

Figura 6.5-4: Equalização após das patas apoio das dianteiras no piso em cima do degrau.

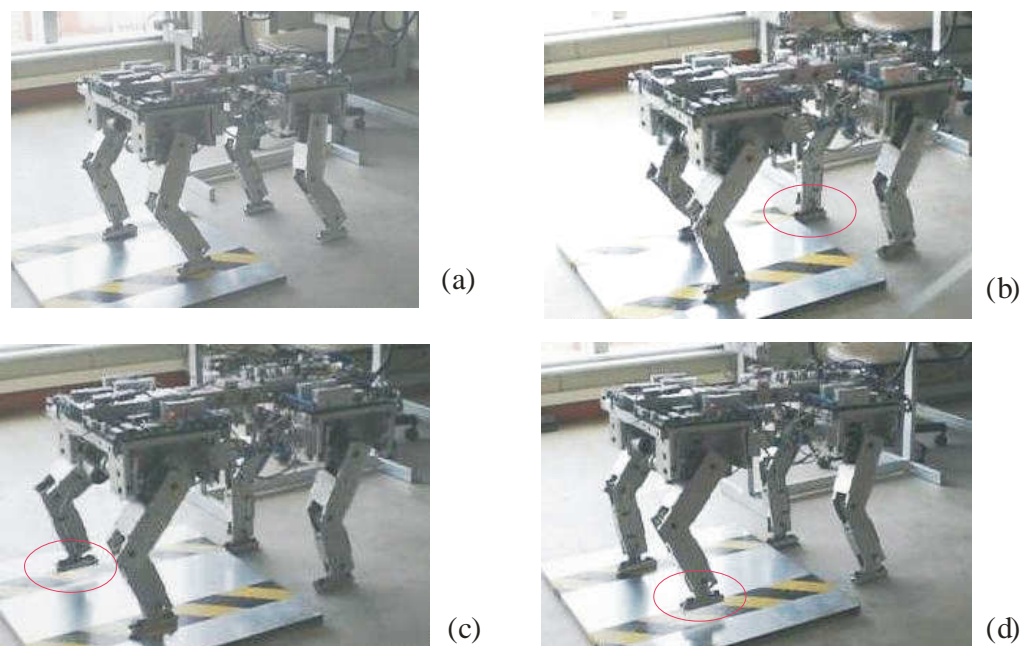


Figura 6.5-5: Andadura mostrando a colisão da pata 3 e a equalização das patas dianteiras.

Para garantir que o robô pode transpor o obstáculo sem que as juntas das pernas atinjam o limite cinemático, a pata é erguida até a altura limite e movida para frente. Se ocorrer uma colisão com a chave **Dedao**, o robô não poderá subir o degrau e acionará o processo **Desviar**.

Após apoiar a pata 2, a plataforma é deslocada para a excursão lateral máxima à direita, conforme mostrado no diagrama de estratégia, para apoiar a pata 0, com a mesma sequência de variáveis e eventos que apoiaram a pata 2.

A Figura 6.5-3 (c) e (d) mostra os movimentos para apoiar a pata 0. O processo é *Ergue e apóia pata direita em cima* conforme mostrado no diagrama de estratégia, Figura 5.2-9. Se não ocorrer colisão com a chave **Dedão**, o processo termina com a pata apoiada no piso em cima do degrau. Caso contrário, é acionado o processo **Desviar**.

Após o apoio das patas dianteiras no degrau em cima, as patas são equalizadas, conforme mostrado na Figura 6.5-4 e no diagrama de estratégia Figura 5.2-9 no processo *Equaliza patas dianteiras*.

Esta operação é executada para posicionar as patas relativamente à pata 2, na posição de melhor margem de estabilidade, para realizar os próximos movimentos do robô.

Após a equalização das patas, com as dianteiras apoiadas no degrau em cima, o processo ativa a variável associada ao processo *Andando*, o controle retorna para a andadura e o robô anda com as patas dianteiras no piso em cima do degrau e com as traseiras no piso embaixo, até ocorrer a colisão de uma pata traseira com o piso elevado, quando novamente o processo retorna o controle para *EmObstaculo*.

A Figura 6.5-5 (a) mostra o robô executando dois dos estágios da andadura com as patas dianteiras apoiadas em cima do degrau, até a colisão da pata 3, cujo detalhe é mostrado na Figura 6.5-5 (b). Na Figura 6.5-5 (c, d) são mostrados os movimentos para equalizar as patas 2 e 0, após a colisão da pata 3.

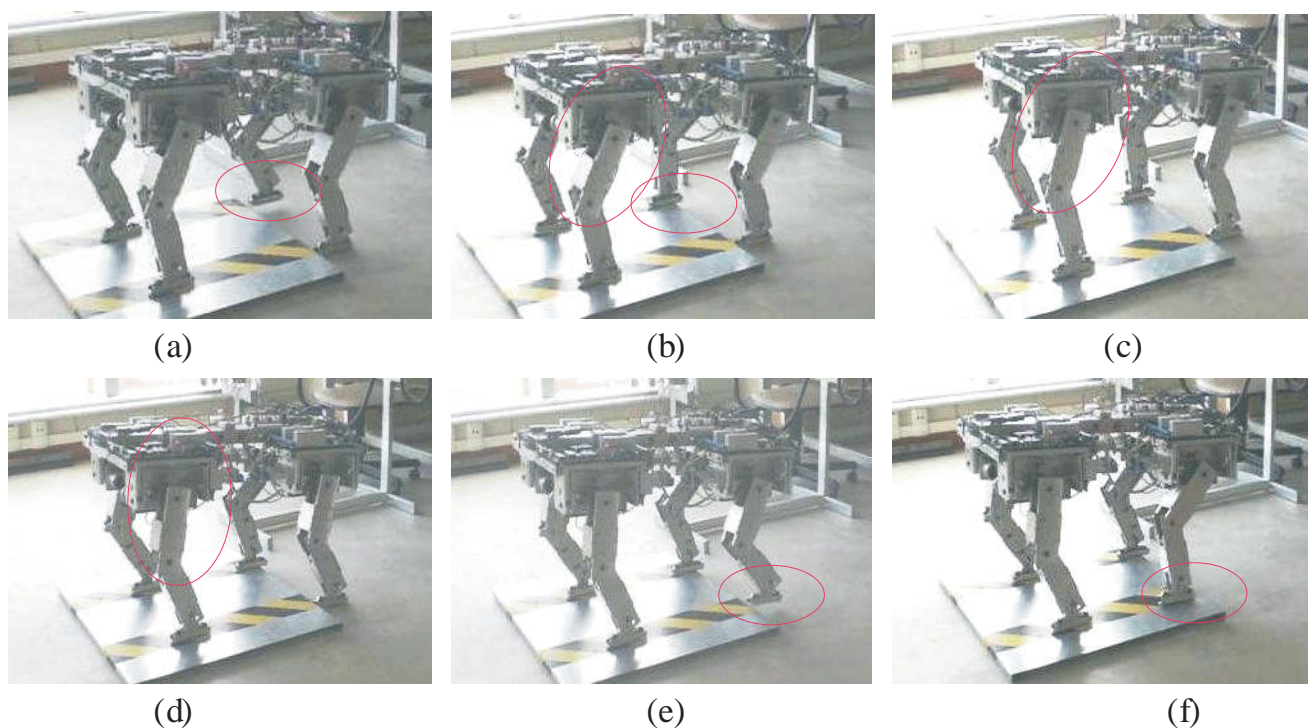


Figura 6.5-6: Apoiando as patas 3 e 1 no piso em cima do degrau.

As patas dianteiras 2, Figura 6.5-5 (c), e 0, Figura 6.5-5 (d), são equalizadas na posição relativa à pata 3 que colidiu, conforme detalhe da Figura 6.5-5 (b), para obter uma posição de equilíbrio para erguer e apoiar a pata 3 no piso em cima do degrau. O processo de equalização das patas dianteiras é o mesmo já descrito anteriormente e independe do número de patas a serem equalizadas. No diagrama de estratégia, Figura 5.2-9, o processo se repete, iniciando em *Move a pata para trás até aliviar a chave Dedao*.

Após a equalização, a plataforma é deslocada para a excursão máxima à esquerda para a subida da pata 3 e depois é deslocada para a excursão máxima à direita e para a frente para a subida da pata 1. O processo segue o diagrama de estratégia na mesma sequência definida para as patas dianteiras.

Na Figura 6.5-6(a, b) são mostrados os movimentos para apoiar a pata 3 no piso em cima do degrau, com a plataforma na posição de excursão lateral máxima à esquerda. A variável de operação da pata é *SubindoDegrau* e as variáveis de movimento são *Erguendo*, *MovendoAfrente* e *Baixando*, respectivamente, conforme descrito anteriormente.

A Figura 6.5-6 (b, c, d), mostra no detalhe da perna 0 dianteira esquerda, que a plataforma está sendo deslocada à frente para obter melhor posição de equilíbrio para erguer e apoiar a pata 1, Figura 6.5-6 (e, f).

A variável de operação da plataforma é *MovendoAfrente*. Para apoiar as patas no piso em cima do degrau, as variáveis de movimento das patas são *Erguendo*, *MovendoAfrente* e *Baixando*, mostrados na Tabela 5.2-4, conforme descritos anteriormente.

Após executar o processo *Apóia pata esquerda em cima* e apoiar a pata esquerda 1 o processo *Equaliza patas dianteiras* é executado e a variável associada ao processo *Andando* é ativada, conforme mostrado no diagrama de estratégia, Figura 5.2-9.

6.6 Discussão

A andadura reta do Guará foi desenvolvida com base em três princípios: a geração incremental das coordenadas no espaço operacional, que após o mapeamento no espaço de juntas e conversão para a contagem dos conversores A/D já são os *setpoints* enviados para o robô; a definição em forma matricial das andaduras, que possibilita um nível de padronização do modelo; e a geração do movimento lateral, vinculando-o à ordem do *setpoint* em execução.

A geração incremental permite que o número de setpoints seja escolhido de acordo com necessidade de precisão na geração da trajetória da pata e com a estabilidade do movimento do robô. A única restrição ao número de pontos é que ele seja um múltiplo inteiro do número de colunas da matriz da andadura. A restrição quanto à estabilidade está associada,

principalmente, ao número de *setpoints* com quatro patas em empuxo quando é feita a movimentação transversal da plataforma de uma extremidade da excursão lateral à outra.

A definição em forma matricial das andaduras é feita em uma interface gráfica desenvolvida para a construção da matriz de andadura, o que permitiu que diversas configurações fossem testadas, variando-se o número de pontos em empuxo e vôo, bem como a distribuição dos trechos de empuxo e vôo da matriz.

A geração do movimento lateral vinculado à ordem do *setpoint* permite que ela seja feita fora do processamento da andadura e seja padronizada em função dos endereços necessários para identificar a ordem da andadura sendo executada, o estágio, que é a coluna da matriz de andadura, e o *setpoint* dentro do estágio em curso. Os dados da andadura são gravados em um arquivo e as variáveis e os processos estão encapsulados em uma classe no programa.

Além disso, o vetor de coordenadas do movimento lateral é adicionado ao vetor de coordenadas da andadura no espaço operacional da perna, não interferindo nos algoritmos de andadura reta e curva.

O modelo é interdependente, porém, o número de *setpoints* na geração incremental e o número de pontos em empuxo e vôo da andadura são independentes, o que permite utilizar o modelo como ferramenta de desenvolvimento de novas andaduras para robôs quadrúpedes, que caminham em equilíbrio estaticamente estável.

O Guará é um robô quadrúpede que caminha apoiado sempre em três ou quatro patas, e seu movimento é baseado no acionamento cooperativo das pernas, ou seja, os movimentos das patas devem ser cinematicamente compatíveis. Na andadura em reta esta é uma restrição naturalmente satisfeita porque as trajetórias das patas são retas paralelas ao eixo longitudinal e formam o plano de apoio do robô.

Na andadura em curva, ao acionar qualquer controle de direção, o sistema constrói os vetores posição com base no raio de curvatura do CG e determina o centro instantâneo da curvatura, que poderá permanecer fixo por um ou mais *setpoints*.

Definidos o centro e o raio de curvatura do CG, as trajetórias das patas são linhas tangentes às circunferências com centro no centro de curvatura e raio definido pela posição da origem do sistema de referência da perna.

A cada *setpoint* pode haver um novo raio, um novo vetor posição, um novo centro de curvatura e uma nova tangente unindo o ponto de apoio anterior ao atual.

O empuxo máximo é atribuído às patas externas à curva para assegurar a não superação do limite cinemático das pernas. A plataforma é um corpo rígido e o incremento de deslocamento angular em relação ao centro de curvatura a cada *setpoint* pode ser definido em função do empuxo e do raio de curvatura da trajetória instantânea das patas externas à curva.

Conforme já mencionado, a andadura em um piso firme e plano apresenta uma cadeia de eventos bastante previsível e o espaço de alternativas fora do padrão tem dimensão reduzida.

A estratégia desenvolvida neste trabalho para a superação dos obstáculos, cujo espaço de alternativas de seqüências tem dimensão elevada, baseia-se na redução da dimensão utilizando processos e grupos de movimentos padronizados.

Os processos e grupos de movimento utilizados para realizar a subida e a descida de um degrau no piso foram implementados com movimentos padronizados e em um espaço de alternativas de seqüências de menor dimensão que asseguram a execução da tarefa.

Alguns processos e movimentos intermediários impostos ao robô provavelmente aumentam o tempo de execução da tarefa, apesar de não ser conhecido o tempo de implementação e de execução de modelos mais sofisticados para realizar as mesmas tarefas, mas assegura que as tarefas sejam executadas com maior estabilidade do robô.

Capítulo 7

Conclusões e Sugestões

Este capítulo apresenta as conclusões gerais deste trabalho e as sugestões para trabalhos futuros. Algumas conclusões sobre partes do sistema implementado podem ser encontradas nas seções *Discussões* dos capítulos correspondentes.

O principal objetivo deste trabalho foi desenvolver, implementar e testar um esquema de detecção local de obstáculos em um robô de quatro patas, utilizando apenas informações de sensores de contato instalados nas patas para perceber o obstáculo e gerenciar uma sequência autônoma de movimentos complexos integrados para posicionar o robô e transpor o obstáculo, ou desviar-se dele, preservando o equilíbrio e a estabilidade do robô.

A maneira utilizada para abordar o problema no presente trabalho foi a separação das ações em espaços de dimensões menores, aproveitando as características de sequência que os movimentos apresentam.

Apesar de não ser conhecido o tempo de implementação e de execução de modelos mais sofisticados para realizar as mesmas tarefas, esta estratégia provavelmente torna a transposição mais lenta, mas garante uma estabilidade maior para o robô.

Os resultados experimentais apresentados para a subida de um degrau elevado no piso, mostram a viabilidade da metodologia proposta, como estratégia complementar, para ser utilizada na navegação autônoma de robôs móveis a pernas.

Através de movimentos sequenciados pelos eventos dos sensores de contato, como se o robô tateasse cegamente, foi possível aproximar-se e detectar um obstáculo tipo degrau ascendente, posicionar o robô de forma a garantir seu equilíbrio durante as manobras e subir o degrau.

Os resultados obtidos mostraram que a estratégia proposta é uma alternativa para a abordagem de aplicações com contornos muito livres, onde a máquina dispõe de muitas alternativas enquanto interage com o ambiente, e a implementação de um modelo fechado

não é possível ou viável. A estratégia proposta pode, por exemplo, ser utilizada na abordagem de ações mais complexas com um robô a pernas, tais como subir ou descer uma escada.

O robô Guará, um quadrúpede com quatro graus de liberdade por perna, foi desenvolvido, fabricado e construído para o presente trabalho, e já está sendo utilizado como plataforma para pesquisa e desenvolvimento na área de robótica móvel a pernas na UFES.

A construção das pernas do Guará, com um grau de liberdade de rolagem formando uma junta esférica no ponto de fixação da perna à plataforma, foi motivada pelo estudo apresentado por MACMAHON (1984), que faz uma análise do andar humano decompondo-o em 6 movimentos característicos que foram definidos em 1953 por SAUDERS, INMAM E EBERHART. Essa configuração tornou possível realizar, durante a andadura, um dos movimentos efetuados pelos mamíferos, que é o deslocamento lateral do CG durante a andadura. No Guará, esse movimento desloca o CG para dentro do polígono de apoio, o que resulta em maior estabilidade para o robô durante a andadura.

Os movimentos das andaduras reta e curva são gerados de forma incremental e o movimento final é composto através da soma vetorial dos componentes no espaço operacional. Com esta estrutura o empuxo pode ser variado tanto através do número de pontos da trajetória como através dos incrementos de coordenadas utilizado. Além disso, é possível variar a velocidade do robô através de interface amigável na tela de *Interface para andar* do programa do *Nível Supervisor*.

O programa de controle do *Nível Supervisor* foi desenvolvido sob o paradigma de *Programação Orientada à Objetos (POO)*, com a distribuição de funções, interfaces e estruturas de dados em classes que agrupam elementos comuns.

Devido às necessidades de simulação de andaduras e ao fato de o programa ter sido implementado paralelamente ao projeto e construção do Guará, foram construídas interfaces para a construção de andaduras, acionamento do robô, movimentação direta das pernas, calibração dos ângulos de juntas, diagnóstico de comunicação, estado de chaves e cadastro de dados operacionais do robô. Assim atualmente é possível a movimentação do robô com as patas em contato com o piso e o acionamento em plataforma para simulação, verificação e ajuste de sincronismo.

A interface que permite movimentar individualmente as pernas incorpora uma facilidade importante que é o ajuste semi-automático dos zeros dos potenciômetros, com o auxílio de um

prumo, e a gravação em arquivo do valor lido. O arquivo de zeros é carregado sempre que o programa de controle do *Nível Supervisor* é iniciado.

A realização de trajetória em reta e em curva em um robô quadrúpede com dirigibilidade nas quatro patas, requer a movimentação cooperativa de quatro manipuladores resultando nos movimentos de empuxo e guinada da plataforma.

A utilização de segmentos de reta no espaço de coordenadas, durante a geração da andadura reta normal e em sincronismo com o movimento de deslocamento lateral da plataforma, possibilita a realização de curvas com o mesmo critério de estabilidade da andadura reta. Com isso os valores de deslocamento radial e tangencial para a realização de curvas são adicionados vetorialmente aos valores gerados na andadura reta, antes do mapeamento das coordenadas do espaço operacional no espaço de juntas.

Esta abordagem eliminou uma camada de geração da trajetória que seria executada para o CG do robô em relação ao sistema de referência de bordo e mapeada para os sistemas de referência das pernas, através dos vetores posição dos pontos de fixação das pernas à plataforma e da matriz de rotação instantânea. A geração da trajetória de cada pata está vinculada à movimentação da plataforma através do esquema de dirigibilidade, mas é gerada no espaço operacional da pata, não demandando nenhuma operação de rotação de matrizes e reduzindo o esforço de processamento.

Considerando o volume de processamento necessário para gerar trajetórias para as quatro patas no espaço operacional e mapeá-las no espaço de juntas, uma estratégia de andadura cuja saída pode ser gerada ao nível do acionamento dos graus de liberdade do robô, permite um ganho de desempenho para o controlador do robô, sendo uma contribuição importante do presente trabalho.

Uma opção de trabalho futuro muito importante é a integração do trabalho de LIMA (2006) com o *Nível Supervisor* do Guará que, através de um algoritmo de controle Fuzzy e um acelerômetro instalado a bordo, o habilitará para reagir a imprevistos que possam colocar em risco o seu equilíbrio.

É possível também, transferir alguns processos utilizados na estratégia atual que rodam no *Nível Supervisor* para o *Nível Coordenador*, formando uma biblioteca de funções que poderiam agilizar a pesquisa e construção de novas alternativas para ampliar a capacidade de navegação autônoma do Guará.

Outra opção de trabalho futuro é a integração do trabalho de ORLANDI (2007), um sistema de identificação de obstáculos baseado em laser que gera um mapeamento 3D do ambiente à frente do robô, o que permitirá expandir a capacidade de navegação autônoma fornecendo informações prévias da presença, posição, forma e dimensões de obstáculos na trajetória do robô.

ANEXOS

A1. Dados técnicos do Guar

Tabela A 1: Dimenses e peso.

Comprimento entre pernas	415 [mm]
Largura entre pernas	305 [mm]
Link superior da perna	150 [mm]
Link inferior da perna	150 [mm]
Altura do tornozelo	50 [mm]
Peso total	16 [kgf]

Tabela A 2: Potencimetros

Excurso mxima:	10 voltas = 3600 graus
Tenso de alimentao:	18 V
Converso A/D :	10 bits
Relao de converso:	$(18 \text{ V}/3600) \cdot (1023/5) = 1023/1000$

Tabela A 3: Relaes de transmisso mecnica.

Junta	Engrenagem		Relao de transmisso mecnica	Relao de transmisso final
	<i>Motriz</i>	<i>Acionada</i>		
0	19	120	0,158333	0,154773
1	19	120	0,158333	0,154773
2	19	108	0,175925	0,171970
3	60	120	0,500000	0,488758

Tabela A 4: Faixa dos incrementos do controle de direção.

<i>Faixa de raio</i>	<i>Incremento</i>
$R \leq 10$	1
$10 < R \leq 100$	10
$100 < R \leq 1000$	50
$1000 < R \leq 10000$	200
$10000 < R$	1000

Tabela A 5: Limites cinemáticos das juntas.

<i>Perna</i>	<i>Junta</i>	<i>Ângulo em graus</i>		<i>Excursão dos potenciômetros</i>		
		<i>Mín</i>	<i>Máx</i>	<i>Mín</i>	<i>Zero</i>	<i>Máx</i>
0	0	-18	18	390	506	622
	1	-54	54	164	512	860
	2	-42	42	267	511	755
	3	-45	45	430	522	614
1	0	-18	18	396	512	628
	1	-54	54	158	506	854
	2	-42	42	269	513	757
	3	-45	45	422	514	606
2	0	-18	18	396	512	628
	1	-54	54	167	515	863
	2	-42	42	270	514	758
	3	-30	30	452	513	574
3	0	-18	18	393	509	625
	1	-54	54	165	513	861
	2	-42	42	269	513	757
	3	-45	45	420	512	604

A2. Programa de controle do Nível Supervisor do Guar

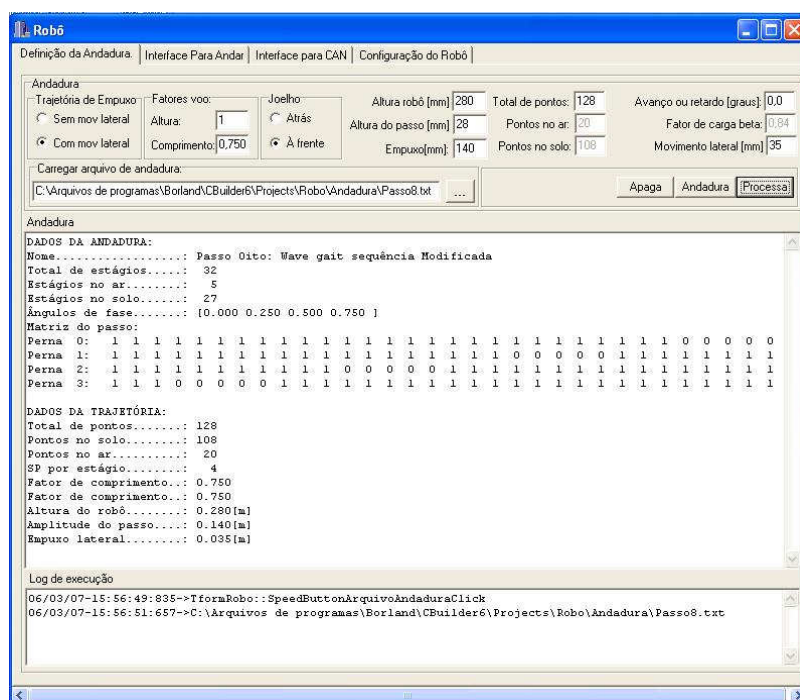


Figura A# 1: Tela inicial, de escolha de andadura e entrada de parmetros.

Atrves desta interface  definida a andadura e os parmetros que sero utilizados pelo modelo para gerar o movimento do rob.

Aps carregada a andadura  necessrio definir o nmero total de pontos da trajetria, a altura do rob, a altura mxima da pata no vo e o empuxo mximo. Os dados so verificados quanto  sua coerncia cinemtica e, em caso de incoerncia, so sugeridos outros valores. H um conjunto de valores padro que so apresentados na abertura da tela.

Na janela *Andadura* da tela, so apresentados os dados da andadura definida e os parmetros para gerao do movimento que sero utilizados pelo modelo.

Na janela *Log de execuo* so apresentadas as mensagens de *log* e de abertura de arquivo.

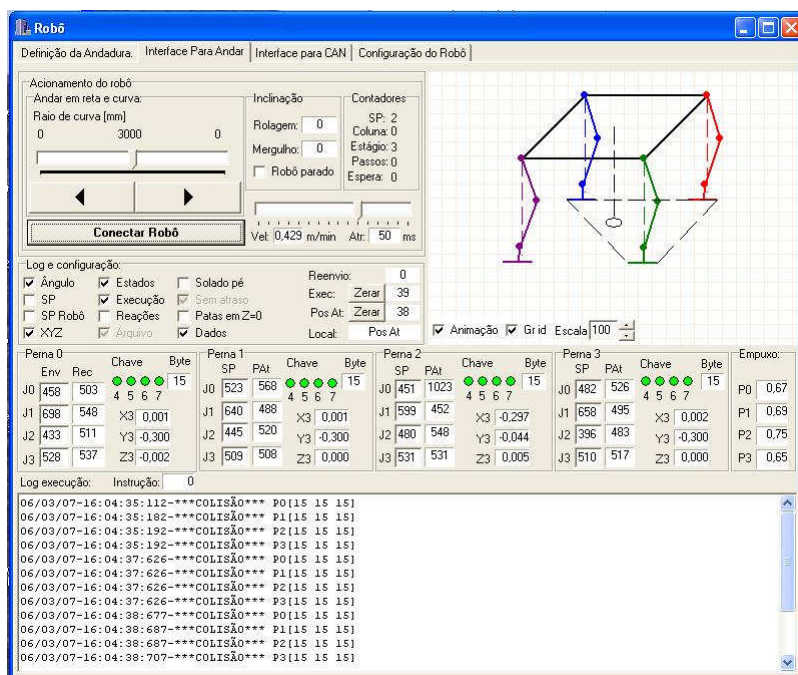


Figura A# 2: Tela de interface para andar.

Esta é a interface de comando do robô. Os controles principais são os botões *Conectar Robô*, que inicia o movimento do robô, a barra de variação de velocidade e as setas que definem o raio de curvatura instantâneo do robô. Caso não tenha sido definida a andadura e os parâmetros para o modelo, um alarme é emitido e é solicitada confirmação.

Diversas funções de arquivo são habilitadas através de *check boxes* e há contadores de comunicação e sinalizadores de instrução para diagnóstico em caso de parada.

Um conjunto de chaves de compilação permite habilitar o registro em arquivo de todas as variáveis geradas no modelo e enviadas para o robô, bem como, os valores recebidos de coordenadas de juntas e os valores de coordenadas no espaço operacional.

Há uma área de interface com o estado das chaves de cada perna, as coordenadas enviadas e recebidas dos potenciômetros e as coordenadas da junta 3 no espaço operacional e em relação ao referencial da perna.

Na janela inferior são apresentadas mensagens de diagnóstico e de depuração, que são ativadas por chaves de compilação e ou pelos *check boxes* da tela. Mensagens críticas que possam colocar em risco o robô provocam a parada do robô e são sempre apresentadas juntamente com alarme sonoro e visual e requerem confirmação para continuar o movimento.

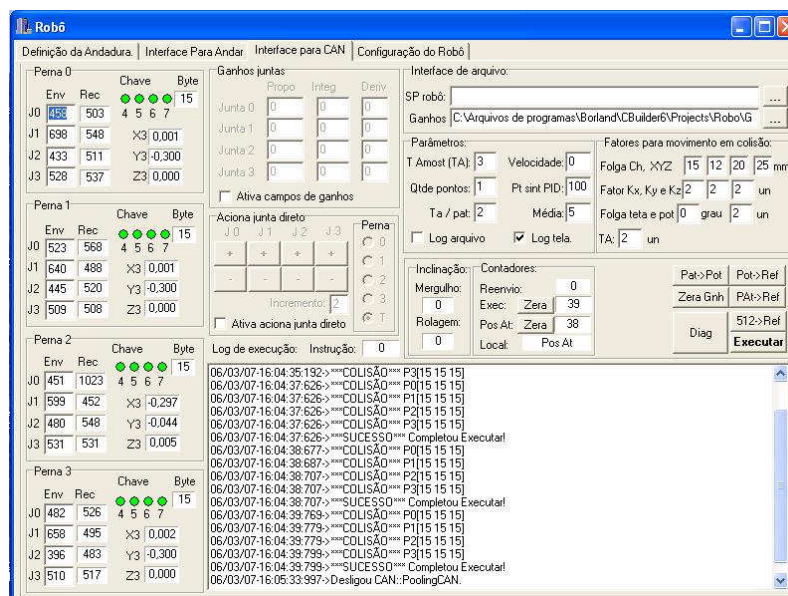


Figura A# 3: Tela de acesso direto ao comando dos graus de liberdade e diagnóstico.

Esta é a interface de comando dos graus de liberdade do robô. Os controles permitem carregar ganhos, alterar parâmetros e *setpoints*, bem como, acionar diretamente cada grau de liberdade do robô.

Esta interface permite também realizar a calibração semi-automática dos potenciômetros, utilizando um prumo para posicionar as pernas verticalmente, e um botão para gravação no arquivo de zeros lidos dos potenciômetros.

Robô

Definição da Andadura | Interface Para Andar | Interface para CAN | **Configuração do Robô**

Dimensões do Robô:
 Comprimento: 415 [mm]
 Largura: 305 [mm]
 Altura: 30 [mm]
 Raio: 300000 [mm]

Dimensões das pernas:
 Link 2: fêmur: 150 [mm]
 Link 3: tibia: 150 [mm]
 Link 4: tornozelo: 50 [mm]
 Pé [mm]: 50 [mm]

Massas discretas:
 Plataforma: [dropdown]
 Qtd: 10
 No.: [Kg] X Y Z [mm]
 0 401 402 403 404

Configuração:
 SP completo: 20 Pos: 20 Tol[mm]: 4 SP discreto: 3 Colunas:
 Escala: 750 Folia: X 90 Y 50 Z 200
 Perspectiva: Inc. XYZ: Inc. ângulo:
 45.0 [Graus] 0.002 [mm] 10 [graus]

Limites cinemáticos das pernas:

Junta	Perna 0: dianteira esquerda			Perna 1: traseira esquerda			Perna 2: Dianteira direita			Perna 3: Traseira esquerda		
	Ângulo	Potenciômetro:		Ângulo	Potenciômetro:		Ângulo	Potenciômetro:		Ângulo	Potenciômetro:	
	Min.	Zero	Máx.	Min.	Zero	Máx.	Min.	Zero	Máx.	Min.	Zero	Máx.
0	18	0	18	388	504	620	18	0	18	453	569	685
1	-54	0	54	201	549	897	-54	0	54	141	489	837
2	-42	0	42	267	511	755	-42	0	42	276	520	764
3	-45	0	45	436	528	620	-45	0	45	417	509	601

Origem dos referenciais das pernas:

Perna 0:		Perna 1:		Perna 2:		Perna 3:	
Diant esquerda	Tras esquerda	Diant esquerda	Tras esquerda	Diant direita	Tras direita	Diant esquerda	Tras esquerda
X: 225 [mm]	X: -225 [mm]	X: 225 [mm]	X: -225 [mm]	X: 225 [mm]	X: -225 [mm]	X: 225 [mm]	X: -225 [mm]
Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]	Y: 125 [mm]
Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]	Z: 0 [mm]

Engrenagens:

	M	A	I
J0	19	120	6,316
J1	19	120	6,316
J2	19	108	5,684
J3	60	120	2,000

Arquivo de dados do robô:

Salva Carrega

Log:

Figura A# 4: Tela de configuração de parâmetros do robô.

Esta é a interface de dados do robô. Todos os dados dimensionais do robô, a excursão máxima e os zeros das juntas em graus e em coordenadas dos potenciômetros, a origem dos sistemas de referência das pernas e outros parâmetros do modelo podem ser alterados nesta interface.

Figura A# 5: Tela de definição de uma nova andadura, mostrando a andadura 8.

Esta interface permite montar uma matriz de andadura a partir da especificação do número de estágios em empuxo e vôo.

O programa monta a planilha com as 4 linhas correspondentes à pata e as colunas correspondentes ao número de estágios do passo. A partir daí devem ser especificados os valores das células sendo 1 para empuxo e 0 para vôo. Os demais dados da andadura são definidos conforme a formulação definida no Capítulo 2.

Os botões da tela permitem carregar uma andadura existente e salvar uma andadura editada.

As definições geradas nesta classe são utilizadas na tela seleção de andadura, onde são inseridos os demais parâmetros.

A3. Tabela de combinações do ciclo principal de controle

A Tabela A6 é a tabela verdade com as variáveis de entrada e as ações de saída que determinam quando a sequência de processamento no ciclo principal de controle do Guará, mostrado na Figura 4.3-1, deve executar o processo *Andando* ou comutar para o processo *Colisão*.

No exame da Tabela A 1, deve-se observar que o padrão elétrico do sinal é ativo baixo, conforme mostrado na Tabela 2.3-1.

A ocorrência da linha 1 é o padrão esperado para aterrissagem da pata após o período em vôo e a ação é a aterrissagem normal da pata que consiste apenas em comutar a variável funcional de *EmEmpuxo* para *EmVoo*. A linha 0 é uma aterrissagem com um valor da chave fora do padrão esperado. Foi observado que estas ocorrências devem-se à diferenças mecânicas no comportamento das chaves, irregularidades no piso, pequena inclinação da pata em relação ao piso e a balanços do robô. Elas são tratadas como aterrissagens normal até o *setpoint* seguinte, quando normalmente ocorre nova mudança de estado das chaves e o padrão correto é recebido no *Nível Supervisor*, ou é ativada a variável *EmObstaculo*, caso contrário.

As linhas 2 a 15, são configurações que não ocorrem devido ao movimento da pata em vôo não permitir a transposição de um obstáculo e o toque da chave calcanhar na parede oposta deste obstáculo.

Tabela A 6: Combinações de chaves e variáveis da andadura e ações do ciclo de controle principal do Guará.

Entradas								Saídas				Observação
Linha	Combinação	Ch dedão	Ch sola à frente	Ch sola atrás	Ch calcanhar	Mom Crítico	Empuxo/Voo	Empuxo	Voo	Aterrissagem	Colisão	
0	18	0	1	0	0	1	0	0	0	1	0	Aterrissagem normal
1	26	0	1	1	0	1	0	0	0	1	0	Aterrissagem normal
2	12	0	0	1	1	0	0	0	0	0	1	Chave calcanhar incompatível
3	14	0	0	1	1	1	0	0	0	0	1	Chave calcanhar incompatível
4	20	0	1	0	1	0	0	0	0	0	1	Chave calcanhar incompatível
5	22	0	1	0	1	1	0	0	0	0	1	Chave calcanhar incompatível
6	28	0	1	1	1	0	0	0	0	0	1	Chave calcanhar incompatível
7	30	0	1	1	1	1	0	0	0	1	0	Chave calcanhar incompatível

Tabela A6: (Continuação).

<i>Entradas</i>								<i>Saídas</i>				<i>Observações</i>
<i>Linha</i>	<i>Combinação</i>	<i>Ch dedão</i>	<i>Ch sola à frente</i>	<i>Ch sola atrás</i>	<i>Ch calcanhar</i>	<i>Mom Crítico</i>	<i>Empuxo/Vôo</i>	<i>Empuxo</i>	<i>Vôo</i>	<i>Aterrissagem</i>	<i>Colisão</i>	
8	36	1	0	0	1	0	0	0	0	0	1	Chave calcanhar incompatível
9	38	1	0	0	1	1	0	0	0	0	1	Chave calcanhar incompatível
10	46	1	0	1	1	1	0	0	0	0	1	Chave calcanhar incompatível
11	48	1	1	0	0	0	0	0	0	0	1	Chave calcanhar incompatível
12	52	1	1	0	1	0	0	0	0	0	1	Chave calcanhar incompatível
13	54	1	1	0	1	1	0	0	0	0	1	Chave calcanhar incompatível
14	60	1	1	1	1	0	0	0	0	0	1	Chave calcanhar incompatível
15	62	1	1	1	1	1	0	0	0	0	1	Chave calcanhar incompatível
16	2	0	0	0	0	1	0	0	0	0	1	Colisão: canal ou descendo degrau
17	4	0	0	0	1	0	0	0	0	0	1	Colisão: canal ou descendo degrau
18	6	0	0	0	1	1	0	0	0	0	1	Colisão: canal ou descendo degrau
19	16	0	1	0	0	0	0	0	0	0	1	Colisão: degrau ascendente
20	32	1	0	0	0	0	0	0	0	0	1	Colisão: degrau ascendente
21	34	1	0	0	0	1	0	0	0	0	1	Colisão: degrau ascendente
22	40	1	0	1	0	0	0	0	0	0	1	Colisão: degrau ascendente
23	42	1	0	1	0	1	0	0	0	0	1	Colisão: degrau ascendente
24	44	1	0	1	1	0	0	0	0	0	1	Colisão: degrau ascendente
25	50	1	1	0	0	1	0	0	0	0	1	Colisão: degrau ascendente
26	56	1	1	1	0	0	0	0	0	0	1	Colisão: degrau ascendente
27	58	1	1	1	0	1	0	0	0	0	1	Colisão: degrau ascendente
28	10	0	0	1	0	1	0	0	0	0	1	Colisão: degrau descendente
29	8	0	0	1	0	0	0	0	0	0	1	Colisão: degrau descendente
30	24	0	1	1	0	0	0	0	0	0	1	Colisão: degrau descendente
31	1	0	0	0	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
32	5	0	0	0	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
33	9	0	0	1	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
34	13	0	0	1	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
35	17	0	1	0	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
36	21	0	1	0	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
37	25	0	1	1	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
38	29	0	1	1	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
39	33	1	0	0	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
40	37	1	0	0	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
41	41	1	0	1	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
42	45	1	0	1	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
43	49	1	1	0	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
44	53	1	1	0	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave
45	57	1	1	1	0	0	1	1	0	0	0	Empuxo, ignora mudança de chave
46	61	1	1	1	1	0	1	1	0	0	0	Empuxo, ignora mudança de chave

Tabela A6: (Continuação).

<i>Entradas</i>								<i>Saídas</i>				<i>Observações</i>
<i>Linha</i>	<i>Combinação</i>	<i>Ch dedão</i>	<i>Ch sola à frente</i>	<i>Ch sola atrás</i>	<i>Ch calcanhar</i>	<i>Mom Crítico</i>	<i>Empuxo/Vôo</i>	<i>Empuxo</i>	<i>Vôo</i>	<i>Aterrissagem</i>	<i>Colisão</i>	
47	3	0	0	0	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
48	7	0	0	0	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
49	11	0	0	1	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
50	15	0	0	1	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
51	19	0	1	0	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
52	23	0	1	0	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
53	27	0	1	1	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
54	31	0	1	1	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
55	35	1	0	0	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
56	39	1	0	0	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
57	43	1	0	1	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
58	47	1	0	1	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
59	51	1	1	0	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
60	55	1	1	0	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
61	59	1	1	1	0	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
62	63	1	1	1	1	1	1	1	0	0	0	Momento crítico não ocorre em empuxo
63	0	0	0	0	0	0	0	0	1	0	0	Vôo normal

As ocorrências das linhas 16 a 30 são capturadas pelo filtro primário de colisão no início do ciclo de controle principal que ativa a variável de processamento associada a *EmObstaculo* que reconhece o tipo de obstáculo e dispara o processamento correspondente.

As ocorrências da linha 31 até a 46, são capturadas pelo filtro primário de colisão e são apenas sinalizadas, já que a pata encontra-se em empuxo. Elas são tratadas como empuxo normal até o *setpoint* seguinte quando normalmente ocorre nova mudança de estado das chaves e o padrão correto é recebido no *Nível Supervisor*.

As ocorrências da linha 47 até a 63, não são possíveis porque só pode existir a condição da pata em momento crítico, que define o ponto de aterrissagem esperado pelo modelo, quando a mesma estiver em vôo para novo posicionamento.

A ocorrência da linha 63 refere-se ao vôo normal da pata, sinalizada por uma mudança de chave sem a ocorrência de colisão.

A4. Cinemática Direta da perna do robô

A configuração com um grau de liberdade de rolagem e um de mergulho na junta fixa ao robô, possibilitam imitar o movimento lateral do quadril dos animais bípedes e quadrúpedes, realizando o balanço que desloca o CG na direção da perna que está apoiada.

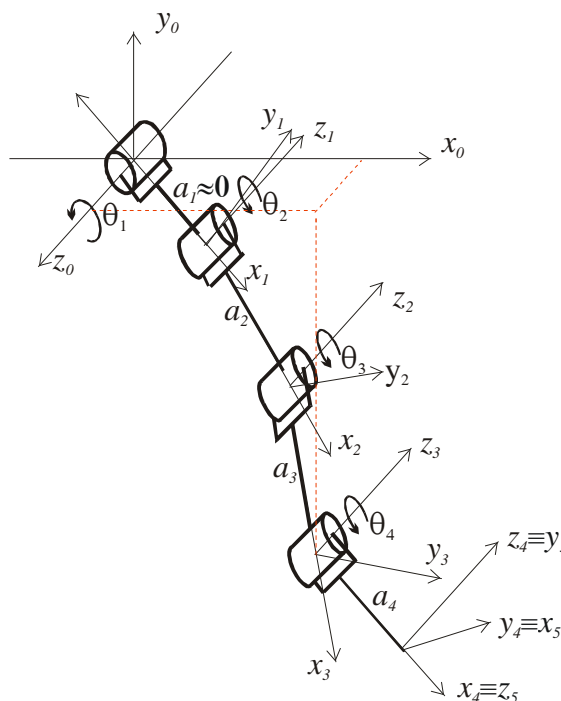


Figura A# 6: Diagrama cinemático da perna com o grau de liberdade de rolagem no quadril.

Esta configuração também permite realizar as manobras da perna com o movimento de guinada, embora com uma maior sobrecarga do motor de acionamento do grau de liberdade de rolagem: giro sobre si mesmo; curvas; e capacidade de movimentar-se paralelamente ao eixo longitudinal, e estacionar em locais bastante restritos. Utilizando-se a convenção de Denavit Hatenberg, pode-se obter as matrizes das transformações homogêneas A_i , para uma perna com 4 graus de liberdade, mostrada na Figura A# 6, conforme a seguir:

Tabela A 7: Parâmetros DH de uma perna com 1 grau de liberdade de rolagem.

<i>Segmento</i>	a_i	α_i	d_i	θ_i
1	0	$\pi/2$	d_1^*	θ_1
2	a2	0	0	θ_2
3	a3	0	0	θ_3
4	a4	0	0	θ_4
5	0	$\pi/2$	0	$-\pi/2$

$$A1 = \begin{bmatrix} c_1 & 0 & s_1 & 0 \\ s_1 & 0 & -c_1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (A4.6.)$$

$$A2 = \begin{bmatrix} c_2 & -s_2 & 0 & a_2 c_2 \\ s_2 & c_2 & 0 & a_2 s_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (A4.7.)$$

$$A3 = \begin{bmatrix} c_3 & -s_3 & 0 & a_3 c_3 \\ s_3 & c_3 & 0 & a_3 s_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad ($$

A4.8.)

$$A4 = \begin{bmatrix} c_4 & -s_4 & 0 & a_4 c_4 \\ s_4 & c_4 & 0 & a_4 s_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (A4.9.)$$

$$A5 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (A4.10.)$$

A5. Cinemática inversa da perna do robô

Um método típico para a obtenção da cinemática inversa para um manipulador com 6 graus de liberdade consiste em obter a posição do referencial do punho (*wrist*) a partir da posição e orientação desejadas do efetuador e, a seguir, utilizar métodos geométricos para as três primeiras juntas seguidas das 3 últimas. (SPONG; VIDYASAGAR, 1992) (FU; GONZALES; LEE, 1987)

Com 4 graus de liberdade, a perna é um problema de cinemática inversa não usual e o método acima não funciona porque não temos a matriz de rotação completa do efetuador. Neste caso, entretanto, é possível resolver a cinemática inversa, geometricamente, diretamente a partir da junta 1, uma vez que as três últimas juntas são paralelas entre si e perpendiculares à junta 1. BONITZ (1997), utiliza esta técnica para obter a cinemática inversa do manipulador com 4 graus de liberdade, utilizado no *MVACS (Mars Volatiles and Climate Surveyor)*, lançado pela NASA em 1999.

Em relação à Figura A# 6, aplicando-se a lei dos co-senos para o triângulo $O_1O_2O_3$, tem-se:

$$\cos\theta_3 = -\frac{x_3^2 + y_3^2 + z_3^2 - (a_2^2 + a_3^2)}{2 * a_2 * a_3} \quad (A5.5.)$$

De maneira semelhante, θ_2 é a diferença entre o ângulo formado pela reta O_1O_3 com a reta O_1AB e o ângulo formado pela reta O_1O_3 com a reta O_1O_2 , conforme a seguir:

$$\theta_2 = a \tan\left(\frac{z_3}{\sqrt{x_3^2 + y_3^2}}\right) - a \tan\left[\frac{a_3 * \sin\theta_3}{a_2 + a_3 * \cos\theta_3}\right] \quad (A5.6.)$$

A rotação θ_1 é obtida diretamente das coordenadas da extremidade da pata (x_3, y_3, z_3) conforme a seguir:

$$\theta_1 = a \tan\left(\frac{x_3}{y_3}\right) \quad (A5.7.)$$

A pata se desloca no ar sempre com θ_4 conhecido, o que permite obter a coordenada da junta 3 em relação às coordenadas de apoio da pata (x_4, y_4, z_4) conforme a seguir:

$$\begin{bmatrix} x_3 \\ y_3 \\ z_3 \end{bmatrix} = \begin{bmatrix} x_4 - a_4 * \text{sen } \theta_4 * \cos \theta_1 \\ Y_4 - a_4 * \text{sen } \theta_4 * \text{sen } \theta_1 \\ z_4 - a_4 * \text{sen } \theta_4 \end{bmatrix} \quad (\text{A5.8.})$$

Referências

- (ALEXANDER, 1984) ALEXANDER, M. N.; “The gaits of bipedal and quadrupedal animals”, The Int. Journal Robotic Research, vol. 3, no. 2, 1984;
- (ARKIN, 1993) ARKIN, R. C.; “Modeling Neural Function at the Schema Level: Implications and Results for Robotic Control”, Proceedings of the Workshop on "Locomotion Control in Legged Invertebrates" on Biological Neural Networks in Invertebrate Neuroethology and Robotics, National Academy of Sciences Study Centre, Chapter XVII, pp. 383 – 410, Massachusetts, USA, 1993;
- (BARES; WETTERGREEN, 2000) BARES, J. E.; WETTERGREEN, D. S.; “Dante II: Technical Description, Results, and Lessons Learned”, The Int. Journal Robotic Research, vol. 19, no. 28 pp. 743-761, August 2000;
- (BAI; LOW, 2000) BAI, S.; LOW, K. H.; “Terrain evaluation and its application to path planning for walking machines”, Advanced Robotics, vol. 15, no. 7 pp. 729-748, 2001;
- (BAI; LOW; ZIELINSKA, 1999) BAI, S.; LOW, K.H.; ZIELINSKA, Tereza; “Quadruped Free Gait Generation Combined with Body Trajectory Planning”, IEEE Proceedings of the First Robot Motion and Control RoMoCo 99, pp. 165-170, Kiekrz, Poland, 1999;
- (BENTO FILHO; et al., 2000) BENTO FILHO, Antônio; AMARAL, Paulo F. S., FREIRE FILHO, Teodiano B. “Verificação teórica e experimental da resposta de um controlador PID aplicado a uma perna de robô”, Anais do XIII CBA, Setembro 2000, Florianópolis, SC, Brasil;
- (BENTO FILHO; et al., 2001) BENTO FILHO, Antônio; AMARAL, Paulo F. S., PINTO, Benedito G. M. “Cinemática de Equilíbrio para um Robô Quadrúpede”, Anais do COBEM - Congresso Brasileiro de Engenharia Mecânica, Novembro 2001, Uberlândia, MG, Brasil;
- (BENTO FILHO; et al., 2004) BENTO FILHO, Antônio; AMARAL, Paulo F. S., PINTO, Benedito G. M., LIMA, Luis E. M. “Uma Metodologia para a Localização Aproximada de um Robô Quadrúpede”, Anais do XV CBA, Setembro 2004, Gramado, RS, Brasil;
- (BERKEMEYER, 1998) BERKEMEYER, Mathew D.; “ Modeling Dynamics of Quadrupedal Running”, The Int. Journal Robotic Research, vol. 19, no. 9, September, pp. 971-985, 1998;
- (BORENSTEIN; KOREN, 1989) BORENSTEIN; J.; KOREN, Y; “Real Time Obstacle Avoidance for Fast Mobile Robots”, IEE Transactions on Systems, Man and Cybernetics, Vol. 19, No. 5, pp. 1179-1187, Sept/Oct, 1989,
- (BORENSTEIN; KOREN, 1991) BORENSTEIN; J.; KOREN, Y; “Them vector field histogram – Fast Obstacle Avoidance for Mobile Robots”, IEE Transactions on Robotics and Automation, Vol. 7, No. 3, pp. 278-288, June, 1991,
- (BOSTON, 2006) “Big Dog”, disponível em www.bostondynamics.com, USA, Acesso em 2006;

- (BONITZ, 1997) BONITZ, Robert G.; “Mars Surveyor 98 Lander MVACS Robotic Arm Control System Design Concepts”, Proc. of the IEEE Int. Conf. on Robotics and Automation, pp. 2465-2470, April 1997. Albuquerque, New Mexico;
- (BURKHARD et al., 2002) BURKHARD, H.D.; DUHAUT, D.; FUJITA, M.; LIMA, P.; MURPHY, R.; ROJAS, R.; “The road to RoboCup 2050”, Inst. of Informatics, Humboldt-Univ., Berlin; Robotics & Automation Magazine, vol 9, no. 2, pp. 31-38, 2002;
- (BROOKS, 1986) BROOKS, R A; “Robust Layered Control System for a Mobile Robot”, IEEE Journal of Robotics and Automation, -pp.14- 23, Mar 1986
- (CHEN; QUINN, 1997)
- (DE CICCO, 2006) DE CICCO, Lúcia Helena S.; www.saudeanimal.com.br/extinto11.htm, 2006
- (ELFES, 1985) ELFES, A.; “Sonar Based Mapping and Navigation System”, Carnegie-Mellon University, The Robotics Institute, Technical Report, pp. 25-30, 1985;
- (FOX et al., 1998) FOX, Dieter; BURGARD, Wolfram; THRUD, Sebastian; CREMERS, Armin B.; "A hybrid Collision Avoidance Method For Mobile Robots", Proceedings of the 1998 IEEE International Conference on Robotics & Automation, pp. 1238-1243, Leuven Belgium - May 1998;
- (FU; GONZALES; LEE, 1987) FU, K.; GONZALES, R.; LEE, C.; “Robotics: Control, Sensing, Vision and Intelligence”, CAD/CAM, Robotics and Computer Vision, McGraw Hill, 1987;
- (GOLDWINE; BURDICK, 2002) GOLDWINE, B.; BURDICH, W.; “Motion Planning for Kinematic Stratified Systems With Applications to Quasi-Static Legged Locomotion and Finger Gaiting”, IEEE Transactions on Robotics and Automation, Vol. 18, No. 2, April, 2002;
- (HIROSHI; MASAYOSHI, 2003) HIROSHI, Higarashi; MASAYOSHI, Kakikurak; “Obstacle Recognition for a Quadruped Robot by Distance Sensors”, Proceedings of the 2003 IEEE International Conference on Robotics, Intelligent Systems and Signal Processing, pp. 501-506, Changsha, China - October 2003;
- (IGARASHI; MASAYOSHI, 2002) IGARASHI, H.; MASAYOSHI, Kakikura; “Path planning for a quadruped robot in 3d complex environment: verification for trajectory following gate”, Proceedings of JSME Conference on Robotics and Mechatronics, Japan, 2002;
- (ISHIHARA, 1998) ISHIHARA, Wilson M.; “Desenvolvimento e construção de um robô de 4 pernas”, Dissertação de Mestrado em Engenharia Elétrica – Robótica, Universidade Federal do Espírito Santo, Brasil, 1998;
- (KHATIB, 1985) KHATIB, O.; “Real-Time Obstacle Avoidance for Manipulators and Móbile Robots”, Proceedings of the IEEE International Conference On Robotics and Automation, pp. 500-505, St. Louis, MO, 1985;
- (KROGH, 1984) KROGH, B.; “A Generalized potential Field Approach to Obstacle Avoidance Control”, SME-TI Technical Paper MS84-484, Society Of Manufacturing Engineers, Dearbor, Michigan, USA, 1984;
- (JARVIS; BYRNE, 1986)

- (LEE; SONG, 1991) LEE, J. K.; SONG, S. M.; "Path planning and gait of walking machine in an obstacle-strewn environment", *Journal of Robotic System*, Vol. 8, No. 6, pp. 801-827, 1991;
- (LEPINAY; ZAPATA; JOUVENCEL, 1993) LEPINAY, P.; ZAPATA, R.; JOUVENCEL, B.; "Sensor-Based Control of the Reactive Behaviors of Walking Machines", *Proceedings of the International Conference ON Industrial Electronics, Control, and Instrumentation IECON '93*, 1993, vol.3, pp. 1583-1587, Maui, HI, USA, Nov 1993;
- (LIN; SONG, 1999) LIN, J. N., SONG, S. M., "Modeling gait transitions of quadruped based on gait kinematics and CMAC neural networks", *Department of Mechanical Engineering, University of Illinois at Chicago*, 1999, USA;
- (LIMA, 2006) LIMA, Luiz E. de; "Uma arquitetura de controle distribuído para um robô móvel quadrúpede com uso do barramento CAN", *Projeto de Tese de Doutorado aprovado pelo Programa de Pós Graduação Em Engenharia Elétrica – Robótica, Universidade Federal do E. Santo*, 2006, BRASIL;
- (MARTINS FILHO, 1998) MARTINS FILHO, L.; "Une Approche de Supervision de Marché pour Robots Quadrupèdes Intégrant Commande et Décision", *Thèse de Doctorat, Université Paul Sabatier, Toulouse, FRANCE*;
- (MCGHEE, 1985) MCGHEE, R. B.; "Vehicular legged locomotion", *Advances in Automation and Robotics*, Edited by G. N. Saridis, Jai Press Inc, Greenwich, Connecticut, USA;
- (MCGHEE; ISWANDHI, 1979) MCGHEE, R B; ISWANDHI, G I; "Adaptive locomotion of a multilegged robot over rough terrain", *IEEE Transactions on Systems, Man, and Cybernetics*. Vol. SMC-9, pp. 176-182. Apr. 1979;
- (MCMAHON, 1984) MCMAHON, T. A.; "Mechanics of Locomotion", *The Intl. Journal of Robotics Research*, Vol. 3 No 2, Summer, 1984;
- (ORLANDI, 2007) ORLANDI, José Geraldo; "Sistema Sensorial a Laser para Mapeamento de Superfícies e Obstáculos de Robôs Quadrúpedes", *Projeto de Tese de Doutorado a ser submetido ao Programa de Pós Graduação Em Engenharia Elétrica – Robótica, Universidade Federal do E. Santo*, 2007, BRASIL;
- (ROWE 1999) ROWE, Patrick Scan; "Adaptive Motion Planning for Autonomous Mass Excavation", *PhD Thesis, The Robotics Institute, Carnegie Mellon University, USA*, 1999;
- (SONG; WALDRON, 1989) SONG, S. M; WALDRON, K. J.; "Machines that Walk: The Adaptive Suspension Vehicle", *The MIT Press, Cambridge, Massachusetts*, 1989;
- (SPONG; VIDYASAGAR, 1992) SPONG, M.W.; VIDYASAGAR, M.; "Robot Dynamics and Control", *John Wiley & Sons, New York*, 1992;
- (ZELINSKY , 1992) ZELINSKY, A., "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robots", *IEE Transactions on Robotics and Automation*, Vol. 8, No. 6, pp. 707-717, Dec, 1992;

Livros Grátis

(<http://www.livrosgratis.com.br>)

Milhares de Livros para Download:

[Baixar livros de Administração](#)

[Baixar livros de Agronomia](#)

[Baixar livros de Arquitetura](#)

[Baixar livros de Artes](#)

[Baixar livros de Astronomia](#)

[Baixar livros de Biologia Geral](#)

[Baixar livros de Ciência da Computação](#)

[Baixar livros de Ciência da Informação](#)

[Baixar livros de Ciência Política](#)

[Baixar livros de Ciências da Saúde](#)

[Baixar livros de Comunicação](#)

[Baixar livros do Conselho Nacional de Educação - CNE](#)

[Baixar livros de Defesa civil](#)

[Baixar livros de Direito](#)

[Baixar livros de Direitos humanos](#)

[Baixar livros de Economia](#)

[Baixar livros de Economia Doméstica](#)

[Baixar livros de Educação](#)

[Baixar livros de Educação - Trânsito](#)

[Baixar livros de Educação Física](#)

[Baixar livros de Engenharia Aeroespacial](#)

[Baixar livros de Farmácia](#)

[Baixar livros de Filosofia](#)

[Baixar livros de Física](#)

[Baixar livros de Geociências](#)

[Baixar livros de Geografia](#)

[Baixar livros de História](#)

[Baixar livros de Línguas](#)

[Baixar livros de Literatura](#)
[Baixar livros de Literatura de Cordel](#)
[Baixar livros de Literatura Infantil](#)
[Baixar livros de Matemática](#)
[Baixar livros de Medicina](#)
[Baixar livros de Medicina Veterinária](#)
[Baixar livros de Meio Ambiente](#)
[Baixar livros de Meteorologia](#)
[Baixar Monografias e TCC](#)
[Baixar livros Multidisciplinar](#)
[Baixar livros de Música](#)
[Baixar livros de Psicologia](#)
[Baixar livros de Química](#)
[Baixar livros de Saúde Coletiva](#)
[Baixar livros de Serviço Social](#)
[Baixar livros de Sociologia](#)
[Baixar livros de Teologia](#)
[Baixar livros de Trabalho](#)
[Baixar livros de Turismo](#)