

**SYLVIO CELSO TARTARI FILHO**

**MODELAGEM E OTIMIZAÇÃO DE UM ROBÔ DE  
ARQUITETURA PARALELA PARA APLICAÇÕES  
INDUSTRIAIS**

*Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de Mestre  
em Engenharia.*

São Paulo

2006

**SYLVIO CELSO TARTARI FILHO**

**MODELAGEM E OTIMIZAÇÃO DE UM ROBÔ DE  
ARQUITETURA PARALELA PARA APLICAÇÕES  
INDUSTRIAIS**

*Dissertação apresentada à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de Mestre  
em Engenharia.*

*Área de Concentração:  
Engenharia Mecatrônica*

*Orientador:  
Prof. Dr. Eduardo Lobo Lustosa Cabral*

São Paulo

2006

## **FICHA CATALOGRÁFICA**

**Tartari Filho, Sylvio Celso**

**Modelagem e otimização de um robô de arquitetura paralela para aplicações industriais / S.C. Tartari Filho. -- São Paulo, 2006.**

**p.**

**Dissertação (Mestrado) - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.**

**1.Robôs (Aplicações industriais) 2.Arquitetura paralela I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t.**

Aos meus pais, Sylvio e Virgina, por todo o apoio e enorme incentivo em todas as etapas da minha vida e da minha carreira. Que possa tê-los ainda por muitos anos.



## **AGRADECIMENTOS**

Ao professor, orientador e amigo Dr. Eduardo Cabral por toda a orientação, apoio e compreensão pessoal neste trabalho.

À Escola Politécnica da USP, pelas oportunidades dadas. Que continue símbolo da mais alta qualidade na formação de engenheiros no Brasil.

A todas as pessoas, que não de menor importância, contribuíram direta ou indiretamente neste trabalho.

## RESUMO

Este trabalho trata do estudo de robôs de arquitetura paralela, focando na modelagem e otimização dos mesmos. Não foi construído nenhum tipo de protótipo físico, contudo os modelos virtuais poderão, no futuro, habilitar tal façanha. Após uma busca por uma aplicação que se beneficie do uso de um robô de arquitetura paralela, fez-se uma pesquisa por arquiteturas viáveis já existentes ou relatadas na literatura. Escolheu-se a mais apta e prosseguiu-se com os estudos e modelagem cinemática e dinâmica, dando uma maior ênfase na cinemática e dinâmica inversa, esta última utilizando a formulação de Newton - Euler. Foi construído um simulador virtual em ambiente MATLAB 6.5, dotado de várias capacidades como interpolação linear e circular, avanço e uso de múltiplos eixos coordenados. Seu propósito principal é o de demonstrar a funcionalidade e eficácia dos métodos utilizados. Depois foi incorporado ao simulador um algoritmo de cálculo do volume de trabalho da máquina que utiliza alguns dados do usuário para calcular o volume, que pode ser aquele atrelado a uma postura em particular ou o volume de trabalho de orientação total. Algoritmos para medir o desempenho da máquina quanto à uniformidade e utilização da força dos atuadores foram construídos e também incorporados ao simulador, que consegue mostrar o elipsóide de forças ao longo de quaisquer movimentos executados pela plataforma móvel. Quanto à otimização, parte do ferramental previamente construído foi utilizado para que se pudesse chegar a um modelo de uma máquina que respeitasse restrições mínimas quanto ao tamanho e forma de seu volume de trabalho, mas ainda mantendo o melhor desempenho possível dentro deste volume.

## **ABSTRACT**

This work is about the study of parallel architecture robots, focusing in modeling and optimization. No physical prototypes were built, although the virtual models can help those willing to do so. After searching for an application that could benefit from the use of a parallel robot, another search was made, this time for the right architecture type. After selecting the architecture, the next step was the kinematics and dynamics analysis. The dynamics model is developed using the Newton – Euler method. A virtual simulator was also developed in MATLAB 6.5 environment. The simulator's main purpose was to demonstrate that the methods applied were correct and efficient, so it has several features such as linear and circular interpolations, capacity to use multiple coordinate systems and others. After finishing the simulator, an algorithm to calculate the machine workspace was added. The algorithm receives as input some desired requirements regarding the manipulator pose and then calculates the workspace, taking into consideration imposed constraints. Lastly, algorithms capable to measure the manipulator's performance regarding to its actuator and end-effector force relationship were also incorporated into the simulator that calculates the machine's force ellipsoid during any movement, for each desired workspace point. For the optimization procedures, some previously developed tools were used, so that the resulting model was capable to respect some workspace constraints regarding size and shape, but also maintaining the best performance possible inside this volume.

# SUMÁRIO

## LISTA DE FIGURAS

## LISTA DE TABELAS

## LISTA DE ABREVIATURAS E SIGLAS

## LISTA DE SÍMBOLOS

<b>1. INTRODUÇÃO</b>	<b>1</b>
<b>1.1. Objetivos</b>	<b>1</b>
<b>1.2. Justificativas</b>	<b>1</b>
<b>1.3. Definição de um Robô</b>	<b>2</b>
<b>1.4. Elementos Básicos Constituintes de um Robô</b>	<b>4</b>
<b>1.5. Histórico</b>	<b>5</b>
1.5.1. O Surgimento dos Termos Robô e Robótica	5
1.5.2. O Primeiro Robô Industrial	6
1.5.3. Os Robôs de Arquitetura Paralela	7
<b>1.6. Classificação dos Robôs</b>	<b>9</b>
<b>1.7. O Panorama Atual dos Robôs (Acadêmico e de Mercado)</b>	<b>13</b>
<b>1.8. Algumas Aplicações dos Robôs de Arquitetura Paralela</b>	<b>14</b>
<b>1.9. Conceitos Básicos para este Trabalho</b>	<b>17</b>
<b>1.10. Comparação entre robôs de Arquitetura Seriada e Arquitetura Paralela</b>	<b>19</b>
<b>2. REVISÃO BIBLIOGRÁFICA</b>	<b>21</b>
<b>2.1. Cinemática</b>	<b>21</b>
2.1.1. Cinemática Inversa	22
2.1.2. Cinemática Direta	22
2.1.3. Matrizes Jacobiano e Singularidades	25
<b>2.2. Dinâmica</b>	<b>26</b>
<b>2.3. Volume de trabalho</b>	<b>28</b>
<b>2.4. Síntese e Otimização de Robôs de Arquitetura Paralela</b>	<b>29</b>

2.5.	Controle	33
<b>3.</b>	<b>METODOLOGIA</b>	<b>34</b>
<b>3.1.</b>	<b>Passos da Metodologia</b>	<b>35</b>
3.1.1.	Escolher uma Aplicação que Necessite um Robô de Arquitetura Paralela	35
3.1.2.	Avaliar as Exigências da Aplicação	35
3.1.3.	Investigar as Arquiteturas Existentes mais Promissoras	36
3.1.4.	Selecionar a Arquitetura mais Apta para a Aplicação Escolhida	36
3.1.5.	Estudar a Cinemática da Máquina, Formas de Abordagem e de Solução	36
3.1.6.	Realizar a Análise Cinemática	37
3.1.7.	Construir um Simulador Gráfico para a Arquitetura	37
3.1.8.	Estudar a Dinâmica da Máquina, Formas de Abordagem e Solução	38
3.1.9.	Construir um Algoritmo para Resolver a Dinâmica da Arquitetura Escolhida	38
3.1.10.	Estudar os Objetivos a serem atingidos numa Otimização	39
3.1.11.	Decidir a melhor Abordagem para a Otimização	39
3.1.12.	Construir o Algoritmo de Otimização da Máquina	39
3.1.13.	Estruturar e Apresentar os Resultados Finais	40
3.1.14.	Definir os Próximos Passos (Trabalhos Futuros)	40
<b>4.</b>	<b>A ESCOLHA DA APLICAÇÃO E DA ARQUITETURA</b>	<b>41</b>
<b>4.1.</b>	<b>A Necessidade de um Robô de Arquitetura Paralela</b>	<b>41</b>
<b>4.2.</b>	<b>Investigação de Arquiteturas Existentes</b>	<b>43</b>
4.2.1.	Arquiteturas Utilizadas em Usinagem	43
4.2.2.	Arquiteturas Utilizadas em Pick – and – Place e Montagem	48
<b>4.3.</b>	<b>A Escolha da Aplicação e seus Requerimentos em uma Análise Qualitativa</b>	<b>55</b>
<b>4.4.</b>	<b>A Escolha da Arquitetura: uma Análise de Foco Qualitativo</b>	<b>56</b>
4.4.1.	Comentários sobre as Arquiteturas	57
4.4.2.	A Arquitetura Escolhida	59
4.4.3.	Uma Breve Revisão da Literatura da Hexa	59
<b>5.</b>	<b>ANÁLISE CINEMÁTICA INVERSA</b>	<b>62</b>
<b>5.1.</b>	<b>O Modelo da Cinemática Inversa da Hexa: Análise de Posições</b>	<b>62</b>
<b>5.2.</b>	<b>Movimentações da Plataforma no Espaço Tridimensional: Questões do Simulador Virtual</b>	<b>71</b>
<b>5.3.</b>	<b>Volume de Trabalho</b>	<b>71</b>
5.3.1.	Tipos de Volume de Trabalho	72

5.3.2.	Algoritmo para Cálculo do Volume de Trabalho _____	74
<b>5.4.</b>	<b>O Modelo da Cinemática Inversa da Hexa: Equações de Velocidades e de Acelerações _____</b>	<b>80</b>
<b>6.</b>	<b><i>ANÁLISE DO JACOBIANO E DA CINEMÁTICA DIRETA</i> _____</b>	<b>87</b>
<b>6.1.</b>	<b>A Matriz Jacobiano _____</b>	<b>87</b>
<b>6.2.</b>	<b>Condições de Singularidade _____</b>	<b>88</b>
6.2.1.	Singularidades da Cinemática Inversa _____	88
6.2.2.	Singularidades da Cinemática Direta _____	89
6.2.3.	Singularidades combinadas _____	89
<b>6.3.</b>	<b>Cálculo da Matriz Jacobiano da Hexa _____</b>	<b>90</b>
<b>6.4.</b>	<b>Rigidez e Flexibilidade _____</b>	<b>92</b>
6.4.1.	Matriz de Rigidez e Matriz de Flexibilidade _____	92
<b>6.5.</b>	<b>Elipsóides de Forças, de Velocidades e de Deslocamentos _____</b>	<b>95</b>
6.5.1.	Um Exemplo Numérico do Elipsóide de Forças _____	97
<b>6.6.</b>	<b>Análise da Cinemática Direta _____</b>	<b>100</b>
<b>7.</b>	<b><i>ANÁLISE DINÂMICA INVERSA</i> _____</b>	<b>102</b>
<b>7.1.</b>	<b>A Abordagem para a Obtenção do Modelo _____</b>	<b>102</b>
<b>7.2.</b>	<b>Metodologia e Conceitos na Formulação de Newton-Euler da Hexa _____</b>	<b>103</b>
<b>7.3.</b>	<b>Equações da Análise Dinâmica da Hexa _____</b>	<b>108</b>
<b>8.</b>	<b><i>OTIMIZAÇÃO</i> _____</b>	<b>115</b>
<b>8.1.</b>	<b>Uma Introdução aos Conceitos de Otimização e de Síntese _____</b>	<b>115</b>
<b>8.2.</b>	<b>A Otimização da Hexa _____</b>	<b>119</b>
8.2.1.	Otimização Completa de Robôs de Arquitetura Paralela _____	119
8.2.2.	Abordagens para a Otimização da Hexa _____	120
8.2.3.	Equacionamento do Problema de Otimização da Hexa e Método _____	121
<b>9.</b>	<b><i>FERRAMENTAS CONSTRUÍDAS E RESULTADOS</i> _____</b>	<b>126</b>
<b>9.1.</b>	<b>O Modelo Virtual da Hexa e seus Parâmetros Geométricos _____</b>	<b>126</b>
<b>9.2.</b>	<b>Cálculo do Volume de Trabalho _____</b>	<b>130</b>
<b>9.3.</b>	<b>Simulador Virtual _____</b>	<b>135</b>

<b>9.4.</b>	<b>Elipsóide de Forças</b>	<b>140</b>
<b>9.5.</b>	<b>Cinemática Direta: Análise de Posições</b>	<b>147</b>
<b>9.6.</b>	<b>Análise Dinâmica</b>	<b>150</b>
<b>9.7.</b>	<b>Otimização</b>	<b>158</b>
9.7.1.	Otimização Através do Elipsóide de Forças	160
9.7.2.	Otimização Através do Elipsóide de Velocidades	162
9.7.3.	Comentários das Soluções Encontradas e Observações	163
<b>10.</b>	<b><i>CONCLUSÕES E PRÓXIMOS PASSOS</i></b>	<b>165</b>
10.1.	Conclusões	166
10.2.	Sugestão de trabalhos futuros	167
<b>11.</b>	<b><i>REFERÊNCIAS BIBLIOGRÁFICAS</i></b>	<b>169</b>
<b><i>APÊNDICE A: DEDUÇÕES COMPLEMENTARES DA ANÁLISE</i></b>		
<b><i>CINEMÁTICA INVERSA DA HEXA</i></b>		
		<b>178</b>
<b><i>APÊNDICE B: DEDUÇÃO DA EQUAÇÃO GERAL DE UM ELIPSÓIDE DE</i></b>		
<b><i>DIMENSÃO N</i></b>		
		<b>181</b>
<b><i>APÊNDICE C: FUNDAMENTOS MATEMÁTICOS PARA O SIMULADOR</i></b>		
<b><i>VIRTUAL</i></b>		
		<b>186</b>
<b><i>APÊNDICE D: FUNDAMENTOS TEÓRICOS PARA OTIMIZAÇÃO</i></b>		
		<b>196</b>

## LISTA DE FIGURAS

<i>Figura 1: Destaques de alguns elementos básicos de um robô (Figura de <b>ABB, 2005</b>). _____</i>	5
<i>Figura 2: O Unimate - primeiro robô industrial (<b>RRG, 2005</b>). _____</i>	7
<i>Figura 3: O protótipo de Gough (à esquerda) e o modelo de Stewart (à direita) – Figuras de <b>Merlet (2000)</b>. _____</i>	8
<i>Figura 4: Robôs segundo o tipo e forma de suas cadeias cinemáticas. As figuras dos robôs foram extraídas do site da ABB – <b>ABB, 2005</b> (1 e 3), do <b>PI, 2005</b> (2), do <b>RRG, 2005</b> (4), de <b>Hannan e Walker, 1999</b> (5). _____</i>	10
<i>Figura 5: Classificação dos robôs seriados quanto aos três primeiros graus de liberdade (As figuras de 1 a 4 são de <b>METU, 2005</b>, notas de aula de ME445 – Integrated Manufacturing Systems). A figura do robô articulado foi criada pelo próprio autor desta dissertação. _____</i>	12
<i>Figura 6: À esquerda, o IRB 940 Tricept da ABB; à direita, o F-200i da FANUC (fotos da <b>ABB, 2005</b> e da <b>FANUC, 2005</b>, respectivamente). _____</i>	14
<i>Figura 7: Da esquerda para a direita: uma máquina ferramenta baseada na arquitetura Hexapod (<b>Stankóczy, 1999</b>); a máquina-ferramenta Hexapod da Ingersoll (<b>Ingersoll, 2004</b>) e o Tricept do Neos Robotics (<b>Robotics Online, 2005</b>) _____</i>	15
<i>Figura 8: À esquerda, o FlexPicker<sup>TM</sup> da ABB (<b>ABB, 2005</b>) e à direita a máquina da SIG (<b>Robotics Online, 2005</b>), ambas arquiteturas Delta, inventada por Reymond Clavel. _____</i>	15
<i>Figura 9: À esquerda, o Hexapod M-850 do IPA, que pode carregar diversos instrumentos (possui repetibilidade de 1-2 µm e resolução dos atuadores de 5 nm); à direita, um modelo para cirurgias cardíacas (fotos, respectivamente, de <b>PI, 2005</b> e <b>Lazarevic, 2000</b>). _____</i>	16
<i>Figura 10: Simulador de voo do Boeing 737, visão interior (à esquerda) e visão exterior (no centro) – <b>Ansett Flight Simulator Centre, 2004</b>. À direita um simulador para bicicletas para estudo de vibrações – Laboratório de Controle de Vibrações do <b>KAIST (2004)</b>. _____</i>	16
<i>Figura 11: Fluxo de Atividades a serem seguidas durante o projeto _____</i>	34
<i>Figura 12: Da esquerda para a direita – o F-200i da FANUC (Hexapod), o IRB 340 da ABB (Delta) e o Tricept da Neos Robotics (Tricept) – Fotos respectivamente de <b>FANUC (2005)</b>, <b>ABB (2005)</b> e <b>Robotics Online (2005)</b>. _____</i>	42
<i>Figura 13: Esquemas das arquiteturas Eclipse-I (esquerda) e II (direita) - <b>Kim et. al., 1999</b>. _____</i>	44
<i>Figura 14: Esquemas das arquiteturas Hexaglide (à esquerda) e Hexapod – visão inferior (à direita). _____</i>	45
<i>Figura 15: Esquema da arquitetura HexaM (à direita) – <b>LIRMM, 2004</b>. Esquema da arquitetura Tricept (à esquerda). _____</i>	46
<i>Figura 16: Máquina do tipo Eclipse-I (à esquerda) e do tipo Eclipse-II (à direita). Fotos de <b>IPMP, 2002</b>. _____</i>	46
<i>Figura 17: À esquerda uma máquina com arquitetura Hexapod, da Micromat (<b>Blümlein, 1999</b>), e à direita a IWF-Hexaglide (<b>Knapp e Cobet, 2000</b>). _____</i>	47



<i>Figura 18: À esquerda, a HexaM da Toyoda Machine Works Ltda. (Parallemic, 2002) e à direita o Tricept da Neos Robotics (Robotics Online, 2004).</i>	47
<i>Figura 19: À esquerda, esquema do robô Delta (Pierrot et al., 1991a). À direita, a organização das juntas das cadeias cinemáticas da arquitetura (Company e Pierrot, 1999).</i>	48
<i>Figura 20: À esquerda o MicroDelta 240 e à direita o Delta 720 com direct drive (fotos do LSRO, 2005).</i>	50
<i>Figura 21: O Delta fabricado pela Demarex Robotique et Microtechnique SA em operação (foto extraída do LSRO, 2005).</i>	50
<i>Figura 22: O Delta fabricado pela empresa sueca Elekta num ambiente hospitalar (foto do LSRO, 2005).</i>	51
<i>Figura 23: Três exemplos de modelos da família H4 (de cima para baixo, os dois primeiros são de Company e Pierrot, 1999 e o último do site do François Pierrot, 2005) e a organização de suas cadeias cinemáticas (figuras de Company e Pierrot, 1999).</i>	52
<i>Figura 24: O robô da família H4 desenvolvido no LIRMM (François Pierrot, 2005) em vista lateral e em vista inferior.</i>	53
<i>Figura 25: Esquema da arquitetura Hexa (à esquerda) e a estrutura de suas cadeias cinemáticas (à direita). A figura da estrutura das cadeias foi extraída de LIRMM, 2004.</i>	54
<i>Figura 26: À esquerda, a Hexa da Toyoda (Space Machines Laboratory, 2004) e à direita a Hexa do IWF (2004).</i>	54
<i>Figura 27: Volumes de trabalho da Delta (à esquerda) e da Hexa (à direita) para tamanhos de máquina semelhantes (IFM, 2005)</i>	57
<i>Figura 28: Esquema detalhado da arquitetura Hexa.</i>	63
<i>Figura 29: Base fixa da Hexa (o eixo Y não necessariamente passa pelo ponto 6).</i>	64
<i>Figura 30: As soluções da cinemática inversa da Hexa. A figura da esquerda é a solução eliminada.</i>	67
<i>Figura 31: Modelo da Plataforma Móvel da Hexa.</i>	70
<i>Figura 32: Algoritmo para cálculo do volume de trabalho.</i>	76
<i>Figura 33: Ilustração para cálculo da distância entre dois ligamentos (A figura de fundo é da Hexa do IWF, 2004).</i>	77
<i>Figura 34: Casos de cálculo de distâncias entre segmentos de reta.</i>	78
<i>Figura 35: Restrição do ângulo das articulações da plataforma móvel.</i>	80
<i>Figura 36: Esquema vetorial de uma cadeia cinemática da Hexa.</i>	81
<i>Figura 37: Robô seriado de dois graus de liberdade.</i>	98
<i>Figura 38: Elipsóide de forças no efetuador do robô serial de dois graus de liberdade.</i>	99
<i>Figura 39: Passos para obtenção do modelo dinâmico pela formulação de Newton-Euler.</i>	104
<i>Figura 40: Esquema dos elementos principais da Hexa.</i>	104
<i>Figura 41: Diagrama de Corpo Livre da Hexa.</i>	105
<i>Figura 42: Elementos básicos do algoritmo Darwin2K (Leger, 1999).</i>	117
<i>Figura 43: Robô que opera no espaço (em gravidade zero). Figura de Leger (1999).</i>	117

<b>Figura 44:</b> <i>Requerimentos para um transportador de materiais (Leger, 1999).</i>	118
<b>Figura 45:</b> <i>População inicial e resultado de três simulações do Darwin2K (Leger, 1999).</i>	118
<b>Figura 46:</b> <i>Seqüência para obter uma estrutura ótima.</i>	119
<b>Figura 47:</b> <i>Restrição do volume de trabalho na otimização.</i>	123
<b>Figura 48:</b> <i>Janela inicial da ferramenta desenvolvida em Matlab.</i>	127
<b>Figura 49:</b> <i>Janela de Ajuda.</i>	127
<b>Figura 50:</b> <i>Modelo virtual da Hexa e a sua plataforma móvel em detalhes.</i>	129
<b>Figura 51:</b> <i>Volumes de trabalho de orientação fixa e orientação total da Hexa em duas vistas diferentes.</i>	131
<b>Figura 52:</b> <i>Acima o volume de trabalho de orientação fixa e abaixo o de orientação total.</i>	132
<b>Figura 53:</b> <i>Volumes de trabalho da Hexa onde o de orientação fixa utilizou a plataforma com ângulo de 45° com a vertical (rotação no eixo X).</i>	133
<b>Figura 54:</b> <i>Volumes de trabalho de orientação fixa e total com algoritmo de detecção de colisões para diferentes diâmetros dos ligamentos passivos. (1) 20 mm; (2) 50 mm; (3) 60 mm; (4) 80 mm; (5) 95 mm.</i>	134
<b>Figura 55:</b> <i>Comparação de volumes de trabalho de orientação fixa da Hexa. À esquerda, o volume da máquina deste trabalho e à direita tem-se uma figura de IFM, 2005.</i>	135
<b>Figura 56:</b> <i>Barra de progresso para o cálculo dos volumes de trabalho, indicando o percentual já calculado (existe uma para cada volume: orientação fixa e total).</i>	135
<b>Figura 57:</b> <i>Janela do Simulador Virtual.</i>	136
<b>Figura 58:</b> <i>Volumes de trabalho de orientação fixa e total (acima) e exemplo de volume escolhido pelo usuário (abaixo).</i>	137
<b>Figura 59:</b> <i>Sistemas de coordenadas (acima) e trajetórias lineares executadas (abaixo, linhas em azul claro).</i>	138
<b>Figura 60:</b> <i>Trajétoria circular num plano qualquer (acima) e movimentos utilizando o sistema da ferramenta.</i>	139
<b>Figura 61:</b> <i>Elipsóide de forças para o caso em que os torques valem zero e a plataforma está em sua posição nominal (acima) e no ponto <math>X = -150</math> mm (abaixo).</i>	141
<b>Figura 62:</b> <i>Elipsóide de torques para o caso em que os torques valem zero e a plataforma está em sua posição nominal (acima) e no ponto <math>X = -150</math> mm (abaixo).</i>	142
<b>Figura 63:</b> <i>Elipsóide de forças para <math>X = -480</math> mm (acima) e <math>X = -485.21558</math> mm (abaixo). Percebe-se o tamanho diminuto deste último (praticamente uma linha).</i>	143
<b>Figura 64:</b> <i>Elipsóide de forças no ponto <math>X = -485.21559</math>. Acima, o modelo ampliado e abaixo o elipsóide visto (um dos eixos dele é inexistente – destacado com linha pontilhada).</i>	144
<b>Figura 65:</b> <i>Elipsóide de forças em <math>X = -486</math> mm.</i>	146
<b>Figura 66:</b> <i>Ponto <math>X, Y, Z = [250 150 -150]</math> no volume de trabalho de orientação fixa.</i>	149
<b>Figura 67:</b> <i>Ferramenta de análise dinâmica.</i>	152
<b>Figura 68:</b> <i>Forças nos pontos da base fixa em duas bases – global (acima) e dos ligamentos atuados (abaixo).</i>	153

<b>Figura 69:</b> Forças nos centros da articulação entre ligamentos atuados e passivos em duas bases diferentes, sendo acima a global e abaixo a dos ligamentos atuados. _____	154
<b>Figura 70:</b> Forças nos centros da articulação entre ligamentos atuados e passivos na base dos ligamentos passivos (acima) e torque aplicado pelos atuadores (abaixo). _____	155
<b>Figura 71:</b> Forças nos pontos da plataforma móvel em duas bases – global (acima) e dos ligamentos passivos (abaixo). _____	156
<b>Figura 72:</b> Momentos nos pontos da base fixa em duas bases - global (acima) e dos ligamentos atuados (abaixo). _____	157
<b>Figura 73:</b> Ferramenta de Otimização. _____	159
<b>Figura 74:</b> Acima a lista que aparece após a execução de uma otimização. Abaixo uma janela de aviso que aparece quando se deseja fechar a lista de opções. _____	159
<b>Figura 75:</b> À esquerda a solução da otimização através do elipsóide de forças e à direita a solução para o elipsóide de velocidades. _____	163
<b>Figura 76:</b> Possíveis ângulos de solução de uma tangente. _____	180
<b>Figura 77:</b> Exemplo numérico para uma transformação de coordenadas. _____	187
<b>Figura 78:</b> Figura ilustrativa da rotação tridimensional em torno de um eixo qualquer. _____	192
<b>Figura 79:</b> Ângulos de Roll – Pitch – Yaw. _____	192

## LISTA DE TABELAS

<b>Tabela 1:</b> Comparação entre robôs de arquitetura seriada e arquitetura paralela. _____	19
<b>Tabela 2:</b> Tabela comparativa de velocidades e acelerações máximas de modelos de arquiteturas paralelas para pick-and-place / montagens leves. _____	57
<b>Tabela 3:</b> Itens da janela inicial da ferramenta construída. _____	128
<b>Tabela 4:</b> Valor dos volumes de trabalho da Hexa nos diferentes casos. _____	133
<b>Tabela 5:</b> Componentes do simulador virtual. _____	136
<b>Tabela 6:</b> Jacobiano avaliado no ponto $X = -485.21558$ , $Y = 0$ e $Z = 0$ ; autovetores e autovalores da matriz característica do elipsóide de forças no ponto e o comprimento dos eixos do elipsóide (baseado nos autovalores). $Ang = \text{Angular}$ , $Inf = \text{Infinito}$ . _____	145
<b>Tabela 7:</b> Jacobiano avaliado no ponto $X = -485.21559$ , $Y = 0$ e $Z = 0$ ; autovetores e autovalores da matriz característica do elipsóide de forças no ponto e o comprimento dos eixos do elipsóide (baseado nos autovalores). $Ang = \text{Angular}$ , $Inf = \text{Infinito}$ , $i = \text{Imaginário}$ . _____	145
<b>Tabela 8:</b> Iterações da otimização através do conceito do elipsóide de forças. _____	161
<b>Tabela 9:</b> Iterações da otimização através do conceito do elipsóide de velocidades. _____	162
<b>Tabela 10:</b> Avaliação em dois pontos do conceito da função objetivo do elipsóide de forças. _____	164
<b>Tabela 11:</b> Comparação entre os métodos gerais de otimização. _____	200

## LISTA DE ABREVIATURAS E SIGLAS

R. U. R.	Obra <b>Rob</b> ôs <b>Un</b> iversais de <b>R</b> ossum, de Karel Chapek
SCARA	<b>S</b> elective <b>C</b> ompliance <b>A</b> ssembly <b>R</b> obot <b>A</b> rm
RIA	<b>R</b> obotics <b>I</b> ndustries <b>A</b> ssociation
IFR	<b>I</b> nternational <b>F</b> ederation of <b>R</b> obotics
IPA	Fraunhofer- <b>I</b> nstitut für <b>P</b> roduktionstechnik und <b>A</b> utomatisierung
IWF	<b>I</b> nstitut für <b>W</b> erkzeugmaschinen und <b>F</b> ertigungstechnik
PI	<b>P</b> hysikinstrumente
PKM	<b>P</b> arallel <b>K</b> inematic <b>M</b> achine
SPM	Stewart like <b>P</b> arallel <b>M</b> anipulator
IPMP	<b>I</b> nnovative <b>P</b> arallel <b>M</b> echanism <b>P</b> latforms
LIRMM	<b>L</b> aboratoire d' <b>I</b> nformatique, de <b>R</b> obotique et de <b>M</b> icroélectronique de <b>M</b> ontpellier
RRG	<b>R</b> obotics <b>R</b> esearch <b>G</b> roup
PID	<b>P</b> roportional <b>I</b> ntegral <b>D</b> erivativo
Lig.	<b>L</b> igamento
Trab.	<b>T</b> rabalho
Vol.	<b>V</b> olume
KAIST	<b>K</b> orea <b>A</b> dvanced <b>I</b> nstitute of <b>T</b> echnology

## LISTA DE SÍMBOLOS

### *Capítulo 5: Análise Cinemática Inversa e Simulador Virtual*

$Pa_j$	Centro da articulação esférica entre os ligamentos passivo e os atuado da Hexa, onde $j$ é o número da cadeia cinemática, sendo $j = 1, 2, \dots, 6$ .
$Ps_j$	Centro do cruzamento dos ligamentos atuados com seus respectivos motores, onde $j$ é o número da cadeia cinemática.
$Pi_j$	Centro das articulações esféricas da plataforma móvel da Hexa, onde $j$ é o número da cadeia cinemática.
$Pc$	Ponto de interesse (ou central) do efetuador (ou ferramenta) no modelo da Hexa.
$\theta_j$	Ângulo de atuação (com relação ao plano da base fixa da Hexa), onde $j$ é o número da cadeia cinemática.
$h_j$	Ligamentos passivos na forma vetorial, onde $j = 1, 2, \dots, 6$ .
$l_j$	Ligamentos atuados na forma vetorial, onde $j = 1, 2, \dots, 6$ .
$t$	Lado da plataforma móvel da Hexa, que é hexagonal.
$h_{tool}$	Distância do plano da plataforma móvel hexagonal da Hexa ao ponto de interesse da ferramenta ( $Pc$ ).
$q$	Distância do centro da base fixa da Hexa a qualquer uma das três linhas que unem dois pontos $Ps_j$ de ligamentos atuados paralelos.
$d$	Distância entre dois ligamentos atuados paralelos da Hexa.
$o-x'y'z'$	Sistema de eixos coordenados auxiliar (usado em vários itens).
$o-xyz$	Sistema de eixos coordenados solidário à plataforma móvel da Hexa com centro em $Pc$ .
$O-XYZ$	Sistema de eixos coordenados fixo à base da Hexa, com centro no ponto central da mesma. O eixo $X$ é perpendicular à linha $\overline{Ps_1Ps_2}$ .
$Pk_{j(z)}$	Coordenada $X$ do ponto qualquer $Pk_j$ nas coordenadas de $O-XYZ$ .
$Pk_{j(y)}$	Coordenada $Y$ do ponto qualquer $Pk_j$ nas coordenadas de $O-XYZ$ .
$Pk_{j(x)}$	Coordenada $Z$ do ponto qualquer $Pk_j$ nas coordenadas de $O-XYZ$ .
$R$	Raio máximo de busca do algoritmo de dicotomia do vol. de trabalho.
$\rho$	Raio de busca do algoritmo de dicotomia do volume de trabalho.

$\varphi$	Ângulo que com $\psi$ definem a direção de busca do volume de trabalho.
$\psi$	Ângulo que com $\varphi$ definem a direção de busca do volume de trabalho.
$B(\rho, \varphi, \psi)$	Pontos da borda do volume de trabalho da Hexa.
$\varepsilon$	Erro máximo para o algoritmo de cálculo do volume de trabalho.
$P_j$	Pontos para o algoritmo de detecção de colisões entre ligamentos.
$\lambda_1$	Variável auxiliar para o algoritmo de colisões entre ligamentos.
$\lambda_2$	Variável auxiliar para o algoritmo de colisões entre ligamentos.
$\mathbf{n}$	Vetor auxiliar para o algoritmo de colisões entre ligamentos.
$\phi$	Ângulo entre lig. passivos e ativos para o algoritmo de colisões.
$\mathbf{u}$	Vetor auxiliar para o cálculo do valor do volume de trabalho ( $\text{m}^3$ ).
$\mathbf{v}$	Vetor auxiliar para o cálculo do valor do volume de trabalho ( $\text{m}^3$ ).
$\mathbf{w}$	Vetor auxiliar para o cálculo do valor do volume de trabalho ( $\text{m}^3$ ).
$O'-X'Y'Z'$	Sistema de coordenadas auxiliar para transformações de coordenadas.
$V_{\text{Pirâmide}}$	Volume de cada pirâmide que compõe o volume de trabalho.
$\mathbf{i}, \mathbf{j}, \mathbf{k}$	Versores de $O-XYZ$ .
$\mathbf{i}', \mathbf{j}', \mathbf{k}'$	Versores de $O'-X'Y'Z'$ .
${}^b\mathbf{M}_a$	Matriz de transformação de coordenadas do sistema $a$ para o $b$ .
$\alpha$	Ângulo que com $\beta$ definem o eixo para uma rotação 3D qualquer.
$\beta$	Ângulo que com $\alpha$ definem o eixo para uma rotação 3D qualquer.
$\theta$	Ângulo de rotação em torno do eixo definido por $\alpha$ e $\beta$ .
$\mathbf{R}_{\mathbf{n},\theta}$	Matriz de rotação em torno de um eixo qualquer, definida por $\alpha, \beta, \theta$ .
$\phi$	Ângulo de <i>Yaw</i> .
$\theta$	Ângulo de <i>Pitch</i> .
$\varphi$	Ângulo de <i>Roll</i> .
$\mathbf{R}_{\varphi,\theta,\varphi}$	Matriz de <i>Roll – Pitch – Yaw</i> .
$\mathbf{v}_{Pc}$	Velocidade do ponto $Pc$ pertencente à plataforma móvel.
$\dot{\mathbf{v}}_{Pc}$	Aceleração do ponto de interesse $Pc$ pertencente à plataforma móvel.
$\boldsymbol{\omega}_p$	Velocidade angular da plataforma móvel.
$\dot{\boldsymbol{\omega}}_p$	Aceleração angular da plataforma móvel.
$\mathbf{b}_j$	Vetores que partem do ponto de interesse $Pc$ e vão até cada um dos centros das articulações em $Pi_j$ (vértices da plataforma móvel).

$\dot{\theta}_j$	Velocidade angular das articulações atuadas.
$\ddot{\theta}_j$	Aceleração angular das articulações atuadas.
$v_{Ps_j}$	Velocidade dos pontos $Ps_j$ .
$\dot{v}_{Ps_j}$	Aceleração dos pontos $Ps_j$ .
$v_{Pa_j}$	Velocidade dos pontos $Pa_j$ .
$\dot{v}_{Pa_j}$	Aceleração dos pontos $Pa_j$ .
$v_{Pi_j}$	Velocidade dos pontos $Pi_j$ .
$\dot{v}_{Pi_j}$	Aceleração dos pontos $Pi_j$ .
$\omega_{h_j}$	Velocidade angular dos ligamentos $h_j$ .
$\dot{\omega}_{h_j}$	Aceleração angular dos ligamentos $h_j$ .

### ***Capítulo 6: Análise do Jacobiano e da Cinemática Direta***

$q$	Vetor cujos elementos são os ângulos de atuação.
$\dot{q}$	Vetor cujos elementos são as velocidades de rotação dos motores.
$x$	Vetor cujos elementos são a postura da máquina (posição de $Pc$ e orientação da máquina).
$\dot{x}$	Vetor cujos elementos são a velocidade da plataforma móvel (linear e angular).
$J$	Matriz Jacobiano.
$Jx$	Matriz Jacobiano da cinemática direta.
$Jq$	Matriz Jacobiano da cinemática inversa.
$\tau$	Vetor cujos elementos são os torques nos atuadores.
$\chi$	Matriz diagonal cujos elementos são a rigidez das articulações atuadas.
$\Delta q$	Vetor de variação da posição dos atuadores.
$\Delta x$	Vetor de variação da posição e orientação da plataforma móvel.
$\delta W$	Variação infinitesimal do trabalho (energia) da Hexa.
$\delta q$	Variação infinitesimal da posição dos atuadores.



$\delta \mathbf{x}$	Variação infinitesimal da posição e orientação da plataforma móvel.
$\lambda_j$	Autovalor da matriz característica do elipsóide de forças.
$\zeta_i$	Componente $i$ do autovetor $j$ cujo respectivo autovalor é $\lambda_j$ .

### **Capítulo 7: Análise Dinâmica Inversa**

$G_l$	Centro de gravidade do ligamento $\mathbf{l}_j$ .
$G_h$	Centro de gravidade do ligamento $\mathbf{h}_j$ .
$G_P$	Centro de gravidade da plataforma móvel.
$m_l$	Massa do ligamento $\mathbf{l}_j$ .
$m_h$	Massa do ligamento $\mathbf{h}_j$ .
$m_P$	Massa da plataforma móvel.
$cl_j$	Vetor de mesma direção e sentido do ligamento $\mathbf{l}_j$ que parte do início de $\mathbf{l}_j$ até seu centro de gravidade.
$ch_j$	Vetor de mesma direção e sentido do ligamento $\mathbf{h}_j$ que parte do início de $\mathbf{h}_j$ até seu centro de gravidade.
$c_P$	Vetor que parte de $P_c$ e aponta para o centro de massa da plat. móvel.
$\mathbf{I}_k$	Matriz de inércia do elemento $k$ , onde $k$ é genérico.
$I_x, I_y, I_z$	Momento de inércia de um corpo com relação aos eixos $X, Y$ e $Z$ .
$I_{xy}, I_{xz}, I_{yz}$	Produtos de inércia com relação
$\mathbf{g}$	Aceleração da gravidade.
$\mathbf{F}_{Pij}$	Forças de ação / reação internas à Hexa no ponto $P_{ij}$ .
$\mathbf{F}_{Paj}$	Forças de ação / reação internas à Hexa no ponto $P_{aj}$ .
$\mathbf{F}_{Psj}$	Forças de reação com o solo ou base fixa, nos pontos $P_{sj}$ .
$\mathbf{M}_{Psj}$	Momentos de reação com a base fixa, nos pontos $P_{sj}$ .
$\mathbf{F}_{Ext}$	Forças externas aplicadas à plataforma móvel.
$\mathbf{M}_{Ext}$	Momento externo aplicado à plataforma móvel.
$\boldsymbol{\omega}_P$	Velocidade angular da plataforma móvel.
$\dot{\boldsymbol{\omega}}_P$	Aceleração angular da plataforma móvel.
$(\mathbf{H}_{h_j})_{Pa_j}$	Momento angular do ligamento $\mathbf{h}_j$ no ponto $P_{aj}$ .
$(\dot{\mathbf{H}}_{h_j})_{Pa_j}$	Derivada do momento angular do ligamento $\mathbf{h}_j$ no ponto $P_{aj}$ .

$(\mathbf{M}_{\mathbf{h}_j})_{Pa_j}$	Somatória dos momentos relativos a $\mathbf{h}_j$ no ponto $Pa_j$ .
$\mathbf{I}_{\mathbf{h}_j}$	Matriz de inércia do ligamento $\mathbf{h}_j$ em seu baricentro.
$\mathbf{I}_{\mathbf{l}_j}$	Matriz de inércia do ligamento $\mathbf{l}_j$ em seu baricentro.
$\mathbf{I}_p$	Matriz de inércia da plataforma móvel em seu baricentro.
$\mathbf{v}_{Gh_j}$	Velocidade do baricentro do ligamento $\mathbf{h}_j$ .
$\dot{\mathbf{v}}_{Gh_j}$	Aceleração do baricentro do ligamento $\mathbf{h}_j$ .
$\mathbf{v}_{Gl_j}$	Velocidade do baricentro do ligamento $\mathbf{l}_j$ .
$\dot{\mathbf{v}}_{Gl_j}$	Aceleração do baricentro do ligamento $\mathbf{l}_j$ .

### **Capítulo 8: Otimização**

$f(x)$	Função contínua na variável $x$ .
$x^*$	Ponto ótimo local (mínimo ou máximo).
H	Matriz Hessiana.
$\lambda$	Multiplicador de Lagrange.
L	Lagrangeano.
$\xi_{\text{Max}}$	Maior dos eixos de um elipsóide de forças ou de velocidades.
$\xi_{\text{Min}}$	Menor dos eixos de um elipsóide de forças ou de velocidades.

## 1. INTRODUÇÃO

Esta introdução explica os preceitos para o completo entendimento deste trabalho. Serão apresentados os objetivos, as justificativas, os conceitos básicos do âmbito da robótica, as definições e as formas de classificação dos robôs de um modo geral, para mostrar em que meio os robôs de arquitetura paralela estão inseridos, e as diferenças entre as arquiteturas seriadas tradicionais e as paralelas de maneira geral. Há também um breve histórico a respeito da robótica, em especial, da robótica de arquitetura paralela.

### 1.1. Objetivos

Este trabalho tem por objetivos o estudo, a modelagem cinemática e dinâmica, e a otimização de um robô de arquitetura paralela para aplicações industriais.

A aplicação e a arquitetura serão detalhadas mais adiante no Capítulo 4.

### 1.2. Justificativas

Algumas justificativas principais são dadas para sustentar os objetivos:

- Comparativamente com os robôs seriados os de arquitetura paralela possuem vantagens quanto à velocidade de operação, exatidão, rigidez e capacidade de carga, tornando-os ideais em várias aplicações (embora ainda existam muitas questões práticas a serem estudadas e trabalhadas);
- A utilização de robôs de uma forma geral cresceu muito nos últimos anos e ainda está crescendo. Esse tipo de robô é visto em muitos casos como a nova geração em matéria de conceito mecânico, dando uma maior flexibilidade à manufatura atual e resolvendo diversos problemas. Academicamente, a importância de trabalhos com o foco nesses robôs também cresceu;

- Ainda existem muitas questões mecânicas, eletrônicas e de modelagem a serem trabalhadas nesse tipo de arquitetura. Particularmente na questão de otimização ainda são encontrados poucos trabalhos;
- A modelagem e a otimização de uma forma geral é o primeiro passo na construção de um protótipo virtual, necessário atualmente, pois reduz em muito os custos de desenvolvimento de um produto. A modelagem através de protótipos físicos é muito custosa e lenta;
- A investigação cinemática é necessária para a construção de simuladores e algoritmos de controle. A investigação dinâmica é necessária ao projeto mecânico, habilita otimizações mais complexas e é parte fundamental do controle do robô;
- Além de melhorar o desempenho de robôs com esta arquitetura (que pode ser entendido de diversas formas, como maximizar a rigidez ou a velocidade de operação, dentre muitos outros exemplos), a solução oriunda de algoritmos de otimização também pode permitir que sejam diminuídos ou eliminados o efeito de alguns problemas encontrados nessas arquiteturas, como as singularidades. Dada a difícil concepção e interpretação da arquitetura, mapear o volume de trabalho não é uma tarefa tão simples, tornando-se ainda mais complexa quando se exige que a máquina respeite requerimentos mínimos de tamanho do volume de trabalho e / ou desempenho dentro desse espaço.

### 1.3. Definição de um Robô

Existem várias definições de diferentes pesquisadores e estudiosos sobre o que é um robô. Algumas definições são:

*“Um manipulador reprogramável e multifuncional projetado para movimentar materiais, partes, ferramentas ou outros aparelhos especializados através de vários movimentos programados a fim de realizar uma grande variedade de tarefas”.*

Instituto Americano de Robótica (1979)

*“Aparelho automático, com aspecto de boneco, capaz de executar diferentes tarefas, incluindo algumas geralmente feitas pelo homem”.*

Dicionário Michaelis

A definição do Instituto Americano de Robótica é uma das mais claras e aceitas no meio científico e será a definição utilizada aqui. Pode-se então definir um robô de arquitetura paralela. Para esta definição é necessário também o entendimento de alguns conceitos (definições baseadas na obra de **Tsai, 1999**):

- **Cadeia Cinemática:** É um conjunto de ligamentos conectados entre si através de juntas;
- **Cadeia Cinemática Fechada:** É uma cadeia cinemática onde cada ligamento está conectado a pelo menos dois outros;
- **Mecanismo:** Uma cadeia cinemática onde pelo menos um dos ligamentos está preso ao solo ou a um elemento de referência (que pode estar também em movimento).

De acordo com os conceitos apresentados, tem-se que um robô de arquitetura paralela é:

*“Um robô constituído de mecanismos de cadeias cinemáticas fechadas no qual o efetuator é ligado a uma base fixa por diversas cadeias cinemáticas independentes (mínimo de duas). A atuação dos  $n$  graus de liberdade do efetuator se dá através de  $n$  atuadores”.*

Jean-Pierre Merlet (**Merlet, 2000**)

Finalmente, deve-se definir um robô de características totalmente paralelas. Essa definição é particularmente importante porque algumas propriedades e abordagem para métodos aplicam-se apenas a eles (**Merlet, 2000**):

*“Um robô paralelo no qual o número de cadeias cinemáticas é exatamente igual ao número de graus de liberdade do efetuator é chamado de robô totalmente paralelo”.*

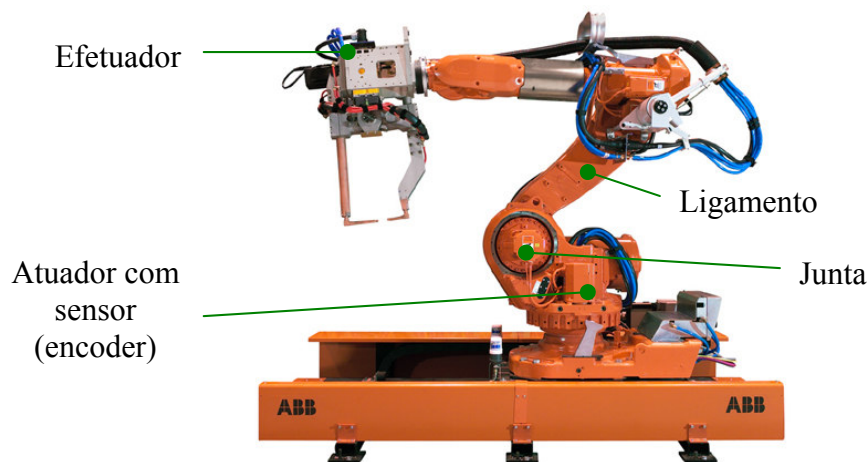
Jean-Pierre Merlet (**Merlet, 2000**)

## 1.4. Elementos Básicos Constituintes de um Robô

Geralmente, uma máquina classificada como um robô possui seis grupos de elementos básicos: Ligamentos, Atuadores, Controladores, Efetuador, Sensores e Juntas. A Fig. 1 apresenta alguns desses elementos.

Frente também à definição do dicionário Michaelis, além de descrever os elementos, pode-se fazer uma analogia com o ser humano:

- *Ligamentos*: seriam os braços e antebraços. São elementos rígidos ligados entre si através de juntas;
- *Atuadores*: São os músculos do robô, dando-lhe a capacidade de se movimentar e levar o Efetuador até o seu destino. Dentre os muitos tipos de atuadores, podem ser citados os motores elétricos, pistões e motores hidráulicos, bem como as formas utilizadas para transferir torque ou força, como cabos, redutores harmônicos, redutores planetários, parafusos sem fim, pinhão e cremalheira, guias lineares e muitos outros;
- *Controlador*: É o cérebro do robô, capaz de receber e de processar os comandos desejados e transcrevê-los em sinais eletrônicos os quais podem ser entendidos pelos atuadores. Além de hardware, os controladores possuem softwares;
- *Efetuator*: É o elemento que realmente executa e dá a função de uso ao robô. Ao contrário do que se pode pensar, o robô apenas transporta o efetuador a algum lugar, contudo é o efetuador que executa o trabalho. Com exceção de robôs arborescentes, geralmente cada robô trabalha com apenas um. Seria equivalente às mãos humanas;
- *Sensores*: Os olhos, os ouvidos e as células de tato do robô. Fornecem os dados necessários para que a máquina perceba a si mesma (*encoders*, por exemplo, que captam a posição dos motores) e / ou ao ambiente, interagindo com ele. Existem obrigatoriamente em robôs com malhas de controle fechadas;
- *Juntas*: Existem dois tipos de juntas (ou articulações) – as passivas e as ativas (ou atuadas). As primeiras existem apenas nos robôs de arquitetura paralela.



**Figura 1:** Destaques de alguns elementos básicos de um robô (Figura de ABB, 2005).

## 1.5. Histórico

### 1.5.1. O Surgimento dos Termos Robô e Robótica

A palavra “robô”, da forma mais próxima a que conhecemos hoje, foi primeiro utilizada pelo escritor tcheco de peças teatrais Karel Capek em sua peça R.U.R. “*Robôs Universais de Rossum*”, que estreou em Praga, em janeiro de 1921. A palavra robô significa “*trabalhador escravo*” ou “*servo*”. A peça foi um sucesso e logo estreou também nos Estados Unidos e na Europa, com seu tema inovador, mostrando a desumanização das pessoas em uma sociedade tecnologicamente avançada. Surpreendentemente, Karel acreditava que os robôs seriam criados através de meios químicos e não mecânicos. Existem algumas evidências também que a palavra na verdade foi uma sugestão de seu irmão Josef, também escritor.

Karel Capek foi indicado ao prêmio Nobel pelos seus trabalhos de enorme influência como escritor. Embora um homem de grande potencial, seus futuros trabalhos serão sempre um mistério, pois Karel morreu em 1938, após ter sido capturado pela Gestapo (polícia secreta alemã) devido a sua postura antinazista (Karel Chapek Website, 2004).

Quanto ao termo robótica, que se refere ao estudo dos robôs, este fora primeiro utilizado pelo cientista americano (nascido russo) Isaac Asimov, conhecido pelos seus trabalhos de ficção científica. O termo foi utilizado em uma de suas

histórias: *Runaround*, publicada em 1942, onde Asimov propôs as chamadas “Leis da Robótica” (três leis e depois foi acrescentada outra lei: a de número zero).

**Lei Zero:** Um robô nunca deve causar mal à humanidade ou, por não agir, permitir que seja causado algum mal à humanidade.

**Lei Um:** Um robô nunca deve ferir um ser humano, ou, por não agir, permitir que seja causado algum mal a um ser humano, a menos que isso viole uma lei de ordem mais alta.

**Lei Dois:** Um robô deve obedecer às ordens dadas a ele por seres humanos, exceto quando essas ordens entrem em conflito com uma lei de ordem mais alta.

**Lei Três:** Um robô deve proteger sua própria existência enquanto tal proteção não entre em conflito com uma lei de ordem mais alta.

### **1.5.2. O Primeiro Robô Industrial**

Após a explosão tecnológica durante a Segunda Guerra Mundial, em 1956 um encontro histórico ocorreu onde se reuniram George C. Devol, um empreendedor e bem sucedido inventor, e um engenheiro chamado Joseph F. Engelberger, onde acabaram discutindo os trabalhos sobre robôs de Asimov.

Os dois então uniram forças para desenvolver um robô real que pudesse ser empregado em diversos trabalhos. Engelberger então abriu a empresa de manufatura *Unimation*, que se dedicava à automação. O primeiro robô desenvolvido, o *Unimate* (Fig. 2), que foi instalado numa fábrica da General Motors, garantiu a Engelberger o título de “Pai da Robótica”.





**Figura 2:** O Unimate - primeiro robô industrial (RRG, 2005).

### **1.5.3. Os Robôs de Arquitetura Paralela**

O primeiro robô deste tipo surgiu como uma solução eficiente para o problema do “*movimento esférico*”, tratado na competição *Le Prix Vaillant* ocorrida na França no início da década de 1900 e organizada pela *Académie des Sciences*. O problema era determinar sobre que condições um corpo rígido pode demonstrar a capacidade de executar movimentos contínuos quando alguns de seus pontos são obrigados a permanecer sobre esferas fixas. O prêmio foi dividido por dois participantes: **Borel (1908)** e **Bricard (1906)**. Apenas depois de algumas décadas, Gough estabeleceu os princípios básicos para um mecanismo de estrutura de cadeias cinemáticas fechadas e construiu um protótipo (**Gough, 1956 e 1962**). Seu primeiro propósito era utilizar essa máquina para teste de pneus de automóveis.

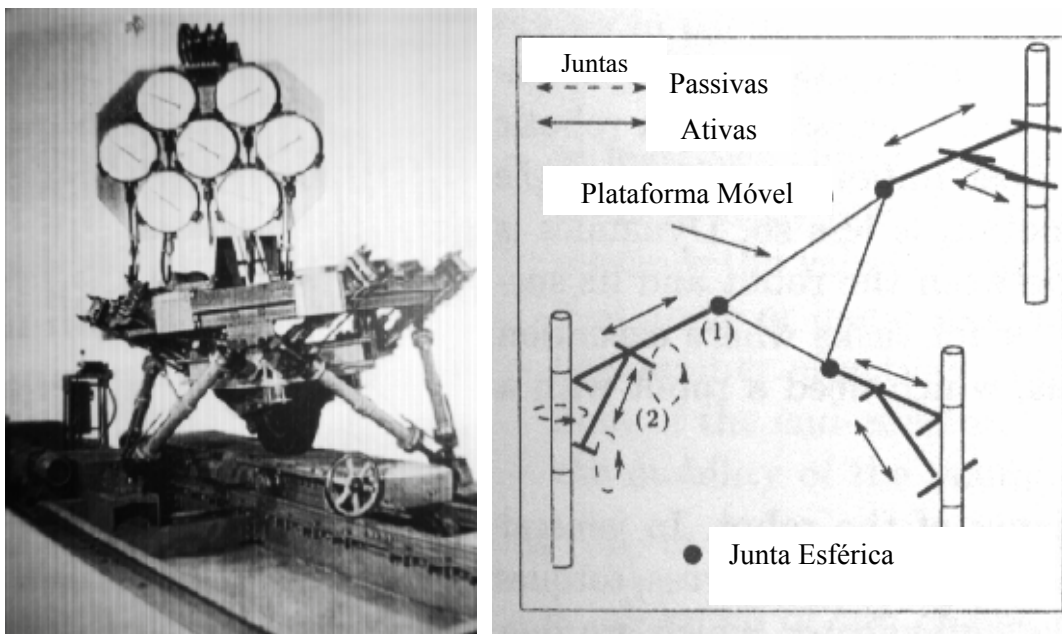
O mecanismo de Gough apresentou um novo conceito ao mundo científico da época com imensas vantagens sobre os mecanismos seriais, os únicos conhecidos até o momento. Nesse mecanismo, uma base hexagonal era ligada através de articulações esféricas a hastes dotadas da capacidade de alterar seu comprimento e cada uma dessas hastes eram por sua vez ligadas a uma junta universal presa ao solo, conforme mostra a Fig. 3.

Na década de 1960, com o crescimento da indústria aeronáutica, o custo de treinamento de pilotos e a necessidade de se testar novos equipamentos sem a

necessidade de estar em voo forçaram os pesquisadores da época a estudar exaustivamente alternativas de mecanismos para se construir simuladores com seis graus de liberdade. Os manipuladores seriais conhecidos eram extremamente limitados para essa tarefa, pois quando a massa dos manipuladores aumenta, as forças dinâmicas envolvidas aumentam consideravelmente e a carga que os manipuladores deveriam suportar era alta: uma carlinga (em inglês, *cockpit*) inteira.

**Stewart (1965)** sugeriu um novo mecanismo para o problema, um mecanismo paralelo de seis graus de liberdade, com uma arquitetura bastante exótica. O modelo era formado por uma estrutura triangular, onde cada vértice era ligado a uma seqüência de dois mecanismos, cada um sendo dotado de um grau de liberdade linear (o esquema do mecanismo aparece na Fig. 3).

Curiosamente, o modelo de Gough é conhecido como “*Plataforma de Stewart*”, enquanto que o modelo de Stewart ainda não possui uso (**Merlet, 2000**). O impacto da máquina de Gough na área de simuladores foi tão intenso que inúmeros modelos utilizam até hoje essa arquitetura.



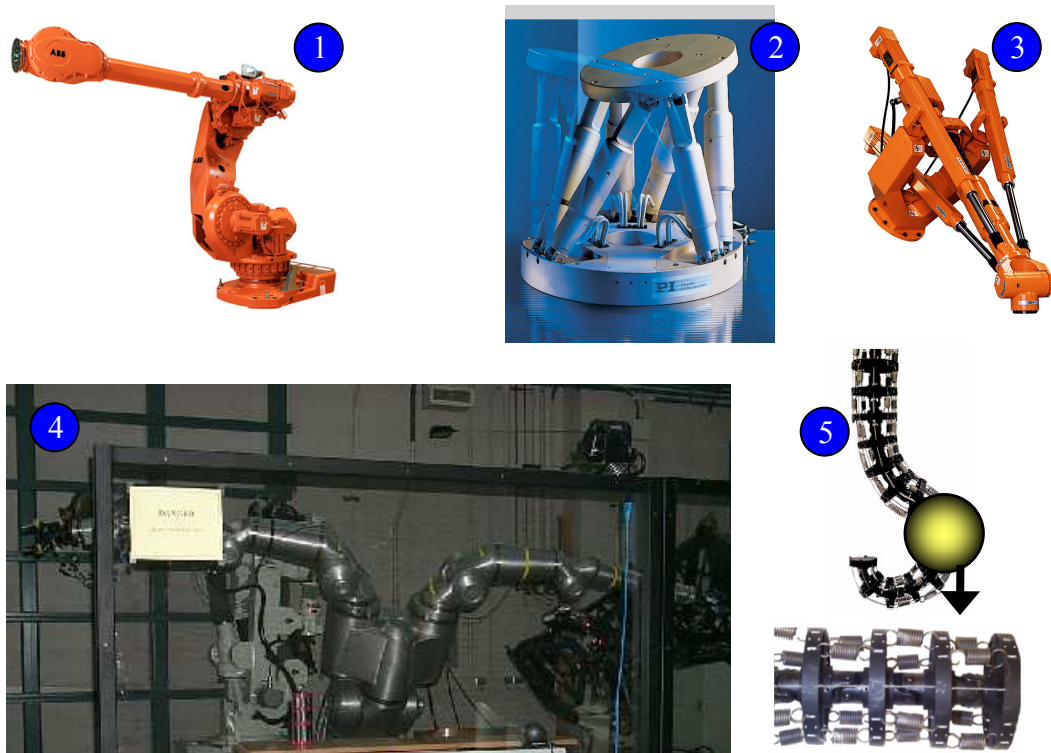
**Figura 3:** O protótipo de Gough (à esquerda) e o modelo de Stewart (à direita) – Figuras de **Merlet (2000)**.

## 1.6. Classificação dos Robôs

Existem várias formas e particularidades com o que se pode classificar um robô, como a forma de atuação, tipo do controlador e muitas outras. Contudo, neste trabalho, vai-se ater aos itens que classificam geometricamente os robôs de forma geral. É apresentado também um tipo de classificação específica para robôs de arquitetura seriada, baseado nos três primeiros graus de liberdade, e outra específica da arquitetura paralela que é uma classificação bastante conhecida e aceita.

No tocante ao tipo de cadeias cinemáticas, os robôs podem ser de um dos cinco tipos descritos a seguir e exemplificados na Fig. 4 (robôs móveis não serão abordados). Dentro dos parênteses está o número do robô correspondente na figura:

- *Seriados (1)*: Possuem uma única cadeia cinemática aberta onde uma das pontas é fixa e a outra carrega o efetuador. É o tipo mais conhecido de robôs;
- *Paralelos (2)*: Possuem várias cadeias cinemáticas e seus ligamentos (ou conjunto de ligamentos) possuem as primeiras extremidades ligadas a uma base fixa e as outras a uma plataforma móvel que carrega o efetuador;
- *Híbridos (3)*: São uma combinação dos dois primeiros, tendo alguns graus de liberdade baseados em arquitetura seriada e outros em arquitetura paralela;
- *Arborescentes (4)*: São robôs cujas cadeias cinemáticas sempre compartilham parte da estrutura (alguns ligamentos), lembrando um tronco de uma árvore e seus galhos. Carregam mais de um efetuador;
- *Hiper - redundantes (5)*: São robôs seriados de muitos graus de liberdade (o número de graus de liberdade é superior ao número de eixos onde ocorrerá movimento). São quase sempre semelhantes a uma tromba.



**Figura 4:** Robôs segundo o tipo e forma de suas cadeias cinemáticas. As figuras dos robôs foram extraídas do site da ABB – ABB, 2005 (1 e 3), do PI, 2005 (2), do RRG, 2005 (4), de Hannan e Walker, 1999 (5).

No caso dos robôs seriados, os três primeiros graus de liberdade costumam ser os responsáveis por definir a forma do volume de trabalho e, por conseguinte, por definir as funções as quais o robô será mais apto. Os tipos possíveis mais conhecidos são (a Fig. 5 ilustra os casos citados):

- *Robô cilíndrico:* feito para operar principalmente em aplicações que se beneficiem do uso de coordenadas cilíndricas e de um volume de trabalho com tal formato;
- *Robô esférico ou polar:* feito para operar principalmente em aplicações que se beneficiem do uso de coordenadas esféricas e de um volume de trabalho com tal formato;
- *Robô cartesiano ou retangular:* feito para operar principalmente em aplicações que se beneficiem do uso de coordenadas cartesianas e de um volume de trabalho com o formato de um cubo ou de um paralelepípedo;
- *Robô SCARA (Selective Compliance Assembly Robot Arm):* os motores deste robô não precisam suportar seus próprios pesos ou o peso dos ligamentos

seguintes como os outros desta lista, já que todo o efeito oriundo da ação de forças gravitacionais é resistido pela própria estrutura do robô (ligamentos e base fixa). São extremamente rápidos e voltados a operações *pick-and-place* e de montagem, embora ainda devam compensar a inércia dos motores que se movimentam com a estrutura;

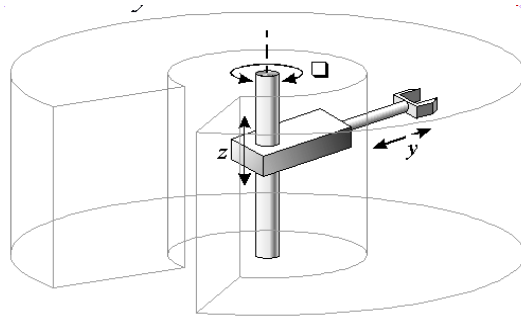
- *Robô articulado*: é o mais comumente visto na literatura, tentando imitar um braço humano. Geralmente são dotados de um mecanismo atuado de punho esférico na ponta que dá a orientação ao efetuador. Dessa forma, passam a ter seis graus de liberdade.

Especificamente para robôs de arquitetura paralela, existe um tipo de classificação bastante aceita que utiliza um conjunto de letras para representar a seqüência de articulações de um ligamento ou conjunto de ligamentos desde a base fixa até a base móvel, sublinhando as articulações atuadas. Para esta classificação, vale a premissa que todas as cadeias cinemáticas são idênticas entre si. O seguinte código de letras costuma ser utilizado na representação das articulações:

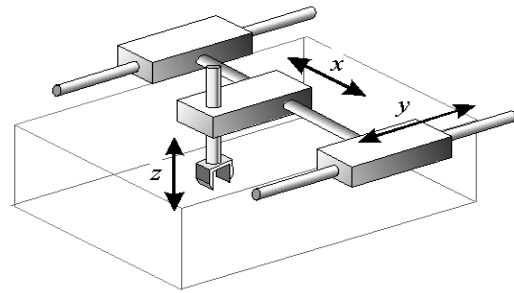
- (R) *Revolute* – Articulação de revolução;
- (P) *Prismatic* – Articulação prismática como, por exemplo, um pistão;
- (U) *Universal* – Articulação do tipo junta universal;
- (S) *Spherical* – Articulação esférica ou punho esférico não atuado;
- (Pa) *Parallelogram* – Articulação do tipo paralelogramo;
- (H) *Helical* – Articulação helicoidal.

Como exemplo, pode-se tomar o robô de Gough (Fig. 3). De acordo com a descrição da máquina, tem-se, partindo do solo até a plataforma móvel, uma junta universal, uma prismática atuada e uma esférica. Assim, trata-se de um robô UPS.

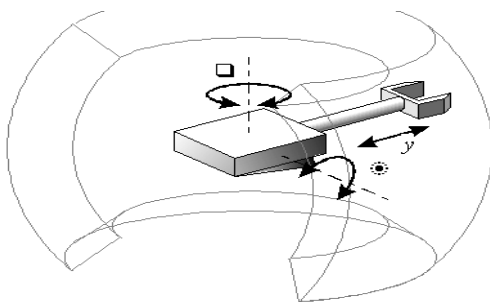
A classificação também vale para robôs com cadeias cinemáticas diferentes, mas nesse caso todas as cadeias devem ser mostradas. Por exemplo, o robô de seis graus de liberdade de **Zlatanov et al. (1992)**, apresentado por **Merlet (2000)**, é do tipo (RRR), (RRR), (RRR).



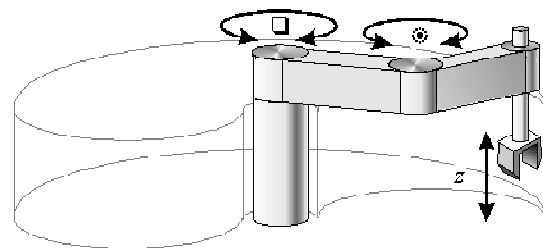
1) Robô Cilíndrico



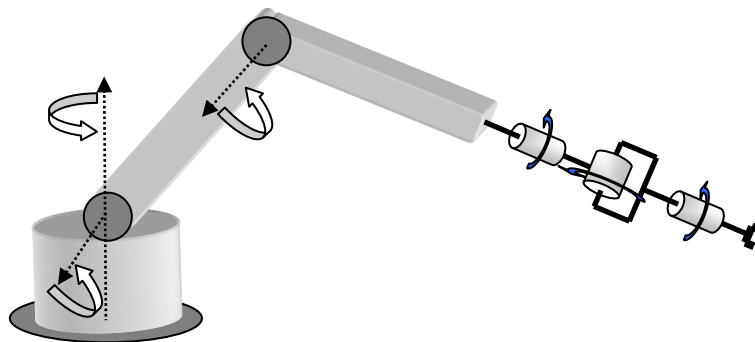
2) Robô Cartesiano



3) Robô Esférico



4) Robô SCARA



5) Robô Articulado com Punho Esférico

**Figura 5:** Classificação dos robôs seriados quanto aos três primeiros graus de liberdade (As figuras de 1 a 4 são de METU, 2005, notas de aula de ME445 – Integrated Manufacturing Systems). A figura do robô articulado foi criada pelo próprio autor desta dissertação.

## 1.7. O Panorama Atual dos Robôs (Acadêmico e de Mercado)

Ao contrário do que se pensa por ser um tema relativamente antigo, robôs ainda são responsáveis por muitos estudos. Um dos motivos é sua crescente utilização na indústria. Uma pesquisa da IFR (*International Federation of Robotics*), apresentada por **Krefft (2002)** em seu trabalho, mostra que existe uma tendência de crescimento da ordem de 15% ao ano na utilização de robôs pela indústria.

Segundo a RIA (*Robotic Industries Association*), em agosto de 2005 os fornecedores de robôs norte americanos puderam constatar um aumento de pedidos na ordem de 36%, colocando a indústria no rumo de um ano recorde de vendas.

As empresas Norte Americanas no período de janeiro a junho de 2005 encomendaram um total de 10.712 robôs avaliados em US\$ 638,9 milhões. Incluindo as exportações, o número chega a 11.381 robôs e US\$ 673,1 milhões. Segundo o Vice – Presidente executivo da RIA, Donald A. Vincent, o segundo quartil de 2005 foi o melhor da história, tomando como base as referências da associação, que acompanha o mercado desde 1983 e conta com cerca de 90% das empresas fabricantes e distribuidoras de robôs, associadas ao órgão (235 empresas no total).

A RIA estima que cerca de 150 mil robôs estejam em funcionamento apenas nos EUA, que perde em número apenas para o Japão (**Robotics Online, 2005**).

No meio acadêmico, particularmente para os robôs de arquitetura paralela, **Sima'an (1998)** menciona que a representatividade das publicações de robôs desse tipo com relação ao número de publicações sobre robôs de uma forma geral aumentou. Em 1985, trabalhos sobre robôs de arquitetura paralela representavam 1% de todos os trabalhos sobre robôs. Em 1996, o percentual era de 9%.

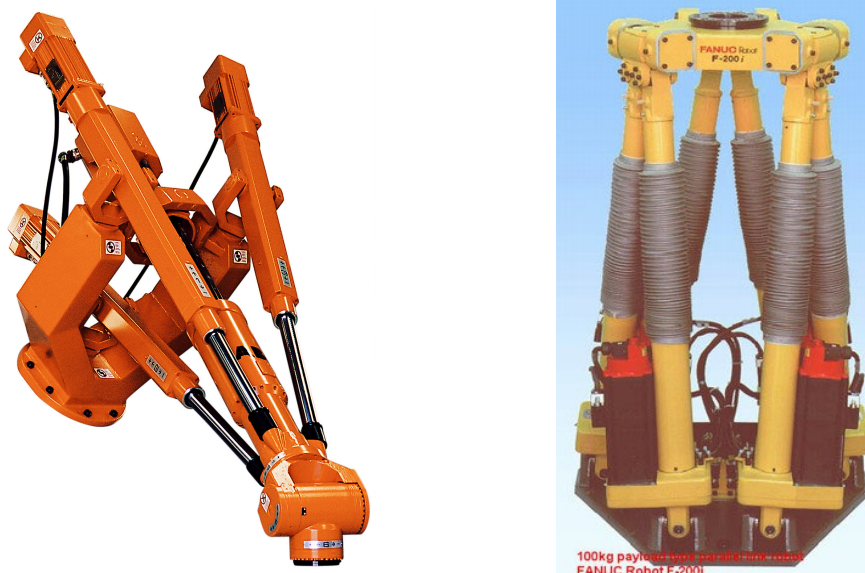
**Leal (2005)**, em sua dissertação de mestrado, realiza uma análise bastante completa do mercado mundial de robôs e também relata a falta de dados precisos e confiáveis para o mercado brasileiro, além da existência de fontes contraditórias. Uma das fontes que apresenta relata que em 2001 deveriam existir pouco menos de oito mil robôs em operação no Brasil, com um crescimento anual bastante acentuado. Segundo o autor, o primeiro robô instalado no Brasil foi em 1982 na Volkswagen. **Leal (2005)** ainda relata que não existem fabricantes nacionais de robôs industriais. Os três maiores fornecedores para o Brasil são a ABB, a FANUC e a KUKA.

## 1.8. Algumas Aplicações dos Robôs de Arquitetura Paralela

Desde 1965, com a utilização destes robôs em simuladores de vôo por Stewart, que o interesse na arquitetura em diversas áreas aumenta. Atualmente eles vêm sendo utilizados não apenas na indústria, mas também na medicina, em simuladores para entretenimento ou treinamento, em orientação de antenas, em micromáquinas e em muitas outras aplicações. As figuras de 6 a 11 mostram alguns dos robôs de arquitetura paralela que existem atualmente.

### • Robôs Industriais para Aplicações Genéricas

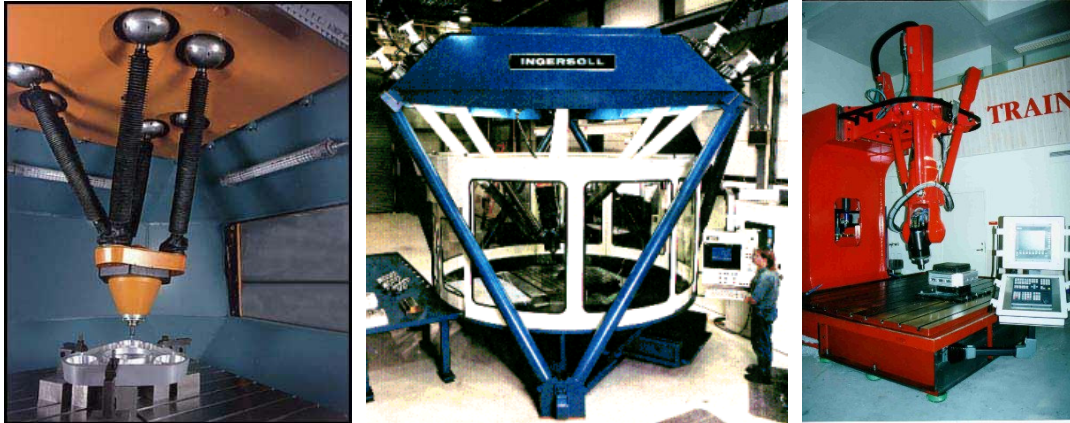
Dois dos maiores fabricantes de robôs do mundo, a ABB e a FANUC, apostam nas arquiteturas paralelas e já possuem alguns produtos para aplicações industriais genéricas: o IRB 940 Tricept (da ABB), com três graus de liberdade paralelos e três seriados (arquitetura híbrida), e o F-200i (da FANUC), de arquitetura Hexapod (similar ao modelo de Gough da Fig. 3), com capacidade de carga de 100 kg, são dois exemplos de produtos já disponíveis.



**Figura 6:** À esquerda, o IRB 940 Tricept da ABB; à direita, o F-200i da FANUC (fotos da ABB, 2005 e da FANUC, 2005, respectivamente).



- Máquinas de Usinagem



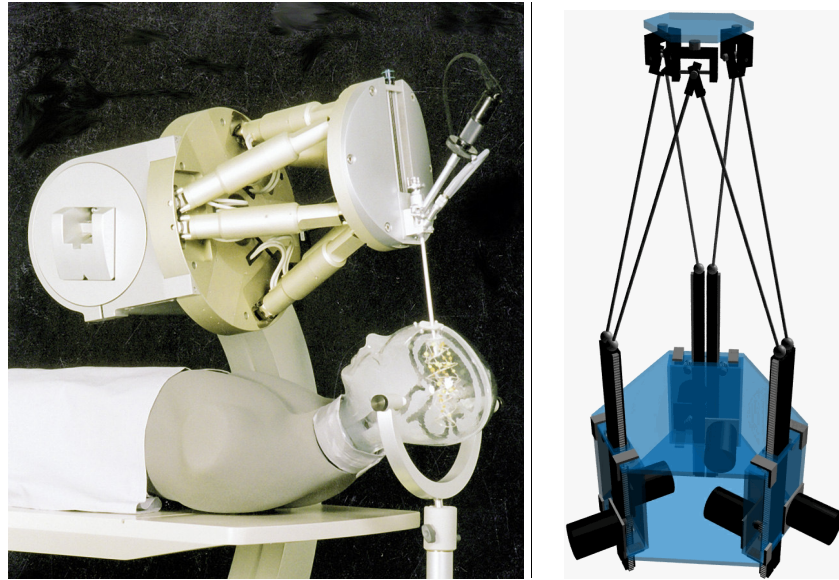
**Figura 7:** Da esquerda para a direita: uma máquina ferramenta baseada na arquitetura Hexapod (Stankóczy, 1999); a máquina-ferramenta Hexapod da Ingersoll (Ingersoll, 2004) e o Tricept do Neos Robotics (Robotics Online, 2005)

- Máquinas para Operações de *Pick – and – Place*



**Figura 8:** À esquerda, o FlexPicker™ da ABB (ABB, 2005) e à direita a máquina da SIG (Robotics Online, 2005), ambas arquiteturas Delta, inventada por Raymond Clavel.

- Robôs Cirúrgicos



**Figura 9:** À esquerda, o Hexapod M-850 do IPA, que pode carregar diversos instrumentos (possui repetibilidade de 1-2  $\mu\text{m}$  e resolução dos atuadores de 5 nm); à direita, um modelo para cirurgias cardíacas (fotos, respectivamente, de **PI, 2005** e **Lazarevic, 2000**).

- Simuladores



**Figura 10:** Simulador de vôo do Boeing 737, visão interior (à esquerda) e visão exterior (no centro) – **Ansett Flight Simulator Centre, 2004**. À direita um simulador para bicicletas para estudo de vibrações – Laboratório de Controle de Vibrações do **KAIST (2004)**.

## 1.9. Conceitos Básicos para este Trabalho

Neste item são descritos os conceitos básicos utilizados neste trabalho. Alguns outros conceitos serão descritos ao longo do estudo quando se mostrarem necessários.

- *Cinemática*: trata dos aspectos geométricos e temporais relativos aos movimentos sem se preocupar com torques ou forças. A cinemática trata da posição, velocidade, aceleração e outras derivadas de mais alta ordem da posição com respeito ao tempo ou outra variável;
- *Análise Cinemática*: trata da dedução ou obtenção das equações de movimentos relativos entre os vários corpos rígidos de um manipulador. Existem dois tipos de análise cinemática: a inversa e a direta. Quando se programa um robô, geralmente são indicados pontos no espaço para onde o efetuador deve ir, para que execute uma função. Nesse caso tem-se a posição do efetuador, mas não a posição dos motores, o que é uma questão da cinemática inversa ou cinemática “para trás”. Contudo, em outra situação, pode-se desejar a posição do efetuador lendo a posição dos atuadores através de sensores (como *encoders*). Esse então é um problema tratado pela cinemática direta ou cinemática “para frente”;
- *Síntese Cinemática*: trata-se do inverso da análise cinemática. Síntese, de uma forma geral, é o novo grande desafio da robótica industrial, onde um projetista deve encontrar o manipulador que possua certas características cinemáticas dadas;
- *Estática*: trata das questões relativas ao equilíbrio de forças, o que depende da posição e configuração do robô, ou a sua *postura* (combinação da posição e orientação do efetuador e dos ligamentos). A estática não lida com as forças oriundas do movimento como as forças inerciais;
- *Dinâmica*: trata das forças e dos torques requeridos para gerar um movimento num sistema de corpos, tendo como foco o efeito das forças inerciais;
- *Análise Dinâmica*: trata da dedução das equações de movimento do manipulador. Analogamente à análise cinemática, existem dois tipos de análise dinâmica: a inversa e a direta. Na primeira, tem-se a trajetória de

movimento do efetuador como função do tempo e deseja-se encontrar as forças nos atuadores que levaram o efetuador a executar essa trajetória com essas características de movimento. Na segunda, tem-se o oposto: dadas as forças nos atuadores, deseja-se encontrar o efeito no efetuador. Os modelos de análise da dinâmica direta são utilizados majoritariamente em simuladores e em testes. Já os modelos para a dinâmica inversa são utilizados em tempo real em diversos robôs para a obtenção de algoritmos de controle eficientes;

- *Síntese Dinâmica*: é o inverso da análise dinâmica. É um problema extremamente complexo e representa hoje o estado da arte quanto à concepção mecânica de robôs;
- *Otimização*: trata da busca não exaustiva do ponto ótimo de uma função, ou do ponto onde há o melhor desempenho, respeitando algumas restrições impostas. As buscas são não exaustivas porque não visam varrer todo o espaço possível de soluções e sim buscar as tendências na função que levam a um ponto melhor ou a um ótimo. As funções podem ser as mais variadas: relação rigidez / peso, relação torque no efetuador / torque nos motores, relação tamanho da máquina / tamanho do volume de trabalho, dentre muitas. Algoritmos de otimização e de síntese estão intrinsecamente ligados permitindo novos saltos na engenharia, na medida em que habilitam soluções melhores para casos extremamente complexos;
- *Singularidade*: é uma posição onde ocorre um problema na cinemática do robô, seja na cinemática inversa ou direta, seja por uma limitação física (por exemplo, limite do volume de trabalho) ou no algoritmo (múltiplas soluções possíveis);
- *Volume de trabalho*: o espaço alcançável pelo efetuador. Pode ser classificado de diversas maneiras e está intrinsecamente ligado à postura do robô.

## 1.10. Comparação entre robôs de Arquitetura Seriada e Arquitetura Paralela

Existem muitas diferenças entre os robôs de arquitetura seriada e paralela, além das diferenças geométricas óbvias.

Essas diferenças podem ser ilustradas conforme apresentado na Tabela 1, que utilizou como base o trabalho de **D'Souza (2003)**. É importante lembrar que as comparações são para robôs com o mesmo número de graus de liberdade e mesma qualidade de elementos (atuadores, articulações, outros).

Item	Arquitetura Seriada	Arquitetura Paralela
Volume de Trabalho	<b>Grande</b>	Pequeno
Capacidade de Carga	Baixa	<b>Alta</b>
Exatidão do Efetuador	Baixa	<b>Alta</b>
Juntas Passivas	<b>Não tem</b>	Tem
Cinem. / Din. Inversa	Complexa, múltiplas soluções	<b>Simples, múltiplas soluções*</b>
Cinem. / Din. Direta	<b>Simples, solução única</b>	Complexa, múltiplas soluções
Aplicações Típicas	Soldagem, montagem, pintura, pick-and-place, manuseio de materiais	Simuladores, montagem, usinagem, posicionamento de precisão, pick-and-place

**Tabela 1:** Comparação entre robôs de arquitetura seriada e arquitetura paralela.

**Volume de Trabalho:** Nos robôs de arquitetura paralela todas as cadeias cinemáticas trabalham em paralelo para executar um movimento qualquer do efetuador. Assim, percebe-se claramente que o volume de trabalho fica bastante reduzido se comparado ao de uma arquitetura seriada onde o movimento de cada atuador em particular contribui individualmente para o movimento do efetuador.

**Capacidade de Carga:** Os robôs seriais clássicos apresentam vários problemas como uma baixa rigidez e o fato de que cada segmento deve suportar o peso dos próximos segmentos, implicando numa baixa relação *carga transportada /*

---

\* Em geral a solução é bastante simples, mas algumas arquiteturas a deixam razoavelmente complexa.

*peso do robô*. Mesmo quando comparados aos robôs SCARA, que possuem um formato tal que os segmentos não precisam suportar o peso dos próximos, os robôs de arquitetura paralela ainda levam vantagem (Merlet, 2000).

**Exatidão:** Tendo-se como premissa que a estrutura mecânica dos robôs é infinitamente rígida, pode-se dizer que nos robôs de arquitetura seriada a exatidão é a soma das curvas de exatidão do efeito de cada uma das juntas atuadas ao término de seu respectivo ligamento. Já nos robôs de arquitetura paralela, essa exatidão se aproxima daquela do elemento de pior exatidão. No entanto, para que se concretize essa verdade teórica são necessários ainda trabalhos para calibragem dos protótipos.

**Cinemática e Dinâmica Direta:** Nos robôs de arquitetura seriada essa é uma tarefa geralmente simples de se realizar, com uma única **solução**, já que definindo-se a posição, velocidade e aceleração de cada atuador pode-se definir todos os dados do efetuador, compondo o efeito de cada um independentemente. No caso dos robôs de arquitetura paralela, tem-se uma tarefa muito complexa, onde poucos modelos apresentam solução analítica, sendo necessários métodos numéricos ainda custosos computacionalmente para encontrar as soluções possíveis.

**Cinemática e Dinâmica Inversa:** Nos robôs de arquitetura paralela, essa é uma tarefa bem mais simples comparativamente com os robôs seriados. Todavia, o cálculo ainda pode ser complexo dependendo da arquitetura em estudo e pode demandar de algum algoritmo de decisão para a posição do atuador no caso de, matematicamente, as equações possuírem mais de uma solução.

As acelerações possíveis, a capacidade de carga, a rigidez e a exatidão elevadas garantem aos robôs de arquitetura paralela seu espaço em diversas aplicações conforme apresentado anteriormente. Todavia, as limitações no volume de trabalho e as dificuldades de projeto e calibração os impedem de ter uma aceitação mais rápida e mais intensa no mercado.

## 2. REVISÃO BIBLIOGRÁFICA

Os principais pontos de estudo e revisão da literatura para o escopo deste trabalho serão a dinâmica, a cinemática (direta e inversa), o cálculo do volume de trabalho e a otimização de modelos. Controle e outros tópicos são mencionados, mas apenas superficialmente, já que fogem ao escopo.

### 2.1. Cinemática

A cinemática de mecanismos paralelos é fundamentalmente diferente da cinemática de mecanismos seriados. Nos mecanismos seriados cada ligamento carrega todas as forças dos ligamentos que se seguem e é possível utilizar a sobreposição de efeitos para cálculos de movimentos. Geralmente também é muito fácil de evitar pontos de singularidade (**Knapp e Cobet, 2000**).

No caso dos mecanismos seriados, geralmente utilizam-se métodos já conhecidos e bem sedimentados, como o de Denavit-Hartenberg. Todavia, para os mecanismos paralelos, tal afirmação não pode ser feita. Devido à complexidade de suas arquiteturas, surgiram e ainda surgem vários trabalhos buscando maior eficiência computacional ou uma metodologia mais rica – principalmente na cinemática direta (**Tsai, 1999**).

Cada arquitetura paralela possui características cinemáticas particulares, dificultando a aplicação de equações generalistas. Obter as equações cinemáticas de mecanismos paralelos requer a análise das cadeias de forma a determinar as equações de cada uma delas, relacionando os ligamentos com a plataforma móvel e a base fixa.

A cinemática inversa é uma tarefa essencial para o controle de posição de qualquer robô e para a construção de simuladores. Ela consiste em calcular as coordenadas das articulações do robô correspondentes a uma dada configuração específica do efetuador (posição e orientação). A cinemática direta consiste em determinar a configuração do efetuador a partir da posição dos atuadores. Ambas as cinemáticas, matematicamente, possuem múltiplas soluções, todavia a direta é muito mais complexa para os mecanismos de arquitetura paralela.

### **2.1.1. Cinemática Inversa**

Felizmente, conforme percebido por diversos cientistas, a análise da cinemática inversa de grande parte dos mecanismos de arquitetura paralela é uma tarefa simples e pode ser realizada por observação geométrica. **Tsai (1999)** apresenta um método de resolução do problema através de uma forma muito eficiente utilizando equações vetoriais da geometria da arquitetura e o aplica, dentre outros modelos, à plataforma de Stewart. Para esta arquitetura em particular as equações já são obtidas diretamente, já que o atuador relaciona a base fixa e a plataforma móvel através de uma expressão geométrica de distância entre dois pontos.

**Hao et al. (1996)** utilizaram um método diferente para solucionar a cinemática inversa da plataforma de Stewart, construindo um modelo equivalente na forma de um robô de arquitetura seriada para então resolver o sistema. Em especial, a resolução do problema desta forma facilitou a obtenção do volume de trabalho.

**Bruzzone et al. (2002)** resolveram a cinemática inversa do seu robô Delta para aplicações de corte com laser, baseados no trabalho do inventor do modelo (Reymond Clavel), e também não obtiveram problemas ou dificuldades.

### **2.1.2. Cinemática Direta**

O cálculo da cinemática direta é particularmente complexo para robôs paralelos, principalmente para os de seis graus de liberdade. Neste aspecto tem-se o oposto das máquinas seriais, onde essa atividade é muito fácil de ser executada. Para os robôs de arquitetura paralela, as equações formam geralmente sistemas não lineares com variáveis acopladas e resultam em múltiplas soluções. Felizmente, devido à sua complexidade, a tarefa de se realizar a cinemática direta em robôs paralelos tem sido motivo de muitos trabalhos, facilmente encontrados na Internet, periódicos e livros.

Segundo **Merlet (2000)**, poucos mecanismos possuem uma formulação explícita, como no caso do manipulador mencionado por Dafaoui (**Dafaoui et. al., 1994**), ou o mecanismo de Rometi estudado por **Fioretti (1994)**.



**Tsai (1999)** apresenta um método de solução genérico, aplicando-o a algumas arquiteturas como a plataforma de Stewart. Ele muda o centro de coordenadas tanto da base fixa quanto da plataforma móvel e utiliza artifícios matemáticos para eliminar algumas das variáveis não-lineares que aparecem.

**Zhang e Song (1991)** estudaram a cinemática direta de robôs de seis graus de liberdade, mais especificamente, variantes da plataforma de Stewart. Eles perceberam que em arquiteturas onde é possível desacoplar um dos graus de liberdade de rotação dos outros cinco, é possível obter equações mais simples e construir algoritmos muito eficientes. Geometricamente, esta condição foi satisfeita deixando-se cinco dos seis pontos de apoio dos atuadores na plataforma móvel numa única linha reta. Uma das possíveis variantes foi comparada a uma plataforma de Stewart e em termos computacionais o algoritmo de solução foi cerca de 40 vezes mais eficiente.

**Liu et al. (1993)** estudaram a cinemática direta de uma plataforma de Stewart analiticamente e propuseram um método aonde através de relações geométricas e de várias passagens algébricas chega-se a um sistema de apenas três equações não-lineares. Trabalhos anteriores mencionados pelos autores conseguem o feito com um sistema de 30 equações não lineares ou um polinômio de alta ordem ( $24^a$ ). Para resolver o sistema de três equações, é utilizado um algoritmo baseado no método de Newton-Raphson.

Frente às dificuldades de se calcular analiticamente a cinemática direta, Maurine (**Maurine et al., 1999**) utiliza um algoritmo iterativo com a matriz Jacobiano inversa no cálculo da cinemática direta. **Nguyen et al. (1991)** já havia feito um estudo semelhante ao de Maurine, preparando as equações da cinemática inversa para que pudessem ser utilizadas num algoritmo iterativo baseado no método de Newton-Raphson para obter a solução da cinemática direta.

Han, Chung e Youm (**Han et al., 19XX**) estudaram robôs de arquitetura paralela do tipo SPM (*Stewart like Parallel Manipulator*) e sugeriram a utilização de sensores extras. Assim, utilizando-se os conceitos de “*Local Structurization*” e “*Mechanism Partition*” apenas uma solução foi encontrada. O primeiro conceito refere-se a quando uma parte do manipulador pode ser resolvida cinematicamente com respeito às coordenadas da base se uma entrada ou algum outro dado cinemático

de um ponto ou outra informação relevante for fornecido. O segundo é um procedimento de decomposição do mecanismo original em componentes menores (como ligamentos, plataforma, etc.).

Na mesma linha, **Bonev e Ryu (1999)** apresentam um método para solução da cinemática direta da plataforma de Stewart utilizando três sensores extras (além dos seis sensores padrão de cada pistão). Utilizando tal abordagem, o sistema de equações resultantes se torna bem mais simples de se resolver e os pesquisadores apresentam alguns exemplos numéricos aleatórios e demonstram a existência de apenas uma única solução.

Também existem abordagens de solução através de outros algoritmos, como redes neurais. Um exemplo é o trabalho de **Geng e Haynes (1991)** onde utilizam e comparam diferentes tipos de redes neurais na solução da cinemática direta de plataformas de Stewart.

Conforme mencionado por **Merlet (2000)**, conseguem-se encontrar todas as soluções da cinemática direta para um dado robô, mas o problema é saber em tempo real qual dessas soluções representa a posição em que o robô se encontra, pois a solução fornece todas as configurações possíveis da plataforma móvel. Sem um método rápido é impossível se programar um sistema de controle que utilize a cinemática direta.

Frente às dificuldades apresentadas, **Merlet (2000)** sugere que atualmente apenas duas abordagens são viáveis para se programar algoritmos da cinemática direta em tempo real:

- Utilizar sensores para detectar a solução correta da máquina, contudo, existem divergências quanto ao tipo e à disposição dos sensores, bem quanto à exatidão exigida deles;
- Utilizar métodos numéricos iterativos programados para funcionar em computadores extremamente poderosos com processamento paralelo, próprios para solução dos algoritmos da cinemática direta (este último realizado por **Knapp e Cobet, 2000**).

### 2.1.3. Matrizes Jacobiano e Singularidades

Quando o determinante da matriz Jacobiano é zero, o sistema de equações que transforma as forças aplicadas na plataforma móvel em forças nos atuadores não pode mais ser resolvido. Isso implica no fato de que ou a rigidez da plataforma se torna zero em alguma direção ou a tensão em um dos ligamentos é infinita (**Knapp e Cobet, 2000**).

**Gosselin e Angeles (1990)** descobriram que a matriz Jacobiano pode ser separada em duas: uma associada com a cinemática direta e outra com a inversa. Utilizando-se essa descoberta, pode-se dizer se a posição de singularidade é da cinemática direta, da inversa ou de ambas. Exemplos e aplicações são tratados detalhadamente em **Tsai (1999)**.

**Kim et al. (1998)** realizam um estudo analítico das posições de singularidade em robôs de arquitetura semelhante à plataforma de Stewart com o uso de sensores extras. Singularidades previamente descobertas por outros estudiosos como Merlet e Hunt são então apresentadas como fruto da aplicação do método.

Yang e Chen (**Yang e Chen, 2001**) propõem um método de detecção de singularidades através do estudo da velocidade instantânea das juntas passivas (não-atuadas) focando em robôs de seis graus de liberdade, mas com apenas três “pernas”. Caso as velocidades instantâneas das juntas passivas e da plataforma móvel possam ser determinadas, então não existe singularidade na cinemática direta. Complementando o raciocínio, se as velocidades instantâneas das juntas passivas e atuadas podem ser determinadas a partir da velocidade da plataforma móvel, então não existem singularidades na cinemática inversa. Sendo assim, apenas algumas matrizes 3 x 3 devem ser analisadas ao invés das matrizes Jacobiano 6 x 6.

**Chan e Ebert-Uphoff (2001)** apresentam um estudo sobre a combinação de velocidades lineares e angulares no efetuador para identificar posições de singularidade perigosas que podem fazer com que o robô passe para uma nova posição singular. Além disso, seu estudo sugere o uso de atuadores redundantes para evitar singularidades. O trabalho apresenta um estudo de um robô paralelo de três graus de liberdade, formando um mapa das direções de singularidades com a forma de uma espiral.

A abordagem de motores extras é uma opção bem sucedida que já foi utilizada na construção da máquina Eclipse pela Universidade Nacional de Seul (**Ryu et al., 1998**). Isso é, na verdade, o oposto do trabalho de **Knapp e Cobet (2000)** que otimizaram a máquina (no caso a Hexaglide, uma máquina de seis graus de liberdade que utiliza guias lineares paralelas como forma de atuação) para que seu volume de trabalho ficasse isento de singularidades.

## 2.2. Dinâmica

Há três abordagens tradicionais utilizadas no cálculo da dinâmica dos robôs de arquitetura paralela: formulação de Newton-Euler, de Lagrange e de D'Alembert (ou princípio do trabalho virtual). A primeira requer que as equações de movimento sejam escritas para cada um dos corpos rígidos que compõem a máquina, resultando num modelo muito custoso computacionalmente (**Tsai, 1999**). A abordagem de Lagrange permite que os modelos sejam mais eficientes, mas devido à complexidade dos robôs, a formulação é difícil (**Tsai, 1999**). Para a abordagem de D'Alembert, **Tsai (1998 e 1999)** lembra que parece ser a mais eficiente e de melhor relação custo (computacional) / benefício (informações obtidas).

**Dasguta e Mruthyunjaya (1998a e 1998b)** utilizaram a formulação de Newton-Euler para a dinâmica inversa de uma plataforma de Stewart e utilizaram o modelo no sistema de controle. No primeiro trabalho, eles realizaram simulações utilizando o MATLAB, mas não comentam a respeito do custo computacional. No segundo, alguns detalhes adicionais são tratados, como a influência da dinâmica dos ligamentos (efeito da inércia destes elementos). Os resultados mostraram que o efeito da inércia dos ligamentos poderia chegar a 50% da força dos atuadores. Para esses resultados, foi utilizado como o valor da massa da plataforma um valor cerca de 10 vezes maior que a de cada ligamento. Obviamente a inércia dos ligamentos não pode ser desprezada, pelo menos na arquitetura estudada e com os parâmetros propostos. O modelo final foi simplificado para reduzir o custo computacional. Segundo **Merlet (2000)**, a inércia dos ligamentos é um ponto que até hoje não houve consenso, ou

seja, muitos defendem que dependendo do uso e da arquitetura pode ser um efeito desprezível.

**Ji (1993)** também havia ressaltado a importância de considerar a inércia dos ligamentos quando estudava a mesma arquitetura, no entanto dependendo das condições de movimento e dos algoritmos de controle a inércia poderia ser desprezada resultando em algoritmos mais eficientes computacionalmente.

Kosuge (**Kosuge et al. 1993**) também utiliza o método de Newton – Euler na formulação de um modelo dinâmico computacional para a plataforma de Stewart. Ele parte da metodologia empregada para robôs seriais e a aplica nos robôs de arquitetura paralela, cortando as cadeias cinemáticas e tratando-as individualmente. Embora faltem dados para que se possa formular uma comparação quanto ao desempenho com outros métodos, sua abordagem é interessante para a solução de problemas semelhantes.

**Do e Yang (1988)** utilizam o método de Newton-Euler para encontrar as equações dinâmicas para um robô genérico. Consideram que a plataforma móvel é um disco e que os centros de massa dos ligamentos estão exatamente no meio dos ligamentos, simplificando em muito as matrizes de inércia. Dessa maneira, o cálculo das forças nas articulações resume-se à solução de um sistema linear de 6 equações. Também sugerem um algoritmo para gerar a trajetória utilizando o modelo dinâmico.

**Knapp e Cobet (2000)** utilizaram a formulação de Lagrange para a máquina Hexaglide, mas desprezaram a dinâmica dos ligamentos. Um modelo completo da máquina seria extremamente demandante computacionalmente. As simplificações feitas possibilitam que o modelo seja utilizado em tempo real nos algoritmos de controle da máquina. **Lebret et al. (1993)** também utilizaram com sucesso a formulação de Lagrange para a modelagem e o controle de uma plataforma de Stewart, mas consideraram os ligamentos como massas concentradas, justificando que não se incorre em erros significativos.

**Codourey e Burdet (1997)** estudaram uma forma de obter equações dinâmicas lineares para serem utilizadas em tempo-real em robôs de arquitetura paralela. O método apresentado utilizou o princípio do trabalho virtual, considerando cada um dos corpos rígidos separadamente. O algoritmo foi aplicado no robô Delta como exemplo e foram feitas algumas simplificações para que as equações pudessem

ser escritas na forma linear, como a distribuição das massas dos ligamentos em alguns pontos específicos (1/3 da massa de cada ligamento foi colocada na plataforma móvel e 2/3 no cotovelo) reduzindo o número de corpos rígidos de 7 para 4 (3 ligamentos foram eliminados). Embora os autores afirmem que o método é difícil de ser aplicado, sugerem o uso do mesmo para robôs mais complexos de seis graus de liberdade.

Também com o intuito de se obter um modelo dinâmico rápido, **Honneger et al. (1997)** utilizaram o método de D'Alembert na máquina Hexaglide. As equações simplificadas habilitaram um algoritmo de controle adaptativo não linear baseado na minimização do erro de posicionamento da ferramenta.

**Merlet (2000)** cita ainda trabalhos de outros métodos como o de Zanganeh (**Zanganeh et al 1997**) que utilizou um método misto baseado nas formulações de Lagrange e Newton-Euler para obter as equações dinâmicas de um robô de 6 graus de liberdade.

### 2.3. Volume de trabalho

**Merlet (2000)** apresenta os diferentes tipos de volume de trabalho, como por exemplo, o Volume de trabalho de Orientação Total e o Volume Máximo. É fornecida uma base sólida para o entendimento de outros trabalhos a respeito deste assunto.

Muitos trabalhos focam em métodos numéricos de análise, dividindo um volume do espaço numa malha de pontos e então é verificado se o robô é capaz de atingir os pontos e com que orientação da plataforma (**Merlet, 2000**).

Outros algoritmos mais inteligentes como o de Conti (**Conti et al, 1997**), utilizam o método da dicotomia para encontrar a borda do volume de trabalho partindo de um ponto central definido. O algoritmo leva em consideração inclusive algumas restrições como possíveis colisões entre ligamentos, o limite dos comprimentos dos mesmos (curso dos pistões) e o limite angular das articulações. A máquina estudada foi uma Hexapod (plataforma de Stewart).

**Masory e Wang (1993)** que também estudaram a plataforma de Stewart comentam algumas conclusões sobre o assunto, como o fato de que o tamanho do volume de trabalho é aproximadamente proporcional ao cubo do curso dos atuadores e que o volume de trabalho é máximo quando a base fixa e a plataforma móvel possuem as mesmas dimensões (embora o robô fique numa posição de singularidade quando em sua posição nominal). O volume de trabalho é pouco influenciado pela posição das articulações na plataforma móvel e os limites mecânicos das articulações são muito importantes de serem levados em consideração, pois geralmente são a maior restrição.

**Bonev e Gosselin (Bonev e Gosselin, 2000)** apresentam um método para o cálculo de volumes de trabalho de orientação constante para máquinas do tipo 6-RUS (cada cadeia cinemática possui uma articulação de rotação atuada, uma junta universal e uma junta esférica; esta última ligada à plataforma móvel). O algoritmo é baseado numa estratégia puramente geométrica de solução onde o espaço alcançável pelos pontos de interesse (juntas, plataforma móvel) é calculado e as intersecções dos volumes (toróides, esferas) são então medidas. As singularidades são descobertas qualitativamente e as restrições, por exemplo, angulares, são impostas às equações. Os autores também sugerem o uso do método em ferramentas CAD.

#### **2.4. Síntese e Otimização de Robôs de Arquitetura Paralela**

Muitos são os parâmetros com o qual se determina completamente um robô. Na verdade, **Merlet (2000)**, que estudou o trabalho de **Masory et al (1993)**, conta que 132 parâmetros são necessários para descrever completamente um robô genérico de seis graus de liberdade.

Frente aos problemas encontrados no assunto, pode-se citar a própria escolha da arquitetura para um dado propósito. **Merlet (2000)** comenta sobre o número reduzido de trabalhos a respeito da síntese e da otimização de robôs de cadeias fechadas, principalmente para seis graus de liberdade. Outro problema é a avaliação do desempenho que até agora só se tem meias respostas. Isso se trata não apenas da

influência do comprimento dos ligamentos ou tipos e limitações das articulações, mas também do que e como se deseja otimizar.

**Brogårdh (2002)**, um pesquisador da ABB, em seu trabalho sobre desenvolvimento de produtos em robótica, lista os fatores que podem influenciar o projeto e sugere uma seqüência de passos a ser seguida para que se tenha um projeto ótimo de uma dada arquitetura paralela. Como fatores que influenciam o projeto podem ser citados dentre outros o tipo de atuadores (rotativos, lineares ou ambos), o número de graus de liberdade e a existência de graus de liberdade redundantes, de atuadores redundantes, de ligamentos passivos e a geometria dos ligamentos e da plataforma móvel.

Encontrando-se uma arquitetura que se encaixe bem perante as necessidades de cada aplicação em particular, alguns passos devem ser levados em consideração para que se complete o projeto da máquina:

1. Depois de escolhida a arquitetura, deve-se transferir a máquina para um ambiente CAD (com uso do ADAMS, DADS, ou outro) onde ela possa ser estudada e as demandas da arquitetura sejam detalhadas como número de graus de liberdade, volume de trabalho, singularidades, padrão de movimentação, acessibilidade e outros.
2. Seguem então as seqüências de otimização, sendo a primeira, a otimização cinemática. Como funções objetivo, são sugeridas variáveis como o volume de trabalho, propagação e identificação de erros, padrão de movimento e acessibilidade. Algumas dessas variáveis também podem ser utilizadas como restrições. Quanto aos parâmetros de projeto, pode ser utilizado o comprimento dos ligamentos, a posição dos pontos de apoio na base fixa ou na plataforma móvel, as limitações das articulações, as dimensões da plataforma, as folgas dos componentes, as tolerâncias de manufatura e a precisão de montagem. São sugeridos métodos como algoritmos genéticos para uma otimização global, mas métodos de programação matemática para otimizações locais.
3. A próxima otimização seria quanto à dinâmica de corpos rígidos. Para este algoritmo pode ser utilizado um modelo simplificado de elementos finitos ou aproximações da estrutura através de tubos, vigas, placas e outros elementos. Exemplos de funções objetivo são a máxima aceleração ou a máxima velocidade



- possível, enquanto as forças nos elementos ficam como restrições. Como variáveis de projeto podem ser utilizados torques ou forças nos atuadores, geometria e / ou formato dos ligamentos. Métodos baseados em otimização topológica são sugeridos para projetos de longo prazo. O estudo da dinâmica principalmente nas trajetórias mais solicitadas do robô é de extrema importância.
4. A última otimização a ser executada é então a de flexibilidade onde se podem utilizar como função objetivo a rigidez e as frequências naturais. O modelo deve comportar parâmetros como rigidez dos elementos (plataforma, ligamentos, juntas, atuadores, redutores), a derivada das acelerações (*Jerk* ou tranco) e outros. Esse modelo também é necessário para compensação de erros de posicionamento estático como a gravidade.
  5. Os passos 1 a 4 são integrados para que façam parte de um único algoritmo de otimização.

**Smith e Nguyen (1991)** estudaram uma plataforma de Stewart levando em consideração a relação de compromisso entre a rigidez e tamanho da plataforma móvel, que afetam a exatidão de posicionamento e volume de trabalho versus o peso da mesma, e robustez dos ligamentos e atuadores. Minimizando-se a tensão nos ligamentos e conhecendo-se o efeito dessas tensões na deflexão da plataforma, pode-se projetar uma máquina com uma certa exatidão desejada. Os pesquisadores salientam que na verdade o que realizaram não é uma otimização, mas sim uma adequação, já que partem de uma rigidez relativamente baixa e vão gradativamente alterando parâmetros até que a deflexão fique dentro de parâmetros aceitáveis.

**Angeles (2002)** defende que ao se projetar um manipulador paralelo para tarefas envolvendo menos de seis graus de liberdade, sua topologia pode ser mensurada utilizando apenas características qualitativas. Tendo em face inúmeras arquiteturas e pequenas variações praticamente ilimitadas das mesmas, Angeles explica que em se tratando de síntese, ainda se está no período Cambriano da História Natural, mostrando que se está longe de existirem algoritmos ou métodos adequados para resolver o problema (escolher e otimizar uma arquitetura).

No âmbito da otimização mecânica, Bidault e outros (**Bidault et al, 2001**) otimizaram um robô de três graus de liberdade de rotação chamado *Agile Wrist*. O objetivo foi o de reduzir ao máximo as tensões nos ligamentos e minimizar o peso

dos mesmos, permitindo maior resposta dinâmica com os mesmos esforços de atuação. Na modelagem foi utilizado o princípio do trabalho virtual e uma abordagem com MEF (Método dos Elementos Finitos) para se obter as tensões ao longo dos ligamentos. Particularmente para este robô, onde os ligamentos são arcos ao invés de cilindros a otimização mecânica mostrou ser um desafio, mas os resultados comprovam a eficácia do método e a sua necessidade para um projeto mecânico otimizado.

**Kim e Tsai (2002)** otimizaram um robô de 3 graus de liberdade para aplicações em usinagem. O foco da otimização é manter uma alta rigidez média ao longo de todo o volume de trabalho. Eles dividiram o volume numa malha e em cada ponto restrições geométricas são verificadas, evitando que a máquina chegue a uma singularidade, como por exemplo, se uma de suas cadeias cinemáticas estiver totalmente esticada. Os dados de entrada para a simulação apresentada foram os cursos dos atuadores lineares e utilizou-se um algoritmo para programação quadrática seqüencial. Os autores salientaram também que um protótipo estava em construção para maiores estudos do assunto.

**Chablat e Wenger (2003)** otimizaram uma arquitetura semelhante, também de 3 graus de liberdade de translação, a Orthoglide, para as mesmas aplicações em usinagem. Durante a estruturação do problema, salientam a importância de considerar a rigidez, minimizar as massas em movimento, utilizar juntas simples para reduzir custos de manufatura, estudar o volume de trabalho tentando equalizar o desempenho na máquina neste volume e finalmente tentar obter a relação entre esforços nos atuadores e esforços para realização de atividades, próxima a 1. Os autores salientam os resultados alcançados (apenas teóricos) mostrando que todo o volume de trabalho é capaz de ser trabalhado pela máquina, que apresenta em todos os pontos uma rigidez adequada e baixa inércia.

## 2.5. Controle

Os robôs paralelos, devido a sua alta rigidez, permitem com que alguns modelos operem com altíssimas velocidades e acelerações, como por exemplo, os robôs Delta, que podem apresentar acelerações da ordem de mais de 10 vezes a aceleração da gravidade (**Robotics Online, 2005**). A possibilidade dos robôs paralelos de operarem com altas acelerações e alta exatidão gerou a necessidade do desenvolvimento de sistemas de controle de alto desempenho. Esses controladores exigem a solução das equações dinâmicas da máquina em tempo real (**Merlet, 2000**).

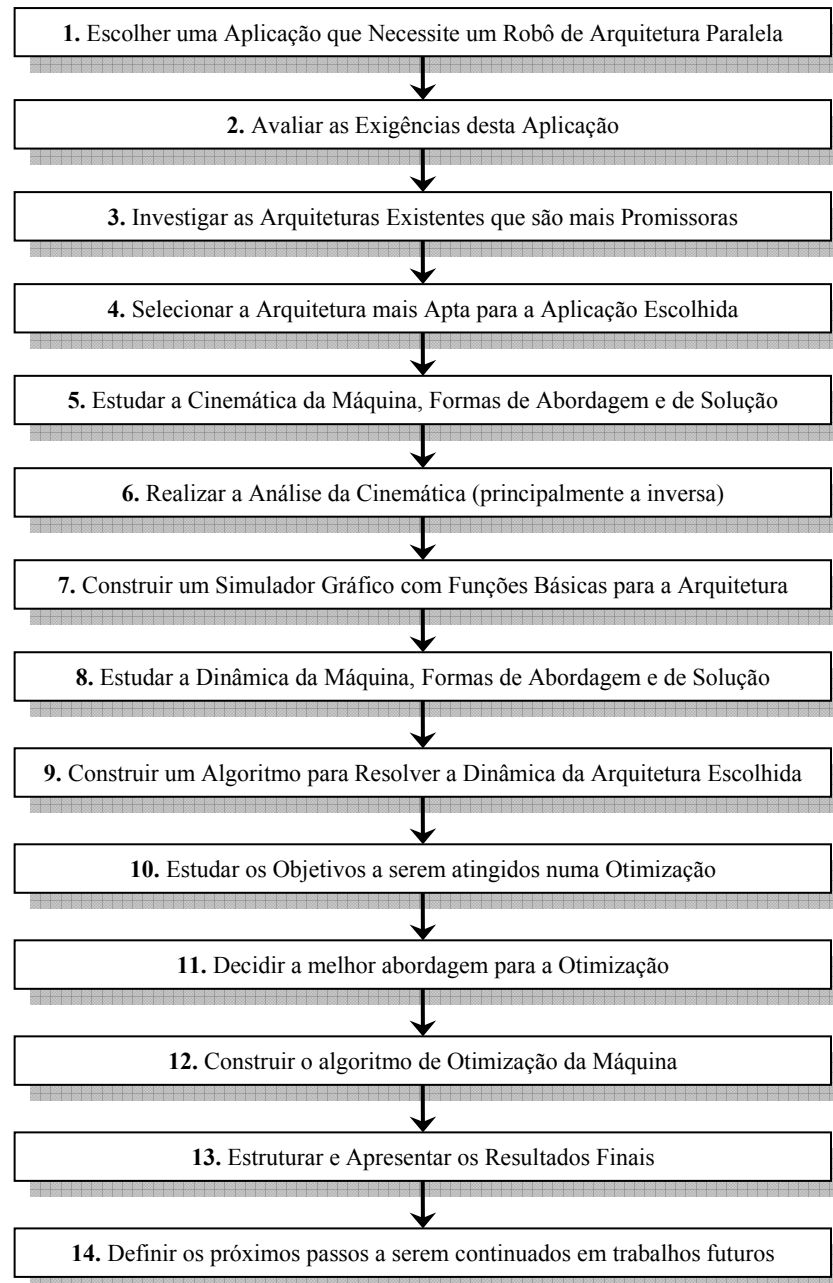
**Honegger et al. (1997)** utiliza as equações dinâmicas do robô Hexaglide em um algoritmo de controle não linear onde alguns parâmetros do modelo dinâmico são corrigidos *on-line* através da medição dos erros.

Devido à grande não linearidade dos mecanismos paralelos e das forças que atuam na ferramenta, estudos mostram que controladores PID individuais para cada motor apresentam uma exatidão limitada (**Honegger, 2000**). **Merlet (2000)** relata que alguns pesquisadores sugerem o uso de controladores não lineares que utilizam informações calculadas em tempo real por um modelo dinâmico da máquina. Outros pesquisadores sugerem ainda aplicar técnicas de controle adaptativo não linear (**Honegger et al., 2000**). Esses estudos justificam o empenho na obtenção de modelos dinâmicos de alto desempenho.

Na medida em que se buscam máquinas de elevada exatidão, além das dificuldades de obtenção de algoritmos de controle de alto desempenho, deve-se medir com extrema exatidão possíveis deformações nos componentes da máquina decorrentes de mudanças de temperatura ou de pequenos desalinhamentos de partes da máquina. Técnicas avançadas para se obter essas medidas foram usadas no projeto da máquina Hexaglide, como relatado por **Honegger et al. (2000)** e por **Knapp e Cobet (2000)**.

### 3. METODOLOGIA

Para realização dos objetivos desse trabalho serão seguidos 14 passos apresentados no fluxograma a seguir:



**Figura 11:** Fluxo de Atividades a serem seguidas durante o projeto

### 3.1. Passos da Metodologia

Nos itens que se seguem, cada um dos passos apresentados na Fig. 11 é descrito em maior detalhe e possíveis riscos ou desvios na metodologia serão indicados.

Todos os passos detalhados, para fins didáticos, também podem ser agrupados em apenas cinco: *Escolher uma arquitetura para uma dada aplicação* (passos 1 a 4), *Realizar a análise cinemática* (passos 5 a 7), *Realizar a análise dinâmica* (passos 8 e 9), *Otimizar a máquina* (passos 10 a 12) e *Preparar os resultados e definir próximos passos* (passos 13 e 14).

#### 3.1.1. Escolher uma Aplicação que Necessite um Robô de Arquitetura Paralela

O primeiro passo deste trabalho deve ser a definição do propósito do robô. Alguns exemplos possíveis para aplicações industriais:

- Usinagem de bloco de motores;
- Montagem de componentes numa placa eletrônica;
- Pick-and-place.

Apenas um objetivo pode ser escolhido, ou pode-se optar pela versatilidade e estudar a possibilidade de mais de uma função.

#### 3.1.2. Avaliar as Exigências da Aplicação

Definida a aplicação, deve-se então avaliar quais as exigências mecânicas ou construtivas da máquina. Sendo assim, pode-se:

- Especificar o número de graus de liberdade exigidos;
- Especificar os requerimentos quanto à amplitude de movimentos de cada grau de liberdade;

- Especificar o tipo de solicitação esperada (momentos, cargas);
- Ter uma primeira avaliação do tipo de arquitetura que se encaixa melhor às especificações exigidas.

### **3.1.3. Investigar as Arquiteturas Existentes mais Promissoras**

Nesta etapa deve-se realizar uma pesquisa quanto às arquiteturas mais promissoras para o decorrer do projeto. Embora seja impossível ter uma análise completa das características das máquinas sem uma análise detalhada de seus modelos, faz-se uma análise baseada nas observações de outros pesquisadores e engenheiros, além de observações próprias quanto à funcionalidade da arquitetura.

### **3.1.4. Selecionar a Arquitetura mais Apta para a Aplicação Escolhida**

Analisam-se as observações e vantagens de cada arquitetura para os requisitos e as características necessárias. A arquitetura mais promissora é então a escolhida.

### **3.1.5. Estudar a Cinemática da Máquina, Formas de Abordagem e de Solução**

Existem duas análises diferentes neste tópico que serão levadas em consideração:

- Cinemática Inversa
- Cinemática Direta

Cada uma delas possui um objetivo em particular. Na cinemática inversa as entradas são os dados da plataforma móvel (posição, orientação) e a saída são os dados dos atuadores. Para cada posição do efetuador, o algoritmo da cinemática inversa deve prover a posição dos atuadores, possibilitando assim a construção de trajetórias para o movimento especificado, item necessário a um simulador.

A cinemática direta é o oposto, ou seja, dadas as posições dos atuadores, deve-se calcular a posição do efetuador. Esta análise é particularmente útil para simuladores, estudos ou mesmo algoritmos de controle, já que pode ser analisado o efeito de um erro de posicionamento nos atuadores com relação ao efetuador e até mesmo magnitudes aceitáveis de sobre-sinal e outros. Para um usuário querendo operar a máquina, é algo irrelevante, já que na maioria das vezes quando se movimenta apenas um atuador, não se sabe os movimentos resultantes da plataforma, muito menos se consegue um movimento complexo.

Para cada uma das análises, os seguintes passos são envolvidos:

- Escolha do método para equacionar a análise cinemática;
- Escolha de um algoritmo apropriado para o modelo.

### **3.1.6. Realizar a Análise Cinemática**

Definido quais e como as análises serão feitas, o modelo é equacionado e o algoritmo implementado. Dependendo do método escolhido, especialmente para a cinemática direta, pode ser necessário um algoritmo para a solução de sistemas de equações não-lineares.

### **3.1.7. Construir um Simulador Gráfico para a Arquitetura**

Para concluir o estudo cinemático deve-se construir um simulador, capaz de, no mínimo, executar alguns movimentos de interpolação linear e interpolação circular. O simulador deve permitir a observação do modelo em funcionamento, como os movimentos ocorrem e a demonstração das capacidades da máquina.

Este tópico também inclui uma análise do volume de trabalho da máquina e como calculá-lo.

### **3.1.8. Estudar a Dinâmica da Máquina, Formas de Abordagem e Solução**

Da mesma forma que na cinemática, existem duas formas de análise dinâmica: a inversa e a direta. No escopo do trabalho é imperativo realizar a Dinâmica Inversa, buscando os esforços nos atuadores e estrutura para um movimento específico do efetuador. No caso da Dinâmica Direta, sua utilização estaria na análise dos esforços introduzidos na estrutura devido a um erro de posicionamento de um dos atuadores e auxiliar no controle (por exemplo, calcular o efeito de uma dada curva de aceleração em um motor).

Novamente, devem-se estudar as abordagens existentes, as vantagens e desvantagens de cada uma e então escolher a mais adequada. Para a dinâmica existem três formulações principais:

- Newton - Euler
- Lagrange
- Princípio de D'Alembert

Outras formulações propostas são em geral casos particulares de uma das formulações acima ou uma composição de mais de uma delas. A escolha do método dependerá de alguns fatores como a complexidade da arquitetura escolhida e os dados que se espera obter.

### **3.1.9. Construir um Algoritmo para Resolver a Dinâmica da Arquitetura Escolhida**

Selecionados a abordagem, o método e o algoritmo, constrói-se o modelo computacional para obter os resultados desejados. Frente ao esforço necessário, resolveu-se definir este passo em separado dos outros. Nesta etapa encerra-se o processo de modelagem do robô para o escopo deste trabalho.



### **3.1.10. Estudar os Objetivos a serem atingidos numa Otimização**

Nesta etapa devem-se definir os objetivos da otimização. Deve-se analisar a relevância de algumas perguntas para o comportamento e desempenho da arquitetura:

- Quais as dimensões da máquina para um certo volume de trabalho esperado?
- Como minimizar esforços na estrutura da máquina em seu volume de trabalho?
- Como minimizar os esforços nos atuadores?
- Que áreas do volume de trabalho permitem uma maior aceleração do efetuador?
- Quais áreas do volume de trabalho contêm singularidades? De que tipo?
- Que dimensões da máquina permitem uma maior rigidez média?

Definidos os objetivos, definem-se também os requisitos de projeto e suas restrições (curso dos atuadores, tamanho máximo dos ligamentos, outros).

### **3.1.11. Decidir a melhor Abordagem para a Otimização**

Esta é a hora de se escolher o método numérico que efetivamente solucionará o modelo de otimização. Deve-se utilizar um método baseado em programação matemática ou um probabilístico? Qual método mais especificamente deve ser utilizado? Como se pode ter certeza de que o ponto obtido é um ponto ótimo? Como saber se é um ótimo local ou um global?

### **3.1.12. Construir o Algoritmo de Otimização da Máquina**

Constrói-se e implementa-se o algoritmo de otimização da máquina. Nesta etapa encerram-se todas as etapas de modelagem e otimização.

### **3.1.13.        *Estruturar e Apresentar os Resultados Finais***

Após a construção de todas as ferramentas, apresentam-se os dados que foram utilizados, os porquês e os resultados, preparando a documentação e encerrando o trabalho.

### **3.1.14.        *Definir os Próximos Passos (Trabalhos Futuros)***

Seria muita pretensão acreditar que terminando este trabalho, encerram-se as possibilidades de estudo neste mesmo assunto com este escopo. Este item visa então apresentar opções de continuidade, avaliando o que ainda precisa ser estudados em maiores detalhes para que se possa construir até mesmo um protótipo. Um exemplo é o projeto mecânico da máquina, principalmente suas articulações.

## 4. A ESCOLHA DA APLICAÇÃO E DA ARQUITETURA

Neste capítulo, uma aplicação que pode se beneficiar da utilização de robôs de arquitetura paralela é escolhida e então é realizada uma análise qualitativa dos requerimentos necessários e desejáveis, seguida da escolha de uma arquitetura.

### 4.1. A Necessidade de um Robô de Arquitetura Paralela

Os robôs de arquitetura paralela ainda não são muito utilizados na indústria, contudo já existem alguns casos de sucesso que serão descritos aqui. Três casos serão relatados (**Robotics Online, 2005**) e as máquinas estão ilustrados na Fig. 12:

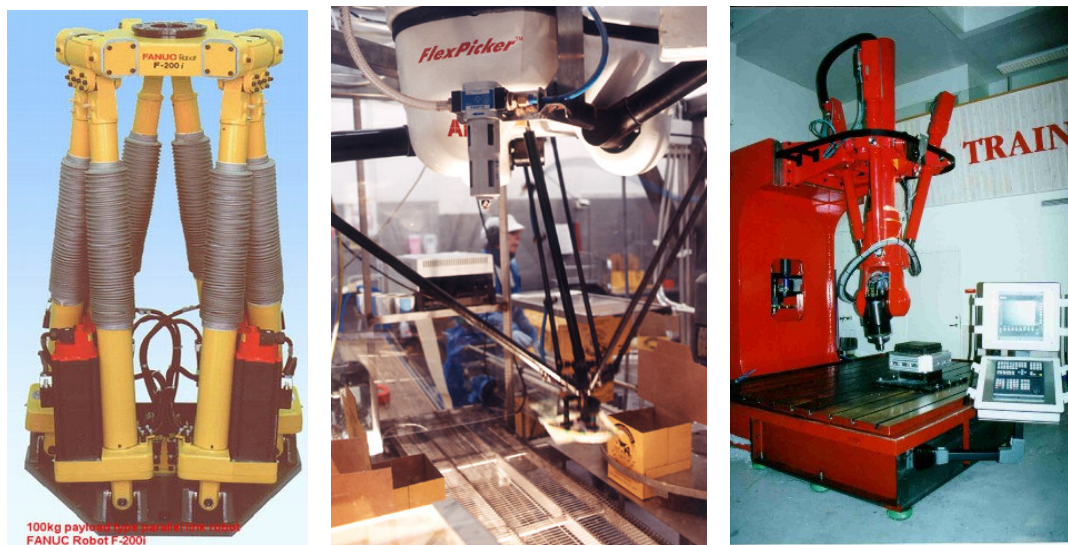
- 1) A demanda por versatilidade na indústria automobilística, para a qual se pôde utilizar o *F-200i* da *FANUC*;
- 2) A demanda por maior velocidade da linha de produção numa indústria de alimentos, onde se utilizou o *FlexPicker IRB 340* da *ABB*;
- 3) A demanda por usinagens de superfícies complexas com alta precisão na indústria aeronáutica, onde se encaixou o *Tricept* da *Neos Robotics*.

Num esforço de explorar as capacidades dos robôs de arquitetura paralela na indústria, a *FANUC Robotics* criou o *F-200i* (uma *Plataforma de Stewart*, com 6 graus de liberdade), capaz de operações como manipulação de partes e ferramentas, operações de soldagem e outras. Segundo o gerente do programa *F-200i*, Allen Grzebyk, o foco da *FANUC* é a indústria automobilística e este robô é capaz de se adequar muito bem na Armação (local onde toda a estrutura externa de chapas de metal do veículo é montada e soldada). O robô já é utilizado em aplicações de soldagem, substituindo equipamentos dedicados em aplicações de fixação onde consegue dar bastante versatilidade ao seu posto de trabalho e reduzir o espaço requerido, e em aplicações de manuseio de peças e partes. O *F-200i* tornou uma célula de manufatura de uma fábrica de autopeças flexível o suficiente para fazer peças da *GM* ou da *Ford*, evitando a utilização de itens dedicados.

Segundo o Vice-Presidente de Engenharia da *Pepperidge Farms*, David Watson, a empresa está bastante satisfeita com a confiabilidade e desempenho do

*FlexPicker* da *ABB*. Por muitos anos foram utilizados robôs *SCARA* para ambas as operações de *pick-and-place* e *sandwiching* na produção dos *cookies* Milano. Os robôs *SCARA*, devido a sua própria inércia, eram capazes de apenas 60 a 65 *picks* por minuto enquanto o *FlexPicker* é capaz de 120. Apenas em 1999 a empresa comprou 32 e em 2000 contava com 46 exemplares do modelo para produção dos seus *cookies*. Os robôs foram instalados pela *SIG Pack Systems*, que os colocou no teto, economizando muito espaço no chão de fábrica.

A arquitetura *Tricept*, criada e patenteada por Karl-Erik Neumann, fundador da *Neos Robotics* influenciou em muito o entendimento que se tinha dos robôs de arquitetura paralela (embora seja, na verdade, uma arquitetura híbrida com 3 graus de liberdade paralelos e 3 seriados). É uma arquitetura própria para usinagens, sendo também de extrema versatilidade. A indústria aeronáutica a usa para usinagens de superfície com contornos complexos como pás de turbinas. Dentre outras aplicações para a *Tricept* estão: polimento, trabalhos em madeira, corte de materiais por jato de água, solda ponto. Além dessa indústria, vários fabricantes de automóveis também já estão utilizando máquinas com essa arquitetura como: *Peugeot*, *Ford*, *Renault*, *Volvo*, *GM*, *BMW* e *Volkswagen*.



**Figura 12:** Da esquerda para a direita – o F-200i da FANUC (Hexapod), o IRB 340 da ABB (Delta) e o Tricept da Neos Robotics (Tricept) – Fotos respectivamente de FANUC (2005), ABB (2005) e Robotics Online (2005).

Um fator interessante observado é a dualidade entre os robôs seriados e paralelos. Conforme relatado por **Bruyninckx (2005)**, essa dualidade vai além de pontos observados na prática, como o fato de que os pontos fracos de um são o forte do outro e vice-versa. Envolve também questões geométricas e matemáticas. Assim, é viável tomar como hipótese que, de forma geral, os robôs de arquitetura paralela desempenham melhor suas funções quando utilizados em aplicações em que os robôs seriados não estão aptos a realizar. Como exemplos, têm-se as aplicações de usinagem e de *pick-and-place*, bem como montagens leves.

## 4.2. Investigação de Arquiteturas Existentes

Neste item busca-se uma investigação das arquiteturas mais conhecidas para as aplicações sugeridas anteriormente. Faz-se a ressalva de que as arquiteturas apresentadas também podem ser utilizadas para outro fim. Por exemplo, a Tricept foi concebida para usinagem pela *Neos Robotics*, contudo o modelo da *ABB* é utilizado para aplicações industriais em geral.

Vale ressaltar que a busca apresentada aqui não é exaustiva. **Pierrot e Company (1999)** sugerem o trabalho de **Innocenti e Parenti-Castelli (19XX)**, onde é apresentada uma busca exaustiva focada em arquiteturas paralelas de seis graus de liberdade (infelizmente não é citado o ano da publicação).

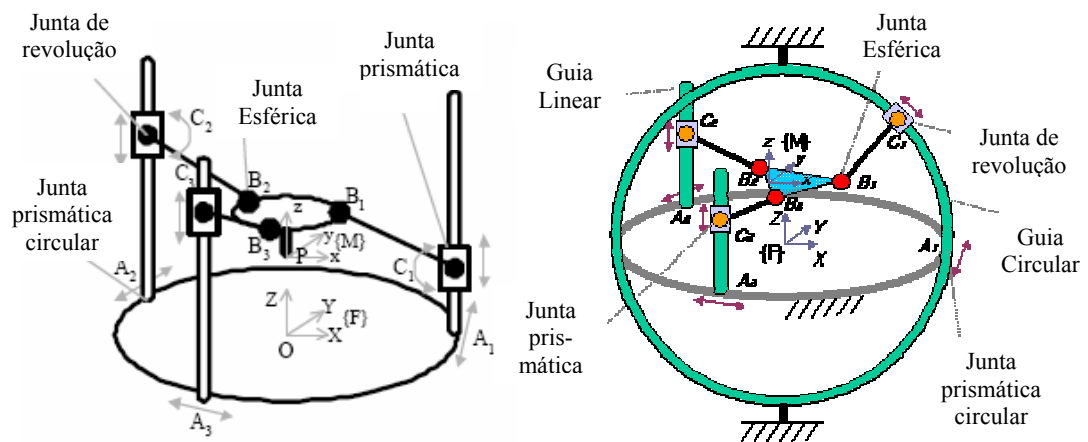
### 4.2.1. Arquiteturas Utilizadas em Usinagem

Na usinagem têm-se algumas arquiteturas paralelas conhecidas e muitas são modelos de máquinas já em comercialização. As arquiteturas investigadas são:

- Eclipse-I (**Kim et al., 1999**);
- Eclipse-II (**IPMP, 2002**);
- Hexapod (**Blümlein, 1999**);
- Hexaglide (**Knapp e Cobet, 2000**);
- HexaM (**LIRMM, 2004**);
- Tricept (**Robotics Online, 2004**);

A Fig. 13 apresenta um esquema das arquiteturas utilizadas nas máquinas Eclipse I e II. Na máquina Eclipse-I a plataforma móvel é conectada a três ligamentos através de juntas esféricas e esses ligamentos, por sua vez, são ligados a hastes verticais por intermédio de duas juntas, uma de revolução e outra prismática, em série. As juntas prismáticas são atuadas, bem como os pontos inferiores das hastes verticais, que são presos a uma guia circular. Na Eclipse-II uma das hastes verticais foi substituída por uma outra guia circular.

Existe, contudo, uma ressalva: as máquinas Eclipse I e II não são arquiteturas de robôs completamente paralelos, já que o número de graus de liberdade é inferior ao de cadeias cinemáticas.

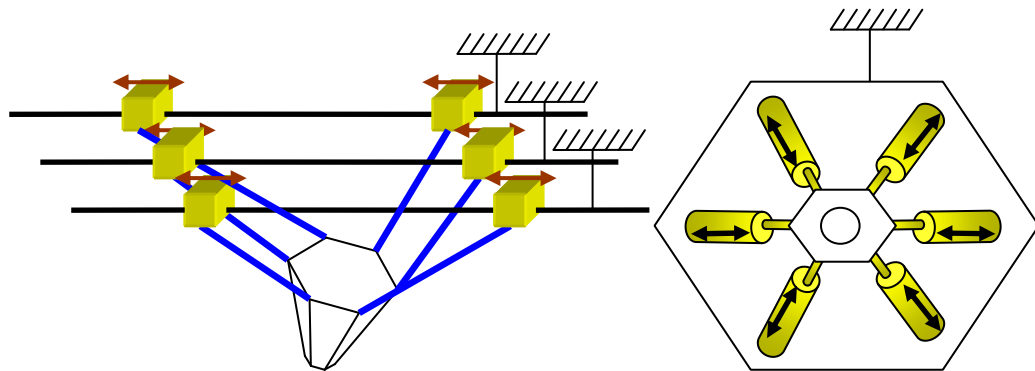


**Figura 13:** Esquemas das arquiteturas Eclipse-I (esquerda) e II (direita) - Kim et. al., 1999.

A Fig. 14 apresenta os esquemas das arquiteturas Hexaglidle e Hexapod. Na Hexaglidle a plataforma móvel é conectada a seis ligamentos rígidos por intermédio de uma junta esférica, ou um punho esférico, e cada um desses ligamentos é preso a um bloco móvel de uma guia linear através de uma junta universal. Os blocos móveis das guias são atuados. Essa estrutura permite muitas variantes, tais como o uso de seis guias paralelas ou o uso de apenas três, cada uma com dois blocos (modelo apresentado na figura).

Os modelos de arquitetura Hexapod consistem de máquinas muito próximas à máquina de Gough. Nesses, a base móvel também é ligada a seis ligamentos utilizando-se juntas universais; cada ligamento é ligado à base fixa por intermédio de

uma junta esférica (ou punho esférico sem atuadores). A atuação se dá alterando o comprimento dos ligamentos.

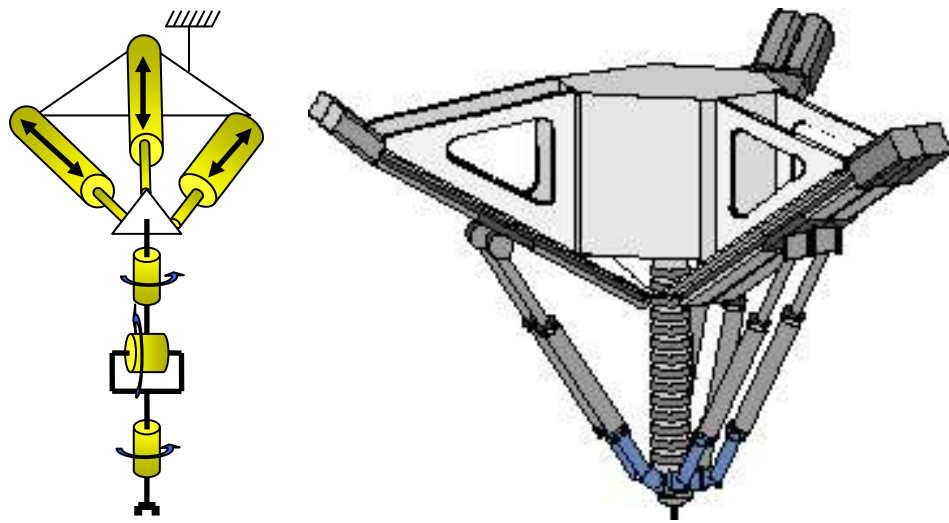


**Figura 14:** Esquemas das arquiteturas Hexaglidle (à esquerda) e Hexapod – visão inferior (à direita).

A Fig. 15 ilustra o modelo da arquitetura HexaM considerada o conceito estendido de outra arquitetura, a Hexa. Na Hexa a plataforma móvel está ligada a seis ligamentos através de articulações esféricas e cada um destes ligamentos está acoplado a um outro por intermédio de articulações esféricas ou juntas universais, existindo essas duas versões de máquinas. Finalmente, estes últimos ligamentos são conectados aos motores, tendo-se assim juntas de revolução atuadas. Os motores ficam presos à base fixa.

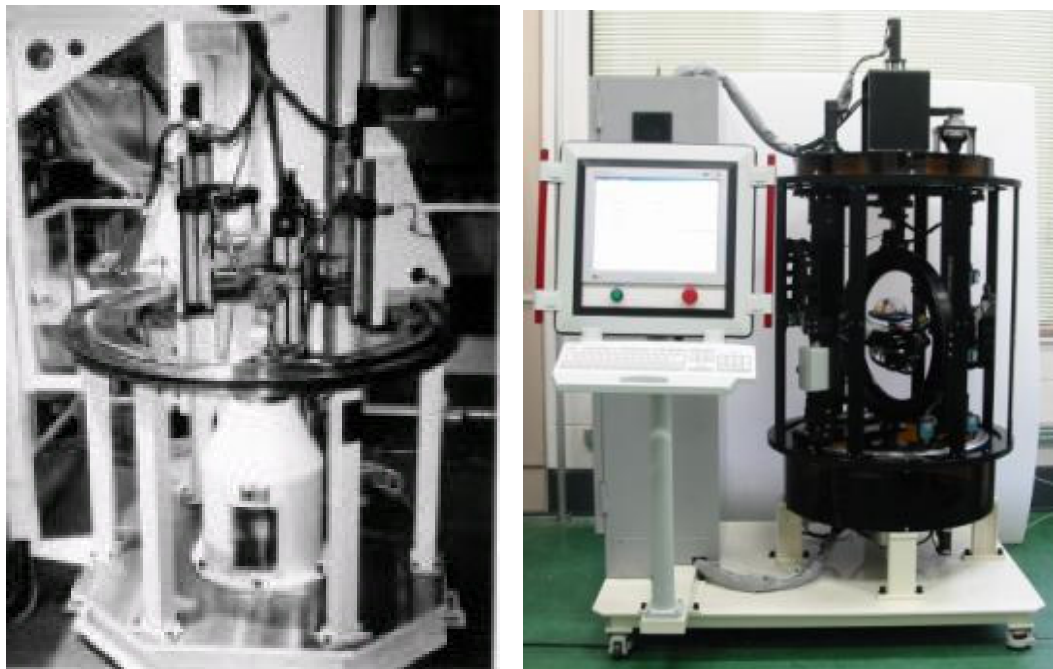
A HexaM possui o esquema de ligações da plataforma móvel idêntico a da Hexa, exceto que os ligamentos acoplados aos motores foram substituídos por guias lineares e se utilizam juntas universais na conexão das guias com os ligamentos. A Toyota, em conjunto com pesquisadores de outras instituições, concebeu a arquitetura e possui uma máquina deste tipo para comercialização. Apenas a título de curiosidade, o modelo possui exatidão de  $4\ \mu\text{m}$ , capacidade de se inclinar com relação à vertical em  $30^\circ$ , máxima aceleração aproximadamente igual à gravidade e velocidade máxima de  $1.7\ \text{m/s}$  (Parallemic, 2002).

Por último, observa-se a arquitetura híbrida Tricept (Fig. 15), onde três graus de liberdade de rotação são executados por um punho esférico motorizado, como nos robôs industriais seriados articulados de seis graus de liberdade.



**Figura 15:** Esquema da arquitetura HexaM (à direita) – LIRMM, 2004. Esquema da arquitetura Tricept (à esquerda).

Alguns exemplos de máquinas que utilizam as arquiteturas citadas estão apresentados nas Fig. 16 a Fig. 18.



**Figura 16:** Máquina do tipo Eclipse-I (à esquerda) e do tipo Eclipse-II (à direita). Fotos de IPMP, 2002.





**Figura 17:** À esquerda uma máquina com arquitetura Hexapod, da Micromat (Blümlein, 1999), e à direita a IWF-Hexaglide (Knapp e Cobet, 2000).



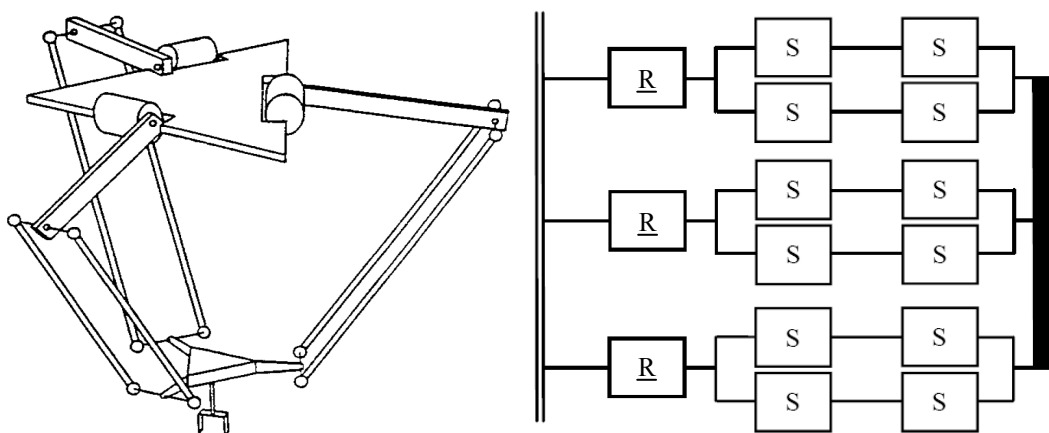
**Figura 18:** À esquerda, a HexaM da Toyoda Machine Works Ltda. (Parallemic, 2002) e à direita o Tricept da Neos Robotics (Robotics Online, 2004).

#### 4.2.2. Arquiteturas Utilizadas em Pick – and – Place e Montagem

Em operações de *pick-and-place* existem algumas arquiteturas conhecidas como a Delta, a família de arquiteturas do tipo H4 e a Hexa, todas apresentadas aqui.

Alguns pesquisadores indicam a arquitetura Delta como a mais vitoriosa de todas as arquiteturas paralelas já fabricadas por sua simplicidade e pelo seu desempenho na indústria. Ela surgiu da necessidade constante de movimentar pequenas massas de alguns poucos gramas a altas cadências (mais de 150 transportes por minuto). Foi considerado também na concepção da arquitetura que a maioria das operações deste tipo necessita apenas de movimentos de translação e rotações na vertical, sendo que este grau de liberdade pode ser realizado pelo próprio efetuador.

A arquitetura Delta possui seis ligamentos passivos iguais que ficam agrupados dois a dois e formam, com as articulações esféricas e os corpos adjacentes, três paralelogramos. Por projeto, as duas articulações esféricas de cada conjunto mais próximas dos motores (acopladas ao ligamento atuado do conjunto) formam uma linha reta que está sempre paralela ao plano da base fixa. Por estarem formando paralelogramos, garante-se que essa linha também estará sempre paralela à plataforma móvel. Conclui-se assim que a plataforma móvel está sempre paralela à base fixa, demonstrando que a Delta apresenta apenas graus de liberdade de translação. Vide a Fig. 19. Na organização das cadeias das figuras com os esquemas das cadeias a linha dupla representa a base fixa e a linha cheia, a plataforma móvel.



**Figura 19:** À esquerda, esquema do robô Delta (Pierrot et al., 1991a). À direita, a organização das juntas das cadeias cinemáticas da arquitetura (Company e Pierrot, 1999).

As siglas das juntas seguem a classificação apresentada no item 1.6 do Capítulo 1 (*R – Revolute, S – Spherical, U – Universal, P – Prismatic*).

Através do site do LSRO: *Laboratoire de Systèmes Robotiques* é possível se obter várias informações a respeito de máquinas Delta já construídas. É nesse laboratório onde trabalha o Prof. Dr. Reymond Clavel, inventor da arquitetura.

Dentre os muitos modelos de Delta já fabricados, podem ser citados os seguintes:

- **MicroDelta 240** (Fig. 23): Um modelo básico da arquitetura, capaz de alcançar uma cadência de três transportes por segundo em distâncias de 100 mm a 150 mm, com uma exatidão de cerca de 0.1 mm. A resolução do equipamento varia de 0.01 mm a 0.02 mm;
- **Delta 720 com *direct drive*** (Fig. 23): É capaz de acelerações que podem ultrapassar  $500 \text{ m/s}^2$  (cerca de 50 vezes a aceleração da gravidade). Sua característica principal são os motores, ligados diretamente aos ligamentos atuados (*direct drives*). A máquina é capaz de cadências superiores a quatro transportes por segundo em distâncias superiores a 250 mm e com exatidão entre 0.1 mm e 0.2 mm;
- **Delta da Demarex** (Fig. 21): este modelo possui atuadores de pequeno porte (100W a 200W), contudo é capaz de uma cadência de até dois e meio transportes por segundo;
- **ABB FlexPicker IRB 340** (Fig. 8): existem, na verdade, seis versões do modelo, mudando algumas características de um para o outro como a capacidade de carga, o material da máquina e a exatidão. Possuem velocidade máxima de operação de 10 m/s, aceleração máxima de  $100 \text{ m/s}^2$  para a versão de 1 kg de capacidade de carga e  $60 \text{ m/s}^2$  para aquela de 2 kg. A repetibilidade do robô é de 0.1 mm e  $0.4^\circ$  ou  $1.5^\circ$  dependendo do modelo. As versões do modelo também foram feitas para resistir a torques de 1 Nm no eixo normal à plataforma móvel. O volume de trabalho da máquina é complexo. Se for considerada uma altura de trabalho de 300 mm, o robô pode trabalhar num diâmetro de 967 mm. Se for considerada uma altura de 250 mm, ele pode trabalhar num diâmetro maior de 1130 mm. A versão comum do robô pesa 140 kg e a versão em aço inoxidável pesa 165 kg.



**Figura 20:** À esquerda o MicroDelta 240 e à direita o Delta 720 com *direct drive* (fotos do LSRO, 2005).



**Figura 21:** O Delta fabricado pela Demareux Robotique et Microtechnique SA em operação (foto extraída do LSRO, 2005).

Apenas a título de curiosidade, a arquitetura também já é utilizada em outros campos. A companhia Elekta fabrica um exemplar que na Fig. 22 está carregando um microscópio de 20 kg. O controlador do robô compensa o efeito da gravidade e o médico pode levar o microscópio até onde desejar (manualmente). O robô também pode receber um ponto de destino de uma ferramenta de diagnóstico e levar o aparato até o ponto desejado automaticamente.

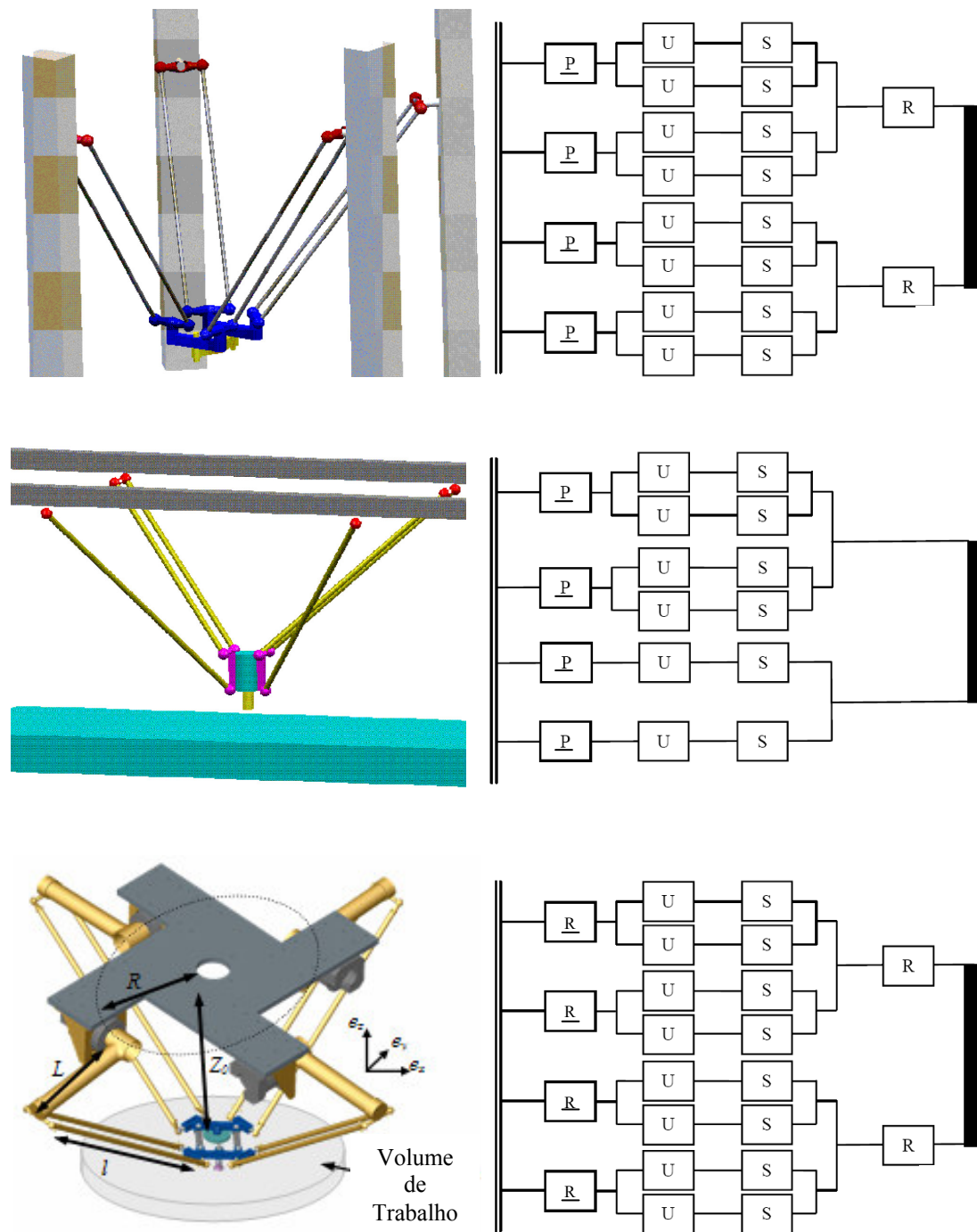


**Figura 22:** O Delta fabricado pela empresa sueca Eleka num ambiente hospitalar (foto do LSRO, 2005).

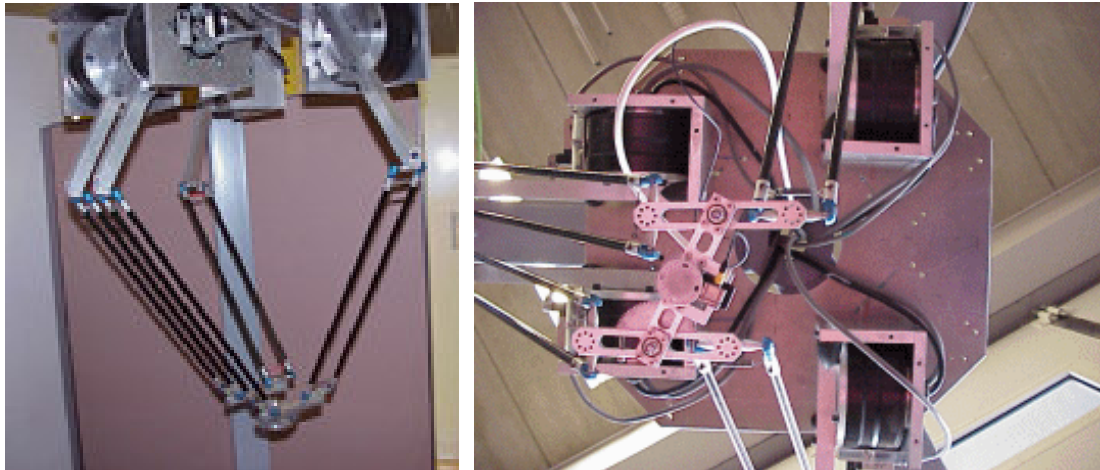
**Pierrot e Company (1999)** acreditam que é necessária uma arquitetura com mais de três graus de liberdade para operações de *pick-and-place* e de *montagem*, contudo ainda mais simples do que arquiteturas de seis graus de liberdade. Por isso propuseram uma nova família de arquiteturas: a família H4. Essa família possui três graus de liberdade de translação e um grau de liberdade de rotação em torno de um eixo definido (não necessariamente o eixo vertical).

A Fig. 23 apresenta alguns modelos da família H4, também inspirada na arquitetura Delta. Já existem protótipos, pelo menos do terceiro modelo de cima para baixo, buscando a vantagem adicional de também poder orientar em alguma direção um objeto transportado. A Fig. 24 apresenta justamente este protótipo que foi construído pelo LIRMM e é capaz de acelerações da ordem de 10 vezes a gravidade e de velocidades máximas acima de 5 m/s.





**Figura 23:** Três exemplos de modelos da família H4 (de cima para baixo, os dois primeiros são de **Company e Pierrot, 1999** e o último do site do **François Pierrot, 2005**) e a organização de suas cadeias cinemáticas (figuras de **Company e Pierrot, 1999**).



**Figura 24:** O robô da família H4 desenvolvido no LIRMM (François Pierrot, 2005) em vista lateral e em vista inferior.

A arquitetura Hexa surgiu como uma expansão do conceito da arquitetura Delta e é anterior em conceito à família H4. A Hexa é dotada de seis graus de liberdade e destina-se também a aplicações mais complexas, como por exemplo, montagens leves em diferentes direções.

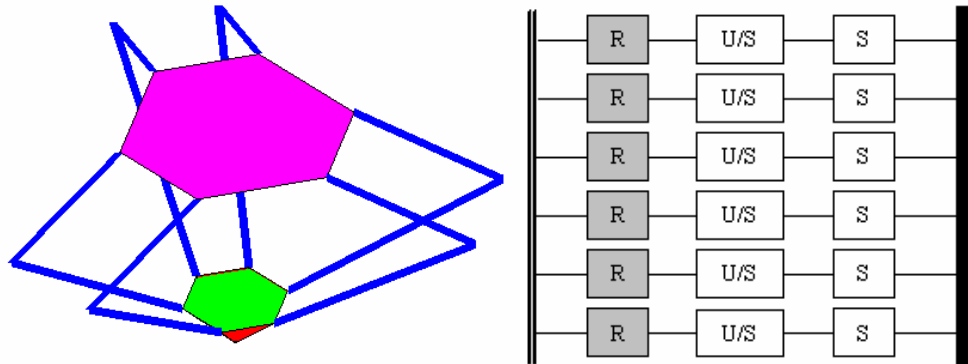
Dentre os principais responsáveis pela concepção da arquitetura estão os pesquisadores François Pierrot e Masaru Uchiyama (Pierrot et al., 1990), onde participou a companhia Toyoda Machine Works. Inspirados no modelo Delta, os pesquisadores expandiram o conceito da arquitetura para um modelo de seis graus de liberdade e criaram a Hexa, que tem patente de agosto de 1994.

O modelo da Toyoda, que utiliza *direct drives*, ainda é motivo de estudos e publicações, atualmente mais focadas ao software de comando e ao controle. Ele possui capacidade de se inclinar com relação a vertical em 30°, máxima velocidade de operação de 6 m/s, máxima aceleração igual a 22 vezes a gravidade e resolução de 0.01 mm. Sua exatidão nunca fora medida, segundo o próprio François Pierrot que foi contatado por e-mail para a obtenção desses dados numéricos e gentilmente se mostrou pronto para responder a qualquer questão ou dúvida sobre o protótipo.

Outros laboratórios também vêm desenvolvendo modelos da Hexa, como o IWF que já está apresentando bons resultados. A Hexa do IWF é capaz de atingir uma velocidade máxima de 5 m/s. Se utilizada uma carga de 1 kg, a máquina pode atingir uma aceleração de 10 vezes a gravidade e com carga total (3,5 kg), seis vezes a gravidade (IWF, 2004). Ela possui uma capacidade limitada de se inclinar com a

vertical ( $15^\circ - 20^\circ$ , dependendo do eixo de rotação), o que aproveita muito pouco os três graus de liberdade de rotação.

Um detalhe interessante é que a Hexa da Toyota é do tipo  $\underline{R}SS$  e a Hexa do IWF é  $\underline{R}US$ . O esquema da arquitetura e os modelos estão apresentados, respectivamente, nas Fig. 25 e 26.



**Figura 25:** Esquema da arquitetura Hexa (à esquerda) e a estrutura de suas cadeias cinemáticas (à direita). A figura da estrutura das cadeias foi extraída de **LIRMM, 2004**.



**Figura 26:** À esquerda, a Hexa da Toyota (**Space Machines Laboratory, 2004**) e à direita a Hexa do IWF (**2004**).



### 4.3. A Escolha da Aplicação e seus Requerimentos em uma Análise Qualitativa

Tendo-se como foco de estudo qualquer uma das aplicações sugeridas, podem ser obtidos excelentes resultados. Contudo, valem algumas observações:

- A versatilidade das aplicações de *pick-and-place* e de montagens leves, utilizadas em diversas indústrias (mecânica, alimentos, médica e outras) é bem maior do que aplicações em usinagem, voltadas mais especificamente à indústria mecânica;
- Aplicações em usinagem exigem um detalhamento muito maior, tanto para entender que forças e torques estão envolvidos nas operações para certas condições de usinagem, bem como na concepção dos modelos matemáticos e físicos;
- Na usinagem importam em muito toda a trajetória da ferramenta e em operações de transporte importam mais os pontos inicial e final do transporte;
- Quanto a questões de exatidão, uma máquina de usinagem é bem mais demandante e costuma ter valores da ordem de alguns poucos  $\mu m$ .

Dessa forma, optou-se por prosseguir o trabalho focando em operações de *pick-and-place* e montagens leves. Qualitativamente, pode-se definir como requerimentos e características da arquitetura que irá desempenhar tais atividades:

- 3 ou 4 graus de liberdade, contudo, 5 ou 6 graus de liberdade é um diferencial, por aumentar a versatilidade, permitindo operações mais complexas. Se for considerado o fato deste trabalho ser também um estudo de cunho acadêmico, o mesmo pode ficar mais rico e completo com os graus adicionais;
- Altas acelerações e velocidades de operação;
- Bons níveis de exatidão e de rigidez;
- Baixa necessidade de capacidade de carga (cerca de 1 kg);
- Baixa magnitude de forças e torques externos;
- Versatilidade para execução de diversas atividades, implicando num amplo volume de trabalho acessível e uniforme (sem predileções por um eixo);

- Grande capacidade de se inclinar com a vertical (para as arquiteturas que possam fazê-lo), aumentando assim a versatilidade nas operações.

#### 4.4. A Escolha da Arquitetura: uma Análise de Foco Qualitativo

Tendo em vista os objetivos da aplicação escolhida, serão levados em consideração os seguintes pontos para analisar as arquiteturas:

- *Aceleração e Velocidade de Operação*: a máquina deve ser muito rápida, já que é a maior demanda nesse tipo de operação;
- *Tamanho do Volume de Trabalho*: o quão extenso e acessível é o volume de trabalho da máquina com relação ao seu próprio tamanho. Obviamente este item depende da orientação do efetuador, contudo, já que serão comparadas arquiteturas de diferentes graus de liberdade, as rotações com a vertical serão analisadas em outro item;
- *Capacidade de Movimentação Angular*: a capacidade do efetuador de se inclinar com relação à vertical;
- *Rigidez*: o quão rígida é a estrutura da máquina com relação ao seu peso, sendo capaz de resistir à forças externas, cargas e forças inerciais;
- *Repetibilidade e Exatidão*: leva em conta a capacidade da máquina de atingir o ponto alvo repetidamente e com exatidão;
- *Facilidade de Projeto e de Fabricação*: o quanto um modelo é fácil de equacionar, realizar estudos e análises, bem como se possui uma estrutura complexa a ponto de tornar a fabricação difícil ou muito custosa;
- *Conhecimento da Indústria*: mede o quanto a arquitetura é conhecida e aceita na indústria, o quanto de material de referência existe, o que ajudará também em pesquisas futuras.

Vale também como premissa que as máquinas devem ser comparadas entre si como se possuíssem elementos de características compatíveis (potência e qualidade dos atuadores, tamanho etc.).

#### 4.4.1. Comentários sobre as Arquiteturas

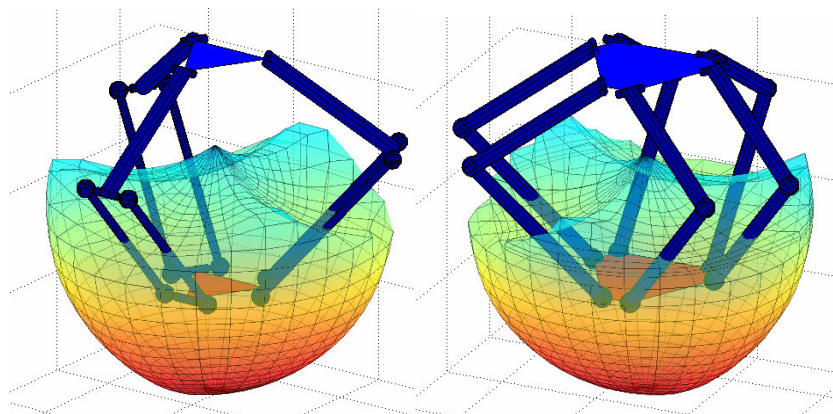
As velocidades e acelerações dos modelos cujos dados foram encontrados estão consolidadas na Tabela 2.

Modelo	Velocidade Máxima	Aceleração Máxima
Delta 720 com <i>direct drive</i>	?	~ 51 G
Delta da ABB (IRB 340)	10 m/s	~ 10 G
Hexa da Toyoda	6 m/s	~ 22 G
Hexa do IWF	5 m/s	~ 10 G
H4 do LIRMM	5 m/s	~ 10 G

**Tabela 2:** Tabela comparativa de velocidades e acelerações máximas de modelos de arquiteturas paralelas para pick-and-place / montagens leves.

Na velocidade máxima destaca-se o Delta da ABB, seguido do Hexa da Toyoda. Embora não se tenha a informação do Delta 720, o modelo também deve ter bons resultados. Para as acelerações, destacam-se o Delta 720 e o Hexa da Toyoda.

Não se encontrou informações a respeito do volume de trabalho da H4 para orientações iguais a zero (apenas o esquema simplista da Fig. 23). Comparando as arquiteturas Delta e Hexa para dimensões de máquina semelhantes, percebe-se que o tamanho do volume praticamente não muda (Fig. 27).



**Figura 27:** Volumes de trabalho da Delta (à esquerda) e da Hexa (à direita) para tamanhos de máquina semelhantes (IFM, 2005)

A arquitetura Delta, já que possui apenas três graus de liberdade de translação, não pode se inclinar em nenhuma direção. A H4 pode se inclinar em torno de um eixo que pode ser escolhido na concepção da máquina e a Hexa, em qualquer direção.

Quanto à rigidez, não foram encontrados dados numéricos, mas é possível fazer uma análise qualitativa. A arquitetura Delta possui três cadeias cinemáticas e cada uma é composta de duas partes: uma com dois ligamentos e outra com apenas um. O mesmo vale para a H4, exceto que possui mais uma cadeia. A Hexa possui seis cadeias, mas com apenas dois ligamentos em cada conjunto. Seguindo este raciocínio, a Hexa deve ser a mais rígida, podendo dividir melhor os esforços externos com as suas seis cadeias cinemáticas, seguida da H4 e da Delta. A alta rigidez da Hexa, tendo em vista sua estrutura, foi destacado por **Sato et al. (2003)**, que realizaram estudos com o modelo da Toyoda (não foram informados números).

Todas as arquiteturas, até mesmo por terem como base a arquitetura Delta, não devem apresentar valores muito diferentes de exatidão e de repetibilidade (em sendo do mesmo porte e utilizando elementos de qualidade compatíveis). As arquiteturas Delta cujos dados foram encontrados apresentam exatidão na ordem de 0.1 mm (com resolução de 0.01 mm). Nenhum dos dois protótipos da Hexa (IWF e Toyoda) teve suas exatidões medidas até hoje, segundo o próprio Pierrot que foi consultado, e o site do IWF.

Sabe-se que a dificuldade de projeto aumenta com o número de graus de liberdade. Problemas, como singularidades, vão ficando mais complexos de se tratar e de se resolver. Sendo assim, a Hexa é a que apresenta as maiores dificuldades e a Delta é a mais simples. Quanto à fabricação, montagem e calibração, novamente a mais complexa é a Hexa.

Quanto ao know-how, a mais famosa de todas e mais bem estudada com certeza é a Delta, além de ser a mais antiga. A Hexa está sendo estudada em alguns laboratórios e sabe-se que pelo menos uma empresa, a Toyoda, demonstrou interesse na máquina. A H4 ainda é muito recente.

#### 4.4.2. A Arquitetura Escolhida

De acordo com os comentários descritos anteriormente, pode-se chegar a um breve resumo das maiores vantagens e desvantagens de cada arquitetura:

**Delta** *Prós*: Altas velocidades e acelerações, eficácia comprovada na indústria.

*Contras*: Apenas três graus de liberdade.

**Hexa** *Prós*: Altas velocidades e acelerações, seis graus de liberdade, bom nível de rigidez.

*Contras*: Complexa.

**H4** *Prós*: Bom equilíbrio entre capacidade de movimentos e complexidade.

*Contras*: Ainda foi pouco estudada.

Para este trabalho, resolveu-se aceitar o desafio de estudar, modelar e otimizar a arquitetura Hexa. Acredita-se que, das arquiteturas sugeridas para operações de *pick-and-place* e montagens leves, a Hexa é a que deve ter a melhor combinação desempenho / versatilidade, embora ainda sejam necessários maiores estudos devido a sua complexidade elevada.

#### 4.4.3. Uma Breve Revisão da Literatura da Hexa

**Pierrot et al., (1991a e 1991b)** apresentam a arquitetura Hexa e exploram o modelo da cinemática inversa e o modelo dinâmico. Na cinemática inversa utilizam relações geométricas que descrevem a arquitetura para criar o modelo, cuja solução fornece o valor dos ângulos de atuação. Na dinâmica inversa, os ligamentos foram tratados como duas massas de mesmo valor concentradas na ponta dos ligamentos. Dessa forma é como se os ligamentos passivos não existissem. Esse modelo simplificado também foi utilizado por **Uchiyama et al. (1994)**.

Como na maioria dos robôs desse tipo (seis graus de liberdade e arquitetura paralela), a Hexa também apresenta alguns problemas como singularidades em seu volume de trabalho. Alguns trabalhos optam por limitar o volume de trabalho de uma

máquina para que ela não encontre pontos de singularidade em sua trajetória. **Queiroz et al. (2005)** sugerem a utilização de um algoritmo “supervisor” dos movimentos da máquina, monitorando-a através de um indicador. Quando o gradiente do indicador indica que a máquina está se aproximando de uma singularidade, ele faz com que ela altere levemente seu curso (mexendo em um ou mais graus de liberdade) e evitando o ponto.

Na questão de calibração, ainda um ponto controverso e centro de muitos estudos, **Last et al. (2005)** sugerem um método de auto-calibração para a Hexa, ou seja, ao invés de se utilizar equipamentos externos para medir uma postura da plataforma móvel com relação a uma posição de atuadores, os pesquisadores utilizaram sensores extras nas juntas passivas permitindo alguns dados redundantes (redundantes pelo menos no modelo teórico). A partir desses dados de posição extra, é possível buscar uma minimização de uma função de resíduos (erros). Para a cinemática direta, apresentada também nesse mesmo trabalho (**Last et al., 2005**), foi utilizado um método iterativo que usa a cinemática inversa. O método é semelhante ao de **Maurine et al. (1999)**, também executado com a Hexa.

Tendo em vista alguns problemas ainda existentes no equacionamento da maioria das arquiteturas paralelas, principalmente as de seis graus de liberdade, como a falta de uma solução analítica da cinemática direta, **Bruyninckx (1997)** sugere uma nova arquitetura chamada 321 – Hexa, onde a plataforma móvel possui apenas três vértices, sendo que em um desses pontos são acoplados três ligamentos passivos, a um outro ponto, dois ligamentos passivos e no último ponto, apenas um ligamento (3 2 1). Essa nova arquitetura passa a possuir solução analítica da cinemática direta.

Também existem outros pesquisadores que buscam outras aplicações para a Hexa. É o caso de **Morizono et al. (2003)** que, inspirados na arquitetura, construíram a W-Hexa (W de *Wearable*, que significa “pode vestir ou usar”). A W-Hexa é um robô para ser utilizado como um exoesqueleto numa articulação de um ombro humano.

No âmbito do controle, **Uchiyama et al. (1996)** apresentam sua estratégia de controle baseada num modelo de flexibilidade que utiliza apenas os dados dos sensores dos motores da Hexa. Levam em consideração alguns itens como o atrito, principalmente a baixas velocidades. É ressaltado que uma das utilidades do modelo

de fricção é que se pode aumentar a exatidão da máquina quando em movimento livre.

**Sato et al. (2001)** apresentam um ambiente *3D* para simulações, testes e programação off-line da Hexa. Neste trabalho, o sistema de controle utilizado contempla de forma integrada posição, força e rigidez. São apresentadas também duas operações básicas da máquina: girar uma manivela e acompanhar uma superfície (no caso, um plano inclinado). São apresentados os resultados para diferentes velocidades de execução das atividades. **Sato et al. (2003)** apresentam uma continuação do trabalho anterior de 2001 onde o foco são atividades complexas sendo executadas rapidamente. São apresentados erros de posicionamento ao longo do tempo nas diferentes coordenadas da máquina, bem como a força de iteração ferramenta – ambiente, onde são impostos limites. Foram reportados alguns problemas devido aos erros geométricos entre o sistema real e o sistema gráfico.

## 5. ANÁLISE CINEMÁTICA INVERSA

Nesse capítulo serão discutidas as equações concernentes à cinemática inversa da máquina e ao seu volume de trabalho. O ambiente virtual para simulações apresentado no capítulo de resultados toma como base principal as equações apresentadas aqui. Fica também como referência o Apêndice C, onde são apresentadas questões específicas do simulador como translações e rotações no espaço 3D, coordenadas homogêneas, transformações de coordenadas, ângulo de *roll-pitch-yaw*, trajetórias lineares e circulares, dentre outros tópicos.

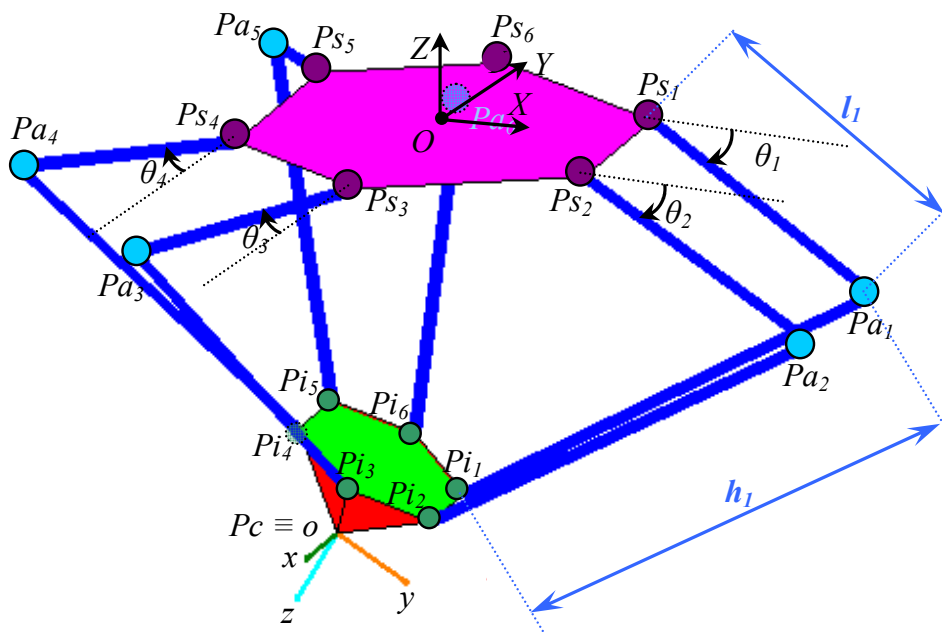
### 5.1. O Modelo da Cinemática Inversa da Hexa: Análise de Posições

As equações da cinemática inversa relacionam a posição do efetuador com as coordenadas dos atuadores. A Fig. 28 mostra um diagrama esquemático da arquitetura Hexa. O conjunto de pontos apresentados como  $P_{ij} = \{P_{i1}, P_{i2}, \dots, P_{i6}\}$  representa os centros das juntas (ou articulações) esféricas que unem os ligamentos passivos à plataforma móvel. O conjunto de pontos  $Pa_j = \{Pa_1, Pa_2, \dots, Pa_6\}$  representa os centros das juntas esféricas que unem os ligamentos passivos aos ativos. O conjunto de pontos  $Ps_j = \{Ps_1, Ps_2, \dots, Ps_6\}$  representa os centros das juntas de revolução atuadas. O ponto  $Pc$  é o ponto em que todos os movimentos são referenciados; o ponto de interesse do efetuador (ou ferramenta). O sistema de coordenadas  $o-xyz$  possui origem em  $Pc$  e é solidário à plataforma móvel e o sistema  $O-XYZ$  é fixo e global, tendo sua coordenada  $X$  perpendicular à linha  $\overline{Ps_1Ps_2}$  e seu centro  $O$  é igual ao centro geométrico da base fixa. Este ponto poderá ser deslocado com relação à coordenada  $Z$  quando conveniente, mas as posições  $X$  e  $Y$  são fixas. Quando a plataforma móvel está em sua posição nominal ( $X = 0$  e  $Y = 0$  e paralela à base fixa), tem-se que os eixos  $X$  e  $y$  ficam paralelos e ambos apontam para a mesma direção. Os ligamentos ativos e passivos são representados respectivamente por  $\mathbf{l}_j = \{\mathbf{l}_1, \mathbf{l}_2, \dots, \mathbf{l}_6\}$  e  $\mathbf{h}_j = \{\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_6\}$ , destacados em negrito por serem considerados



vetores ou matrizes. Se escritos em *itálico*, serão considerados os módulos destes vetores. O lado da plataforma móvel (que é um hexágono de lados iguais) tem valor igual a  $t$ .

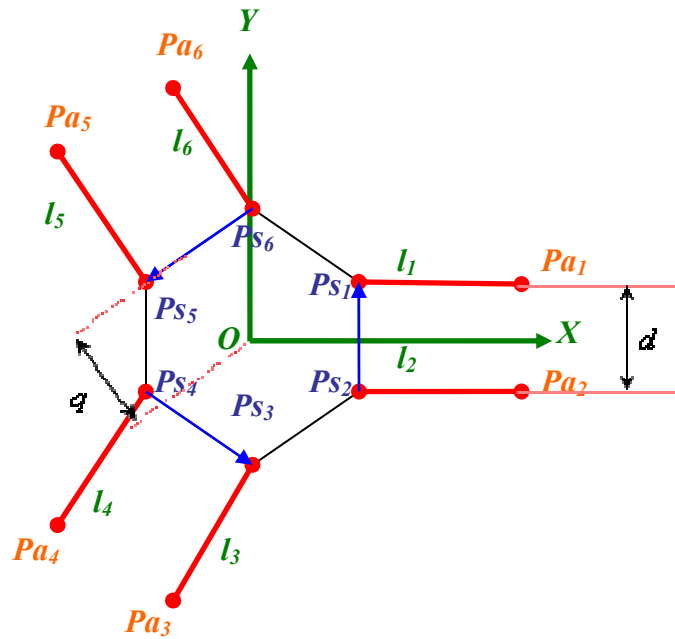
A Fig. 29 mostra em maiores detalhes a base fixa e ilustra as variáveis  $d$  e  $q$ . A primeira representa o módulo das distâncias  $\overline{Ps_1Ps_2} = \overline{Ps_3Ps_4} = \overline{Ps_5Ps_6}$ . A segunda representa o módulo da linha que une qualquer um dos centros das distâncias citadas anteriormente (variável  $d$ ) ao centro da base fixa  $O$ .



**Figura 28:** Esquema detalhado da arquitetura Hexa.

Os ângulos de atuação, definidos por  $\theta_j = \{\theta_1, \theta_2, \dots, \theta_6\}$ , valem zero quando o respectivo ligamento atuado está contido no mesmo plano que a base fixa. Estes são contados conforme também mostram as Fig. 28 e Fig. 29, seguindo a regra da mão direita e utilizando os vetores demonstrados (eixos de rotação dos motores).

Seja também a notação para as coordenadas dos pontos  $Ps_j$  escritos na base fixa  $O-XYZ$  definida como  $Ps_j = [Ps_{j(x)} \ Ps_{j(y)} \ Ps_{j(z)}]^T$ . A notação dos outros pontos é análoga.



**Figura 29:** Base fixa da Hexa (o eixo  $Y$  não necessariamente passa pelo ponto 6).

Tendo em vista os dados fornecidos aqui e nos itens anteriores, chega-se ao seguinte conjunto de equações por pura análise geométrica:

$$\begin{bmatrix} Pa_{1(X)} \\ Pa_{1(Y)} \\ Pa_{1(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{1(X)} \\ Ps_{1(Y)} \\ Ps_{1(Z)} \end{bmatrix} + \begin{bmatrix} l_1 \cos \theta_1 \\ 0 \\ -l_1 \sin \theta_1 \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} Pa_{2(X)} \\ Pa_{2(Y)} \\ Pa_{2(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{2(X)} \\ Ps_{2(Y)} \\ Ps_{2(Z)} \end{bmatrix} + \begin{bmatrix} l_2 \cos \theta_2 \\ 0 \\ -l_2 \sin \theta_2 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} Pa_{3(X)} \\ Pa_{3(Y)} \\ Pa_{3(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{3(X)} \\ Ps_{3(Y)} \\ Ps_{3(Z)} \end{bmatrix} + \begin{bmatrix} -l_3 \cos 60^\circ \cos \theta_3 \\ -l_3 \sin 60^\circ \cos \theta_3 \\ -l_3 \sin \theta_3 \end{bmatrix} \quad (3)$$

$$\begin{bmatrix} Pa_{4(X)} \\ Pa_{4(Y)} \\ Pa_{4(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{4(X)} \\ Ps_{4(Y)} \\ Ps_{4(Z)} \end{bmatrix} + \begin{bmatrix} -l_4 \cos 60^\circ \cos \theta_4 \\ -l_4 \sin 60^\circ \cos \theta_4 \\ -l_4 \sin \theta_4 \end{bmatrix} \quad (4)$$

$$\begin{bmatrix} Pa_{5(X)} \\ Pa_{5(Y)} \\ Pa_{5(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{5(X)} \\ Ps_{5(Y)} \\ Ps_{5(Z)} \end{bmatrix} + \begin{bmatrix} -l_5 \cos 60^\circ \cos \theta_5 \\ l_5 \sin 60^\circ \cos \theta_5 \\ -l_5 \sin \theta_5 \end{bmatrix} \quad (5)$$

$$\begin{bmatrix} Pa_{6(X)} \\ Pa_{6(Y)} \\ Pa_{6(Z)} \end{bmatrix} = \begin{bmatrix} Ps_{6(X)} \\ Ps_{6(Y)} \\ Ps_{6(Z)} \end{bmatrix} + \begin{bmatrix} -l_6 \cos 60^\circ \cos \theta_6 \\ l_6 \sin 60^\circ \cos \theta_6 \\ -l_6 \sin \theta_6 \end{bmatrix} \quad (6)$$

As eq. (1) a eq. (6) estão apresentadas conforme a ordem escolhida de numeração das cadeias cinemáticas da Hexa. Utilizando então a fórmula de distância entre dois pontos, tem-se que:

$$h_j^2 = (Pa_{j(X)} - Pi_{j(X)})^2 + (Pa_{j(Y)} - Pi_{j(Y)})^2 + (Pa_{j(Z)} - Pi_{j(Z)})^2 \quad (7)$$

$$h_1^2 = (Ps_{1(X)} - Pi_{1(X)} + l_1 \cos \theta_1)^2 + (Ps_{1(Y)} - Pi_{1(Y)})^2 + (Ps_{1(Z)} - Pi_{1(Z)} - l_1 \sin \theta_1)^2 \quad (8)$$

$$h_2^2 = (Ps_{2(X)} - Pi_{2(X)} + l_2 \cos \theta_2)^2 + (Ps_{2(Y)} - Pi_{2(Y)})^2 + (Ps_{2(Z)} - Pi_{2(Z)} - l_2 \sin \theta_2)^2 \quad (9)$$

$$h_3^2 = (Ps_{3(X)} - Pi_{3(X)} - l_3 \cos 60^\circ \cos \theta_3)^2 + (Ps_{3(Y)} - Pi_{3(Y)} - l_3 \sin 60^\circ \cos \theta_3)^2 + (Ps_{3(Z)} - Pi_{3(Z)} - l_3 \sin \theta_3)^2 \quad (10)$$

$$h_4^2 = \left( P_{S_{4(X)}} - P_{i_{4(X)}} - l_4 \cos 60^\circ \cos \theta_4 \right)^2 + \left( P_{S_{4(Y)}} - P_{i_{4(Y)}} - l_4 \sin 60^\circ \cos \theta_4 \right)^2 + \left( P_{S_{4(Z)}} - P_{i_{4(Z)}} - l_4 \sin \theta_4 \right)^2 \quad (11)$$

$$h_5^2 = \left( P_{S_{5(X)}} - P_{i_{5(X)}} - l_5 \cos 60^\circ \cos \theta_5 \right)^2 + \left( P_{S_{5(Y)}} - P_{i_{5(Y)}} + l_5 \sin 60^\circ \cos \theta_5 \right)^2 + \left( P_{S_{5(Z)}} - P_{i_{5(Z)}} - l_5 \sin \theta_5 \right)^2 \quad (12)$$

$$h_6^2 = \left( P_{S_{6(X)}} - P_{i_{6(X)}} - l_6 \cos 60^\circ \cos \theta_6 \right)^2 + \left( P_{S_{6(Y)}} - P_{i_{6(Y)}} + l_6 \sin 60^\circ \cos \theta_6 \right)^2 + \left( P_{S_{6(Z)}} - P_{i_{6(Z)}} - l_6 \sin \theta_6 \right)^2 \quad (13)$$

Aplicando-se a eq. (7) nas seis cadeias cinemáticas obtêm-se as equações (8) a (13). Trabalhando-se algebricamente as equações (8) a (13), trabalho este apresentado no Apêndice A, tem-se que a solução de cada uma das relações para as variáveis  $\theta_j$  é equivalente à solução da eq. (14):

$$a = b \cdot \cos \theta + c \cdot \sin \theta \quad (14)$$

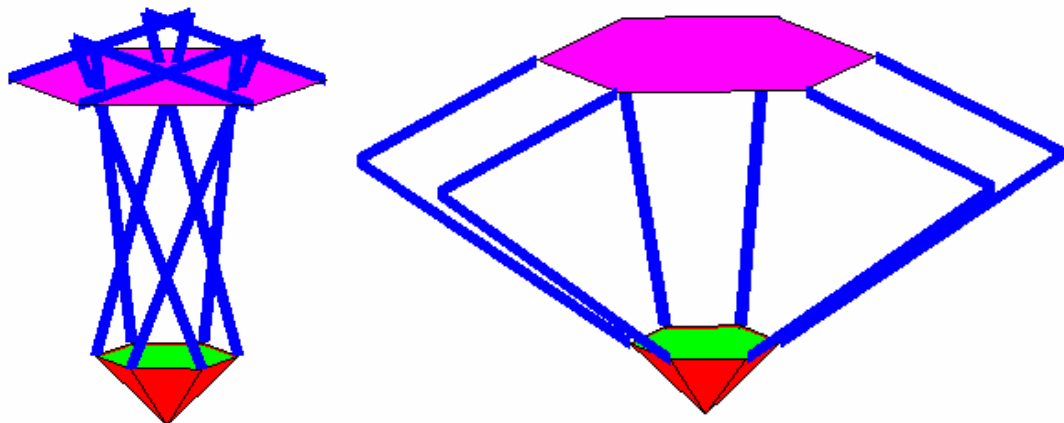
Utilizando o manipulador simbólico do Matlab, tem-se que a eq. (14) apresenta duas respostas. Podem parecer quatro devido ao arco tangente, mas o programa fornece a solução utilizando a função *atan2*, que recebe o numerador e o denominador – seno e co-seno – como entradas, definindo um único ângulo no espaço entre  $[-180^\circ; 180^\circ]$ :

$$\theta = \arctan \left[ \frac{\left[ a \cdot c^2 + b \cdot \sqrt{-c^2 \cdot (a^2 - b^2 - c^2)} \right] / \left[ c \cdot (b^2 + c^2) \right]}{\left[ a \cdot b - \sqrt{-c^2 \cdot (a^2 - b^2 - c^2)} \right] / (b^2 + c^2)} \right] \quad (15)$$

$$\theta = \arctan \left[ \frac{\left[ a \cdot c^2 - b \cdot \sqrt{-c^2 \cdot (a^2 - b^2 - c^2)} \right] / \left[ c \cdot (b^2 + c^2) \right]}{\left[ a \cdot b + \sqrt{-c^2 \cdot (a^2 - b^2 - c^2)} \right] / (b^2 + c^2)} \right] \quad (16)$$

Uma das equações, a eq. (15), refere-se à solução com os ligamentos voltados para dentro, o que não interessa ao modelo. A solução viável é a da eq. (16). Ambas as soluções estão demonstradas na Fig. 30. A maior robustez da solução do manipulador simbólico isenta da programação considerações adicionais quando em diferentes quadrantes. As equações (15) e (16) estão apresentadas com seu numerador e denominador segundo resposta do próprio Matlab.

A eq. (14) também pode ser resolvida algebricamente. A literatura costuma utilizar a tangente do meio arco de atuação como forma de escrever as funções seno e co-seno sob uma única função (tangente do meio arco), que depois é tratada como variável de uma equação de segundo grau. Contudo, esta solução apresenta uma desvantagem, que é o dobro de soluções. Elas devem ser analisadas para que sempre se tenha a solução real da máquina, não importando o quadrante do ângulo de atuação. Vide o Apêndice A para maiores detalhes e para a dedução algébrica.



**Figura 30:** As soluções da cinemática inversa da Hexa. A figura da esquerda é a solução eliminada.

Para o caso particular de cada uma das seis equações que compõem o sistema do robô, tem-se que:

1) Para as cadeias 1 e 2 – eq. (8) e eq. (9)

$$\begin{aligned}
 a &= \frac{h_j^2 - l_j^2 - (Ps_{j(X)} - Pi_{j(X)})^2 - (Ps_{j(Y)} - Pi_{j(Y)})^2 - (Ps_{j(Z)} - Pi_{j(Z)})^2}{2l_j} \\
 b &= (Ps_{j(X)} - Pi_{j(X)}) \\
 c &= -(Ps_{j(Z)} - Pi_{j(Z)})
 \end{aligned} \tag{17}$$

2) Para as cadeias 3 e 4 – eq. (10) e eq. (11)

$$\begin{aligned}
 a &= \frac{h_j^2 - l_j^2 - (Ps_{j(X)} - Pi_{j(X)})^2 - (Ps_{j(Y)} - Pi_{j(Y)})^2 - (Ps_{j(Z)} - Pi_{j(Z)})^2}{l_j} \\
 b &= -\left[ \sqrt{3}(Ps_{j(Y)} - Pi_{j(Y)}) + (Ps_{j(X)} - Pi_{j(X)}) \right] \\
 c &= -2(Ps_{j(Z)} - Pi_{j(Z)})
 \end{aligned} \tag{18}$$

3) Para as cadeias 5 e 6 – eq. (12) e eq. (13)

$$\begin{aligned}
 a &= \frac{h_j^2 - l_j^2 - (Ps_{j(X)} - Pi_{j(X)})^2 - (Ps_{j(Y)} - Pi_{j(Y)})^2 - (Ps_{j(Z)} - Pi_{j(Z)})^2}{l_j} \\
 b &= -\left[ \sqrt{3}(Ps_{j(Y)} - Pi_{j(Y)}) - (Ps_{j(X)} - Pi_{j(X)}) \right] \\
 c &= -2(Ps_{j(Z)} - Pi_{j(Z)})
 \end{aligned} \tag{19}$$

Observa-se que o modelo é robusto o suficiente para levar em consideração quaisquer posições dos pontos de interesse da base fixa e da base móvel, com comprimentos quaisquer dos doze ligamentos envolvidos. O modelo pode ser rapidamente resolvido, o que implica na possibilidade de ser utilizado em tempo real. Apenas duas variáveis ainda não estão definidas:  $Ps_j$  e  $Pi_j$ , que estão descritas a seguir. As equações (20) a (25) apresentam as coordenadas dos pontos da base fixa. As coordenadas  $Z$  destes pontos são iguais entre si e conforme mencionado anteriormente, apenas estas podem mudar dependendo do referencial o qual  $O-XYZ$  é colocado (sempre que ocorrer alguma conveniência para isto).

$$\begin{bmatrix} Ps_{1(X)} \\ Ps_{1(Y)} \\ Ps_{1(Z)} \end{bmatrix} = \begin{bmatrix} q \\ d/2 \\ Z \end{bmatrix} \quad (20)$$

$$\begin{bmatrix} Ps_{2(X)} \\ Ps_{2(Y)} \\ Ps_{2(Z)} \end{bmatrix} = \begin{bmatrix} q \\ -d/2 \\ Z \end{bmatrix} \quad (21)$$

$$\begin{bmatrix} Ps_{3(X)} \\ Ps_{3(Y)} \\ Ps_{3(Z)} \end{bmatrix} = \begin{bmatrix} -q \sin(30^\circ) + (d/2) \cos(30^\circ) \\ -q \cos(30^\circ) - (d/2) \sin(30^\circ) \\ Z \end{bmatrix} \quad (22)$$

$$\begin{bmatrix} Ps_{4(X)} \\ Ps_{4(Y)} \\ Ps_{4(Z)} \end{bmatrix} = \begin{bmatrix} -q \sin(30^\circ) - (d/2) \cos(30^\circ) \\ -q \cos(30^\circ) + (d/2) \sin(30^\circ) \\ Z \end{bmatrix} \quad (23)$$

$$\begin{bmatrix} Ps_{5(X)} \\ Ps_{5(Y)} \\ Ps_{5(Z)} \end{bmatrix} = \begin{bmatrix} -q \sin(30^\circ) - (d/2) \cos(30^\circ) \\ q \cos(30^\circ) - (d/2) \sin(30^\circ) \\ Z \end{bmatrix} \quad (24)$$

$$\begin{bmatrix} Ps_{6(X)} \\ Ps_{6(Y)} \\ Ps_{6(Z)} \end{bmatrix} = \begin{bmatrix} -q \sin(30^\circ) + (d/2) \cos(30^\circ) \\ q \cos(30^\circ) + (d/2) \sin(30^\circ) \\ Z \end{bmatrix} \quad (25)$$

As coordenadas dos pontos  $Pi_j$  são apresentadas relativamente a  $Pc$ . Existe, contudo, a necessidade de considerar translações e rotações. Nesses casos, podem ser aplicadas a todos os pontos as mesmas operações, carregando-se sempre **uma matriz de sete pontos** os quais definem a plataforma móvel. A Fig. 31 e as equações (26) a (28) ilustram as coordenadas relativas dos pontos  $Pi_j$  com relação a  $Pc$ .

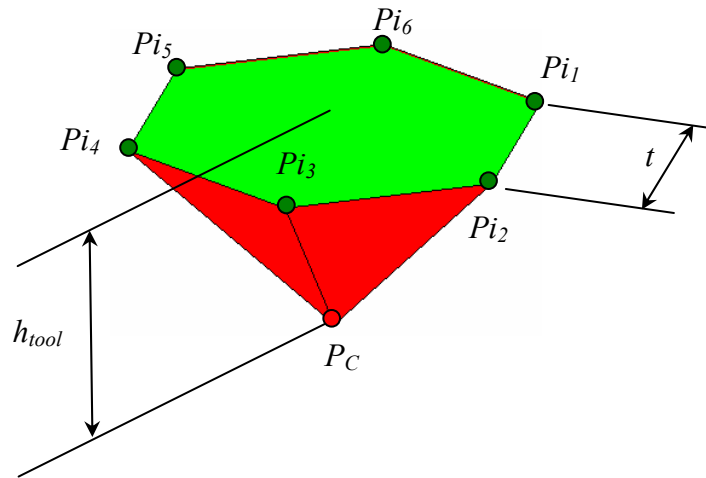


Figura 31: Modelo da Plataforma Móvel da Hexa.

$$\begin{bmatrix} Pi_{1(X)} \\ Pi_{1(Y)} \\ Pi_{1(Z)} \end{bmatrix} - \begin{bmatrix} Pc_{(X)} \\ Pc_{(Y)} \\ Pc_{(Z)} \end{bmatrix} = \begin{bmatrix} t\sqrt{3}/2 \\ t/2 \\ h_{tool} \end{bmatrix} \quad (26)$$

$$\begin{bmatrix} Pi_{2(X)} \\ Pi_{2(Y)} \\ Pi_{2(Z)} \end{bmatrix} - \begin{bmatrix} Pc_{(X)} \\ Pc_{(Y)} \\ Pc_{(Z)} \end{bmatrix} = \begin{bmatrix} t\sqrt{3}/2 \\ -t/2 \\ h_{tool} \end{bmatrix} \quad (27)$$

$$\begin{bmatrix} Pi_{3(X)} \\ Pi_{3(Y)} \\ Pi_{3(Z)} \end{bmatrix} - \begin{bmatrix} Pc_{(X)} \\ Pc_{(Y)} \\ Pc_{(Z)} \end{bmatrix} = \begin{bmatrix} 0 \\ -t \\ h_{tool} \end{bmatrix} \quad (28)$$

Para os pontos  $Pi_4$  a  $Pi_6$  as equações são as mesmas de  $Pi_1$  a  $Pi_3$ , respectivamente, exceto que no lado direito delas todos os sinais para as coordenadas  $X$  e  $Y$  são invertidos.

As questões relativas à posição e à orientação da plataforma são tratadas no item a seguir, já que são de interesse direto do simulador virtual.



## 5.2. Movimentações da Plataforma no Espaço Tridimensional: Questões do Simulador Virtual

O cálculo das posições dos pontos  $P_{ij}$  e do próprio ponto  $P_c$  dependem do conhecimento de coordenadas homogêneas, translações tridimensionais e rotações em torno de um eixo qualquer, tópicos estes bastante conhecidos, ficando no Apêndice C como referência.

Quando se inicia um programa de simulação virtual, têm-se como parâmetros de entrada, além da geometria da máquina, a posição inicial de  $P_c$  e a orientação da plataforma móvel (neste estudo, a posição inicial considerada é sempre o ponto zero para todas as coordenadas do sistema global fixo e as orientações são nulas). Assim, logo na abertura do programa, a matriz dos sete pontos característicos da plataforma é calculada, sendo  $P_c$  dado e  $P_{ij}$  calculados através das equações de (26) a (28).

Para realizar uma *translação* de uma distância  $dt(x,y,z)$  qualquer, aplica-se a matriz de translação a cada um dos sete pontos característicos (a matriz de translação apenas soma  $dt$  a cada ponto).

Para realizar uma *rotação* em torno de um eixo qualquer, definem-se o centro de rotação  $C$ , o eixo de rotação (através de dois ângulos) e o grau de rotação (um ângulo). Definidos estes parâmetros, a matriz de rotação é calculada. Assim, para cada ponto característico (incluindo  $P_c$ ), aplica-se essa matriz de rotação que dará como resultado uma nova matriz de pontos característicos. Os ângulos com relação aos eixos globais são dados por ângulos *roll-pitch-yaw* (RPY), que foram utilizados apenas para visualização ou para quando o usuário desejar centrar a plataforma móvel com respeito a rotações (tornar zero algum ou todos os ângulos RPY).

## 5.3. Volume de Trabalho

Este item tem por objetivos apresentar os diferentes tipos de volumes de trabalho para o caso dos robôs de arquitetura paralela e definir métodos para o cálculo dos mesmos. Obviamente, um foco maior será dado à arquitetura Hexa.

No caso dos robôs seriais de seis graus de liberdade, com os três últimos graus compondo um punho esférico, o volume de trabalho é usualmente representado pelo espaço alcançável pelo centro do punho. Considera-se então que os três últimos graus de liberdade apenas orientam o efetuador.

No caso dos robôs totalmente paralelos de seis graus de liberdade essa separação não é possível, já que os graus de liberdade de translação e de rotação são inseparáveis com relação aos atuadores. Assim, devem-se buscar algoritmos que calculem o volume, dadas algumas características do mesmo.

Robôs de arquitetura paralela têm seus movimentos restritos por três diferentes fatores: limites mecânicos das juntas passivas (não atuadas), limites dos atuadores e interferência entre os ligamentos e partes móveis.

Um exemplo muito claro da limitação das juntas passivas está nos punhos esféricos, juntas universais e articulações esféricas, onde existe um ângulo de rotação máximo que limita o movimento do robô como um todo. Exemplos de limites impostos por atuadores são: comprimento de pistões, comprimento de guias lineares, movimentos angulares limitados e muitos outros.

### **5.3.1. Tipos de Volume de Trabalho**

Primeiro deve-se considerar o ponto  $P_c$  de interesse, solidário ao efetuador, que é o ponto de referência. Considera-se também um conjunto de fatores que tornem possível conhecer a orientação do efetuador.

Conforme explicado anteriormente, faz-se necessário o uso de classificações para diferentes representações do volume de trabalho de robôs de arquitetura paralela. Dentre os vários tipos possíveis, existem algumas classificações mais utilizadas:

- *De orientação constante* ou *translacional*: é definido pelo conjunto de pontos do espaço que podem ser alcançados por  $P_c$ , tendo-se constante a orientação do efetuador;

- *De posição constante ou rotacional*: De forma análoga ao primeiro, define-se o volume de trabalho de posição constante, onde se buscam todas as orientações possíveis do efetuador para um ponto  $P_c$  estacionário;
- *Máximo ou alcançável*: é definido pelo conjunto de pontos do espaço que podem ser alcançados por  $P_c$ , com pelo menos uma orientação do efetuador (ou plataforma móvel). Produz o mapa mais completo da máquina, embora de difícil análise;
- *De orientação limitada*: é definido pelo conjunto de pontos que podem ser alcançados por  $P_c$ , com pelo menos uma orientação pertencente a um conjunto pré-definido de orientações;
- *De orientação total*: é definido pelo conjunto de pontos que podem ser alcançados por  $P_c$  com todas as orientações pertencentes a um conjunto pré-definido de orientações;
- *Ágil*: nos pontos que compõem este volume, a plataforma pode assumir qualquer orientação de  $0^\circ$  a  $360^\circ$  (um caso particular do volume de orientação total);
- *Especializados*: nesses volumes busca-se alguma coisa muito específica. Um exemplo deste caso seria o conjunto de pontos onde o fator de transmissão de todos os ligamentos é maior ou igual a um valor  $k$  (fator de transmissão de um ligamento é definido como o co-seno do ângulo existente entre o vetor da força aplicada pelo ligamento na plataforma e o vetor velocidade da plataforma. Esse parâmetro varia de 0 a 1, obviamente).

Vai-se ater aos volumes de *orientação constante* e *orientação total*, por serem os mais úteis nas análises para as aplicações escolhidas.

### 5.3.2. Algoritmo para Cálculo do Volume de Trabalho

O método mais simples que pode ser utilizado no cálculo de volumes de trabalho é definir um volume suficientemente grande no espaço, dividi-lo numa malha tridimensional e então verificar para cada nó da malha se as condições específicas daquele tipo de volume são respeitadas ponto a ponto. Obviamente este método demanda de um enorme custo computacional dependendo do passo desejado para a malha. O custo aumenta cubicamente para uma melhora uniforme (nas três dimensões) da resolução do volume obtido. Sendo assim, será utilizado um método semelhante ao proposto por **Conti et al. (1997)**, que lembra um método de *dicotomia*.

Seguindo então o método de Conti, com algumas alterações, têm-se os seguintes passos:

- 1) Definir as restrições desejadas (colisão entre ligamentos, limites das articulações da máquina, colisão com a plataforma móvel etc.);
- 2) Criar uma função “*Testar Postura*”, que retorne *verdadeiro* se a plataforma móvel consegue atingir tal postura (posição de  $P_c$  + orientação) sem violar as restrições e *falso* caso contrário;
- 3) Definir o erro máximo aceitável  $\varepsilon$  que também serve como condição de parada do algoritmo. O erro representa o raio da esfera definida ao redor do ponto correto real com que o ponto em questão do volume pode ser encontrado. Contudo, o ponto não poderá constar em toda a região da esfera, pois obrigatoriamente ele respeitará a função que testa a postura. A esfera contém parte de seu volume fora do volume de trabalho;
- 4) Definir o raio inicial  $R$ , que será o valor inicial para  $\rho$ , o raio de busca.  $R$  também representa o limite máximo de busca e deve ser grande o suficiente para permitir uma busca completa. Inicialmente recomenda-se utilizar como limite máximo  $R = 2 * (h + l)$ , contudo, a expressão é arbitrária;
- 5) Determinar um ponto central a partir do qual serão feitas buscas utilizando-se coordenadas polares. Para determiná-lo, a plataforma é centrada, ou seja,  $X$  e  $Y$  valem zero para o ponto  $P_c$ . Variando-se apenas a coordenada  $Z$  a partir de uma altura arbitrária (coordenada  $Z$  de  $P_c$ ) e utilizando-se a rotina de

dicotomia da Fig. 32 e a função de testar posturas, encontram-se o pico e o vale da posição central (admite-se o volume como sendo totalmente fechado e sem vazios internos). O ponto central pode ser qualquer um entre o pico e o vale. Recomenda-se a média aritmética das coordenadas  $Z$  destes pontos para que a resolução em cada célula do volume seja mais ou menos uniforme. Caso o ponto central seja muito próximo do pico ou do vale, corre-se o risco de despender metade do algoritmo numa área pequena da borda do volume de trabalho, que ficará muito detalhada em detrimento do resto;

- 6) Quando o ponto central está sendo calculado, o algoritmo deve estar preparado para responder a casos onde apenas um ponto é encontrado. Isso pode ser resultado de chutes iniciais onde o ponto central ficou abaixo ou acima do volume de trabalho existente. Por isso só se encontra o pico ou só o vale. Assim, basta ser acrescentada uma nova busca a partir do ponto encontrado: um dos “lados” retornará vazio e o outro o ponto que faltou;
- 7) Determinar os passos para  $\varphi$  e  $\psi$ , que são os ângulos diretores do vetor de busca, em coordenadas polares, que tem comprimento  $\rho$ . Depois, inicia-se o algoritmo de busca. Devem ser utilizados  $0^\circ \leq \varphi \leq 180^\circ$  e  $0^\circ \leq \psi \leq 360^\circ$ . Para cada par  $[\varphi, \psi]$  o algoritmo irá iniciar com  $\rho = R$  e aplicar a função *Testar Postura*. Se retornar *verdadeiro*,  $\rho$  será incrementado  $R/2$ , se *falso*, será subtraído  $R/2$ . O algoritmo segue sempre dividindo o fator por dois ( $R/4$ ,  $R/8$ , etc.) até que seja menor ou igual a  $\varepsilon$  e *Testar Postura* retorne *verdadeiro*. Devido ao tamanho de  $R$  a primeira iteração da função deverá sempre retornar *falso*. O processo continua até que todos os pontos da borda sejam encontrados: os pontos  $B(\varphi, \psi, \rho)$ . Caso se esteja buscando o volume de orientação total, deve ser utilizada pelo menos a simetria da Hexa com relação ao plano  $XZ$ , poupando 50% do custo computacional ( $\psi$  varia entre  $0^\circ$  e  $180^\circ$ ). Ver Fig. 32. A rigor, a Hexa possui três planos de simetria, permitindo-se utilizar  $\psi$  entre  $0^\circ$  e  $60^\circ$ , o que reduz o custo em pouco mais de 83%. Essas vantagens computacionais devem ser utilizadas para que se possa melhorar o número de orientações de teste da plataforma móvel na função *Testar Postura* (a orientação da plataforma consiste em três ângulos).

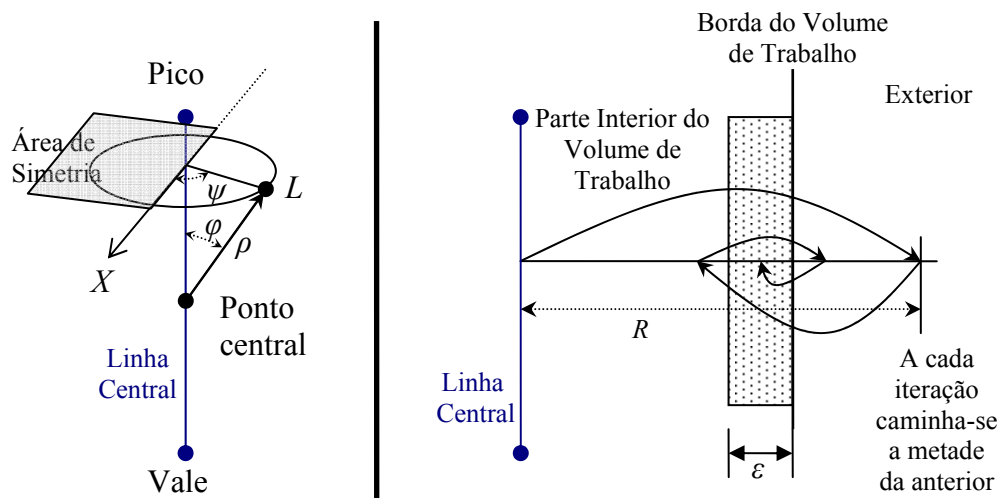


Figura 32: Algoritmo para cálculo do volume de trabalho.

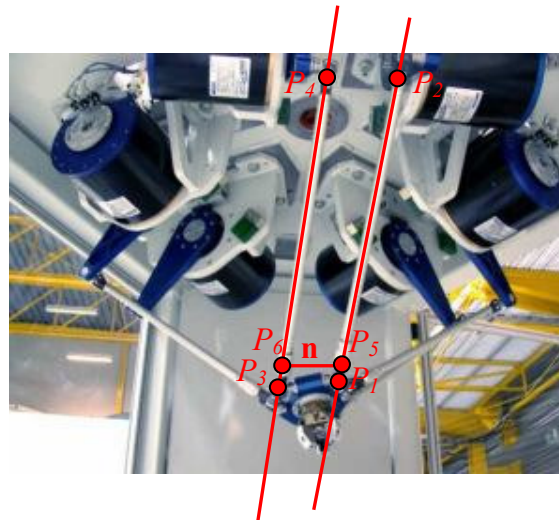
Quanto às restrições, algumas são impostas e outras são limites da própria geometria da máquina. Limites impostos são aqueles oriundos do tipo de volume de trabalho que se deseja, como, por exemplo, o de orientação limitada. Restrições geométricas são:

- Colisões entre os ligamentos;
- Colisões dos ligamentos com a plataforma móvel ou com a base e entre estas;
- Limites angulares das articulações atuadas e passivas, tanto aquelas entre os ligamentos quanto aquelas ligadas à plataforma móvel;

Devido ao elevado custo computacional, a segunda não foi aplicada neste trabalho, mas é conceitualmente idêntica à primeira, apresentada a seguir.

### 5.3.2.1. Detecção de Colisão entre os Ligamentos

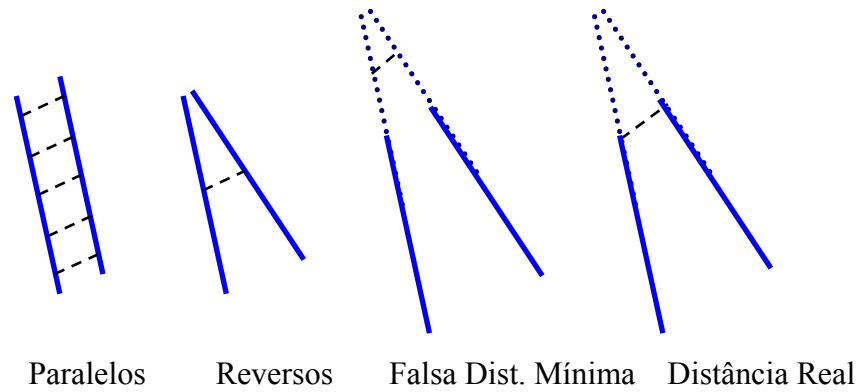
Seja a estrutura desenhada na Fig. 33. As extremidades dos ligamentos estão definidas por  $P_1$  e  $P_2$ , e  $P_3$  e  $P_4$ . A distância entre os dois ligamentos é  $n$  e o vetor  $\mathbf{n}$  é aquele que liga os pontos  $P_5$  e  $P_6$  (pontos dos ligamentos que se referem à distância mínima), apontando para  $P_5$ .



**Figura 33:** Ilustração para cálculo da distância entre dois ligamentos (A figura de fundo é da Hexa do IWF, 2004).

Ligamentos, computacionalmente, não são nada além de segmentos de reta. Desta forma, para que se saiba quando ocorre uma colisão, deve-se calcular qual a distância mínima entre eles e verificar se esta distância é menor que o valor do diâmetro de um ligamento (admitindo que sejam idênticos uns aos outros e cilíndricos).

Para realizar a tarefa de detecção de colisões, primeiro verifica-se se o sistema dos dois ligamentos em questão é linearmente dependente, pois isso indica paralelismo. Nesse caso utiliza-se a fórmula de distância de um ponto a uma reta. Se não forem paralelos, calcula-se normalmente como se fossem duas retas. Entretanto, se a distância mínima encontrada entre dois ligamentos for definida por dois pontos  $P_5$  e  $P_6$  que não pertençam aos ligamentos (devemos lembrar que a reta, por conceito, é infinita), a solução obrigatoriamente envolve uma das extremidades dos ligamentos, ou seja,  $P_5$ ,  $P_6$  ou ambos são coincidentes com  $P_j$ , sendo  $j = 1, 2, 3, 4$ . A Fig. 34 explora alguns casos possíveis. As equações que regem os cálculos estão demonstradas a seguir e são as eq. (29) a eq. (32).



**Figura 34:** Casos de cálculo de distâncias entre segmentos de reta.

$$P_5 = P_1 + \lambda_1(P_2 - P_1) \quad (29)$$

$$P_6 = P_3 + \lambda_2(P_4 - P_3) \quad (30)$$

$$P_5 - P_6 = \pm n \cdot \frac{(P_2 - P_1) \times (P_4 - P_3)}{\|(P_2 - P_1) \times (P_4 - P_3)\|} \quad (31)$$

$$n = \frac{\|(P_2 - P_4) \cdot [(P_2 - P_1) \times (P_4 - P_3)]\|}{\|(P_2 - P_1) \times (P_4 - P_3)\|} \quad (32)$$

Um total de dez equações escalares e dez incógnitas escalares ( $P_5, P_6, \lambda_1, \lambda_2, n$  e um sinal). Resolvendo então o sistema de equações, detecta-se se ocorre ou não colisão entre os ligamentos, estipulando uma distância  $d_{\min}$ , tal que  $n \geq d_{\min}$ . A eq. (32) representa a projeção da distância entre dois pontos quaisquer cada um pertencente a um ligamento no versor diretor da distância entre as retas.

Esse cálculo, entretanto, encontra a distância entre as duas retas e não os segmentos de reta. Sendo assim, deve-se ainda fazer uma verificação dos valores de  $\lambda_1$  e  $\lambda_2$ . Se  $0 \leq \lambda_1 \leq 1$  e  $0 \leq \lambda_2 \leq 1$ , então os pontos encontrados pertencem aos segmentos de reta. Se ocorrer uma violação nessas condições, devem-se verificar



individualmente as opções restantes e então escolher a que possui a menor distância entre os segmentos.

Quando se calcula a distância entre retas paralelas, utiliza-se o conceito de distância de um ponto a uma reta. Seja um triângulo  $ABP$  de base  $AB$ , altura  $h$  e área  $S$ . A distância de um ponto a uma reta é dada pela altura do triângulo, considerando  $AB$  pertencentes a uma reta. Vide eq. (33) e eq. (34).

$$S = \frac{1}{2} \|AP \times AB\| = \frac{1}{2} \|AB\| \cdot h \quad (33)$$

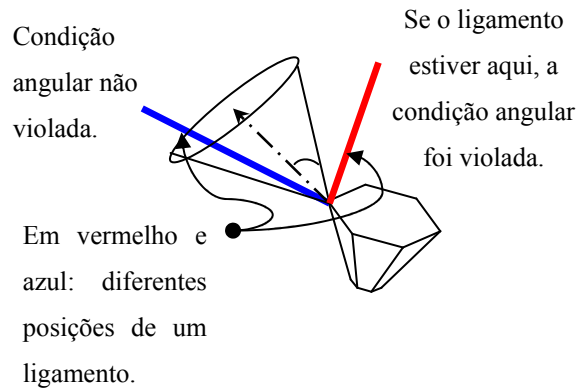
$$h = \frac{\|AP \times AB\|}{\|AB\|} \quad (34)$$

### 5.3.2.2. Limites Angulares das Juntas Atuadas e Esféricas

Da própria cinemática inversa pode-se extrair o valor dos ângulos de atuação. Assim, quaisquer limitações podem ser detectadas e impostas. Os ângulos entre ligamentos passivos e ativos (junta esférica) são obtidos através da equação de ângulo entre vetores. Seja o vetor  $\mathbf{h}$  e o vetor  $\mathbf{l}$ , bem como o ângulo entre eles  $\phi$ . O ângulo entre os dois ligamentos é dado pela eq. (35).

$$\phi = \arccos\left(\frac{\mathbf{l} \cdot \mathbf{h}}{\|\mathbf{l}\| \cdot \|\mathbf{h}\|}\right) \quad (35)$$

Para as articulações entre a plataforma móvel e os ligamentos passivos, definiu-se um eixo de referência no qual ao violar um ângulo máximo de referência, o ângulo limite da articulação e da plataforma são violados. A Fig. 35 ilustra esse caso, onde o cone representa o espaço possível onde o ligamento pode se locomover.



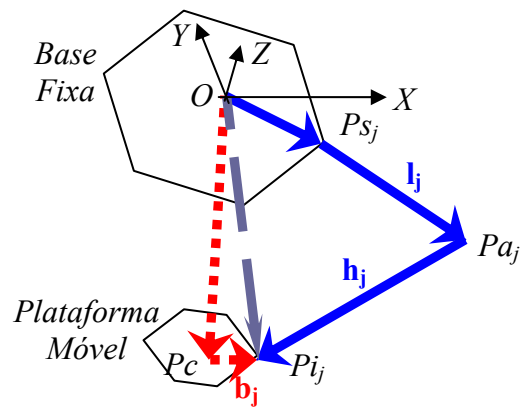
**Figura 35:** Restrição do ângulo das articulações da plataforma móvel.

Quando se obtém toda a malha de pontos da borda do volume de trabalho, os pontos adjacentes são agrupados em triângulos para formar as bases de muitas pirâmides cujos picos são o ponto central. O volume total obtido tem um erro numérico que diminui à proporção que aumenta o detalhe da malha. Seja uma pirâmide de vértices  $ABCD$  onde  $D$  é o ponto central. Se forem considerados os vetores que passam por  $AB$ ,  $AC$  e  $AD$  como sendo, respectivamente,  $\mathbf{u}$ ,  $\mathbf{v}$  e  $\mathbf{w}$ , o volume da pirâmide pode ser calculado segundo a eq. (36).

$$V_{\text{Pirâmide}} = \frac{1}{6} (\mathbf{u} \times \mathbf{v}) \cdot \mathbf{w} \quad (36)$$

#### 5.4. O Modelo da Cinemática Inversa da Hexa: Equações de Velocidades e de Acelerações

Continuando o equacionamento da arquitetura Hexa, tem-se o esquema apresentado na Fig. 36. Ele descreve algumas relações vetoriais que serão utilizadas na dedução das equações de velocidade e de aceleração dos pontos de interesse da arquitetura.



**Figura 36:** Esquema vetorial de uma cadeia cinemática da Hexa.

Será utilizado o sistema  $O-XYZ$  a menos que informado algum outro. Parte-se do pressuposto que as seguintes variáveis são conhecidas, sejam entradas para o problema, sejam obtidas durante a análise de posição. A exceção é para com os módulos da velocidade angular e da aceleração angular das articulações atuadas.

- $\mathbf{v}_{Pc}$  Velocidade do ponto  $Pc$  pertencente à plataforma móvel.
- $\dot{\mathbf{v}}_{Pc}$  Aceleração do ponto de interesse  $Pc$  pertencente à plataforma móvel.
- $\boldsymbol{\omega}_p$  Velocidade angular da plataforma móvel.
- $\dot{\boldsymbol{\omega}}_p$  Aceleração angular da plataforma móvel.
- $\mathbf{b}_j$  Vetores que partem do ponto de interesse  $Pc$  e vão até cada um dos centros das articulações em  $Pi_j$  (vértices da plataforma móvel).
- $\mathbf{h}_j$  Ligamentos passivos na forma vetorial.
- $\mathbf{l}_j$  Ligamentos atuados na forma vetorial.
- $\dot{\boldsymbol{\theta}}_j$  Velocidade angular das articulações atuadas.
- $\ddot{\boldsymbol{\theta}}_j$  Aceleração angular das articulações atuadas.

Podem ser então extraídas relações para o cálculo das velocidades e das acelerações de todas as partes móveis em seus pontos de interesse. As outras variáveis de interesse tratadas aqui são:

- $\mathbf{v}_{Ps_j}$  Velocidade dos pontos  $Ps_j$ .
- $\dot{\mathbf{v}}_{Ps_j}$  Aceleração dos pontos  $Ps_j$ .
- $\mathbf{v}_{Pa_j}$  Velocidade dos pontos  $Pa_j$ .
- $\dot{\mathbf{v}}_{Pa_j}$  Aceleração dos pontos  $Pa_j$ .
- $\mathbf{v}_{Pi_j}$  Velocidade dos pontos  $Pi_j$ .
- $\dot{\mathbf{v}}_{Pi_j}$  Aceleração dos pontos  $Pi_j$ .
- $\boldsymbol{\omega}_{h_j}$  Velocidade angular dos ligamentos  $h_j$ .
- $\dot{\boldsymbol{\omega}}_{h_j}$  Aceleração angular dos ligamentos  $h_j$ .

Sejam dois pontos  $A$  e  $B$  pertencentes a um mesmo corpo rígido. Seja a velocidade angular e aceleração angular do corpo, respectivamente,  $\boldsymbol{\Omega}$  e  $\dot{\boldsymbol{\Omega}}$ . Pode-se dizer a respeito das velocidades e acelerações de cada ponto que:

$$\mathbf{v}_A = \mathbf{v}_B + \boldsymbol{\Omega} \times (A - B) \quad (37)$$

$$\mathbf{v}_B = \mathbf{v}_A + \boldsymbol{\Omega} \times (B - A) \quad (38)$$

$$\dot{\mathbf{v}}_A = \dot{\mathbf{v}}_B + \dot{\boldsymbol{\Omega}} \times (A - B) + \boldsymbol{\Omega} \times [\boldsymbol{\Omega} \times (A - B)] \quad (39)$$

$$\dot{\mathbf{v}}_B = \dot{\mathbf{v}}_A + \dot{\boldsymbol{\Omega}} \times (B - A) + \boldsymbol{\Omega} \times [\boldsymbol{\Omega} \times (B - A)] \quad (40)$$

Utilizando-se estes conceitos, pode-se escrever para os pontos de interesse da Hexa suas velocidades e acelerações. As velocidades e acelerações dos pontos  $Ps_j$  valem zero, já que elas são os vértices da base fixa.

$$\mathbf{v}_{Ps_j} = \dot{\mathbf{v}}_{Ps_j} = 0 \quad (41)$$

As velocidades dos pontos  $Pa_j$  dependem apenas do módulo da velocidade angular das articulações atuadas:

$$\mathbf{v}_{Pa_j} = \dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j \quad (42)$$

Os vértices da plataforma móvel conseguem unir em suas equações várias incógnitas cinemáticas da Hexa. As velocidades dos pontos  $Pi_j$  são dadas por:

$$\mathbf{v}_{Pi_j} = \mathbf{v}_{Pc} + \boldsymbol{\omega}_p \times \mathbf{b}_j \quad (43)$$

$$\mathbf{v}_{Pi_j} = \mathbf{v}_{Pa_j} + \boldsymbol{\omega}_{h_j} \times \mathbf{h}_j \quad (44)$$

Igualando-se ambos os termos do lado direito das equações anteriores das velocidades dos pontos  $Pi_j$  chega-se a:

$$\mathbf{v}_{Pc} + \boldsymbol{\omega}_p \times \mathbf{b}_j = \dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j + \boldsymbol{\omega}_{h_j} \times \mathbf{h}_j \quad (45)$$

A eq. (45), conforme será mostrado no próximo capítulo, pode ser trabalhada algebricamente levando ao cálculo da matriz Jacobiano, que pode ser utilizada para que se calcule o módulo das velocidades angulares das articulações atuadas a partir da velocidade de  $Pc$  e da velocidade angular da plataforma móvel. Tendo-se as velocidades das articulações atuadas, calculam-se as velocidades dos pontos  $Pa_j$  e, por último, a velocidade angular dos ligamentos passivos, calculada através da solução do seguinte sistema linear:

$$\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j = \mathbf{v}_{Pi_j} - \mathbf{v}_{Pa_j} \quad (46)$$

$$\begin{bmatrix} 0 & h_{j,Z} & -h_{j,Y} \\ -h_{j,Z} & 0 & h_{j,X} \\ h_{j,Y} & -h_{j,X} & 0 \end{bmatrix} \begin{bmatrix} \omega_{h_j,X} \\ \omega_{h_j,Y} \\ \omega_{h_j,Z} \end{bmatrix} = \mathbf{v}_{Pi_j} - \mathbf{v}_{Pa_j} \quad (47)$$

Um outro modo de se obter as velocidades é através do método apresentado a seguir para as acelerações dos pontos de interesse. Esse método não foi utilizado para as velocidades por ser mais complexo que simplesmente utilizar o Jacobiano.

As acelerações dos pontos  $Pa_j$  são dadas por:

$$\dot{\mathbf{v}}_{Pa_j} = \ddot{\boldsymbol{\theta}}_j \times \mathbf{l}_j + \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) \quad (48)$$

Para o cálculo das acelerações dos pontos  $Pi_j$ , têm-se novamente duas equações, obtidas derivando as equações de velocidades:

$$\dot{\mathbf{v}}_{Pi_j} = \dot{\mathbf{v}}_{Pc} + \dot{\boldsymbol{\omega}}_p \times \mathbf{b}_j + \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{b}_j) \quad (49)$$

$$\dot{\mathbf{v}}_{Pi_j} = \dot{\mathbf{v}}_{Pa_j} + \dot{\boldsymbol{\omega}}_{h_j} \times \mathbf{h}_j + \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) \quad (50)$$

Substituindo as acelerações de  $Pa_j$  na equação anterior e seguindo com alguma manipulação algébrica, tem-se a seguinte seqüência de equações:

$$\dot{\mathbf{v}}_{Pi_j} = \underbrace{\ddot{\boldsymbol{\theta}}_j \times \mathbf{l}_j + \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j)}_{\dot{\mathbf{v}}_{Pa_j}} + \dot{\boldsymbol{\omega}}_{h_j} \times \mathbf{h}_j + \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) \quad (51)$$

$$\dot{\mathbf{v}}_{Pi_j} - \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) - \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) = \ddot{\boldsymbol{\theta}}_j \times \mathbf{l}_j + \dot{\boldsymbol{\omega}}_{h_j} \times \mathbf{h}_j \quad (52)$$

$$\dot{\mathbf{v}}_{Pi_j} - \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) - \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) = \ddot{\theta}_j \left( \frac{\dot{\boldsymbol{\theta}}_j}{\|\dot{\boldsymbol{\theta}}_j\|} \times \mathbf{l}_j \right) + \dot{\boldsymbol{\omega}}_{h_j} \times \mathbf{h}_j \quad (53)$$

Todavia, sabe-se que tanto na velocidade angular quanto na aceleração angular dos ligamentos  $\mathbf{h}_j$  uma das componentes possui valor arbitrário: aquela ao longo do ligamento. Sendo assim, a equação anterior não possui quatro incógnitas (um escalar e um vetor), mas sim três incógnitas, bastando-se utilizar uma matriz  $\mathbf{M}$  de transformação de coordenadas.

O termo  $lig_{h_j}$  subscripto ou sobrescrito indicará a base dos ligamentos  $\mathbf{h}_j$  (sistema solidário aos ligamentos, onde o eixo  $x_j$  aponta de  $Pa_j$  para  $Pi_j$ ). Embora até o momento não se tenha utilizado o termo *Global* para se referir a *O-XYZ*, a partir deste ponto será utilizado em algumas variáveis para evitar confusão dos termos.

A matriz  $\mathbf{M}$  de transformação de coordenadas é uma matriz de versores e pode ser utilizada sem preocupações quanto a translações com relação aos centros das coordenadas, pois o módulo das variáveis vetoriais que ela transforma não mudará (não se está lidando com pontos). Os versores de *O-XYZ* são chamados de  $[\hat{\mathbf{x}}_{Global} \ \hat{\mathbf{y}}_{Global} \ \hat{\mathbf{z}}_{Global}]$  e os da base do ligamento são:  $[\hat{\mathbf{x}}_{lig_{h_j}} \ \hat{\mathbf{y}}_{lig_{h_j}} \ \hat{\mathbf{z}}_{lig_{h_j}}]$ , ambos escritos com relação a *O-XYZ* (vide o Apêndice C para maiores detalhes sobre transformações de coordenadas):

$${}^{Global}\mathbf{M}_{lig_j} = \begin{bmatrix} \hat{\mathbf{x}}_{lig_{h_j}} \cdot \hat{\mathbf{x}}_{Global} & \hat{\mathbf{y}}_{lig_{h_j}} \cdot \hat{\mathbf{x}}_{Global} & \hat{\mathbf{z}}_{lig_{h_j}} \cdot \hat{\mathbf{x}}_{Global} \\ \hat{\mathbf{x}}_{lig_{h_j}} \cdot \hat{\mathbf{y}}_{Global} & \hat{\mathbf{y}}_{lig_{h_j}} \cdot \hat{\mathbf{y}}_{Global} & \hat{\mathbf{z}}_{lig_{h_j}} \cdot \hat{\mathbf{y}}_{Global} \\ \hat{\mathbf{x}}_{lig_{h_j}} \cdot \hat{\mathbf{z}}_{Global} & \hat{\mathbf{y}}_{lig_{h_j}} \cdot \hat{\mathbf{z}}_{Global} & \hat{\mathbf{z}}_{lig_{h_j}} \cdot \hat{\mathbf{z}}_{Global} \end{bmatrix} \quad (54)$$

Continuando o equacionamento, tem-se que:

$$\begin{aligned} \dot{\mathbf{v}}_{Pi_j} - \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) - \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) &= \ddot{\theta}_j \left( \frac{\dot{\boldsymbol{\theta}}_j}{\|\dot{\boldsymbol{\theta}}_j\|} \times \mathbf{l}_j \right) + \\ + \left( {}^{Global}\mathbf{M}_{lig_{h_j}}{}^{lig_{h_j}}\dot{\boldsymbol{\omega}}_{h_j} \right) \times \mathbf{h}_j \end{aligned} \quad (55)$$

Por simplicidade, a equação é reescrita como:

$$\ddot{\theta}_j \mathbf{T}_j + \left( {}^{Global}\mathbf{M}_{lig_{h_j}}{}^{lig_{h_j}}\dot{\boldsymbol{\omega}}_{h_j} \right) \times \mathbf{h}_j = \mathbf{B}_j \quad (56)$$

Onde  $\mathbf{B}_j$  e  $\mathbf{T}_j$  são conhecidos:

$$\dot{\mathbf{v}}_{Pi_j} - \dot{\boldsymbol{\theta}}_j \times (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) - \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) = \mathbf{B}_j \quad (57)$$

$$\ddot{\theta}_j \left( \frac{\ddot{\theta}_j}{\|\ddot{\theta}_j\|} \times \mathbf{l}_j \right) = \ddot{\theta}_j \mathbf{T}_j \quad (58)$$

Tem-se ainda que:

$$\begin{aligned} & \left( {}^{Global} \mathbf{M}_{lig\ h\ j} {}^{lig\ h\ j} \dot{\omega}_{h\ j} \right) \times \mathbf{h}_j = \\ & = \underbrace{\begin{bmatrix} 0 & h_{j,Z} & -h_{j,Y} \\ -h_{j,Z} & 0 & h_{j,X} \\ h_{j,Y} & -h_{j,X} & 0 \end{bmatrix}}_{\text{Matriz } \mathbf{A}_j} \underbrace{\begin{bmatrix} {}^{lig\ h\ j} \dot{\omega}_{h\ j,X} \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Y} \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Z} \end{bmatrix}}_{\mathbf{0}} \end{aligned} \quad (59)$$

Dessa forma, tem-se que:

$$\ddot{\theta}_j \mathbf{T}_j + \mathbf{A}_j \begin{bmatrix} 0 \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Y} \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Z} \end{bmatrix} = \mathbf{B}_j \quad (60)$$

Agrupando-se as três incógnitas no mesmo vetor de variáveis e fazendo as devidas considerações com o vetor  $\mathbf{T}_j$  e a matriz  $\mathbf{A}_j$ , tem-se que o seguinte sistema de equações:

$$\begin{bmatrix} T_{1j} & A_{12j} & A_{13j} \\ T_{2j} & A_{22j} & A_{23j} \\ T_{3j} & A_{32j} & A_{33j} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_j \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Y} \\ {}^{lig\ h\ j} \dot{\omega}_{h\ j,Z} \end{bmatrix} = \begin{bmatrix} B_{1j} \\ B_{2j} \\ B_{3j} \end{bmatrix} \quad (61)$$



## 6. ANÁLISE DO JACOBIANO E DA CINEMÁTICA DIRETA

Neste capítulo será dada ênfase à análise do Jacobiano do robô. Além de serem apresentadas as matrizes características, análises como as do elipsóide de forças e velocidades serão realizadas. A análise cinemática direta será explorada ao final do capítulo.

### 6.1. A Matriz Jacobiano

É a matriz que escreve a velocidade das juntas atuadas em função das variáveis de velocidade do efetuador. Pode-se escrever que a matriz Jacobiano  $\mathbf{J}$ , nos robôs de arquitetura paralela, respeita a relação:

$$\dot{\mathbf{q}} = \mathbf{J}\dot{\mathbf{x}} \quad (62)$$

Onde  $\dot{\mathbf{q}}$  é o vetor dos módulos das velocidades angulares das articulações atuadas e  $\dot{\mathbf{x}}$  é o vetor da velocidade linear e angular da plataforma móvel, dados por:

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \vdots \\ \dot{\theta}_6 \end{bmatrix} \quad \dot{\mathbf{x}} = \begin{bmatrix} \mathbf{v}_{Pc} \\ \boldsymbol{\omega}_P \end{bmatrix} \quad (63)$$

É importante observar que a matriz  $\mathbf{J}$  (Jacobiano) dos robôs de arquitetura paralela é, por convenção, o inverso da matriz Jacobiano dos robôs seriados. Nesses últimos a matriz Jacobiano segue a relação  $\dot{\mathbf{x}} = \mathbf{J}\dot{\mathbf{q}}$ .

Pode-se observar que essa forma facilita a solução da cinemática mais natural de cada arquitetura. Além disso, conforme explicado mais adiante, é mais fácil obter a inversa de  $\mathbf{J}_q$  se comparada à  $\mathbf{J}_x$ .

## 6.2. Condições de Singularidade

Matematicamente as singularidades ocorrem quando o determinante da matriz Jacobiano vale zero. Embora o usual seja o cálculo do determinante dessa matriz, **Gosselin e Angeles (1990)** estudaram casos de singularidades em robôs de arquitetura paralela e sugeriram a separação da matriz em duas outras: uma associada à cinemática inversa e outra associada à cinemática direta.

Sendo as restrições cinemáticas impostas pelos ligamentos apresentadas na forma de uma função  $f(\mathbf{x}, \mathbf{q}) = \mathbf{0}$ , pode-se definir a relação  $\dot{\mathbf{q}} = \mathbf{J}\dot{\mathbf{x}}$ . A matriz Jacobiano é escrita como:

$$\mathbf{J}_x \dot{\mathbf{x}} = \mathbf{J}_q \dot{\mathbf{q}} \quad (64)$$

$$\text{Onde } \mathbf{J}_x = \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \text{ e } \mathbf{J}_q = -\frac{\partial \mathbf{f}}{\partial \mathbf{q}}.$$

$$\dot{\mathbf{q}} = \underbrace{\mathbf{J}_q^{-1} \mathbf{J}_x}_{\text{Jacobiano}=\mathbf{J}} \dot{\mathbf{x}} \quad (65)$$

Utilizando-se este artifício, podem ser separados os casos de singularidade da cinemática direta e da cinemática inversa através do estudo dos determinantes de cada matriz. Não será feito aqui o equacionamento da dedução da separação das matrizes para um robô genérico, sendo feito apenas para a Hexa (item 6.3).

### 6.2.1. Singularidades da Cinemática Inversa

Ocorre um caso de singularidade na cinemática inversa quando o determinante de  $\mathbf{J}_q$  vale zero. Quando  $\mathbf{J}_q$  é singular e atenta-se para as soluções não triviais, existem alguns vetores  $\dot{\mathbf{q}}$  não nulos que resultam em vetores  $\dot{\mathbf{x}}$  nulos. Assim, movimentos infinitesimais da plataforma móvel ao longo de certas direções não poderão ocorrer, implicando na perda de um ou mais graus de liberdade naquela

posição. Isso também implica na capacidade do robô de resistir a forças ou momentos em alguma direção sem a necessidade de se aplicar força ou torque nos atuadores.

Singularidades da cinemática inversa geralmente ocorrem na fronteira do volume de trabalho alcançável da máquina. Estes casos são similares aos dos robôs seriados.

Em outras palavras, uma singularidade da cinemática inversa resulta numa *rigidez infinita* em alguma direção de movimento da plataforma móvel.

### **6.2.2. Singularidades da Cinemática Direta**

Singularidades da cinemática direta ocorrem quando o determinante da matriz  $\mathbf{J}_x$  vale zero. De forma análoga, existem vetores  $\dot{\mathbf{x}}$  diferentes de zero que resultam em vetores  $\dot{\mathbf{q}}$  iguais a zero. Isso implica na possibilidade de ocorrer movimentos infinitesimais da plataforma em alguma direção específica com os atuadores totalmente parados. Dessa forma a plataforma móvel ganha um ou mais graus de liberdade que não podem ser controlados, ou seja, o robô será incapaz de resistir a esforços (forças e / ou momentos) em alguma direção específica.

Em outras palavras, uma singularidade da cinemática direta resulta numa *rigidez nula* em alguma direção de movimento da plataforma móvel.

### **6.2.3. Singularidades combinadas**

Ocorrem quando o determinante de ambas as matrizes  $\mathbf{J}_q$  e  $\mathbf{J}_x$  valem zero. Segundo **Tsai (1999)**, esses casos são os mais raros, sendo geralmente encontrados apenas em máquinas com arquiteturas especiais. Nessas condições ocorre uma degeneração das equações cinemáticas que regem o movimento do robô, havendo a possibilidade de ocorrer movimento da plataforma móvel com os atuadores travados e a possibilidade de ocorrer movimento dos atuadores com a plataforma móvel parada.

### 6.3. Cálculo da Matriz Jacobiano da Hexa

Neste item é apresentado um método geométrico genérico que pode ser utilizado com facilidade na maioria das arquiteturas paralelas. Como referência é utilizada a Fig. 36.

Para cada cadeia cinemática existem dois conjuntos de vetores partindo de  $O$  e chegando a  $Pi_j$ . O objetivo é relacionar a velocidade do ponto  $Pi_j$  olhando-se através da perspectiva da plataforma móvel e da perspectiva dos atuadores. Sendo assim, pode-se escrever a eq. (66), cuja derivada é a eq. (67). A eq. (66) é a equação  $f(\mathbf{x}, \mathbf{q}) = \mathbf{0}$  definida anteriormente, onde apenas os termos estão dispostos diferentemente de forma a facilitar a dedução das matrizes Jacobiano.

$$\overline{OP_C} + \overline{P_C P_i_j} = \overline{OP_s_j} + \overline{P_s_j P_a_j} + \overline{P_a_j P_i_j} \quad (66)$$

$$\mathbf{v}_{P_C} + \boldsymbol{\omega}_P \times \mathbf{b}_j = \dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j + \boldsymbol{\omega}_{h_j} \times \mathbf{h}_j \quad (67)$$

No entanto, não é interessante para este cálculo que se tenha nas equações o vetor de velocidade angular de cada ligamento. Multiplica-se então a eq. (67) pelo vetor  $\mathbf{h}_j$ , pois  $\mathbf{h}_j \cdot (\boldsymbol{\omega}_{h_j} \times \mathbf{h}_j) = 0$ . Assim, tem-se que:

$$\mathbf{h}_j \cdot \mathbf{v}_{P_C} + \mathbf{h}_j \cdot (\boldsymbol{\omega}_P \times \mathbf{b}_j) = \mathbf{h}_j \cdot (\dot{\boldsymbol{\theta}}_j \times \mathbf{l}_j) \quad (68)$$

Conhece-se também da teoria de álgebra linear que se  $\mathbf{a}$ ,  $\mathbf{b}$  e  $\mathbf{c}$  são vetores de mesma dimensão, valem as relações:

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = -\mathbf{b} \cdot (\mathbf{a} \times \mathbf{c}) \quad (69)$$

$$\mathbf{a} \cdot (\mathbf{b} \times \mathbf{c}) = -\mathbf{a} \cdot (\mathbf{c} \times \mathbf{b}) \quad (70)$$

Aplicando as relações da álgebra linear apresentadas à eq. (68) e prosseguindo com o equacionamento, tem-se que a:

$$\mathbf{h}_j \cdot \mathbf{v}_{Pc} - \boldsymbol{\omega}_P \cdot (\mathbf{h}_j \times \mathbf{b}_j) = -\dot{\theta}_j \cdot (\mathbf{h}_j \times \mathbf{l}_j) \quad (71)$$

$$\mathbf{h}_j \cdot \mathbf{v}_{Pc} + \boldsymbol{\omega}_P \cdot (\mathbf{b}_j \times \mathbf{h}_j) = \dot{\theta}_j \cdot (\mathbf{l}_j \times \mathbf{h}_j) \quad (72)$$

$$\mathbf{h}_j \cdot \mathbf{v}_{Pc} + \boldsymbol{\omega}_P \cdot (\mathbf{b}_j \times \mathbf{h}_j) = \frac{\dot{\theta}_j}{\|\dot{\theta}_j\|} \cdot (\mathbf{l}_j \times \mathbf{h}_j) \dot{\theta}_j \quad (73)$$

Pode-se então escrever a relação de **Gosselin e Angles (1990)** com as matrizes Jacobiano de cada cinemática:

$$\underbrace{\begin{bmatrix} \mathbf{h}_1 & (\mathbf{b}_1 \times \mathbf{h}_1) \\ \mathbf{h}_2 & (\mathbf{b}_2 \times \mathbf{h}_2) \\ \vdots & \vdots \\ \mathbf{h}_6 & (\mathbf{b}_6 \times \mathbf{h}_6) \end{bmatrix}}_{\mathbf{J}_x \ 6 \times 6} \underbrace{\begin{bmatrix} \mathbf{v}_{Pc} \\ \boldsymbol{\omega}_P \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} \frac{\dot{\theta}_1}{\|\dot{\theta}_1\|} \cdot (\mathbf{l}_1 \times \mathbf{h}_1) & 0 & \dots & 0 \\ 0 & \frac{\dot{\theta}_2}{\|\dot{\theta}_2\|} \cdot (\mathbf{l}_2 \times \mathbf{h}_2) & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \dots & 0 & \frac{\dot{\theta}_6}{\|\dot{\theta}_6\|} \cdot (\mathbf{l}_6 \times \mathbf{h}_6) \end{bmatrix}}_{\mathbf{J}_q \ 6 \times 6} \underbrace{\begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \\ \dot{\theta}_5 \\ \dot{\theta}_6 \end{bmatrix}}_{\dot{\mathbf{q}}} \quad (74)$$

Para a cinemática direta, a análise requer mais cuidados. Para a cinemática inversa as singularidades ocorrem se  $\det(\mathbf{J}_q) = 0$ , ou seja:

$$\prod_{j=1}^6 \frac{\dot{\theta}_j}{\|\dot{\theta}_j\|} (\mathbf{l}_j \times \mathbf{h}_j) = 0 \quad (75)$$

Por exemplo, ocorrerá singularidade toda vez que os dois ligamentos de um mesmo “ramo” estiverem formando uma linha reta; ou o plano formado pelos dois ligamentos de cada ramo for perpendicular ao plano de qualquer outro dos cinco ramos restantes; etc. Pode-se perceber que realmente a maior probabilidade de ocorrer tais casos de singularidade é na borda do volume de trabalho (singularidades semelhantes às dos robôs de arquitetura seriada).

## 6.4. Rigidez e Flexibilidade

Quando um robô executa uma tarefa, ele geralmente aplica uma força ou momento no meio externo. Como consequência, seu efetuador sofre uma solicitação de igual magnitude, o que pode provocar um desvio na trajetória do manipulador. Esse desvio é proporcional à rigidez e à magnitude e tipo da solicitação.

A rigidez de um robô depende de vários fatores como a geometria e o comprimento dos ligamentos, o material de sua estrutura, os tipos de atuadores e transmissões mecânicas, as articulações e o próprio controlador. No caso particular dos robôs de arquitetura paralela, o aumento dos comprimentos dos ligamentos é crítico para a rigidez, diminuindo os maiores ganhos teóricos dessas arquiteturas: exatidão e rigidez elevadas. Ao mesmo tempo, é um desejo constante o aumento do volume de trabalho e da velocidade de operação. Assim, deve-se buscar uma relação de compromisso entre todos estes fatores.

Parte importante de estudos no campo da robótica está justamente no item rigidez. Conhecendo-se a rigidez de uma máquina, podem-se utilizar até mesmo estratégias sofisticadas de controle para aumentar a exatidão sem alterar a estrutura (Lebret et al., 1993).

### 6.4.1. Matriz de Rigidez e Matriz de Flexibilidade

Admitindo-se a rigidez das articulações atuadas muitas vezes inferior à rigidez estrutural do robô e tendo-se que  $\tau = [\tau_1 \quad \tau_2 \quad \dots \quad \tau_6]^T$  é o vetor de torques

nas juntas atuadas,  $\chi = \text{diag}[k_1, k_2, \dots, k_6]^T$  é a matriz diagonal de rigidez das articulações atuadas e  $\Delta \mathbf{q} = [\Delta q_1 \quad \Delta q_2 \quad \dots \quad \Delta q_6]$  é o vetor de deflexões dessas articulações, pode-se escrever que:

$$\boldsymbol{\tau} = \boldsymbol{\chi} \cdot \Delta \mathbf{q} \quad (76)$$

Todavia, conforme apresentado anteriormente, em todas as posições, com exceção das singularidades, uma pequena variação da postura (posição e orientação) da plataforma móvel, dada por  $\Delta \mathbf{x} = [\Delta x_1 \quad \Delta x_2 \quad \dots \quad \Delta x_6]$ , está atrelada a uma pequena variação nos ângulos de atuação. A variação de posição no tempo é a velocidade e estas se relacionam entre si através da matriz Jacobiano. Assim sendo, pode-se deduzir que:

$$\Delta \mathbf{q} = \mathbf{J} \cdot \Delta \mathbf{x} \quad (77)$$

Embora sejam relações importantes, ainda não se podem inserir as forças e momentos externos nas equações, dados por  $\mathbf{F} = [F_1 \quad F_2 \quad \dots \quad F_6]^T$ , onde  $F_1$  a  $F_3$  são forças e  $F_4$  a  $F_6$ , momentos. Assim, recorre-se ao princípio do trabalho virtual. Embora para diversos aspectos no estudo dos robôs não se possam fazer estas premissas, para efeito deste item vai-se admitir como sendo desprezíveis quaisquer perdas de energia devido a forças de atrito nas articulações. As forças gravitacionais também serão desconsideradas. Desta maneira, o trabalho virtual de todas as forças ativas pode ser descrito pela seguinte equação (vale zero quando o sistema está em equilíbrio, caso este o estudado aqui):

$$\delta \mathbf{W} = \boldsymbol{\tau}^T \delta \mathbf{q} - \mathbf{F}^T \delta \mathbf{x} = 0 \quad (78)$$

Onde a letra grega  $\delta$  representa uma variação infinitesimal e é comumente utilizada como notação onde envolve o trabalho virtual. Sabe-se também que:

$$\delta \mathbf{q} = \mathbf{J} \cdot \delta \mathbf{x} \quad (79)$$

Logo,

$$\boldsymbol{\tau}^T \mathbf{J} \cdot \delta \mathbf{x} - \mathbf{F}^T \delta \mathbf{x} = \mathbf{0} \quad (80)$$

$$\boldsymbol{\tau}^T \mathbf{J} - \mathbf{F}^T = \mathbf{0} \quad (81)$$

$$\mathbf{F}^T = \boldsymbol{\tau}^T \mathbf{J} \quad (82)$$

$$\mathbf{F} = \mathbf{J}^T \boldsymbol{\tau} \quad (83)$$

Utilizando-se as relações obtidas aqui, pode-se calcular a matriz de rigidez  $\mathbf{K}$  e a matriz de flexibilidade  $\mathbf{C}$ , que é a inversa da matriz de rigidez. Conforme apresentado, sabe-se que:

$$\boldsymbol{\tau} = \chi \cdot \underbrace{\mathbf{J} \cdot \Delta \mathbf{x}}_{\Delta \mathbf{q}} \quad (84)$$

Portanto,

$$\mathbf{F} = \mathbf{J}^T \underbrace{\chi \cdot \mathbf{J} \cdot \Delta \mathbf{x}}_{\boldsymbol{\tau}} \quad (85)$$

$$\mathbf{F} = \mathbf{K} \cdot \Delta \mathbf{x} \quad (86)$$

$$\mathbf{K} = \mathbf{J}^T \chi \mathbf{J} \quad (87)$$

$$\Delta \mathbf{x} = \mathbf{C} \cdot \mathbf{F} \quad (88)$$

Se todas as constantes de rigidez das articulações forem iguais entre si, a matriz de rigidez  $\mathbf{K}$  pode ser escrita como o produto  $k \cdot \mathbf{J}^T \mathbf{J}$ .



## 6.5. Elipsóides de Forças, de Velocidades e de Deslocamentos

Uma boa métrica de desempenho para um robô é o estudo da magnitude de forças no efetuador produzidas por uma unidade de força nos atuadores. Ou seja, deseja-se estudar a equação  $\boldsymbol{\tau}^T \cdot \boldsymbol{\tau} = 1$ . Sabe-se que  $\boldsymbol{\tau} = \mathbf{J}^{-T}\mathbf{F}$  e  $\boldsymbol{\tau}^T = \mathbf{F}^T\mathbf{J}^{-1}$ , assim  $\boldsymbol{\tau}^T \cdot \boldsymbol{\tau} = 1$  pode ser escrito como:

$$\mathbf{F}^T\mathbf{J}^{-1}\mathbf{J}^{-T}\mathbf{F} = 1 \Rightarrow \mathbf{F}^T(\mathbf{J}^T\mathbf{J})^{-1}\mathbf{F} = 1 \quad (89)$$

Essa relação representa a superfície de um elipsóide  $m$ -dimensional, onde  $m$  é o número de graus de liberdade da plataforma móvel. A matriz  $(\mathbf{J}^T\mathbf{J})^{-1}$  é simétrica positiva semi-definida, logo seus autovetores são ortogonais entre si. Os eixos principais do elipsóide coincidem com os autovetores da matriz e seus comprimentos são inversamente proporcionais às raízes quadradas dos autovalores dessa matriz.

Obviamente o elipsóide depende da postura da plataforma móvel. Todavia, o quanto mais próximo este elipsóide for de uma esfera, melhores serão as características gerais de transmissão de forças e de movimentos. Por exemplo, menores serão as derivadas das acelerações (chamadas de *tranco* ou, em inglês *jerk*) do robô quando, em operação, mudar de direção de movimento. Se o elipsóide for, na verdade, uma esfera, diz-se que ocorreu uma transformação *isotrópica*.

Seja  $n$  o número de atuadores. Numa transformação isotrópica, uma esfera  $n$ -dimensional no espaço das juntas atuadas é mapeada numa esfera  $m$ -dimensional no espaço do efetuador. Da mesma maneira, numa posição de singularidade a esfera  $n$ -dimensional no espaço das juntas é mapeada como um cilindro (autovalor vale zero nessa direção) ou como uma figura onde um dos eixos não existe (autovalor infinito nessa direção) no espaço do efetuador. Esses casos podem ser relacionados com os conceitos físicos de singularidade anteriormente apresentados. Assim, quando o autovalor é zero, tem-se uma singularidade da cinemática inversa e, quando é infinito, um caso de singularidade da cinemática direta.

Se os autovalores da matriz característica  $(\mathbf{J}^T\mathbf{J})^{-1}$  são descritos pelo conjunto de valores  $\lambda_1, \lambda_2, \dots, \lambda_6$ ; os autovetores pelo conjunto  $\boldsymbol{\xi} = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \dots, \boldsymbol{\xi}_6\}$ ; e as forças e os

torques no efetuador são dados pelo vetor  $\mathbf{F}$  (já descrito anteriormente), pode-se escrever a equação da borda do elipsóide. Vale lembrar que, pela concepção das equações, o vetor  $\mathbf{F}$  é na verdade  $\{F_X, F_Y, F_Z, T_X, T_Y, T_Z\}$ , onde  $F$  representa forças em  $X$ ,  $Y$  e  $Z$  e  $T$  os torques.  $F_j$  representa a coordenada  $j$  de  $\mathbf{F}$ , bem como  $\xi_{ij}$ , a coordenada  $j$  do autovetor  $i$ . A equação da borda do elipsóide está deduzida no Apêndice B e, para as variáveis descritas, vale:

$$\sum_{i=1}^6 \left\{ \lambda_i \cdot \left[ \sum_{j=1}^6 (\xi_{ij} \cdot F_j) \right]^2 \right\} = 1 \quad (90)$$

A título de representações e estudos, a equação pode ser apresentada para apenas três graus de liberdade. Por exemplo, podem-se estudar casos onde todos os torques valham zero. Nesse caso o elipsóide passa a ser tridimensional e pode ser incluído, dentre usos, em ferramentas gráficas (simuladores).

Quando um robô de arquitetura paralela encontra-se próximo a uma singularidade, uma pequena movimentação de um dos atuadores pode gerar uma alta velocidade no efetuador. Isso significa que a exatidão do posicionamento da ferramenta é mais baixa em certas direções quando próximo a singularidades porque a resolução dos sensores de medição é amplificada em muitas vezes. Além disso, uma amplificação na velocidade em uma direção é equivalente à perda de rigidez naquela direção. Este é o caso do estudo do elipsóide de velocidades.

**Chablat e Wenger (2003)** sugerem o estudo do elipsóide de velocidades. O que é proposto é o estudo de  $\dot{\mathbf{q}}^T \dot{\mathbf{q}} \leq 1$ , ou seja, estudar o mapeamento do elipsóide de velocidades no efetuador a partir das velocidades dos atuadores. De maneira similar, tem-se que  $\dot{\mathbf{x}}^T (\mathbf{J}^T \mathbf{J}) \dot{\mathbf{x}} \leq 1$ . Utiliza-se o sinal menor ou igual para representar todo o elipsóide e não apenas a sua casca ou borda.

Analisando-se  $\mathbf{J}^T \mathbf{J}$ , tem-se equivalentemente que as raízes quadradas dos inversos dos autovalores definirão o comprimento dos eixos principais do elipsóide de velocidades (fatores de amplificação ou redução) e os autovetores definirão estes eixos no espaço.

Quando **Chablat e Wenger (2003)** projetavam seu robô, eles propuseram que os autovalores da matriz característica do elipsóide de velocidades deveriam estar compreendidos entre  $1/\kappa$  e  $\kappa$ , onde  $\kappa$  valia 9 para todos os pontos do volume de trabalho como forma de suavização e otimização da transmissão de velocidades. O estudo também possui uma vantagem matemática, pois evita o cálculo de operações de matrizes inversas, que é custoso. A única ressalva que se faz é que o estudo era para um robô de três graus de liberdade de translação. Quando são misturados graus de liberdade de translação com graus de liberdade de rotação têm-se resultados (autovalores) de diferentes ordens de grandeza, o que torna muito difícil a construção de algoritmos eficientes.

Uma outra maneira de se realizar estudos semelhantes é através da hipótese de deflexão de uma unidade de distância para a plataforma móvel. É também importante lembrar que  $\mathbf{C}^T\mathbf{C}$  também é positiva semi-definida e simétrica. Sendo assim, também é possível escrever que:

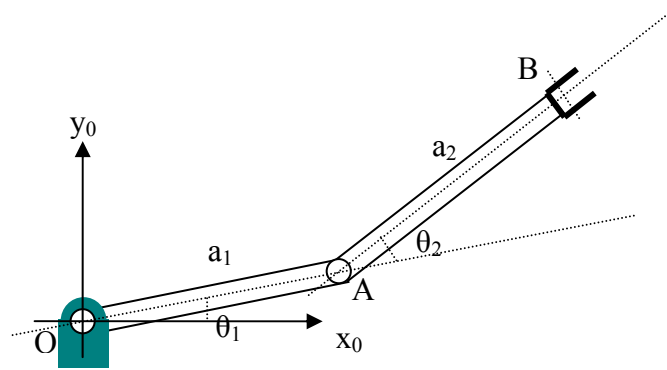
$$(\Delta\mathbf{x})^T(\Delta\mathbf{x})=1 \quad (91)$$

$$\mathbf{F}^T(\mathbf{C}^T\mathbf{C})\mathbf{F}=1 \quad (92)$$

Em resumo, este estudo é um ponto chave também para a otimização de robôs de arquitetura paralela. Será detalhado um procedimento utilizando estes conceitos no capítulo pertinente.

### **6.5.1. Um Exemplo Numérico do Elipsóide de Forças**

Embora seja apresentado apenas o exemplo com o elipsóide de forças, as análises para o elipsóide de velocidades são semelhantes. Mudam os conceitos e os propósitos. O exemplo a seguir foi baseado em **Tsai (1999)**, que consiste em um robô seriado de dois graus de liberdade da Fig. 37.



**Figura 37:** Robô seriado de dois graus de liberdade.

Para os robôs de arquitetura seriada a matriz em questão é  $\mathbf{J}\mathbf{J}^T$ , já que  $\Delta\mathbf{x} = \mathbf{J} \cdot \Delta\mathbf{q}$ , o que implica em  $\boldsymbol{\tau} = \mathbf{J}^T \cdot \mathbf{F}$ . A posição do ponto  $B$  com relação ao centro do sistema de eixos coordenados é dada por (lembrar que  $\theta_{12} = \theta_1 + \theta_2$ ):

$${}^0\mathbf{P}_B = \begin{bmatrix} a_1 \cos \theta_1 + a_2 \cos \theta_{12} \\ a_1 \sin \theta_1 + a_2 \sin \theta_{12} \end{bmatrix} \quad (93)$$

Derivando a equação de posição, tem-se a matriz Jacobiano apresentada logo a seguir.

$$\mathbf{J} = \begin{bmatrix} \frac{\partial^0 P_{B-\text{linha1}}}{\partial \theta_1} & \frac{\partial^0 P_{B-\text{linha1}}}{\partial \theta_2} \\ \frac{\partial^0 P_{B-\text{linha2}}}{\partial \theta_1} & \frac{\partial^0 P_{B-\text{linha2}}}{\partial \theta_2} \end{bmatrix} = \begin{bmatrix} -a_1 \sin \theta_1 - a_2 \sin \theta_{12} & -a_2 \sin \theta_{12} \\ a_1 \cos \theta_1 + a_2 \cos \theta_{12} & a_2 \cos \theta_{12} \end{bmatrix} \quad (94)$$

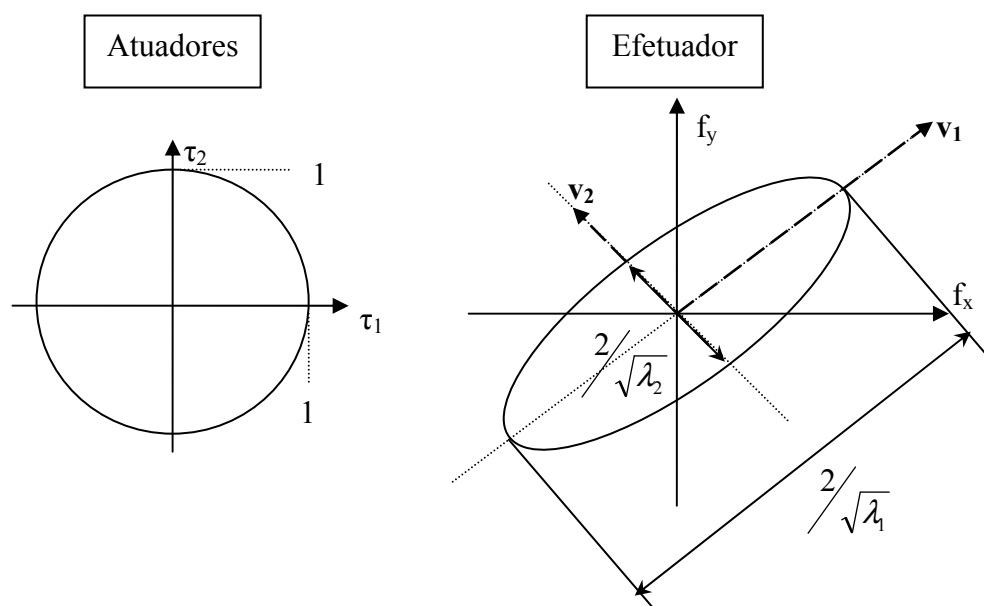
Agora, tendo-se o Jacobiano, dois casos são apresentados para mostrar o elipsóide de forças. Ambos os casos consideram que  $a_1$  e  $a_2$  valem, respectivamente,  $\sqrt{2}$  m e 1 m, e  $\theta_1$  vale  $0^\circ$ .

**Caso 1:**  $\theta_2 = 90^\circ \Rightarrow \mathbf{J} = \begin{bmatrix} -1 & -1 \\ \sqrt{2} & 0 \end{bmatrix} \therefore \mathbf{J}\mathbf{J}^T = \begin{bmatrix} 2 & -\sqrt{2} \\ -\sqrt{2} & 2 \end{bmatrix}$

Os autovalores são  $\lambda_1 = 0.5858$  e  $\lambda_2 = 3.4142$ , e os respectivos autovetores são  $\mathbf{v}_1 = \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$  e  $\mathbf{v}_2 = \begin{bmatrix} -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}$  (eixos principais do elipsóide).

Assim tem-se que a equação da elipse é:

$$0.5858 \left( \frac{f_x}{\sqrt{2}} + \frac{f_y}{\sqrt{2}} \right)^2 + 3.4142 \left( \frac{f_y}{\sqrt{2}} - \frac{f_x}{\sqrt{2}} \right)^2 = 1 \quad (95)$$



**Figura 38:** Elipsóide de forças no efetuador do robô serial de dois graus de liberdade.

**Caso 2:**  $\theta_2 = 0^\circ$  (uma singularidade)

Essa é uma posição de singularidade, onde o robô perde um grau de liberdade (em x). É importante observar que a rigidez teórica em x é infinita.

$$\mathbf{J} = \begin{bmatrix} 0 & 0 \\ 1 + \sqrt{2} & 1 \end{bmatrix} \therefore \mathbf{J}\mathbf{J}^T = \begin{bmatrix} 0 & 0 \\ 0 & 6.8284 \end{bmatrix}$$

Os autovalores são  $\lambda_1 = 0$  e  $\lambda_2 = 6.8284$ , e os respectivos autovetores são  $\mathbf{v}_1 = [1 \ 0]$  e  $\mathbf{v}_2 = [0 \ 1]$ . A equação do elipsóide é, na verdade,  $6.8284(f_y)^2 = 1$ , ou seja, apenas dois valores de  $f_y$  são capazes de fazer com que  $\boldsymbol{\tau}^T \cdot \boldsymbol{\tau} = 1$  seja verdadeiro, independentemente dos valores de  $f_x$ . Nesse caso o eixo  $\mathbf{v}_1$  da elipse é infinito.

## 6.6. Análise da Cinemática Direta

No problema da cinemática inversa, obteve-se uma série de equações não lineares, mas que tinham uma grande vantagem: estavam *desacopladas*. O problema da cinemática direta em robôs de arquitetura paralela já não é tão simples. A maneira tradicional para se chegar a uma solução resulta num conjunto de equações não lineares *acopladas* de difíceis e múltiplas soluções, mesmo quando se utilizam ferramentas computacionais. Do ponto de vista puramente analítico, dependendo da arquitetura, o problema pode ser até mesmo impossível de se resolver.

Na abordagem tradicional devem-se buscar relações geométricas que relacionem os pontos de interesse da máquina e permitam que a postura da plataforma móvel possa ser escrita numa equação cujas entradas sejam os ângulos de atuação. Além das eq. (8) até a eq. (13) e eq. (20) até a eq. (28), devem ser escritas equações de distância dos pontos da plataforma móvel, atrelando cada um de forma a definir realmente seis pontos de um hexágono com distâncias relativas fixas. O sistema de equações resultante terá termos ao quadrado e uma variável multiplicando uma outra. A solução de sistemas deste tipo com este número de variáveis (depende de como é posto o problema, mas seguindo desta forma serão 18) é difícil.

Tendo em vista a complexidade do problema, muitos pesquisadores sugerem o uso de uma abordagem computacional baseada em estimativas da posição e orientação da máquina a partir da matriz Jacobiano. **Maurine et al., (1999)** e **Last et al. (2005)**, por exemplo, utilizaram esse método em seus estudos com a Hexa.

Conforme apresentado na eq. (77),  $\Delta \mathbf{q} = \mathbf{J} \cdot \Delta \mathbf{x}$ . Se for considerado um espaço de tempo muito pequeno, os erros incorridos por essa aproximação não serão altos. Parte-se então do pressuposto que existe uma postura da máquina na qual tanto

a posição dos atuadores bem como as coordenadas que descrevem esta postura são conhecidas. Nessa posição é possível calcular a matriz Jacobiano  $\mathbf{J}_{(x_0, q_0)}$ . Assim, a partir deste ponto trabalha-se com estimativas da posição futura (representadas por “ $\tilde{\cdot}$ ”). O símbolo  $n$  representa o número atual da iteração matemática.

$$\mathbf{q}_1 - \mathbf{q}_0 = \mathbf{J}_{(x_0, q_0)}(\tilde{\mathbf{x}}_1 - \mathbf{x}_0) \quad (96)$$

$$\tilde{\mathbf{x}}_1 = \mathbf{x}_0 + \mathbf{J}_{(x_0, q_0)}^{-1}(\mathbf{q}_1 - \mathbf{q}_0) \quad (97)$$

$$\tilde{\mathbf{x}}_{n+1} = \tilde{\mathbf{x}}_n + \mathbf{J}_{(x_0, q_0)}^{-1}(\mathbf{q}_{n+1} - \mathbf{q}_n) \quad (98)$$

A matriz Jacobiano pode ou não ser atualizada ao longo do tempo, dependendo do custo computacional demandado versus o custo que o controlador da máquina aceita. Se a matriz puder ser calculada em tempo real, ela será dada por  $\mathbf{J}_{(\tilde{x}_n, q_n)}^{-1}$ . Na prática costuma-se atualizar a matriz após algumas iterações.

Uma ferramenta construída apenas a título ilustrativo é apresentada em detalhes no capítulo de resultados. A ferramenta aceita como entrada os pontos de partida e destino para os quais a ponta da ferramenta deve se locomover, entretanto, internamente, utiliza o ponto de partida para criar uma matriz Jacobiano e segue o método proposto sem conhecimento do ponto de destino. Ao final, compara o ponto estimado com o ponto real para dois casos: atualizando o Jacobiano e sem atualizar o Jacobiano.

## 7. ANÁLISE DINÂMICA INVERSA

Parte essencial do projeto do robô é saber seu comportamento dinâmico e conhecer as forças envolvidas nos elementos mecânicos ao longo de sua operação. Esse tipo de estudo permite uma melhora do projeto mecânico de todo o equipamento. Embora não seja um dos tópicos deste trabalho, uma outra função é prover um modelo rápido computacionalmente para ser utilizado no sistema de controle da máquina em tempo real. Alguns dos robôs mais modernos são capazes de medir e corrigir desvios de trajetória provocados por forças externas.

É importante salientar que neste trabalho apenas a dinâmica inversa será descrita, ou seja, conhecendo-se os dados relativos à plataforma fixa, calculam-se os outros parâmetros do robô (forças e torques nas articulações atuadas e não atuadas e nos ligamentos). A dinâmica direta não será estudada.

### 7.1. A Abordagem para a Obtenção do Modelo

A análise dinâmica de um robô de arquitetura paralela é quase sempre mais difícil se comparada à análise dos robôs seriados, devido à existência de múltiplas cadeias cinemáticas.

Muitas abordagens ao problema já foram propostas, incluindo a formulação de Newton-Euler, a formulação de Lagrange e o princípio do trabalho virtual (ou formulação de D' Alembert). Cada abordagem possui suas vantagens e desvantagens.

Na formulação de Newton-Euler a dedução é mais simples e é possível calcular os esforços de reação entre as partes constituintes da máquina (incluindo motores), o que provê um excelente modelo para o projeto mecânico. Todavia, devido ao número de equações e complexidade das mesmas, a eficiência computacional do modelo se torna pobre, dificultando um controle adequado em tempo real.

Já na formulação de Lagrange, por ser possível eliminar os esforços internos que não interessam para a maioria dos algoritmos de controle, consegue-se um modelo mais eficiente. Todavia, devido ao número de restrições impostas por manipuladores de cadeia fechada, torna-se extremamente difícil deduzir equações



com coordenadas generalizadas independentes. Para resolver esse problema, alguns pesquisadores sugerem a adição de coordenadas extras, junto aos multiplicadores de Lagrange (Tsai, 1999).

O princípio do trabalho virtual é o que apresenta melhor relação custo benefício, permitindo-se conhecer as principais informações do modelo sem um custo computacional elevado.

Outras abordagens não serão analisadas neste trabalho, já que geralmente são apenas variações dos outros três métodos, ou se aplicam a apenas um mecanismo em particular, ou ainda são abordagens puramente numéricas e específicas para controle.

Neste trabalho vai-se focar na formulação de Newton-Euler por prover informações no maior grau de detalhe possível. Já que não é objetivo o controle, um baixo custo computacional para aplicações em tempo real não é necessário.

## 7.2. Metodologia e Conceitos na Formulação de Newton-Euler da Hexa

Ao se investigar a análise dinâmica da Hexa, são utilizados os passos descritos na Fig. 39. O robô é composto de uma base fixa, uma plataforma móvel e seis conjuntos compostos por dois ligamentos ( $\mathbf{h}_j$  e  $\mathbf{l}_j$ ). Todavia, estes elementos se unem formando uma máquina onde as peças possuem uma relação forte de dependência, não sendo possíveis cálculos individuais como nos robôs seriados. Sendo assim, esses elementos devem ser analisados um a um e devem formar um grande sistema de equações cuja solução fornece todos os dados desejados. O diagrama de corpo livre está representado na Fig. 41.

A Fig. 40 ilustra a separação e seleção dos elementos de interesse para um conjunto de ligamentos. As articulações serão desconsideradas do modelo (atrito, massa, etc.), contudo as distâncias dadas por  $q$ ,  $d$ ,  $h$ ,  $l$  e  $t$  devem levar em consideração os centros das articulações.

No caso da massa das articulações, pode-se utilizar algum artifício como o de manipular as matrizes de inércia dos ligamentos e da plataforma móvel a fim de diminuir os erros incorridos por esta aproximação.

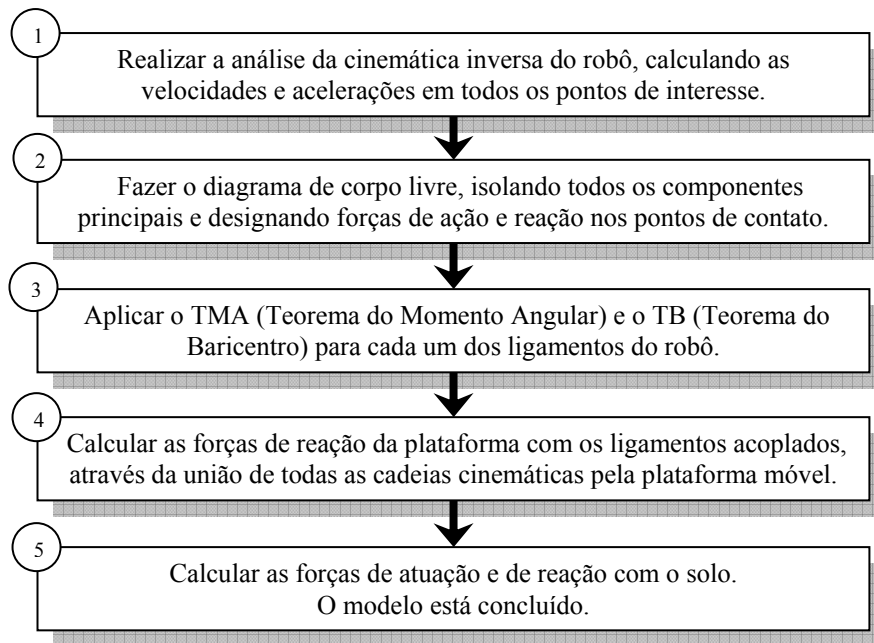


Figura 39: Passos para obtenção do modelo dinâmico pela formulação de Newton-Euler.

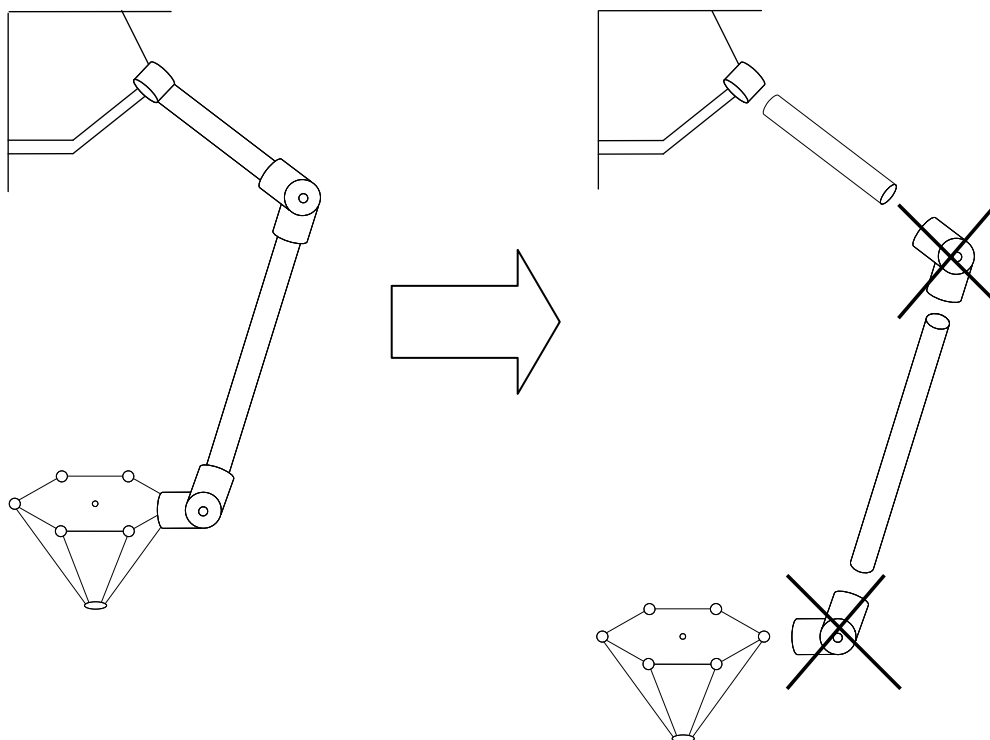


Figura 40: Esquema dos elementos principais da Hexa.

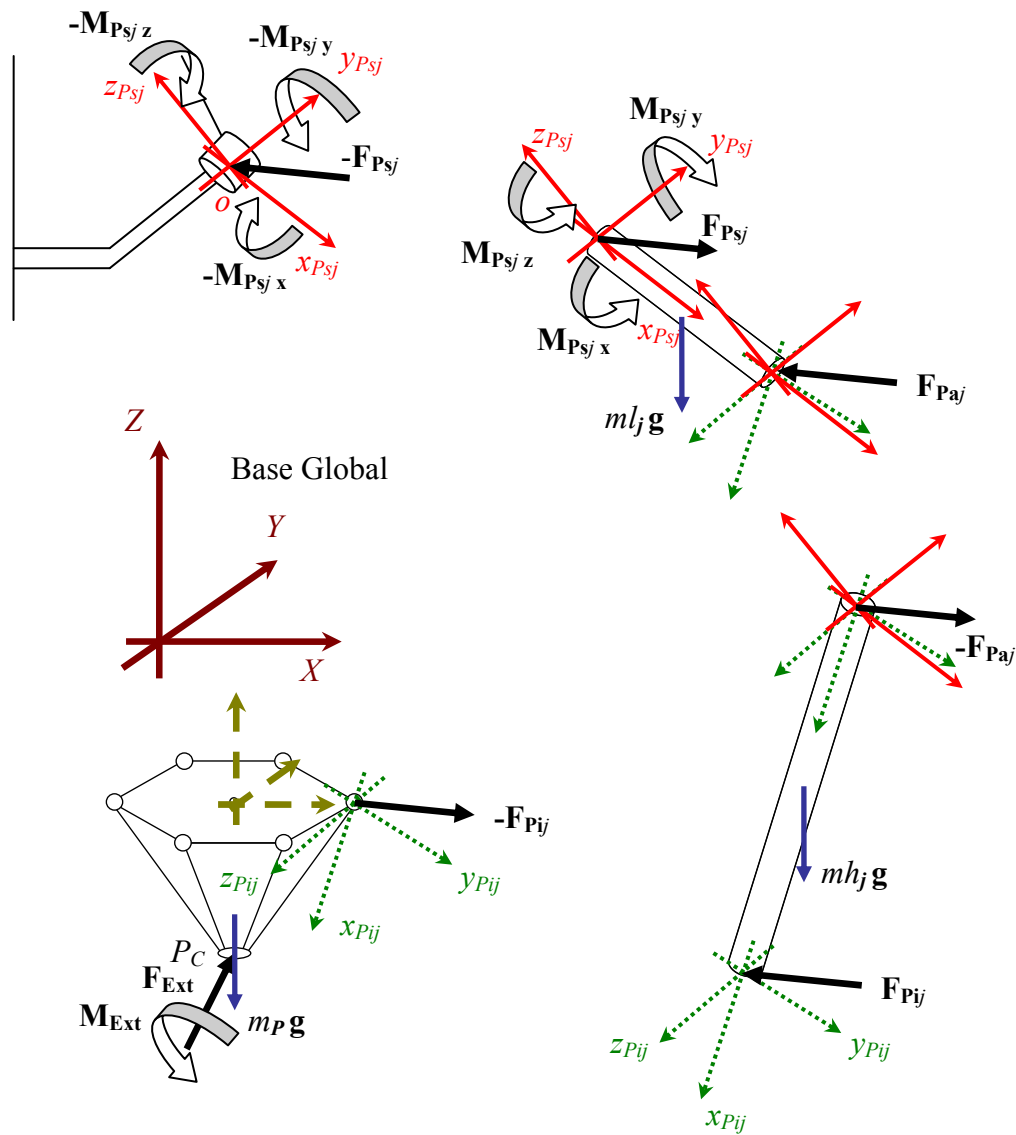


Figura 41: Diagrama de Corpo Livre da Hexa.

No desenho do modelo, vai-se trabalhar com quatro sistemas de coordenadas: o *Global* ( $O-XYZ$ ), o sistema do ponto  $Ps_j$  cujo eixo  $x$  acompanha o ligamento  $\mathbf{l}_j$ , o sistema do ponto  $Pi_j$  cujo eixo  $x$  acompanha o ligamento  $\mathbf{h}_j$  e o sistema do baricentro da plataforma móvel. O ponto  $Pa_j$  pode utilizar qualquer um dos sistemas, não possuindo um específico. Estas bases são chamadas, respectivamente, de:  $lig_{l_j}$  (base cujos eixos estão desenhados cheios),  $lig_{h_j}$  (base pontilhada) e  $plat$  (base tracejada). O sistema *Global*, por ser o sistema padrão, não será referenciado quando utilizado (exceto onde possa gerar confusão). Os outros serão sempre indicados.

Os centros de massa dos ligamentos  $\mathbf{l}_j$  e  $\mathbf{h}_j$  e da plataforma móvel serão representados respectivamente por  $G_{lj}$ ,  $G_{hj}$  e  $G_P$ . As massas desses elementos serão  $m_{lj}$ ,  $m_{hj}$  e  $m_P$ . Os vetores que marcam os centros de massa são dados por  $\mathbf{c}_{lj}$ ,  $\mathbf{c}_{hj}$  e  $\mathbf{c}_P$ , sendo o primeiro partindo de  $P_{Sj}$  e apontando para  $G_{lj}$ . O segundo parte de  $P_{Aj}$  e chega a  $G_{hj}$ . Por último, o vetor da plataforma parte de  $P_C$  e aponta para  $G_P$ .

As forças trocadas entre os elementos principais da Hexa são  $\mathbf{F}_{Pij}$ ,  $\mathbf{F}_{PAj}$  e  $\mathbf{F}_{Psj}$ . Seus índices marcam os pontos a que se referem. Os torques são representados pela letra  $M$  (momento) e o índice  $Ext$  representa externo (força ou momento).

As matrizes de inércia dos ligamentos passivos, dos atuados e da plataforma móvel são, respectivamente,  $\mathbf{I}_{lj}$ ,  $\mathbf{I}_{hj}$  e  $\mathbf{I}_P$ . Elas são sempre referentes aos baricentros de seus corpos rígidos e a base na qual estão escritas será sobrescrita no lado esquerdo, como, por exemplo,  ${}^{lighj}\mathbf{I}_{hj}$ .

Para as equações dos modelos, serão utilizados o Teorema do Momento Angular (TMA) e o Teorema do Baricentro (TB). O primeiro aplicado ao baricentro de um corpo rígido (dado pelo índice  $G$ ), é dado pela eq. (99). O segundo que provém da 2ª Lei de Newton é dado pela eq. (100).

$$\sum \mathbf{M}_G = \dot{\mathbf{H}}_G \quad (99)$$

$$\sum \mathbf{F} = m\mathbf{a}_G \quad (100)$$

O TMA rege que a somatória dos torques (ou momentos) aplicados a um corpo rígido com respeito ao seu centro de massa é igual à derivada do momento angular no mesmo ponto. O TB diz que a somatória de todas as forças aplicadas a um corpo rígido é igual ao produto da massa do corpo pela aceleração linear do seu centro de massa.

Seja  $\mathbf{I}$  a matriz de inércia de um corpo rígido. O momento angular em relação ao baricentro será dado pelo produto da matriz  $\mathbf{I}$  pela velocidade angular deste corpo, segundo mostra a eq. (101). A matriz  $\mathbf{I}$  é dada pela eq. (102) e seus elementos estão descritos nas eq. (103) e eq. (104), respectivamente os momentos de inércia e os produtos de inércia.

$$\mathbf{H}_G = \mathbf{I}\boldsymbol{\omega} \quad (101)$$

$$\mathbf{I} = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix} \quad (102)$$

$$\begin{cases} I_x = \int (y^2 + z^2) dm \\ I_y = \int (x^2 + z^2) dm \\ I_z = \int (x^2 + y^2) dm \end{cases} \quad (103)$$

$$\begin{cases} I_{xy} = \int xy \cdot dm \\ I_{xz} = \int xz \cdot dm \\ I_{yz} = \int yz \cdot dm \end{cases} \quad (104)$$

No cálculo do momento angular utiliza-se a velocidade angular do corpo. Assim, é natural que se utilizem os eixos fixos. No entanto, tem-se uma matriz de inércia  $\mathbf{I}$  variante no tempo, já que o corpo rígido assume uma posição e uma orientação diferentes a cada momento. Esse fato é indesejável, pois o cálculo da derivada do momento angular ficaria lento e muito complexo de se fazer.

Para resolver este inconveniente, deve-se perceber que o sistema de referência pode ser alterado, por exemplo, para um sistema solidário ao baricentro do corpo rígido em estudo. Para tanto, deve-se escrever a velocidade angular do corpo rígido nas coordenadas dos eixos solidários a este corpo. Procedendo desta forma a matriz de inércia torna-se invariante no tempo.

Além disso, deve-se entender a derivada de vetores com relação a eixos em rotação. Embora a equação valha para qualquer vetor, tem-se como exemplo para o momento angular que:

$$\left(\dot{\mathbf{H}}_G\right)_{O-XYZ} = \left(\dot{\mathbf{H}}_G\right)_{o-xyz} + \boldsymbol{\Omega} \times \left(\mathbf{H}_G\right)_{O-XYZ} \quad (105)$$

Onde:

$(\dot{\mathbf{H}}_G)_{o-xyz}$	Derivada do momento angular em relação ao sistema girante solidário ao corpo $o-xyz$ .
$(\dot{\mathbf{H}}_G)_{O-XYZ}$	Derivada do momento angular em relação ao sistema de orientação fixa $O-XYZ$ .
$\boldsymbol{\Omega} \times (\mathbf{H}_G)_{O-XYZ}$	Produto vetorial da velocidade angular do sistema $o-xyz$ e do momento angular do corpo em relação ao sistema fixo $O-XYZ$ .

Uma última relação também importante é a transferência do vetor de momento angular para outro ponto que não o baricentro. Seja  $\mathbf{H}_O$  o momento angular num ponto  $O$  genérico e  $\mathbf{r}_{OG}$  o vetor que parte de  $O$  e aponta para  $G$  (baricentro):

$$\mathbf{H}_O = \mathbf{H}_G + \mathbf{r}_{OG} \times (m\mathbf{v}_G) \quad (106)$$

### 7.3. Equações da Análise Dinâmica da Hexa

Como uma observação inicial, todas as matrizes de inércia serão utilizadas apenas nos baricentros dos corpos.

Nas equações que se seguem, a notação  ${}^Y X_Z$  indicará a função  $X$  com relação a  $Z$  utilizando a base  $Y$ . Apenas para as matrizes de transformação de coordenadas  $\mathbf{R}$ , tem-se a notação  ${}^Y \mathbf{R}_Z$ , indicando uma transformação da base  $Z$  para a  $Y$ . Como exemplo, para a aceleração da gravidade  $\mathbf{g}$  a equação de conversão é dada por:

$${}^{lig\ hj} \mathbf{g} = \left( {}^{lig\ hj} \mathbf{R}_{Global} \right) {}^{Global} \mathbf{g} \quad (107)$$

Recordando a figura do diagrama de corpo livre, e atentando-se ao ponto  $Pa_j$ , calcula-se o momento angular do ligamento  $\mathbf{h}_j$ . A base utilizada será a do ponto  $Pi_j$  cujo eixo  $x$  acompanha o ligamento  $\mathbf{h}_j$  (os eixos aparecem tracejados e na cor verde).

$${}^{lig\ hj}(\mathbf{H}_{h_j})_{Pa_j} = m_{h_j} \left( {}^{lig\ hj} \mathbf{c}_{h_j} \times {}^{lig\ hj} \mathbf{v}_{Gh_j} \right) + {}^{lig\ hj} \mathbf{I}_{h_j} {}^{lig\ hj} \boldsymbol{\omega}_{h_j} \quad (108)$$

$$\begin{aligned} {}^{lig\ hj}(\dot{\mathbf{H}}_{h_j})_{Pa_j} &= m_{h_j} \left( {}^{lig\ hj} \mathbf{c}_{h_j} \times {}^{lig\ hj} \dot{\mathbf{v}}_{Gh_j} \right) + {}^{lig\ hj} \mathbf{I}_{h_j} {}^{lig\ hj} \dot{\boldsymbol{\omega}}_{h_j} + \\ &+ {}^{lig\ hj} \boldsymbol{\omega}_{h_j} \times \left( {}^{lig\ hj} \mathbf{I}_{h_j} {}^{lig\ hj} \boldsymbol{\omega}_{h_j} \right) \end{aligned} \quad (109)$$

$$\mathbf{v}_{Gh_j} = \mathbf{v}_{Pa_j} + \boldsymbol{\omega}_{h_j} \times (G_{h_j} - Pa_j) \quad (110)$$

$$\dot{\mathbf{v}}_{Gh_j} = \dot{\mathbf{v}}_{Pa_j} + \dot{\boldsymbol{\omega}}_{h_j} \times (G_{h_j} - Pa_j) + \boldsymbol{\omega}_{h_j} \times (\boldsymbol{\omega}_{h_j} \times (G_{h_j} - Pa_j)) \quad (111)$$

$$\sum {}^{lig\ hj}(\mathbf{M}_{h_j})_{Pa_j} = m_{h_j} \left( {}^{lig\ hj} \mathbf{c}_{h_j} \times {}^{lig\ hj} \mathbf{g} \right) + {}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j y} + {}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j z} \quad (112)$$

O TMA rege que:

$$\sum {}^{lig\ hj}(\mathbf{M}_{h_j})_{Pa_j} = {}^{lig\ hj}(\dot{\mathbf{H}}_{h_j})_{Pa_j} \quad (113)$$

Logo,

$$\begin{aligned} {}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j y} + {}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j z} &= m_{h_j} \left( {}^{lig\ hj} \mathbf{c}_{h_j} \times {}^{lig\ hj} \dot{\mathbf{v}}_{Gh_j} \right) + \\ &+ {}^{lig\ hj} \mathbf{I}_{h_j} {}^{lig\ hj} \dot{\boldsymbol{\omega}}_{h_j} + {}^{lig\ hj} \boldsymbol{\omega}_{h_j} \times \left( {}^{lig\ hj} \mathbf{I}_{h_j} {}^{lig\ hj} \boldsymbol{\omega}_{h_j} \right) - m_{h_j} \left( {}^{lig\ hj} \mathbf{c}_{h_j} \times {}^{lig\ hj} \mathbf{g} \right) \end{aligned} \quad (114)$$

$${}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j y} + {}^{lig\ hj} \mathbf{h}_j \times {}^{lig\ hj} \mathbf{F}_{Pi_j z} = h_j F_{Pij y} {}^{lig\ hj} \mathbf{z} - h_j F_{Pij z} {}^{lig\ hj} \mathbf{y} \quad (115)$$

Com estas equações é possível encontrar duas componentes das forças nos pontos de contato entre a plataforma móvel e os ligamentos passivos. Contudo, ambos os lados das equações não possuem termos em  $x$ , justamente porque não se podem calcular as forças aplicadas no sentido do ligamento (tração e compressão) com os dados e equações existentes de momento.

A obtenção dos termos em  $x$  é apresentada a seguir, contudo, a próxima equação mostra como calcular os esforços nos pontos  $Pa_j$  (pontos de contato entre os

dois tipos de ligamentos: passivos e atuados). É o TB aplicado aos ligamentos passivos.

$$\mathbf{F}_{\mathbf{P}i_j} - \mathbf{F}_{\mathbf{P}a_j} + m_{h_j} \mathbf{g} = m_{h_j} \dot{\mathbf{v}}_{\mathbf{G}h_j} \quad (116)$$

Agora focando na plataforma móvel e seguindo o mesmo processo realizado para os ligamentos passivos, tem-se que:

$$(\mathbf{H}_{\mathbf{P}})_{\mathbf{G}P} = {}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_{\mathbf{P}} \boldsymbol{\omega}_{\mathbf{P}} \quad (117)$$

$$(\dot{\mathbf{H}}_{\mathbf{P}})_{\mathbf{G}P} = {}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_{\mathbf{P}} \dot{\boldsymbol{\omega}}_{\mathbf{P}} + \boldsymbol{\omega}_{\mathbf{P}} \times ({}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_{\mathbf{P}} \boldsymbol{\omega}_{\mathbf{P}}) \quad (118)$$

Momento de forças:

$$\sum \mathbf{M}_{\mathbf{G}P} = (P_C - G_P) \times \mathbf{F}_{\mathbf{E}xt} + \mathbf{M}_{\mathbf{E}xt} + \sum_{j=1}^6 \left[ (Pi_j - G_P) \times {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_j}) \right] \quad (119)$$

De acordo com o TMA:

$$(\dot{\mathbf{H}}_{\mathbf{P}})_{\mathbf{G}P} = \sum \mathbf{M}_{\mathbf{G}P} \quad (120)$$

$$\sum_{j=1}^6 \left[ (Pi_j - G_P) \times {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_j}) \right] = (P_C - G_P) \times \mathbf{F}_{\mathbf{E}xt} + \mathbf{M}_{\mathbf{E}xt} - \left[ {}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_{\mathbf{P}} \dot{\boldsymbol{\omega}}_{\mathbf{P}} + \boldsymbol{\omega}_{\mathbf{P}} \times ({}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_{\mathbf{P}} \boldsymbol{\omega}_{\mathbf{P}}) \right] \quad (121)$$

Onde:

$$(Pi_j - G_P) \times {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_j}) = \begin{bmatrix} (Pi_{j_x} - G_{P_x}) \\ (Pi_{j_y} - G_{P_y}) \\ (Pi_{j_z} - G_{P_z}) \end{bmatrix} \times \left( {}^{Global} \mathbf{R}_{lig\ h_j} \begin{bmatrix} {}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_{j_x}} \\ {}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_{j_y}} \\ {}^{lig\ h_j} \mathbf{F}_{\mathbf{P}i_{j_z}} \end{bmatrix} \right) \quad (122)$$



$${}^{Global} \mathbf{R}_{lig\ h_j} = \begin{bmatrix} R_{j11} & R_{j12} & R_{j13} \\ R_{j21} & R_{j22} & R_{j23} \\ R_{j31} & R_{j32} & R_{j33} \end{bmatrix} \quad (123)$$

Portanto, tem-se que:

$$\begin{aligned} (P_{i_j} - G_P) \times {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ hj} \mathbf{F}_{P_{i_j}}) &= \\ &= \begin{bmatrix} 0 & -(P_{i_{jz}} - G_{Pz}) & (P_{i_{jy}} - G_{Py}) \\ (P_{i_{jz}} - G_{Pz}) & 0 & -(P_{i_{jx}} - G_{Px}) \\ -(P_{i_{jy}} - G_{Py}) & (P_{i_{jx}} - G_{Px}) & 0 \end{bmatrix} \cdot \\ &\cdot \begin{bmatrix} R_{j11} & R_{j12} & R_{j13} \\ R_{j21} & R_{j22} & R_{j23} \\ R_{j31} & R_{j32} & R_{j33} \end{bmatrix} \begin{bmatrix} {}^{lig\ hj} \mathbf{F}_{P_{i_{jx}}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i_{jy}}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i_{jz}}} \end{bmatrix} \end{aligned} \quad (124)$$

Por simplicidade, na equação anterior, as duas matrizes 3x3 multiplicadas serão chamadas de  $\mathbf{M}_j$ , resultando em:

$$(P_{i_j} - G_P) \times {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ hj} \mathbf{F}_{P_{i_j}}) = \begin{bmatrix} M_{j11} & M_{j12} & M_{j13} \\ M_{j21} & M_{j22} & M_{j23} \\ M_{j31} & M_{j32} & M_{j33} \end{bmatrix} \begin{bmatrix} {}^{lig\ hj} \mathbf{F}_{P_{i_{jx}}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i_{jy}}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i_{jz}}} \end{bmatrix} \quad (125)$$

No sistema de equações descrito pela eq. (120), percebe-se que as incógnitas são os termos em  $x$  buscados. Todavia, o sistema em questão possui apenas três equações, mas contém seis incógnitas ( ${}^{lig\ hj} \mathbf{F}_{P_{i_{jx}}}$ , para  $j = 1$  a 6). Aplicando o TB na plataforma, tem-se a eq. (126), que também pode ser escrita como a eq. (127):

$$-\sum_{j=1}^6 \left[ {}^{Global} \mathbf{R}_{lig\ h_j} ({}^{lig\ hj} \mathbf{F}_{P_{i_j}}) \right] + m_P \mathbf{g} + \mathbf{F}_{Ext} = m_P \dot{\mathbf{v}}_{G_P} \quad (126)$$

$$\sum_{j=1}^6 \left[ {}^{Global} \mathbf{R}_{lig\ h_j} \left( {}^{lig\ h_j} \mathbf{F}_{P_{i_j}} \right) \right] = -m_p \dot{\mathbf{v}}_{G_p} + m_p \mathbf{g} + \mathbf{F}_{Ext} \quad (127)$$

A velocidade e aceleração do baricentro da plataforma móvel são dadas pelas equações:

$$\mathbf{v}_{G_p} = \mathbf{v}_{P_c} + \boldsymbol{\omega}_p \times (G_p - P_c) \quad (128)$$

$$\dot{\mathbf{v}}_{G_p} = \dot{\mathbf{v}}_{P_c} + \dot{\boldsymbol{\omega}}_p \times (G_p - P_c) + \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times (G_p - P_c)) \quad (129)$$

O sistema final de equações da plataforma móvel, apresentado na eq. (120) e na eq. (121) somado às equações lineares do sistema da eq. (127) pode ser apresentado na forma  $\mathbf{Ax} = \mathbf{B}$ , onde:

$$\mathbf{A} = \begin{bmatrix} M_{j=111} & M_{j=211} & M_{j=311} & M_{j=411} & M_{j=511} & M_{j=611} \\ M_{j=121} & M_{j=221} & M_{j=321} & M_{j=421} & M_{j=521} & M_{j=621} \\ M_{j=131} & M_{j=231} & M_{j=331} & M_{j=431} & M_{j=531} & M_{j=631} \\ R_{j=111} & R_{j=211} & R_{j=311} & R_{j=411} & R_{j=511} & R_{j=611} \\ R_{j=121} & R_{j=221} & R_{j=321} & R_{j=421} & R_{j=521} & R_{j=621} \\ R_{j=131} & R_{j=231} & R_{j=331} & R_{j=431} & R_{j=531} & R_{j=631} \end{bmatrix} \quad (130)$$

$$\mathbf{x} = \begin{bmatrix} {}^{lig\ h_j} \mathbf{F}_{P_{i_j=1x}} \\ {}^{lig\ h_j} \mathbf{F}_{P_{i_j=2x}} \\ {}^{lig\ h_j} \mathbf{F}_{P_{i_j=3x}} \\ {}^{lig\ h_j} \mathbf{F}_{P_{i_j=4x}} \\ {}^{lig\ h_j} \mathbf{F}_{P_{i_j=5x}} \\ {}^{lig\ h_j} \mathbf{F}_{P_{i_j=6x}} \end{bmatrix} \quad (131)$$

E para completar o sistema, tem-se que:

$$\mathbf{B} = \begin{bmatrix} (P_C - G_P) \times \mathbf{F}_{\text{Ext}} + \mathbf{M}_{\text{Ext}} - \\ - \left[ {}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_P \dot{\boldsymbol{\omega}}_P + \boldsymbol{\omega}_P \times \left( {}^{Global} \mathbf{R}_{plat} {}^{plat} \mathbf{I}_P \boldsymbol{\omega}_P \right) \right] \\ - m_P \dot{\mathbf{v}}_{G_P} + m_P \mathbf{g} + \mathbf{F}_{\text{Ext}} \end{bmatrix} - \begin{bmatrix} M_{j=1_{12}} & M_{j=2_{12}} & M_{j=3_{12}} & M_{j=4_{12}} & M_{j=5_{12}} & M_{j=6_{12}} \\ M_{j=1_{22}} & M_{j=2_{22}} & M_{j=3_{22}} & M_{j=4_{22}} & M_{j=5_{22}} & M_{j=6_{22}} \\ M_{j=1_{32}} & M_{j=2_{32}} & M_{j=3_{32}} & M_{j=4_{32}} & M_{j=5_{32}} & M_{j=6_{32}} \\ R_{j=1_{12}} & R_{j=2_{12}} & R_{j=3_{12}} & R_{j=4_{12}} & R_{j=5_{12}} & R_{j=6_{12}} \\ R_{j=1_{22}} & R_{j=2_{22}} & R_{j=3_{22}} & R_{j=4_{22}} & R_{j=5_{22}} & R_{j=6_{22}} \\ R_{j=1_{32}} & R_{j=2_{32}} & R_{j=3_{32}} & R_{j=4_{32}} & R_{j=5_{32}} & R_{j=6_{32}} \end{bmatrix} \begin{bmatrix} {}^{lig\ hj} \mathbf{F}_{P_{i\ j=1_y}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=2_y}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=3_y}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=4_y}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=5_y}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=6_y}} \end{bmatrix} - \begin{bmatrix} M_{j=1_{13}} & M_{j=2_{13}} & M_{j=3_{13}} & M_{j=4_{13}} & M_{j=5_{13}} & M_{j=6_{13}} \\ M_{j=1_{23}} & M_{j=2_{23}} & M_{j=3_{23}} & M_{j=4_{23}} & M_{j=5_{23}} & M_{j=6_{23}} \\ M_{j=1_{33}} & M_{j=2_{33}} & M_{j=3_{33}} & M_{j=4_{33}} & M_{j=5_{33}} & M_{j=6_{33}} \\ R_{j=1_{13}} & R_{j=2_{13}} & R_{j=3_{13}} & R_{j=4_{13}} & R_{j=5_{13}} & R_{j=6_{13}} \\ R_{j=1_{23}} & R_{j=2_{23}} & R_{j=3_{23}} & R_{j=4_{23}} & R_{j=5_{23}} & R_{j=6_{23}} \\ R_{j=1_{33}} & R_{j=2_{33}} & R_{j=3_{33}} & R_{j=4_{33}} & R_{j=5_{33}} & R_{j=6_{33}} \end{bmatrix} \begin{bmatrix} {}^{lig\ hj} \mathbf{F}_{P_{i\ j=1_z}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=2_z}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=3_z}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=4_z}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=5_z}} \\ {}^{lig\ hj} \mathbf{F}_{P_{i\ j=6_z}} \end{bmatrix} \quad (132)$$

Com esse sistema, encerra-se o cálculo de  ${}^{lig\ hj} \mathbf{F}_{P_{i\ j}}$ . Essa variável vetorial pode então ser escrita nos seus eixos globais e pode-se aplicar a eq. (116) para se obter  $\mathbf{F}_{P_{a\ j}}$ . Não existem torques nessas juntas, já que os momentos transmitidos serão zero (juntas esféricas ou punhos esféricos não transmitem momento). Falta então calcular as forças de reação com a base fixa. Para isso aplica-se o TB nos ligamentos atuados. As velocidades e acelerações do baricentro são apresentadas logo depois:

$$\mathbf{F}_{P_{s\ j}} + \mathbf{F}_{P_{a\ j}} + m_{l_j} \cdot \mathbf{g} = m_{l_j} \cdot \dot{\mathbf{v}}_{G_{l_j}} \quad (133)$$

Onde a velocidade e a aceleração do baricentro dos ligamentos atuados são, respectivamente, dadas por:

$$\mathbf{v}_{G_{l_j}} = \dot{\boldsymbol{\theta}} \times (G_{l_j} - P_{S_j}) \quad (134)$$

$$\dot{\mathbf{v}}_{G_{l_j}} = \ddot{\boldsymbol{\theta}} \times (\mathbf{G}_{l_j} - \mathbf{P}_{S_j}) + \dot{\boldsymbol{\theta}} \times (\dot{\boldsymbol{\theta}} \times (\mathbf{G}_{l_j} - \mathbf{P}_{S_j})) \quad (135)$$

Aplicando agora o TMA nos ligamentos atuados no centro das articulações atuadas tem-se que:

$$\begin{aligned} {}^{lig lj} (\boldsymbol{\tau}_j)_{P_{S_j}} = & -m_{l_j} {}^{lig lj} \mathbf{c}_{1_j} \times {}^{lig lj} \mathbf{g} - {}^{lig lj} \mathbf{I}_{1_j} \times {}^{lig lj} \mathbf{F}_{Pa_j} + {}^{lig lj} \mathbf{I}_{1_j} {}^{lig lj} \ddot{\boldsymbol{\theta}}_j + \\ & + {}^{lig lj} \dot{\boldsymbol{\theta}}_j \times ({}^{lig lj} \mathbf{I}_{1_j} {}^{lig lj} \dot{\boldsymbol{\theta}}_j) + m_{l_j} ({}^{lig lj} \mathbf{c}_{1_j} \times {}^{lig lj} \dot{\mathbf{v}}_{G_{l_j}}) \end{aligned} \quad (136)$$

Através desta equação têm-se os torques nas articulações. Para se saber o torque nos motores, basta dividir o resultado pela relação de redução existente. Com isso, encerra a análise dinâmica da Hexa pela formulação de Newton – Euler.

O equacionamento do modelo buscou um total de 18 vetores de forças e 6 de torques, sendo quatro vetores por cadeia ( $18 + 6 = 24$ ,  $\times 3 = 72$  variáveis escalares!). Além disso, existem as variáveis intermediárias, como as do modelo cinemático. Apenas os vetores de velocidades e acelerações dos pontos característicos  $P_{i_j}$  e  $P_{a_j}$  (24 vetores) e dos baricentros dos corpos envolvidos (26 vetores) totalizam 50 vetores, ou 150 variáveis escalares!

## 8. OTIMIZAÇÃO

Este capítulo tem por objetivo apresentar algumas abordagens para a otimização de robôs, em especial, a Hexa. Alguns fundamentos teóricos básicos também podem ser encontrados no Apêndice D.

### 8.1. Uma Introdução aos Conceitos de Otimização e de Síntese

O processo de desenho de robôs é muitas vezes prejudicado pela ausência de regras bem conhecidas e estabelecidas capazes de transformar requerimentos em configurações. As mentes de projetistas humanos não conseguem entender plenamente o espaço possível de soluções e o impacto nas variáveis de projeto ou no desempenho dos robôs. Muito da experiência de cada um é empregada na fase de projeto do equipamento. Mesmo quando se está modificando um projeto existente, pequenas alterações podem acarretar em grandes mudanças no todo que são difíceis de prever manualmente: o tamanho do volume de trabalho, o tamanho e a potência dos atuadores, o consumo de energia (importante em algumas aplicações como robôs para operar no espaço), e muitos outros.

Processos manuais de síntese (como foi utilizado aqui para definir que a arquitetura a ser estudada seria a Hexa) envolvem uma forma de projeto onde poucas configurações são criadas e analisadas, os projetos são limitados à experiência relevante disponível, existem poucas regras gerais, existe uma grande dificuldade em prever qual será o desempenho do robô e assim por diante. Algoritmos automáticos de síntese baseiam-se em busca ao invés de experiência, em simulações para avaliar o desempenho de cada solução, em exploração por amostragem do espaço possível de soluções e avaliação computacional do desempenho e não apenas a amostragem do espaço alcançável pela mente do projetista.

Embora muitos pesquisadores considerem síntese e otimização como apenas nomenclaturas diferentes para o mesmo conceito, outros como **Leger (1999)** fazem uma diferenciação. **Leger (1999)** separa bem os termos *síntese de configurações* e *otimização de configurações*. O primeiro refere-se ao processo de criação: definição

do tipo de cinemática e, no mínimo, as dimensões desse novo robô; o segundo considera um espaço de soluções menor e foca em refinar uma solução específica melhorando seu desempenho, sendo ambos os casos muito importantes na concepção e projeto de novos equipamentos. Outros autores também são citados em seu trabalho, como **Rosenman (1997)** que define na teoria evolucionária aplicada à computação os termos *projeto evolucionário criativo* e *otimização do projeto evolucionário*, sendo o primeiro retratado na frase: “Quanto menor o conhecimento das relações existentes entre requerimentos e formas de se satisfazer estes requerimentos, mais um problema se aproxima de um projeto criativo”.

**Leger (1999)**, em sua tese de Doutorado, constrói e apresenta o algoritmo Darwin2K baseado em conceitos de algoritmos genéticos (teoria evolucionária) e possuindo uma arquitetura totalmente voltada a objetos. O algoritmo não realiza análises muito complexas, mas é capaz de realizar análises dinâmicas, calcular deflexões de elementos, consumo de energia, detecção de colisões, saturação de atuadores, projeto de controladores PID e outros, baseando-se em elementos de projeto muito simples, conforme apresentados na Fig. 2041.

Dentre os vários exemplos que **Leger (1999)** apresenta, pode-se citar o de um transportador de materiais, como o da Fig. 44. Na figura, à direita, tem-se uma ilustração do que o algoritmo recebe como requerimentos de projeto. Quanto à Fig. 45, tem-se um exemplo de população inicial e o resultado de três simulações do programa. Os percentuais indicam quanto dos requerimentos puderam ser cumpridos com a solução. Apenas para ilustrar a versatilidade do software, a Fig. 43 também ilustra a concepção de um robô espacial que caminha em gravidade zero.

Essa dissertação não focaliza algoritmos de síntese, já que se optou por um processo de seleção manual da arquitetura. O foco é a otimização da Hexa, ou seja, refinar os seus parâmetros para que ela tenha um melhor desempenho, respeitando algumas restrições impostas.

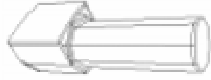
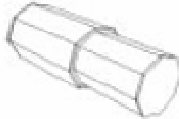


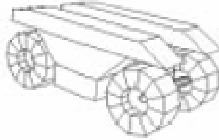


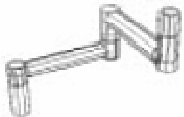
module type	parameters	
rightAngleJoint	components: motor, gearbox, material length between connectors tube diameter wall thickness	
inlineRevolut e2	components: motor, gearbox, material length between connectors tube diameter wall thickness	
prismaticTube	components: motor, gearbox, lead screw, material outer diameter wall thickness segment length	
hollowTube	material selection length outer diameter wall thickness	
oc1Chassis	wheelbase, engine location (front to back), front-to-back position of connector (to which other modules can be attached) connector height	
offsetElbow	components: motor, gearbox, material distance between actuator housing and plate initial joint angle wall thickness	
stackerBase	total number of bins, number of bins vertically, x and y location of connector, x and y location of payload entry point	
scaraElbow	components: motor, gearbox for each of 3 joints; material for all links length, wall thickness, diameter for links joint angle offsets for each joint	

Figura 42: Elementos básicos do algoritmo Darwin2K (Leger, 1999).

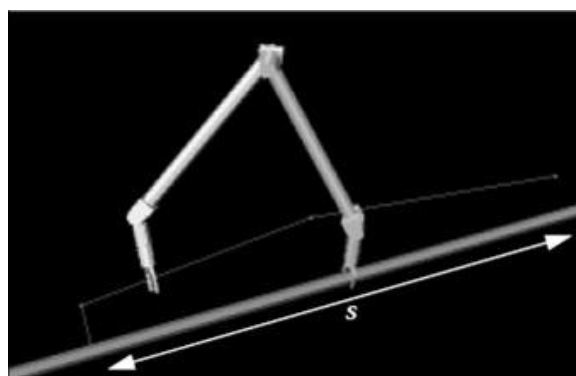


Figura 43: Robô que opera no espaço (em gravidade zero). Figura de Leger (1999).

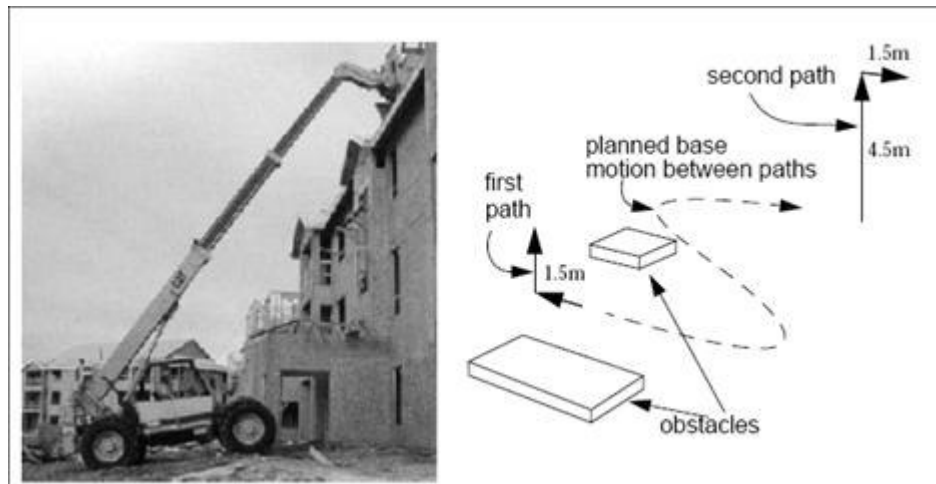


Figura 44: Requerimentos para um transportador de materiais (Leger, 1999).

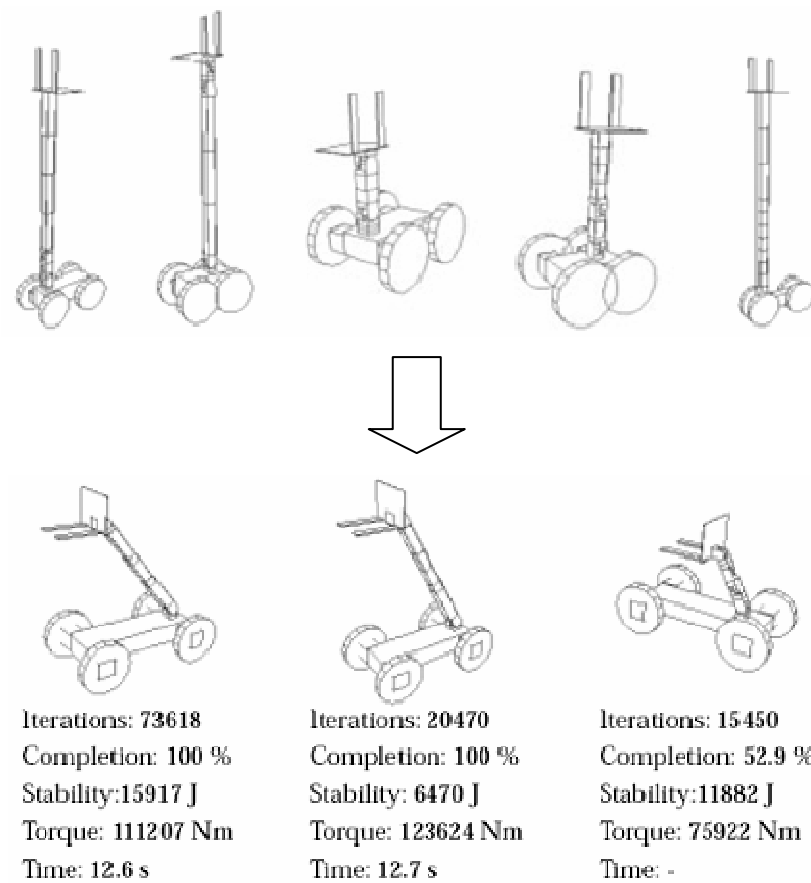


Figura 45: População inicial e resultado de três simulações do Darwin2K (Leger, 1999).



## 8.2. A Otimização da Hexa

### 8.2.1. Otimização Completa de Robôs de Arquitetura Paralela

Essa seqüência já foi explicada no capítulo de revisão bibliográfica, mas está reproduzida aqui de forma mais detalhada. A seqüência de otimizações proposta por **Brogårdh (2002)** está ilustrada na Fig. 46. Segundo ele, essa abordagem também será útil em futuras pesquisas sobre modularização, que para os robôs de arquitetura paralela será uma tarefa mais simples que para robôs seriados, já que utilizam componentes mais simples e com menores interdependências mecânicas. Atenta-se também para as iterações entre os passos do algoritmo: cada uma deve se comunicar com a antecessora e a sucessora.

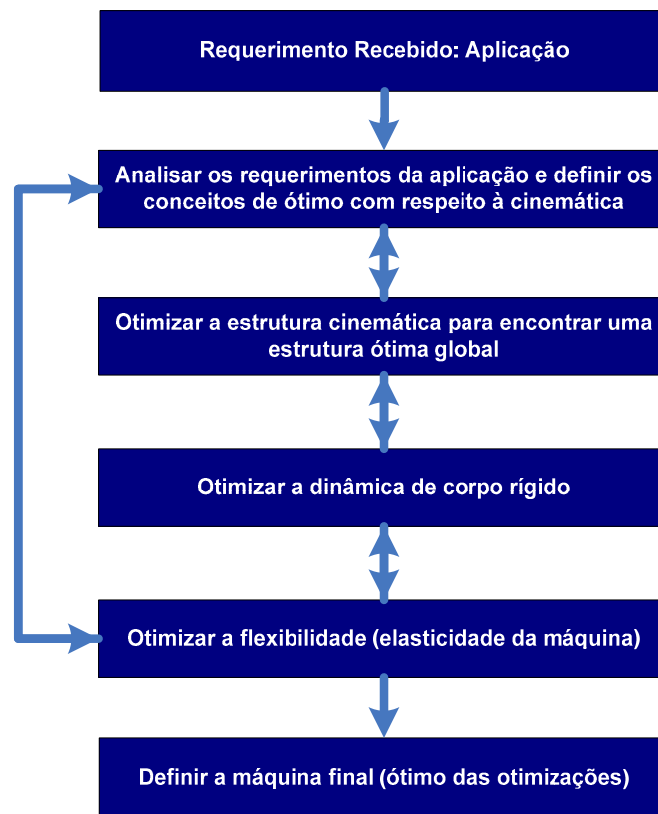


Figura 46: Seqüência para obter uma estrutura ótima.

O trabalho de **Brogårdh (2002)** representa o estado da arte em otimização de robôs de arquitetura paralela, contemplando todos os aspectos inerentes a um modelo mecânico completo. Obviamente, o desenvolvimento deste modelo não é uma tarefa simples e exige a utilização de softwares complexos como o ADAMS (sugerido pelo autor), fato este também reforçado por **Tartari e Marchiori (2003)**, que mostraram a dificuldade de definir funções multiobjetivos em otimizações dinâmicas com restrições complexas e não lineares, utilizando a Hexaglide como foco dos estudos.

### **8.2.2. Abordagens para a Otimização da Hexa**

Neste trabalho vai-se ater à otimização cinemática, focando na proposta de **Chablat e Wenger (2003)**. Os outros passos descritos na Fig. 46 e a ligação entre eles devem ser abordados em trabalhos futuros. Vai-se buscar a otimização do elipsóide de forças / velocidades do robô em seu volume de trabalho, respeitando como restrições um volume mínimo que a máquina deve ter e algumas restrições laterais da geometria da arquitetura.

Embora **Chablat e Wenger (2003)** tenham utilizado a abordagem dos elipsóides no robô Orthoglide de três graus de liberdade de translação, o conceito pode ser expandido para um robô de seis graus de liberdade. Infelizmente, não há menção ao algoritmo de otimização (método computacional) de seu robô.

Quando se analisa o elipsóide de forças, deseja-se que ele seja o mais uniforme possível (perto de uma transformação isotrópica) e também o maior possível, já que quanto maior seu volume, menor serão as forças nos atuadores. Quanto mais uniforme o elipsóide mais uniforme será o movimento do robô (dentre outras vantagens, menores as derivadas das acelerações).

Quando se analisa o elipsóide de velocidades (trabalho de **Chablat e Wenger, 2003**) deseja-se também que ele seja o mais uniforme possível, entretanto não que ele seja o maior possível, mas sim que ele fique dentro de certos padrões para que não se amplifique ou se reduza muito as velocidades no efetuador, o que pode trazer problemas na resolução da máquina também. Para a Orthoglide tem-se uma análise mais simples, já que seus atuadores são lineares e são apenas três graus de liberdade, contudo na Hexa isso já não é verdade.

Existe ainda uma abordagem através da maximização da rigidez, proposta por **Kim e Tsai (2002)**, onde se maximizou a diagonal principal da matriz de rigidez (autovalores) de uma arquitetura de três graus de liberdade. Nessa arquitetura conseguiu-se uma matriz diagonal, o que simplificou bastante o problema.

A respeito do método de **Chablat e Wenger (2003)** vale apenas uma observação a respeito do número de condição da matriz Jacobiano. Na Hexa, devido aos seus seis graus de liberdade, três graus (os de translação) terão autovalores com ordens de grandeza não necessariamente iguais aos outros três (de rotação). Além disso, as juntas atuadas produzem torques e o efetuator responde a forças e torques. Têm-se sempre metros e radianos, m/s e rad/s, Newton e Newton\*metro “misturados” nas equações. **Maurine et al., (1999)**, em estudos para a calibragem da Hexa, relatam terem registrado as posições dos motores para várias posições da máquina. Esses pontos serviram como entrada do algoritmo de otimização construído (para minimizar o erro de posicionamento) e foram eliminados aqueles pontos de entrada que resultaram num número de condição do Jacobiano maior que 300. Esse valor serve como idéia da diferença na ordem de grandeza que se espera encontrar entre os autovalores do modelo.

### **8.2.3. Equacionamento do Problema de Otimização da Hexa e Método**

#### **8.2.3.1. Método Numérico para as Otimizações**

Será utilizada a função *fmincon* do Matlab, que utiliza um algoritmo de otimização baseado no método *SQL* (*Sequential Quadratic Programming*) para encontrar mínimos locais. O programa no cálculo das derivadas (gradiente e Hessiana) utiliza algoritmos de diferenças finitas.

#### **8.2.3.2. Variáveis de Projeto**

Como variáveis de projeto são utilizadas as dimensões da Hexa ( $d, q, l, h, t$  e  $h_{tool}$ ), sendo todas variáveis contínuas, que contabilizam seis no total.

### 8.2.3.3. Restrições

Como restrições, o algoritmo conta com restrições laterais para cada variável. Elas evitam resultados matematicamente possíveis, mas fisicamente incoerentes como comprimentos de ligamentos negativos ou muito grandes.

Além de restrições laterais, é utilizada uma restrição de desigualdade para permitir que um volume de trabalho mínimo de orientação total definido pelo usuário seja respeitado. Este volume, com formato de um paralelepípedo, é definido pelas dimensões  $D_X$ ,  $D_Y$  e  $D_Z$ , onde a coordenada  $X$  varia de  $-D_X/2$  a  $D_X/2$  e a coordenada  $Y$  varia de  $-D_Y/2$  a  $D_Y/2$ .

Embora o complexo volume de trabalho da Hexa seja mais bem aproveitado com a definição de um cilindro, escolheu-se um paralelepípedo por ser mais usual na indústria se utilizar volumes com tais formatos. Dessa forma busca-se otimizar a máquina dentro deste espaço, mas sem restringir seus movimentos a apenas este paralelepípedo escolhido (pode ir para qualquer ponto de seu volume de trabalho).

O algoritmo utiliza o método para cálculo do volume de trabalho explicado no capítulo de cinemática inversa apenas em alguns pontos específicos e sempre na direção  $Z$  (positiva e negativa). Nestes pontos de coordenadas  $[X, Y]$  os limites mínimo e máximo de  $Z$  pertencentes ao volume de trabalho são calculados. Com os limites de todos os pontos determinados, o algoritmo seleciona o “maior dos mínimos”  $[\max(Z_{Min})]$  e o menor dos máximos  $[\min(Z_{Max})]$  e determina o espaço em  $Z$  onde o paralelepípedo pode ser posicionado. A restrição é dada então por:

$$D_Z - [\max(Z_{Min}) - \min(Z_{Max})] \leq 0 \quad (137)$$

Os pontos verificados são os quatro vértices do paralelepípedo definido (ex.:  $[D_X/2, D_Y/2]$ ) e os pontos entre eles (ex.:  $[0, D_Y/2]$ ), resultando num total de oito pontos para verificações. A Fig. 47 ilustra as avaliações e o que se busca com elas.

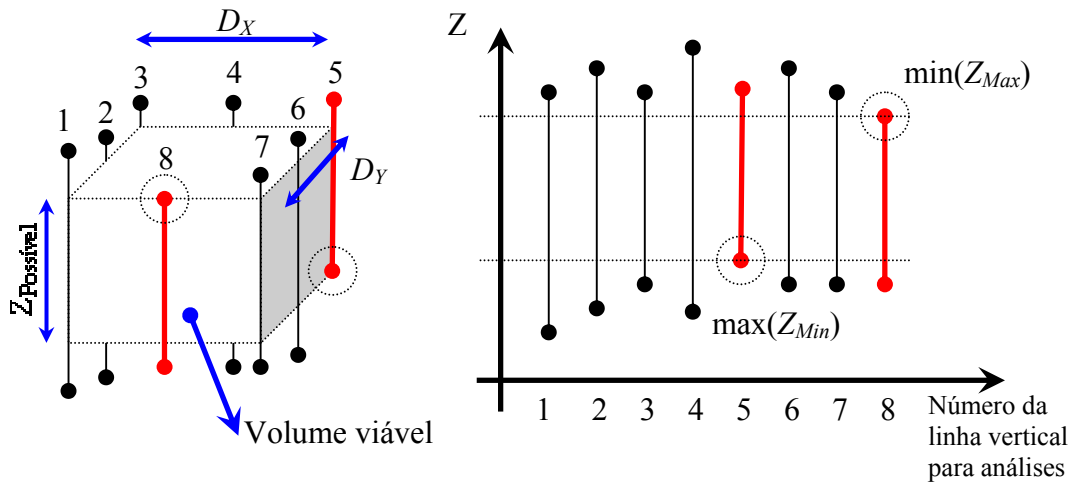


Figura 47: Restrição do volume de trabalho na otimização.

#### 8.2.3.4. Funções Objetivo

Para as funções objetivo poderiam ser utilizadas três abordagens diferentes: uma com o elipsóide de forças, uma com o elipsóide de velocidades e uma com o elipsóide de deslocamentos. Contudo, a análise através do elipsóide de deslocamentos é análoga ao do elipsóide de forças. Os objetivos desejados para ambos são iguais e a matriz característica do elipsóide de deslocamentos é igual à matriz do elipsóide de forças ao quadrado.

Busca-se então otimizar a máquina com cada uma das funções separadamente e analisar os resultados. As equações base para a definição das funções são apresentadas a seguir (elipsóides de forças e de velocidades).

$$\mathbf{F}^T (\mathbf{J}^T \mathbf{J})^{-1} \mathbf{F} \leq 1 \quad (138)$$

$$\dot{\mathbf{x}}^T (\mathbf{J}^T \mathbf{J}) \dot{\mathbf{x}} \leq 1 \quad (139)$$

Nessas funções a análise realizada será a dos autovalores das matrizes características, dadas respectivamente por  $(\mathbf{J}^T \mathbf{J})^{-1}$  e  $\mathbf{J}^T \mathbf{J}$ .

Para a análise das funções objetivo faz-se necessário a verificação em diversos pontos do volume de trabalho qual é o valor dessa função. Assim, o volume escolhido pelo usuário é tratado como uma *malha tridimensional de pontos onde a função é avaliada em cada um*. Para isso são definidos os números de pontos, dados por  $[n_X, n_Y, n_Z]$ , avaliados no paralelepípedo definido em cada dimensão  $X$ ,  $Y$  e  $Z$ . O total de pontos avaliados para calcular a função objetivo é  $n_{Total\ de\ Pontos} = n_X * n_Y * n_Z$ .

Quando a restrição de desigualdade não está ativa, o algoritmo se utiliza dessa “sobra” de espaço para também avaliá-lo e assim poder escolher a melhor posição para posicionar o volume definido dentro do espaço  $Z$  disponível.

Quando a restrição de desigualdade é violada, o algoritmo calcula normalmente a função objetivo escolhida dentro do espaço  $Z$  disponível, com o propósito de não causar problemas ao método numérico, já que descontinuidades na função ou nas suas derivadas (gradiente e Hessiana) podem prejudicar a convergência do algoritmo.

Para o elipsóide de deslocamentos, a equação e a matriz característica são dadas por:

$$\mathbf{F}^T (\mathbf{C}^T \mathbf{C}) \mathbf{F} \leq 1 \quad (140)$$

$$\mathbf{C}^T \mathbf{C} = \left[ (\mathbf{J}^T \mathbf{J})^{-1} \right]^T (\mathbf{J}^T \mathbf{J})^{-1} \quad (141)$$

Conforme é possível se observar, a matriz característica é semelhante ao do elipsóide de forças. Na verdade, caso essa matriz seja utilizada, irá apenas dificultar o algoritmo de otimização que terá que lidar com autovalores muito maiores (ao quadrado).

**Elipsóide de Forças**

Para o elipsóide de forças buscam-se dois objetivos: uma transformação isotrópica (ou seja, eixos do elipsóide do mesmo tamanho) e a maximização do menor eixo do elipsóide (o que implica no aumento de seu volume). Se o maior eixo do elipsóide for  $\xi_{Max}$  e o menor for  $\xi_{Min}$ , então o problema de otimização é dado por:

Transform. Maximizar  
 Isotrópica menor eixo

Pior Ponto  
 Encontrado  
 no Volume

$$\text{Minimizar } f(x) = \frac{\sum_{\text{Pontos do Volume}} \left[ \frac{\xi_{Max}}{\xi_{Min}} \left( 1 + \frac{1}{\xi_{Min}} \right) \right]}{n_{\text{Total de Pontos}}} + \text{Max} \left\{ \frac{\xi_{Max}}{\xi_{Min}} \left( 1 + \frac{1}{\xi_{Min}} \right) \right\}$$

Variáveis de Projeto :  $d, q, l, h, t, h_{tool}$  (142)

Tal que :  $D_Z - [\max(Z_{Min}) - \min(Z_{Max})] \leq 0$

**Elipsóide de Velocidades**

Para o elipsóide de velocidades, busca-se majoritariamente uma transformação isotrópica. Segundo **Chablat e Wenger (2003)** devem-se manter os autovalores dentro de certos parâmetros para que não se incorra também numa perda acentuada de resolução dos atuadores ou ocorram casos onde a máquina se movimenta muito lentamente. Frente às diferentes ordens de grandeza encontradas nos autovalores, optou-se por não acrescentar mais restrições. O problema de otimização é dado por:

$$\text{Minimizar } f(x) = \frac{\sum_{\text{Pontos do Volume}} \frac{\xi_{Max}}{\xi_{Min}}}{n_{\text{Total de Pontos}}} + \text{Max} \left\{ \frac{\xi_{Max}}{\xi_{Min}} \right\}$$

Variáveis de Projeto :  $d, q, l, h, t, h_{tool}$  (143)

Tal que :  $D_Z - [\max(Z_{Min}) - \min(Z_{Max})] \leq 0$

## 9. FERRAMENTAS CONSTRUÍDAS E RESULTADOS

Neste item são apresentadas as ferramentas construídas na plataforma de desenvolvimento Matlab, versão 6.5 para demonstrar os conceitos detalhados neste trabalho. Não é objetivo que se chegue ao projeto de um robô Hexa, mas sim apresentar ferramentas que no futuro poderão habilitar esse desejo.

A ferramenta encontra-se no CD-ROM, chamado de Apêndice E, anexo a este trabalho.

### 9.1. O Modelo Virtual da Hexa e seus Parâmetros Geométricos

Exceto pela cinemática direta, todas as funções desenvolvidas estão integradas numa única ferramenta, cuja janela inicial apresentada na Fig. 48 é ativada no Matlab pelo comando *HexaStart*. Optou-se por utilizar a língua inglesa em todas as janelas, permitindo que a ferramenta seja visualizada por uma quantidade maior de pessoas na comunidade acadêmica e industrial. A Tabela 3 apresenta o significado dos componentes da janela inicial.

A Fig. 49 apresenta a janela de ajuda que aparece quando o botão de ajuda é pressionado. Além disso, na tela de gráficos aparece o esquema de ajuda apresentado na Fig. 48. A Fig. 50 mostra o esquema virtual da Hexa e um zoom na plataforma móvel; telas estas produzidas quando são apertados, respectivamente, os botões de “Machine View” e “Mobile Platform View”.

Foram utilizados como parâmetros de entrada do programa para o propósito de apresentar os resultados os seguintes parâmetros:

- Variáveis da base fixa  $d$  e  $q$  de 200 mm cada uma;
- Comprimento dos ligamentos atuados, dados por  $l$ , de 300 mm;
- Comprimento dos ligamentos passivos, dados por  $h$ , de 500 mm;
- Lado do hexágono da plataforma móvel e altura da ferramenta, dados, respectivamente, por  $t$  e  $h_{tool}$ , de 100 mm cada;
- Ponto de interesse centrado, ou seja, a variável  $C(x,y)$  vale  $[0\ 0]$ ;



- Distância  $H$  entre a base fixa e o hexágono da plataforma móvel de 400 mm.

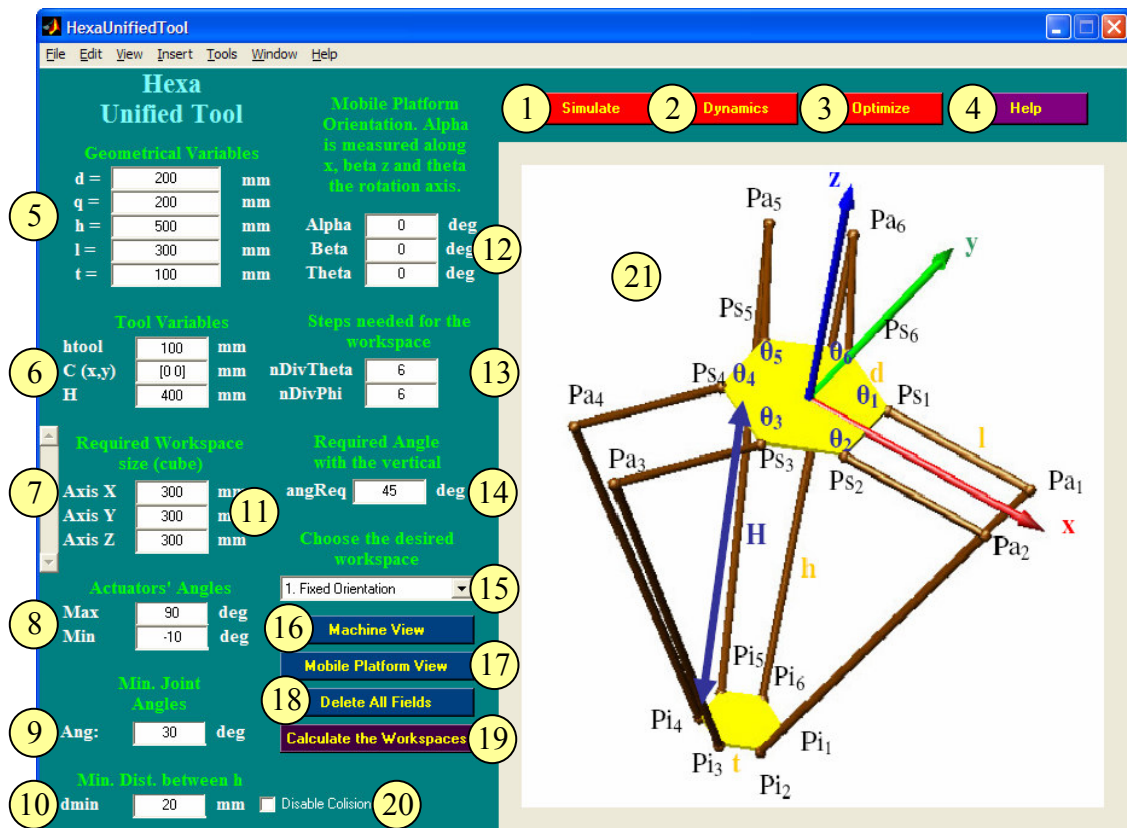


Figura 48: Janela inicial da ferramenta desenvolvida em Matlab.

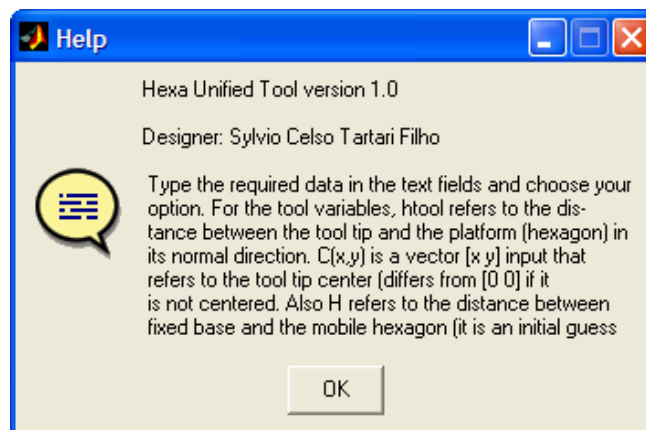


Figura 49: Janela de Ajuda.

Nº	Tipo	Ação / Variável
1	Botão	Ativa a ferramenta do simulador virtual.
2	Botão	Ativa a ferramenta de análise dinâmica.
3	Botão	Ativa a ferramenta de otimização.
4	Botão	Exibe uma janela de texto sobre como utilizar o software.
5	Campo Texto	Variáveis geométricas da estrutura da máquina <sup>‡</sup> .
6	Campo Texto	Variáveis da plataforma móvel <sup>§</sup> .
7	Slider	Posiciona o volume de trabalho desejado na tela gráfica.
8	Campo Texto	Ângulos máximos e mínimos que o atuador pode assumir.
9	Campo Texto	Ângulo mínimo entre os ligamentos do tipo $l$ e do tipo $h$ .
10	Campo Texto	Distância mínima aceitável entre dois ligamentos $h$ quaisquer.
11	Campo Texto	Volume de trabalho desejado (tomado como um paralelepípedo).
12	Campo Texto	Orientação da plat. móvel para o vol. de trab. de orient. fixa.
13	Campo Texto	Número de pontos para o cálculo dos volumes de trabalho <sup>**</sup> .
14	Campo Texto	Ângulo exigido da plataforma com a vertical para o vol. de trab.
15	Combo Box	Escolhe o volume de trabalho a visualizar.
16	Botão	Mostra a máquina conforme definida pelos campos de texto.
17	Botão	Mostra apenas a plataforma móvel.
18	Botão	Apaga o conteúdo de todos os campos de texto na janela.
19	Botão	Calcula os volumes de trabalho (orientação fixa e total).
20	Check Box	Habilita ou desabilita os algoritmos de detecção de colisões.
21	Tela Gráfica	Tela onde são exibidos os volumes de trab. e a máquina.

**Tabela 3:** Itens da janela inicial da ferramenta construída.

<sup>‡</sup> Essas variáveis já foram discutidas previamente no capítulo de cinemática inversa. São elas:

$d \rightarrow$  é a distância entre dois ligamentos atuados paralelos.

$q \rightarrow$  distância entre o centro da base fixa e o centro da linha de  $Ps_1$  a  $Ps_2$  ou  $Ps_3$  a  $Ps_4$  ou  $Ps_5$  a  $Ps_6$ .

$h \rightarrow$  comprimento dos ligamentos passivos.

$l \rightarrow$  comprimento dos ligamentos atuados.

$t \rightarrow$  lado da plataforma móvel (que é um hexágono).

<sup>§</sup> Essas variáveis definem a plataforma móvel e sua posição inicial:

$h_{tool} \rightarrow$  distância da ponta da ferramenta à plataforma móvel, seguindo a reta normal do hexágono.

$C(x,y) \rightarrow$  coordenadas  $x$  e  $y$  da ponta da ferramenta, caso ela não seja centrada (é apenas uma função da ferramenta, mas de utilidades práticas limitadas. Geralmente se usa  $[0\ 0]$  mesmo (centrada).

$H \rightarrow$  distância inicial da base fixa ao hexágono móvel (para fins de desenhar a máquina).

<sup>\*\*</sup> O cálculo do volume utiliza coordenadas polares, assim, esse número de divisões diz respeito ao número de pontos no plano  $xy$  de  $0$  a  $360^\circ$  ( $nDivTheta$ ) e no plano  $xz$  de  $0$  a  $180^\circ$  ( $nDivPhi$ ).

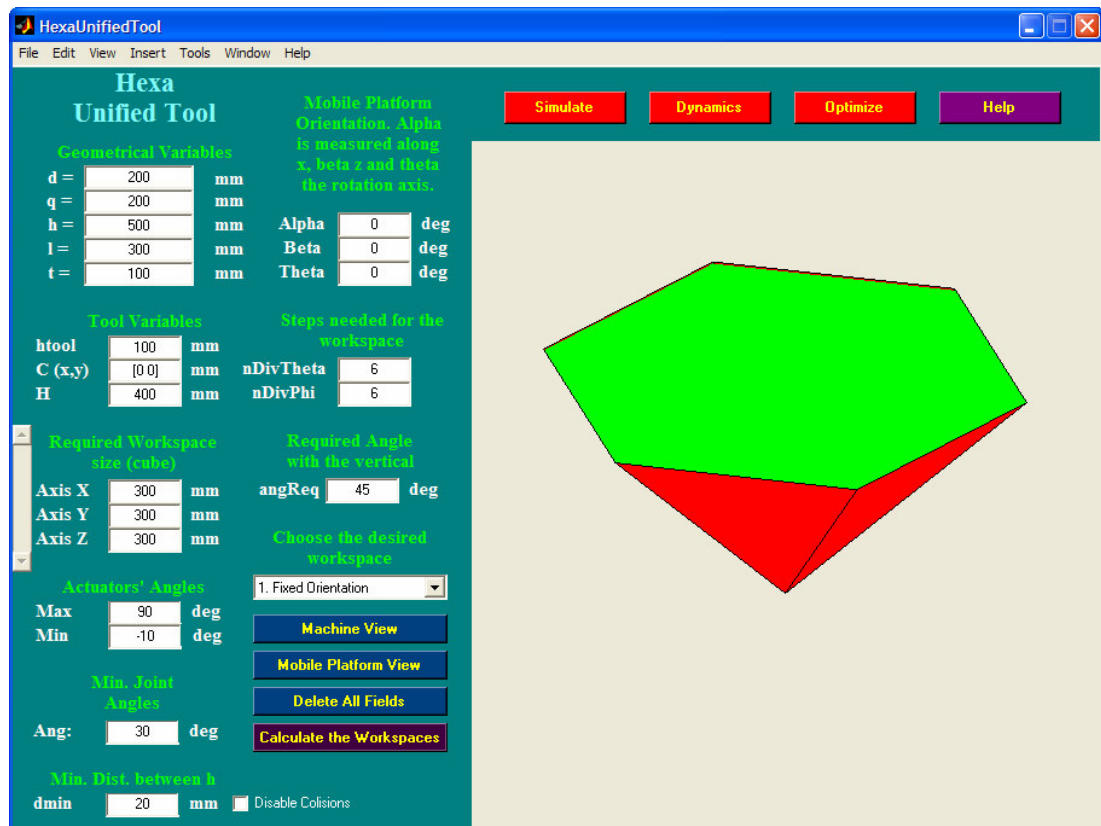
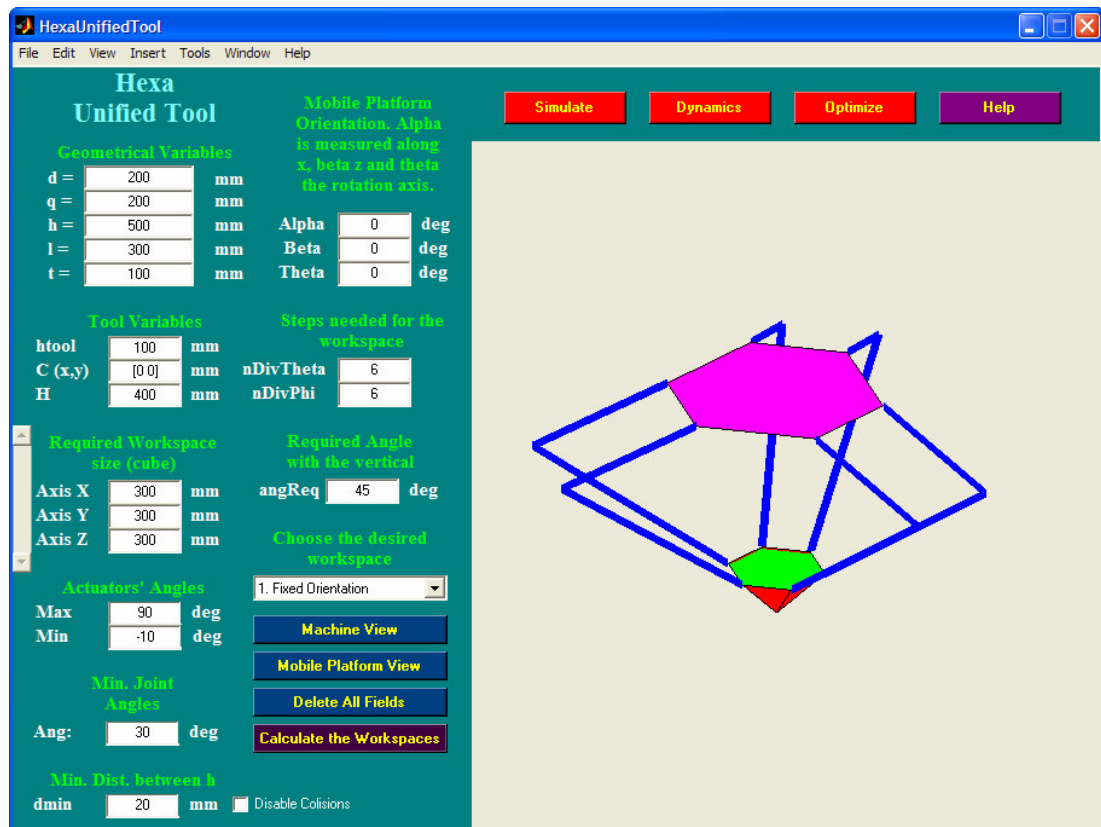


Figura 50: Modelo virtual da Hexa e a sua plataforma móvel em detalhes.

## 9.2. Cálculo do Volume de Trabalho

Neste item estão apresentados alguns exemplos de volumes de trabalho. A Fig. 51 ilustra os volumes de orientação fixa (região mais clara) e de orientação total (região mais escura), sendo este último contido inteiramente no primeiro. O volume de orientação fixa apresentado é para a plataforma sem rotações, ou seja, inteiramente na vertical (por isso a total simetria). Estes volumes estão apresentados individualmente na Fig. 52.

Os parâmetros do modelo para o volume de trabalho foram os mesmos descritos no item anterior. Como restrições, um ângulo mínimo de  $30^\circ$  entre os ligamentos atuados e passivos, e ângulos máximo e mínimo de atuação de  $90^\circ$  e  $-10^\circ$  foram utilizados. Os volumes de orientação total demandaram um ângulo de  $45^\circ$  com a vertical. Os passos utilizados para a definição dos vetores de busca em coordenadas polares foram de 40 e 30, respectivamente, para  $\psi$  e  $\varphi$  (vide Fig. 32 do Capítulo 5).

Na Fig. 53 têm-se os volumes de orientação fixa e de orientação total para a mesma máquina, contudo o ângulo da plataforma para o volume de orientação fixa é de  $45^\circ$  no eixo  $X$ . Desta forma observa-se que o volume não é mais simétrico a cada  $120^\circ$  e que existem pontos de contato com o volume de orientação total.

Num computador com processador Centrino® de 2 GHz e 2 GB de memória o cálculo do volume de trabalho de orientação fixa levou cerca de 30 segundos e o de orientação total, cerca de 7 minutos.

Não foi utilizado o algoritmo de colisões. Quando este algoritmo está ativo, a obtenção dos volumes de trabalho fica muito mais lenta, levando cerca de oito vezes mais tempo. Contudo, para a máquina em questão, com as limitações impostas, as colisões só afetam os volumes se a distância mínima aceita entre os ligamentos for maior que 50 mm. Se este valor for entendido como o diâmetro dos ligamentos, caso sejam cilindros, estes seriam ligamentos demasiadamente grossos para a máquina.

A Fig. 54 ilustra os volumes de trabalho com uma resolução bem menor para vários diâmetros de ligamento. Pode-se observar que apenas quando o diâmetro é maior que 50 mm têm-se um decréscimo no tamanho dos volumes. O volume de trabalho de orientação total é mais afetado e o de orientação fixa é mantido praticamente constante até um diâmetro próximo ao lado do hexágono (variável  $t$ ).

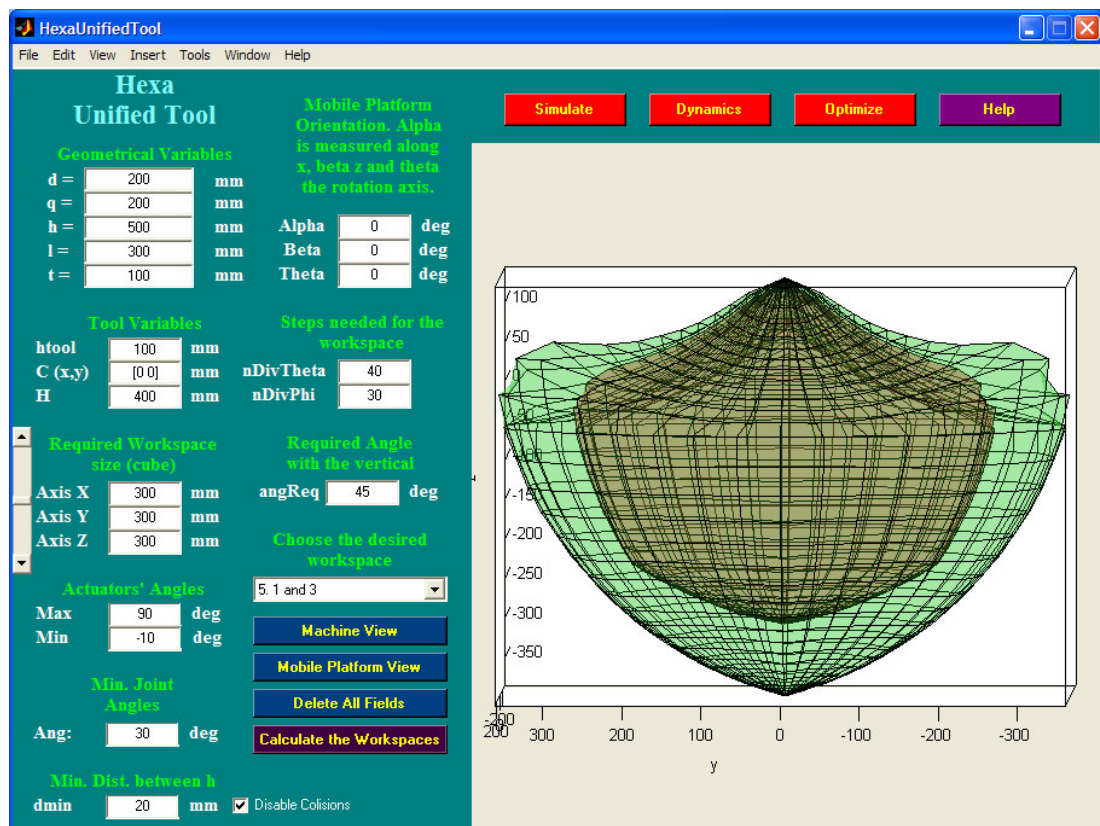
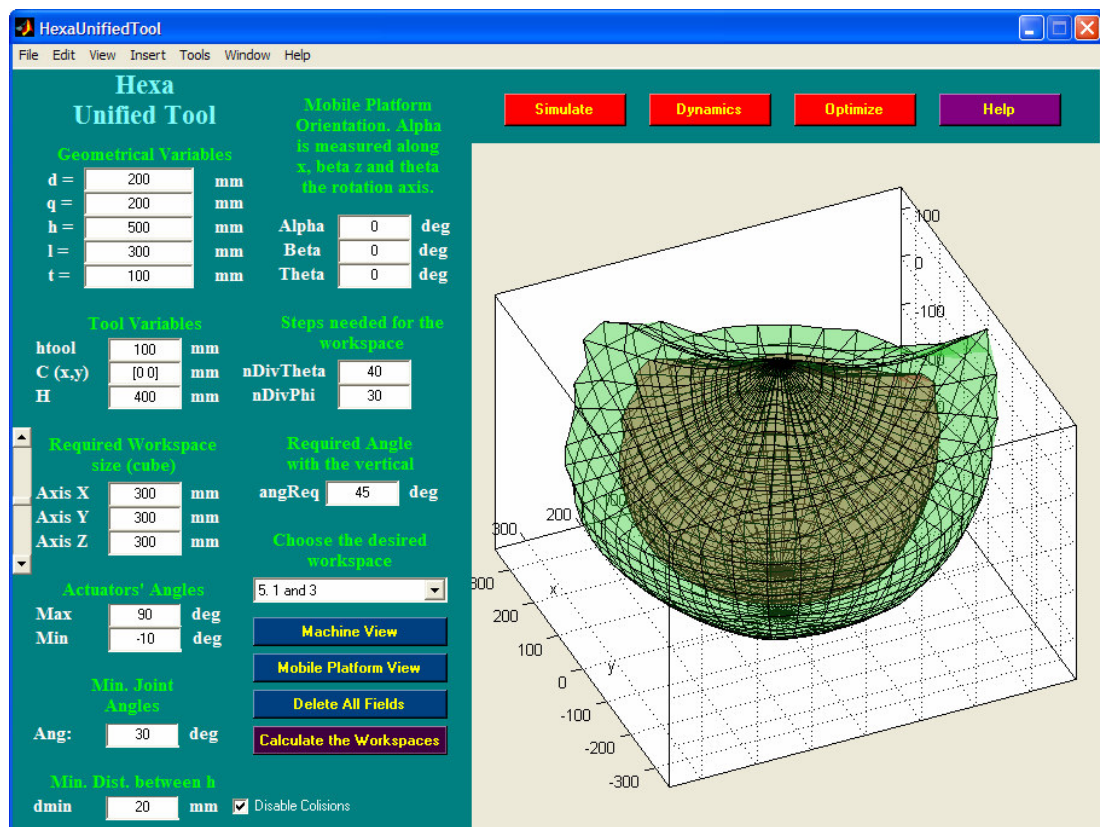


Figura 51: Volumes de trabalho de orientação fixa e orientação total da Hexa em duas vistas diferentes.



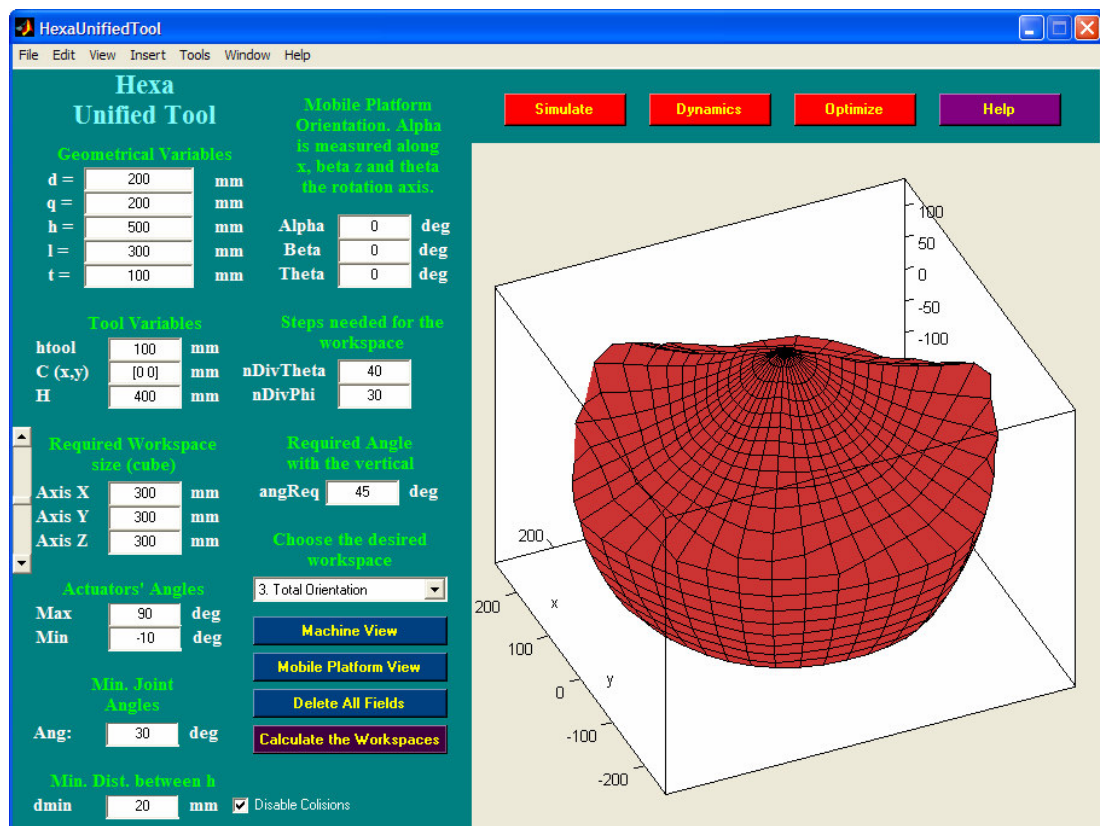
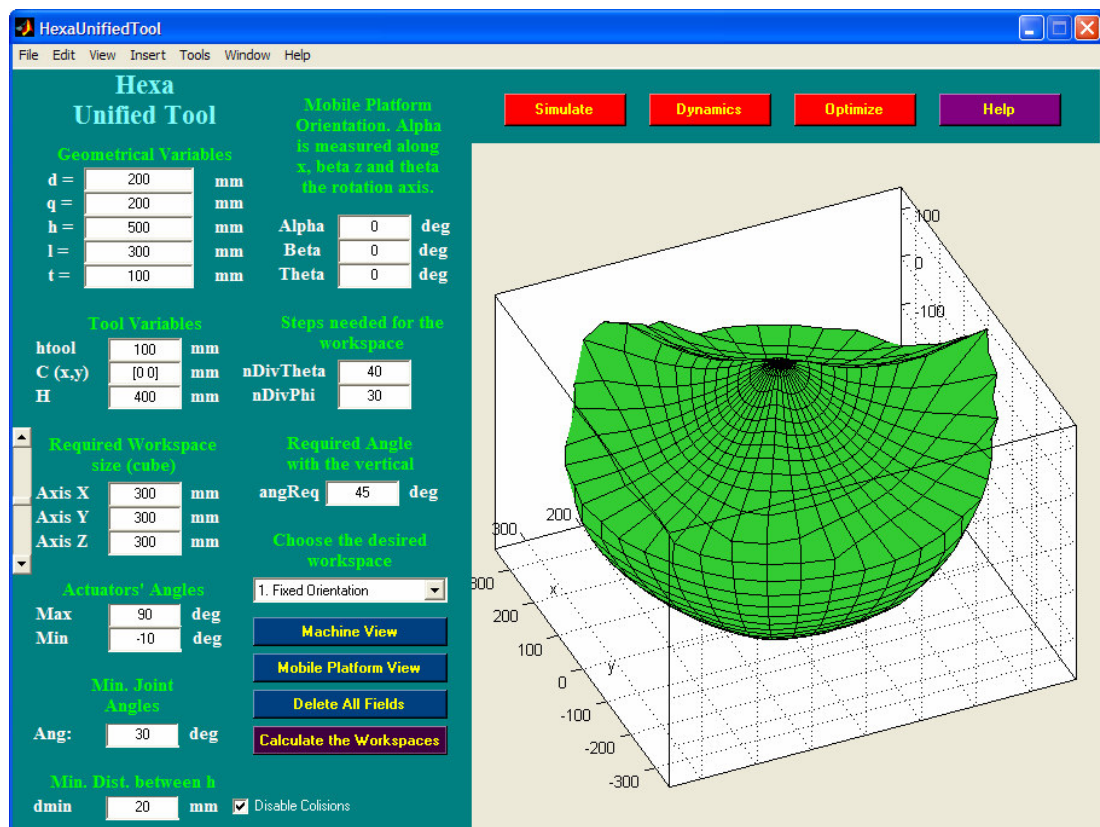
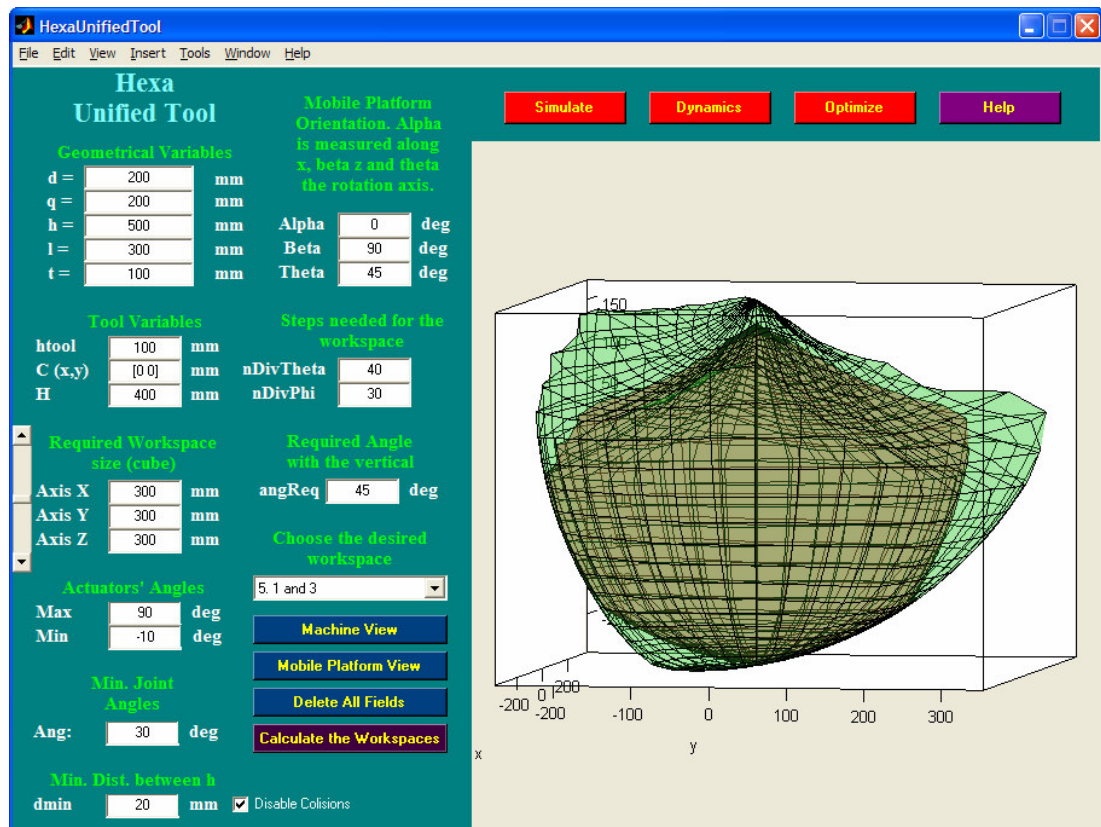


Figura 52: Acima o volume de trabalho de orientação fixa e abaixo o de orientação total.

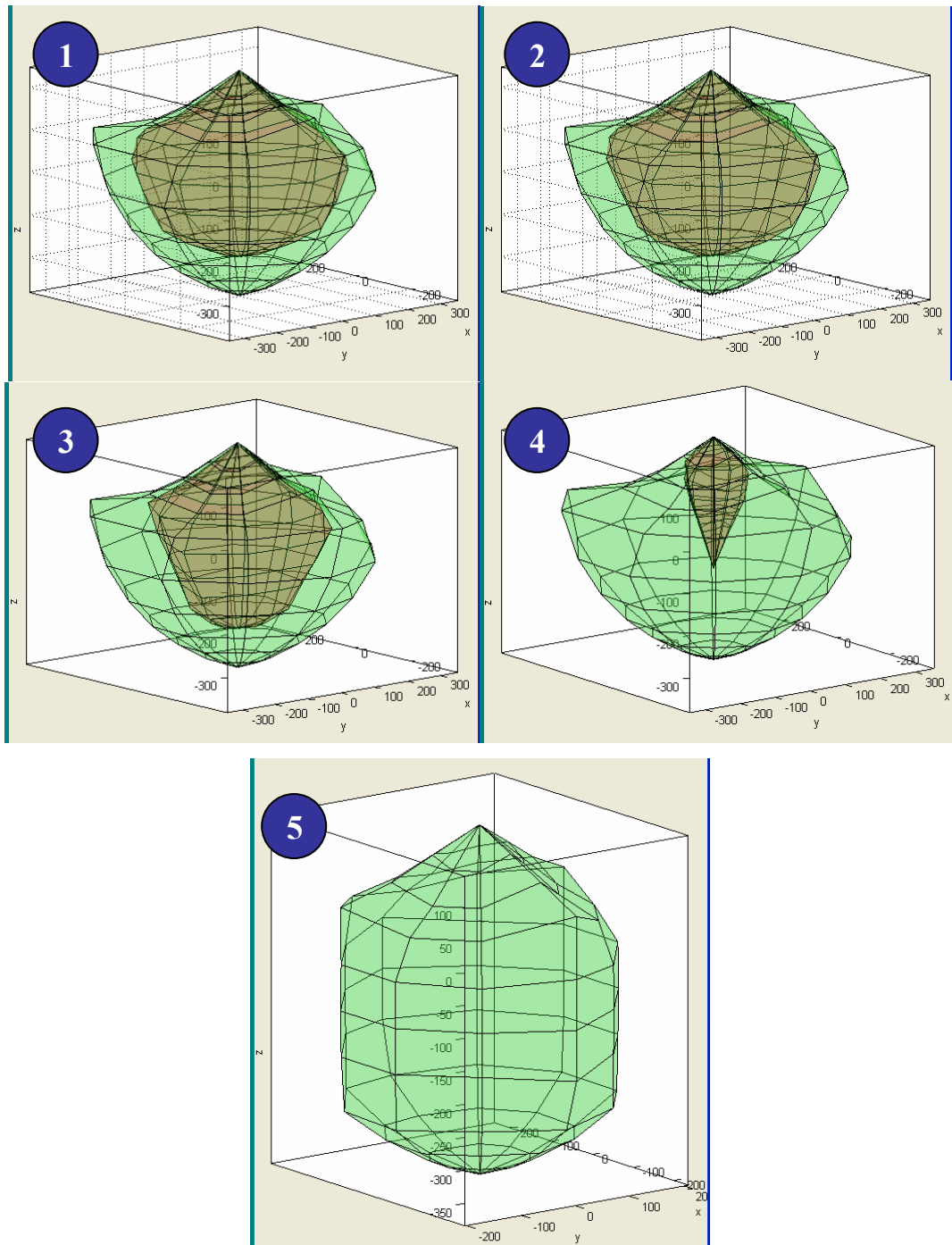


**Figura 53:** Volumes de trabalho da Hexa onde o de orientação fixa utilizou a plataforma com ângulo de 45° com a vertical (rotação no eixo X).

A Tabela 4 apresenta os valores dos volumes de trabalho das figuras 52 e 53. A Fig. 56 ilustra uma barra de progresso para um maior conforto do usuário e finalmente a Fig. 55 apresenta uma comparação do volume de orientação fixa deste trabalho (Fig. 52) e do IFM (2005). Embora não se tenham informações adicionais, percebe-se que possuem formas semelhantes.

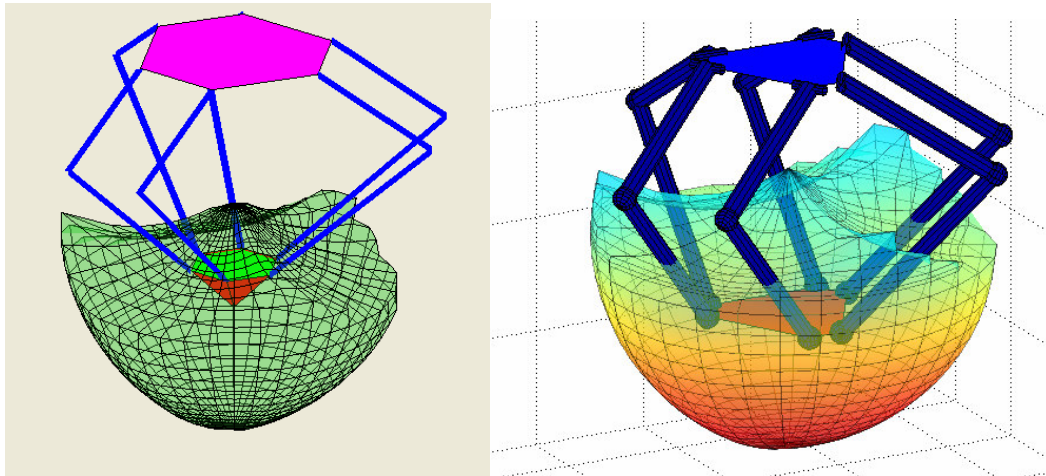
Número da Figura	Volume de Trabalho de Orientação Fixa	Volume de Trabalho de Orientação Total
52	97.98 dm <sup>3</sup>	49.95 dm <sup>3</sup>
53	81.35 dm <sup>3</sup>	49.95 dm <sup>3</sup>

**Tabela 4:** Valor dos volumes de trabalho da Hexa nos diferentes casos.

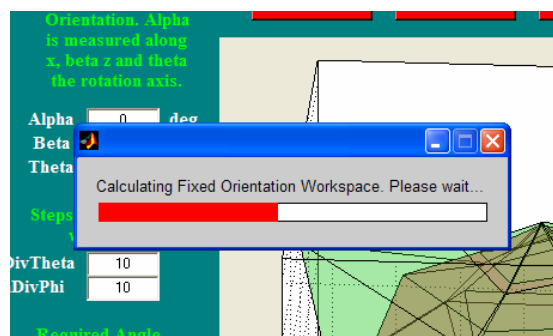


**Figura 54:** Volumes de trabalho de orientação fixa e total com algoritmo de detecção de colisões para diferentes diâmetros dos ligamentos passivos. (1) 20 mm; (2) 50 mm; (3) 60 mm; (4) 80 mm; (5) 95 mm.





**Figura 55:** Comparação de volumes de trabalho de orientação fixa da Hexa. À esquerda, o volume da máquina deste trabalho e à direita tem-se uma figura de IFM, 2005.



**Figura 56:** Barra de progresso para o cálculo dos volumes de trabalho, indicando o percentual já calculado (existe uma para cada volume: orientação fixa e total).

### 9.3. Simulador Virtual

O simulador é ativado quando o botão “Simulate” é pressionado. A Fig. 57 apresenta a janela do simulador, cujos componentes estão detalhados na Tabela 5.

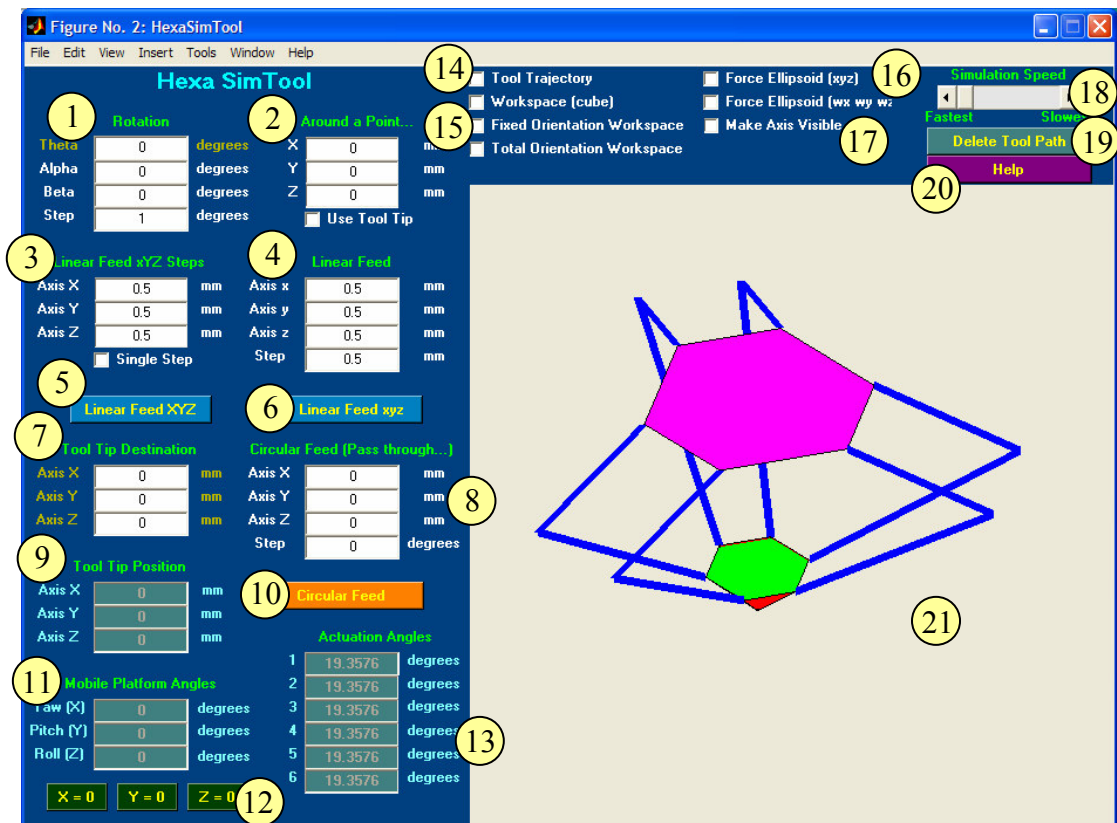


Figura 57: Janela do Simulador Virtual.

Nº	Tipo	Ação / Variável
1	Campos Texto	Conjunto de campos texto para rotações quaisquer.
2	Campos Texto	Centro para rotações. A check box define a ponta da ferramenta.
3	Campos Texto	Passos que são utilizados no avanço linear.
4	Campos Texto	Distância no qual o efetuador irá andar com relação ao sistema xyz.
5	Botão	Executa o avanço linear utilizando o sistema global.
6	Botão	Executa o avanço linear utilizando o sistema da ferramenta.
7	Campos Texto	Destino da ponta da ferramenta no sistema global.
8	Campos Texto	Campos para a interp. circular (passo e ponto da circunferência).
9	Campos Texto	Mostram a posição atual da ferramenta.
10	Botão	Executa a interpolação circular.
11	Campos Texto	Mostram os ângulos de Roll-Pitch-Yaw (RPY) da máquina.
12	Botões	Fazem com que o respectivo ângulo RPY seja zero.
13	Campos Texto	Mostram os ângulos dos atuadores.
14	Check Box	Quando ticado desenha o caminho percorrido pela ferramenta.
15	Check Boxes	Mostram / Escondem os volumes de trabalho.
16	Check Boxes	Mostram / Escondem os elipsóides de forças para translação ou rot.
17	Check Box	Mostra / Esconde os sistemas de eixos coordenados.
18	Slider	Acelera / Desacelera as simulações (compensa um PC rápido).
19	Botão	Apaga a trajetória desenhada com 14 ticado.
20	Botão	Botão de Ajuda.
21	Tela Gráfica	Local onde as simulações ocorrem.

Tabela 5: Componentes do simulador virtual.

A Fig. 58 mostra os volumes de trabalho anteriormente apresentados e a Hexa, além de um volume com formato de um cubo escolhido pelo usuário. As figuras 59 e 60 apresentam outras particularidades como trajetórias lineares e circulares, movimentação em diferentes sistemas de eixos coordenados e os sistemas da ferramenta e global.

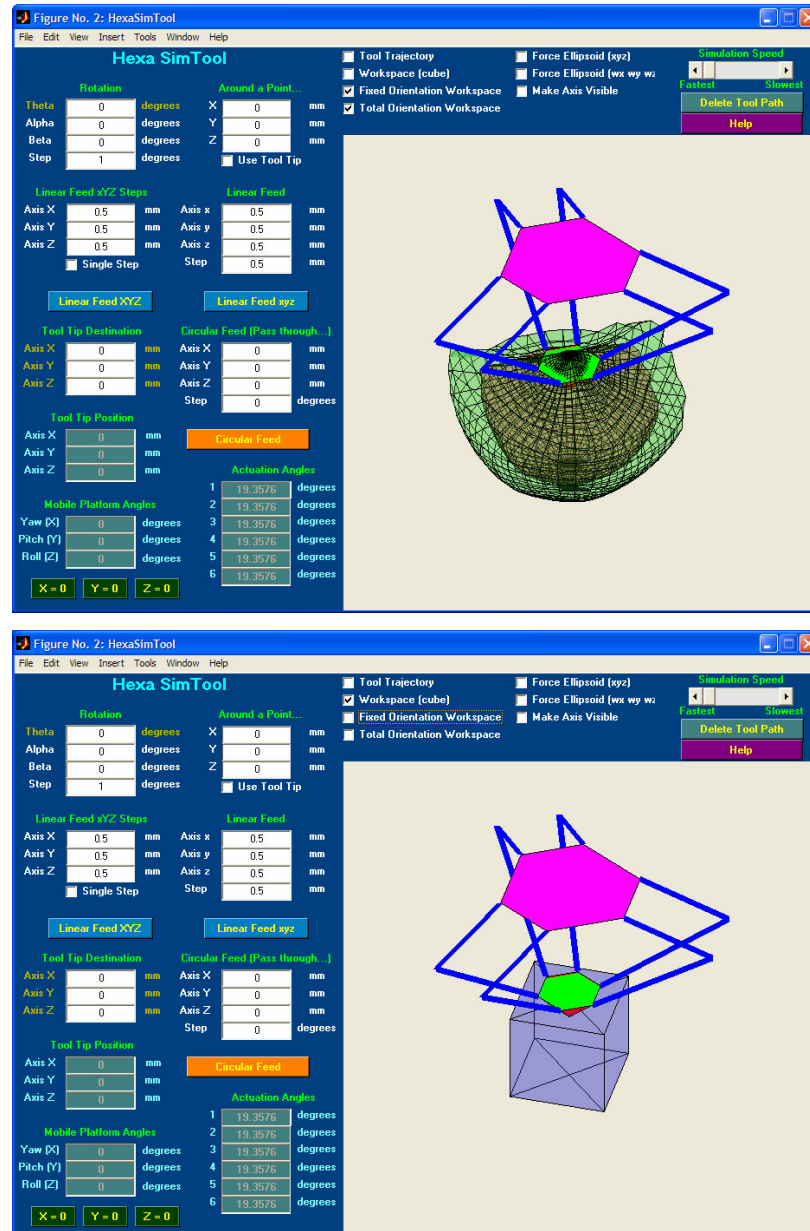


Figura 58: Volumes de trabalho de orientação fixa e total (acima) e exemplo de volume escolhido pelo usuário (abaixo).

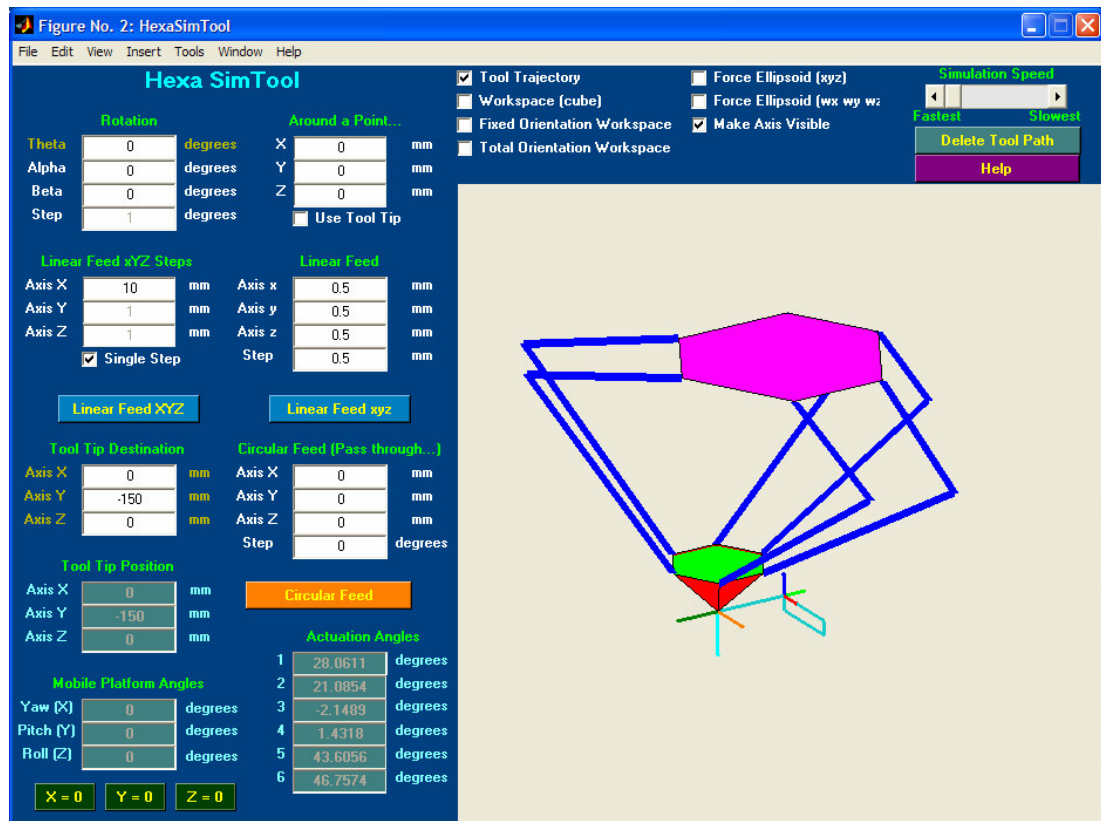
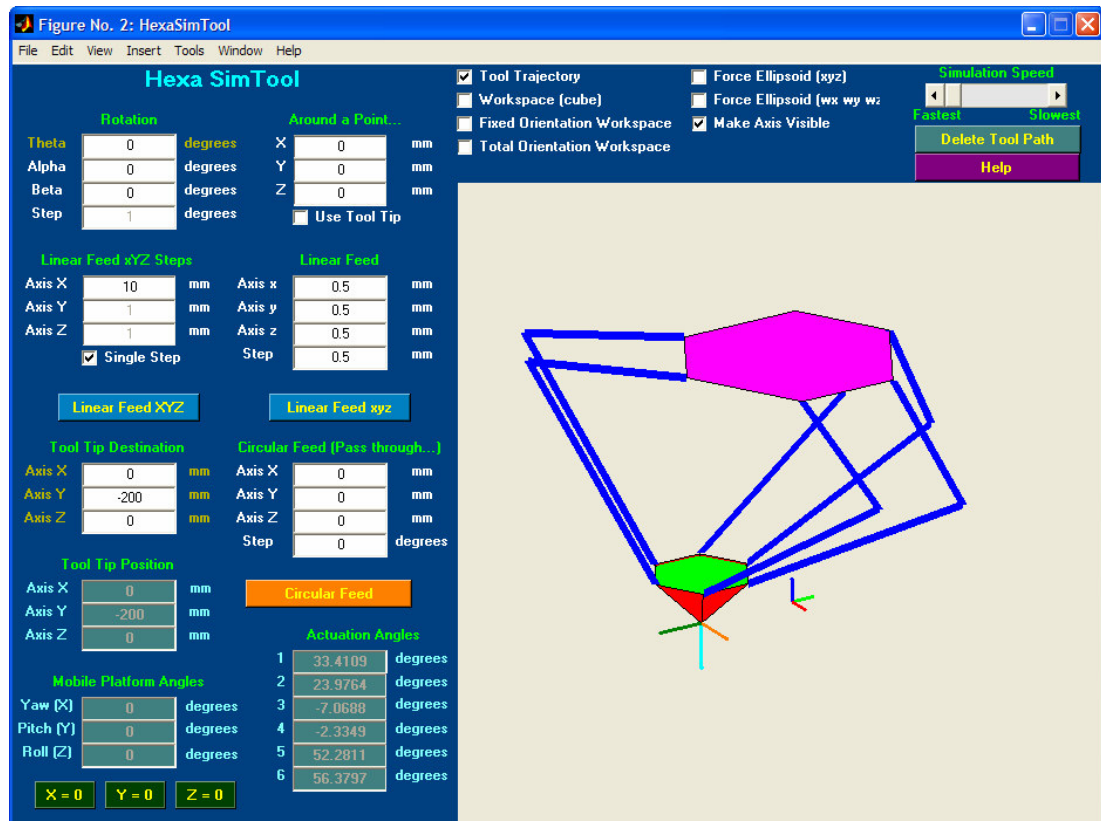


Figura 59: Sistemas de coordenadas (acima) e trajetórias lineares executadas (abaixo, linhas em azul claro).



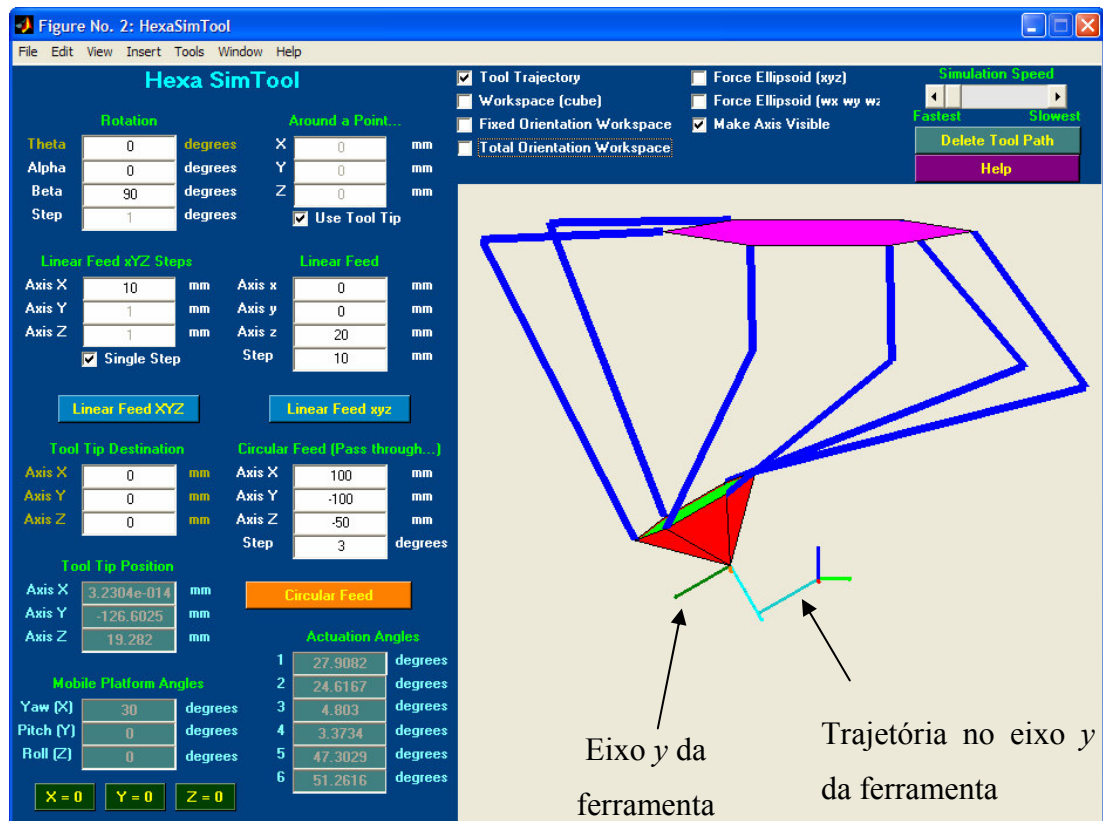
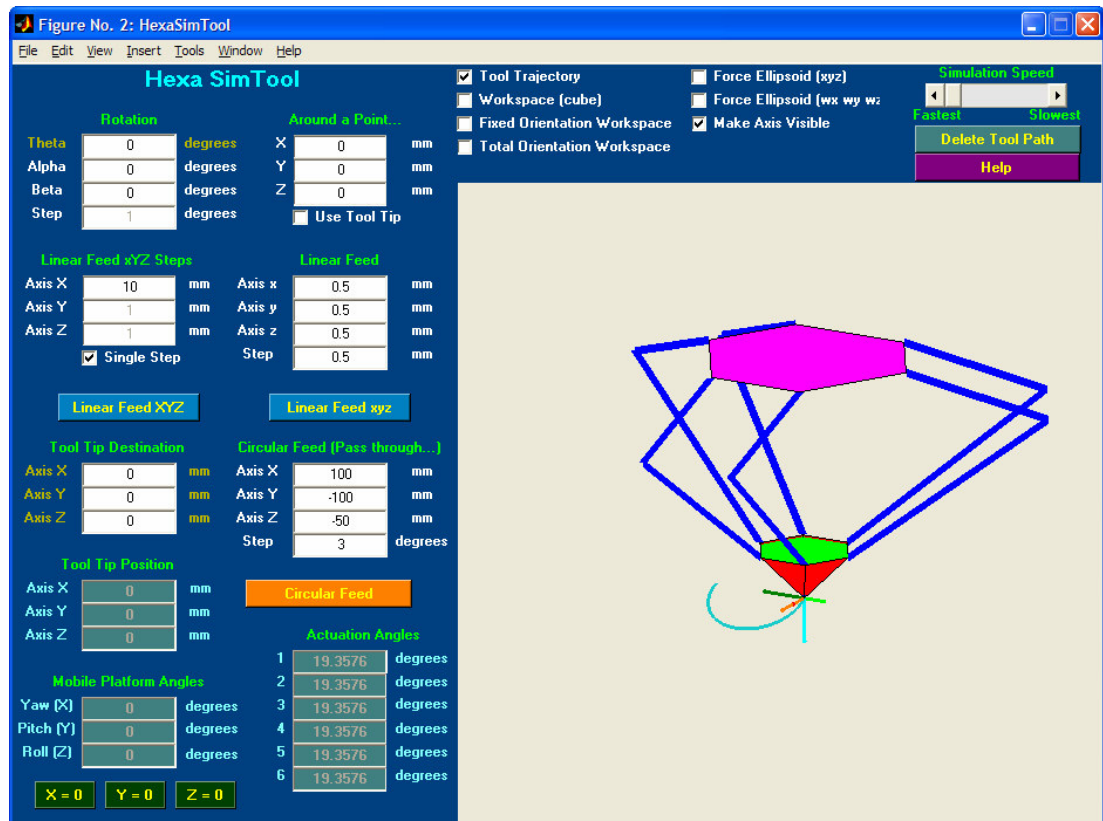


Figura 60: Trajetória circular num plano qualquer (acima) e movimentos utilizando o sistema da ferramenta.

## 9.4. Elipsóide de Forças

Também foram adicionadas ao simulador algumas funções para o cálculo do elipsóide de forças com torques iguais a zero e com forças iguais a zero (para que o elipsóide a ser mostrado tivesse apenas três dimensões).

A Fig. -89 apresenta o elipsóide de forças no ponto  $X, Y, Z = [0, 0, 0]$ , que é a posição nominal da máquina, para o caso onde os torques valem zero. Para ilustrar, também é apresentado o ponto onde  $X = -150$  mm. Pode-se atentar para as deformações no elipsóide que parte de uma total simetria em  $X$  e  $Y$  e vai gradativamente se deformando à proporção em que caminha para a borda de seu volume de trabalho.

A Fig. 62 apresenta os mesmos pontos, no entanto considera que todas as forças externas são nulas, apresentando assim um elipsóide de torques. Vale lembrar que não se deve atentar para o tamanho absoluto do elipsóide, já que o mesmo foi ampliado para possibilitar uma fácil observação. Devem ser observadas suas diferenças de forma e de tamanho quando a Hexa se movimenta de um ponto a outro.

A Fig. 63 apresenta mais dois pontos onde é observado o elipsóide de forças. Desde a posição nominal que o elipsóide vai diminuindo até que em  $X = -480$  mm ele já está bem pequeno (este ponto está fora do volume de trabalho definido anteriormente, violando a restrição de limite máximo do ângulo de atuação). Num ponto próximo,  $X = -485.21558$  mm, o elipsóide praticamente some, ficando apenas uma linha. Essa postura da plataforma está muito perto de uma singularidade.

Algo interessante ocorre quando se imprime uma leve movimentação para outro ponto próximo  $X = -485.21559$  mm (vide Fig. 64). Embora a matriz característica seja positiva semi-definida, um dos autovalores nesse ponto é negativo. Possivelmente esse fato seja devido a algum erro numérico fruto de graves problemas de mau condicionamento da matriz característica neste ponto crítico (o próprio Matlab avisa em seu *prompt* dessa possibilidade quando se chega ao ponto).

Em  $X = -485.21559$  mm tem-se ainda que os determinantes de  $\mathbf{Jq}$  e  $\mathbf{Jx}$  valem, respectivamente,  $2,49 \cdot 10^{-7}$  e  $1,19 \cdot 10^{-14}$ . As Tabelas 6 e 7 apresentam os autovalores e autovetores da matriz característica em cada ponto, além do tamanho dos eixos do elipsóide de seis dimensões. O Jacobiano quase não muda.

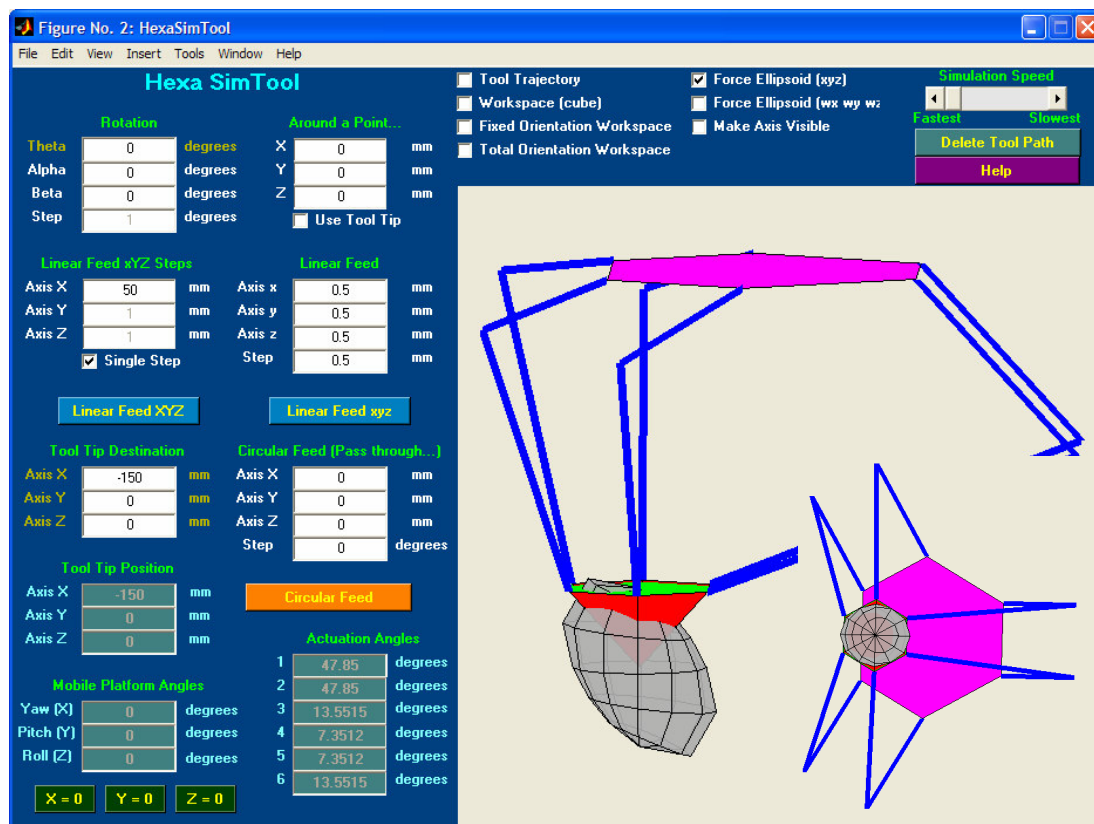
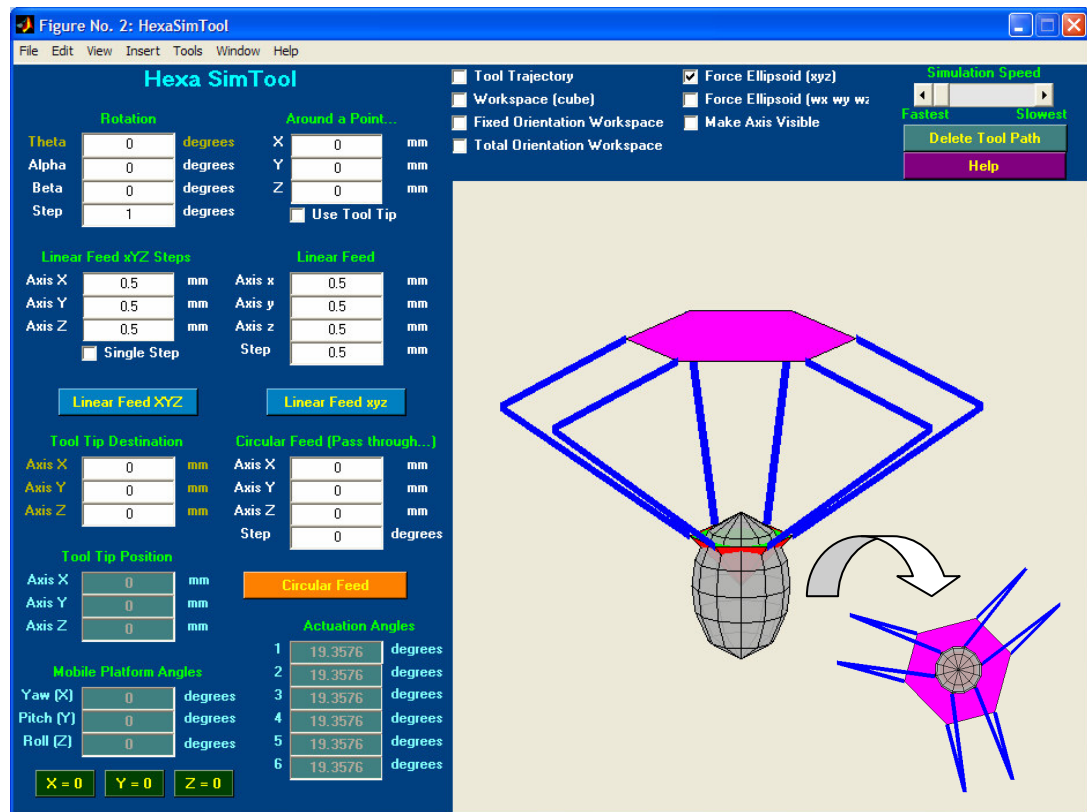


Figura 61: Elipsóide de *forças* para o caso em que os torques valem zero e a plataforma está em sua posição nominal (acima) e no ponto  $X = -150$  mm (abaixo).

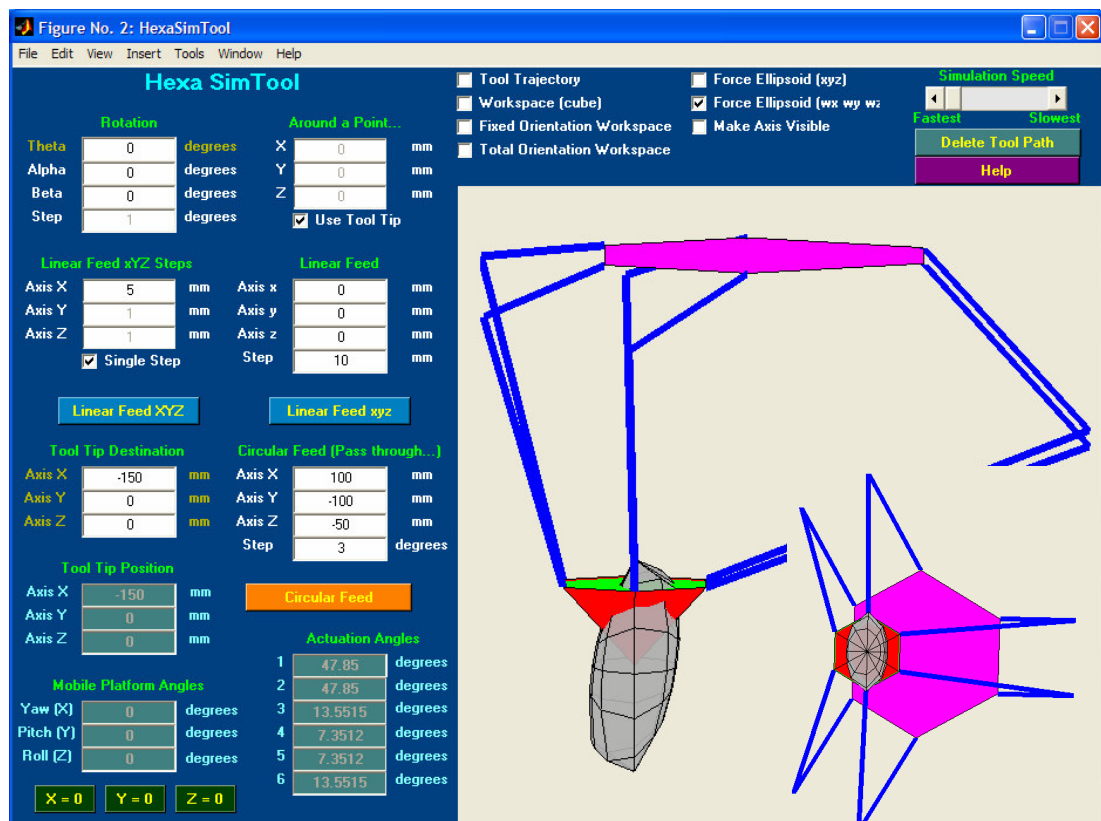
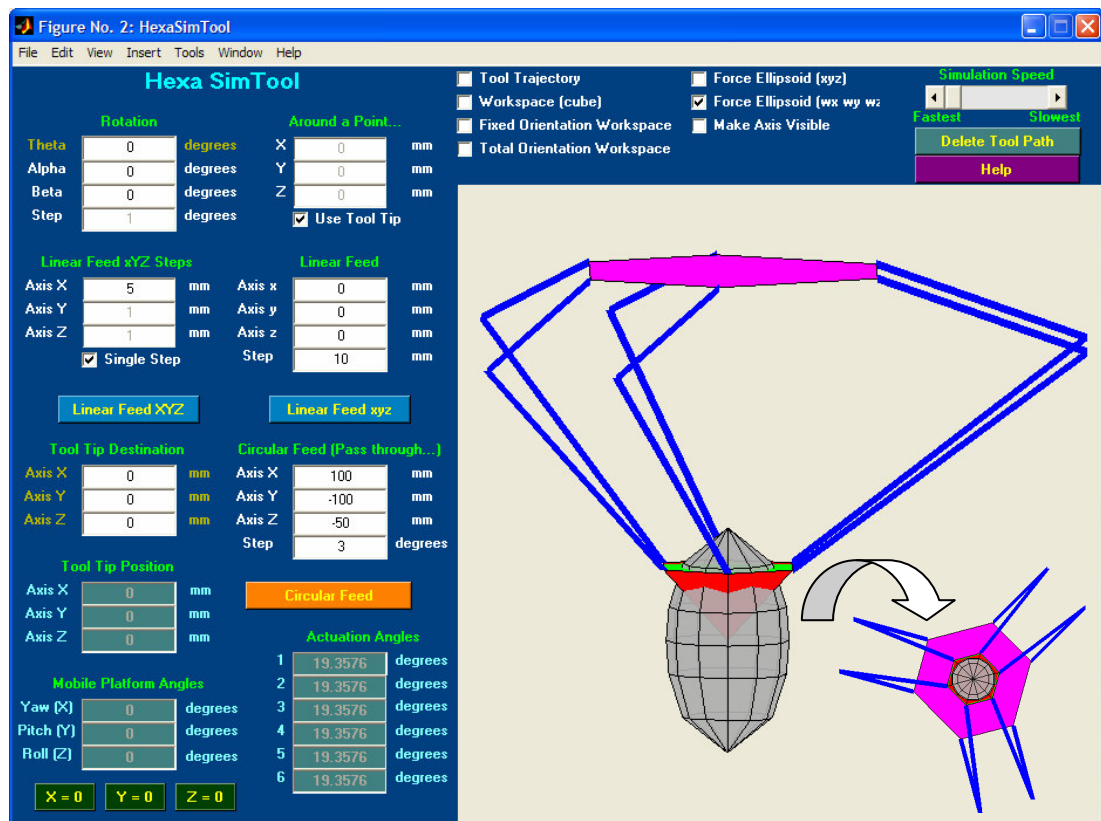
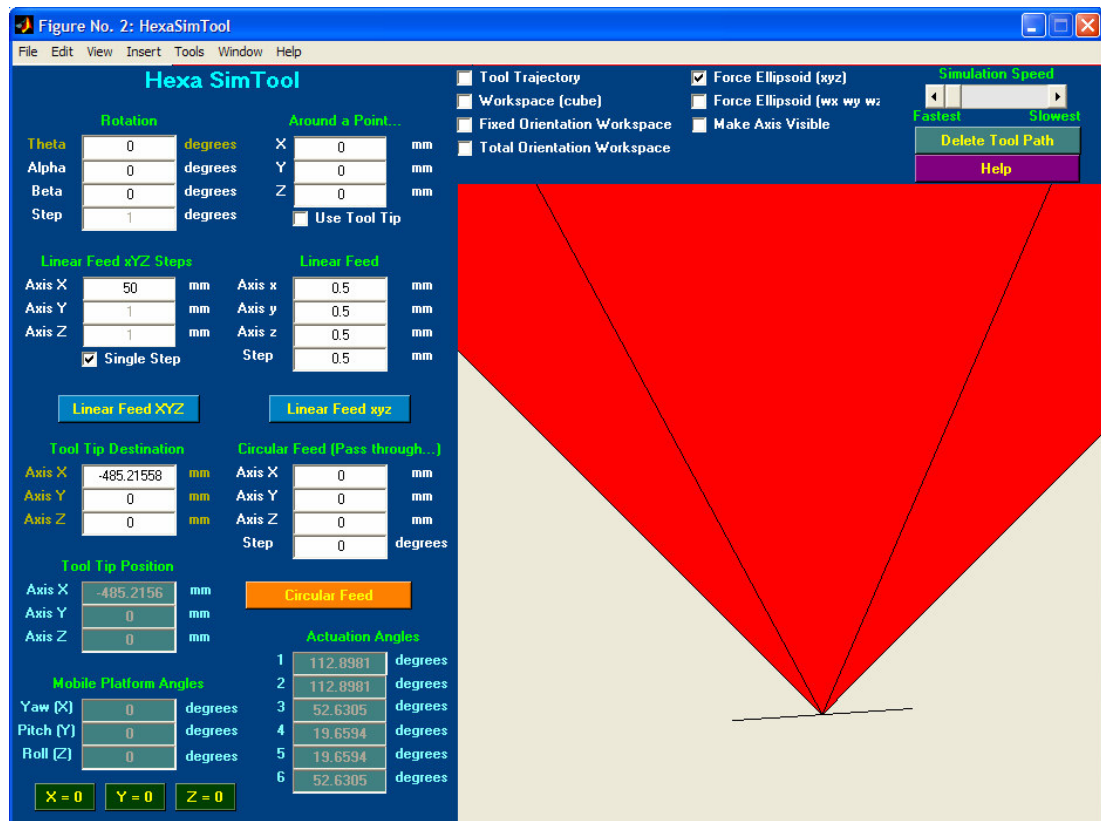
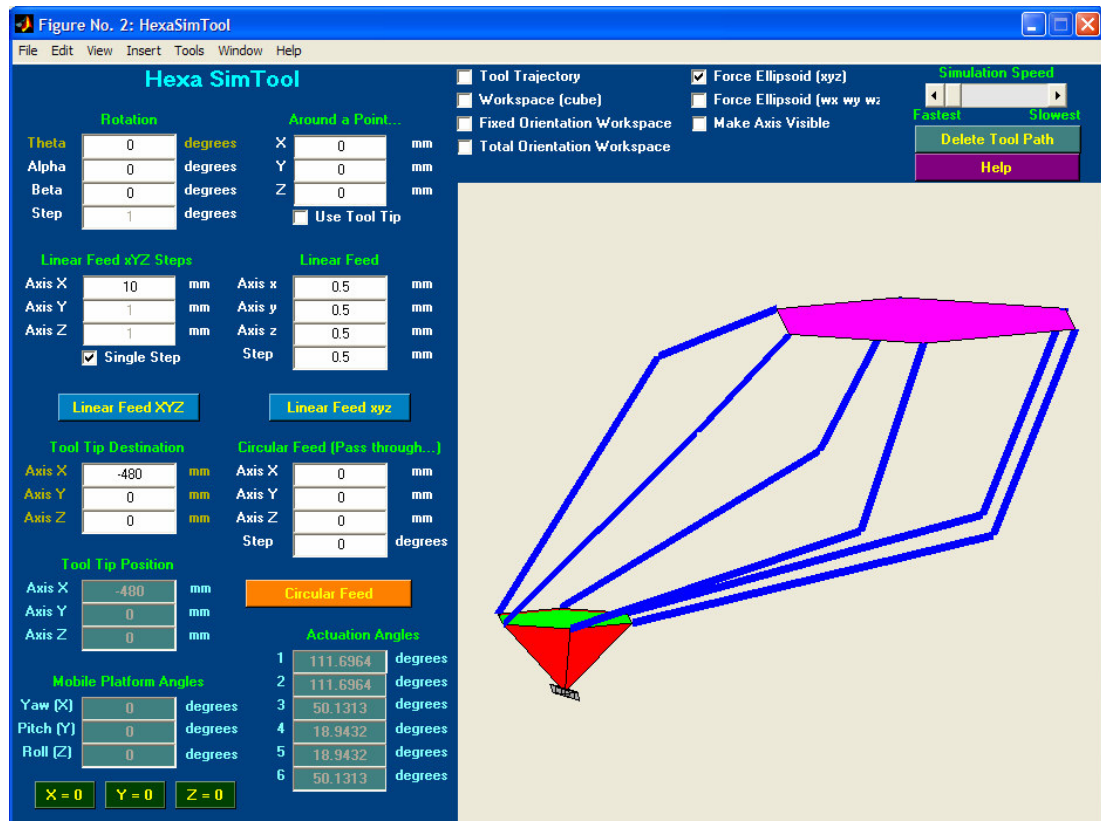


Figura 62: Elipsóide de *torques* para o caso em que os torques valem zero e a plataforma está em sua posição nominal (acima) e no ponto X = -150 mm (abaixo).





**Figura 63:** Elipsóide de forças para X = -480 mm (acima) e X = -485.21558 mm (abaixo).  
 Percebe-se o tamanho diminuto deste último (praticamente uma linha).

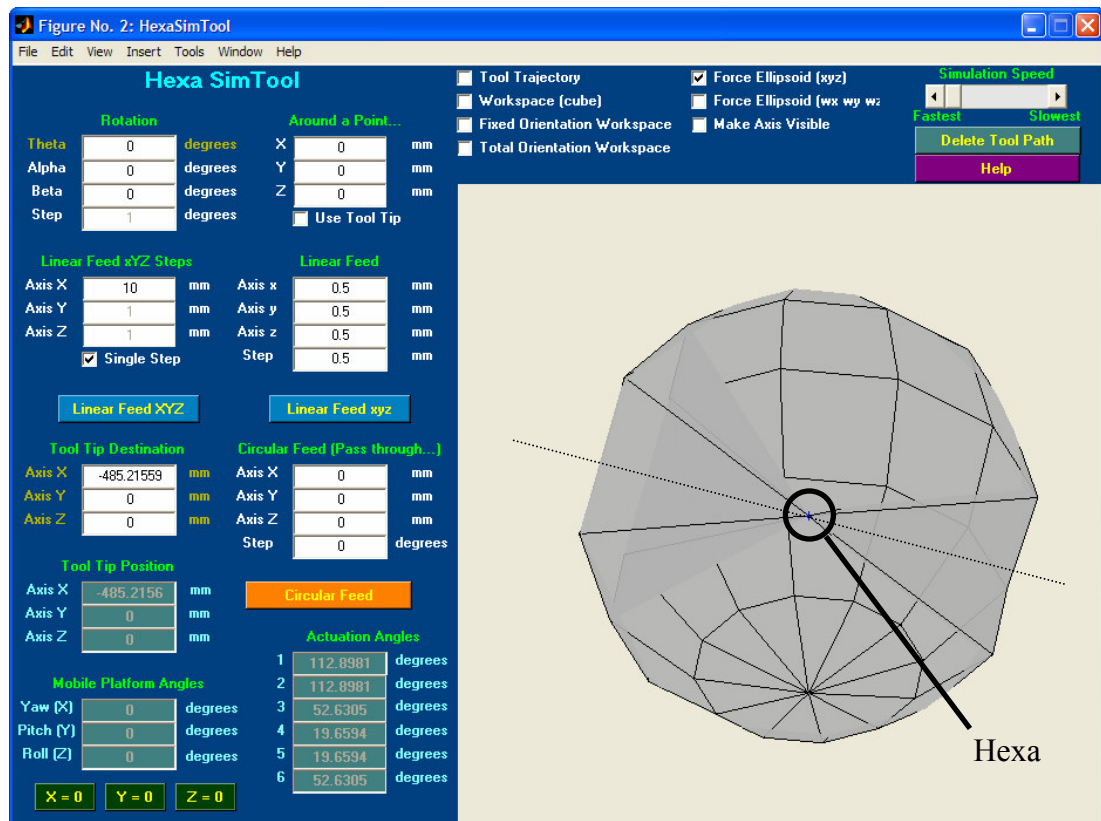
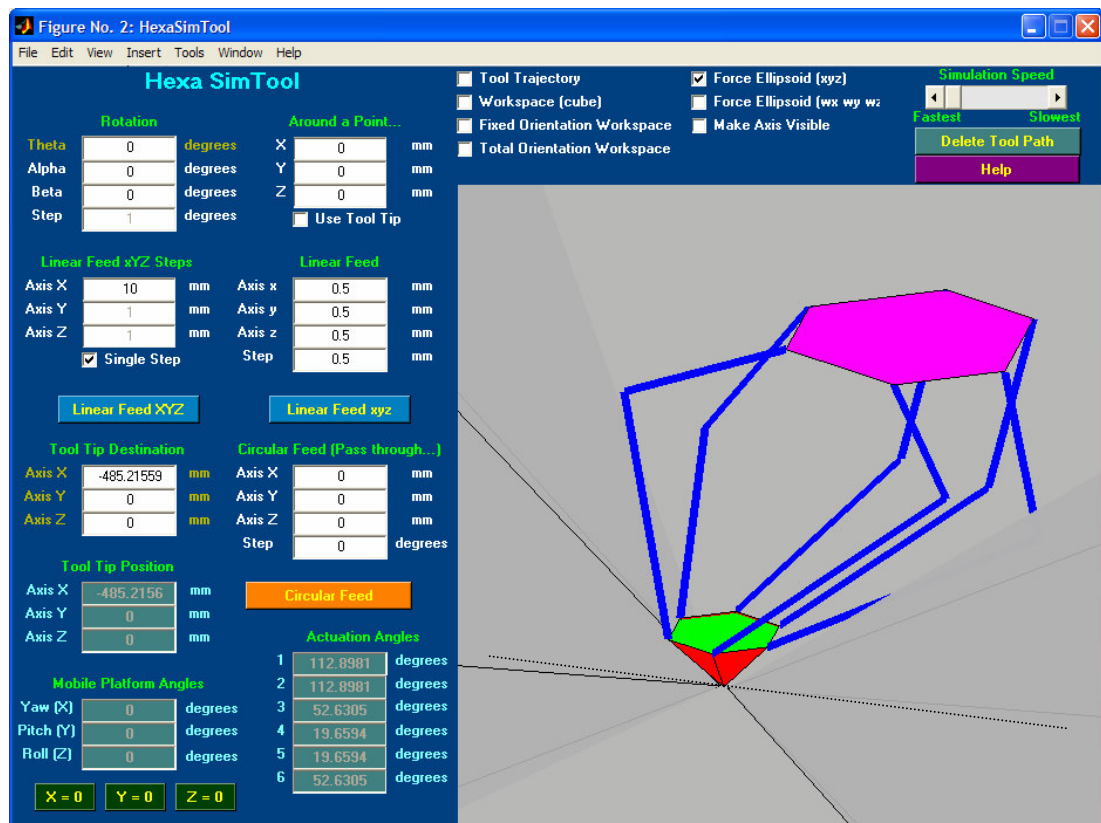


Figura 64: Elipsóide de forças no ponto  $X = -485.21559$ . Acima, o modelo ampliado e abaixo o elipsóide visto (um dos eixos dele é inexistente – destacado com linha pontilhada).

Jacobiano	1	2	3	4	5	6
1	-4.0583	-0.4211	-1.0413	-0.01	-0.3157	0.1664
2	-4.0583	0.4211	-1.0413	0.01	-0.3157	-0.1664
3	-9.0601	6.6836	-3.8446	-0.2839	-0.906	-0.906
4	-2.4405	3.1798	-2.9918	-0.1684	-0.5031	-0.3974
5	-2.4405	-3.1798	-2.9918	0.1684	-0.5031	0.3974
6	-9.0601	-6.6836	-3.8446	0.2839	-0.906	0.906

Matriz Característica	Autovetores (cada coluna é um autovetor)					
	1	2	3	4	5	6
X	-0.9991	0.0434	0	0	-0.4273	0.1472
Y	0	0	0.083	0.1129	0	-0.9793
Z	0.0057	0.1312	0	0	0.8985	0.0012
X (Ang)	0	0	0.9485	-0.3138	0	0.0433
Y (Ang)	-0.043	-0.9904	0	0	0.1003	0.0066
Z (Ang)	0	0	0.3058	0.9427	0	0.1317
<b>Autovalores</b>	<b>0</b>	<b>7.20E+14</b>	<b>~0</b>	<b>~0</b>	<b>~0</b>	<b>~0</b>
<b>Eixos</b>	<b>Inf</b>	<b>0</b>	<b>0.0287</b>	<b>0.1682</b>	<b>2.9805</b>	<b>10.5886</b>

**Tabela 6:** Jacobiano avaliado no ponto  $X = -485.21558$ ,  $Y = 0$  e  $Z = 0$ ; autovetores e autovalores da matriz característica do elipsóide de forças no ponto e o comprimento dos eixos do elipsóide (baseado nos autovalores). Ang = Angular, Inf = Infinito.

Jacobiano	1	2	3	4	5	6
1	-4.0583	-0.4211	-1.0413	-0.01	-0.3157	0.1664
2	-4.0583	0.4211	-1.0413	0.01	-0.3157	-0.1664
3	-9.0601	6.6836	-3.8446	-0.2839	-0.906	-0.906
4	-2.4405	3.1798	-2.9918	-0.1684	-0.5031	-0.3974
5	-2.4405	-3.1798	-2.9918	0.1684	-0.5031	0.3974
6	-9.0601	-6.6836	-3.8446	0.2839	-0.906	0.906

Matriz Característica	Autovetores (cada coluna é um autovetor)					
	1	2	3	4	5	6
X	0.9991	-0.0434	0	0.0001	-0.9503	-0.4562
Y	0	0	-0.083	0.1129	-0.2942	0
Z	-0.0057	-0.1312	0	-0.0001	-0.0772	0.8846
X (Ang)	0	0	-0.9485	-0.3138	0.013	0
Y (Ang)	0.043	0.9904	0	0	-0.0519	0.0972
Z (Ang)	0	0	-0.3058	0.9427	0.0396	0
<b>Autovalores</b>	<b>0</b>	<b>-1.22E+16</b>	<b>~0</b>	<b>~0</b>	<b>~0</b>	<b>~0</b>
<b>Eixos</b>	<b>Inf</b>	<b>i</b>	<b>0.0287</b>	<b>0.1682</b>	<b>10.5886</b>	<b>2.9435</b>

**Tabela 7:** Jacobiano avaliado no ponto  $X = -485.21559$ ,  $Y = 0$  e  $Z = 0$ ; autovetores e autovalores da matriz característica do elipsóide de forças no ponto e o comprimento dos eixos do elipsóide (baseado nos autovalores). Ang = Angular, Inf = Infinito, i = Imaginário.

A singularidade apresentada pode ser uma singularidade combinada de ambas as cinemáticas. Trata-se apenas de uma hipótese, pois também pode estar ocorrendo um caso onde o ponto crítico está bem próximo de dois outros pontos de singularidade, sendo cada um relativo a um tipo diferente de singularidade.

Infelizmente não se podem fazer maiores análises visuais deste caso, principalmente porque, devido aos seis graus de liberdade, as imagens são imprecisas quanto ao que realmente está acontecendo com o elipsóide real de seis dimensões. A busca pelo ponto de singularidade foi manual e não foram feitos algoritmos automáticos de detecção de pontos desse tipo.

Quando a plataforma passa desse ponto de singularidade, o elipsóide volta a crescer. A Fig. 65 ilustra o ponto  $X = -486$  mm, cujo elipsóide já é maior.

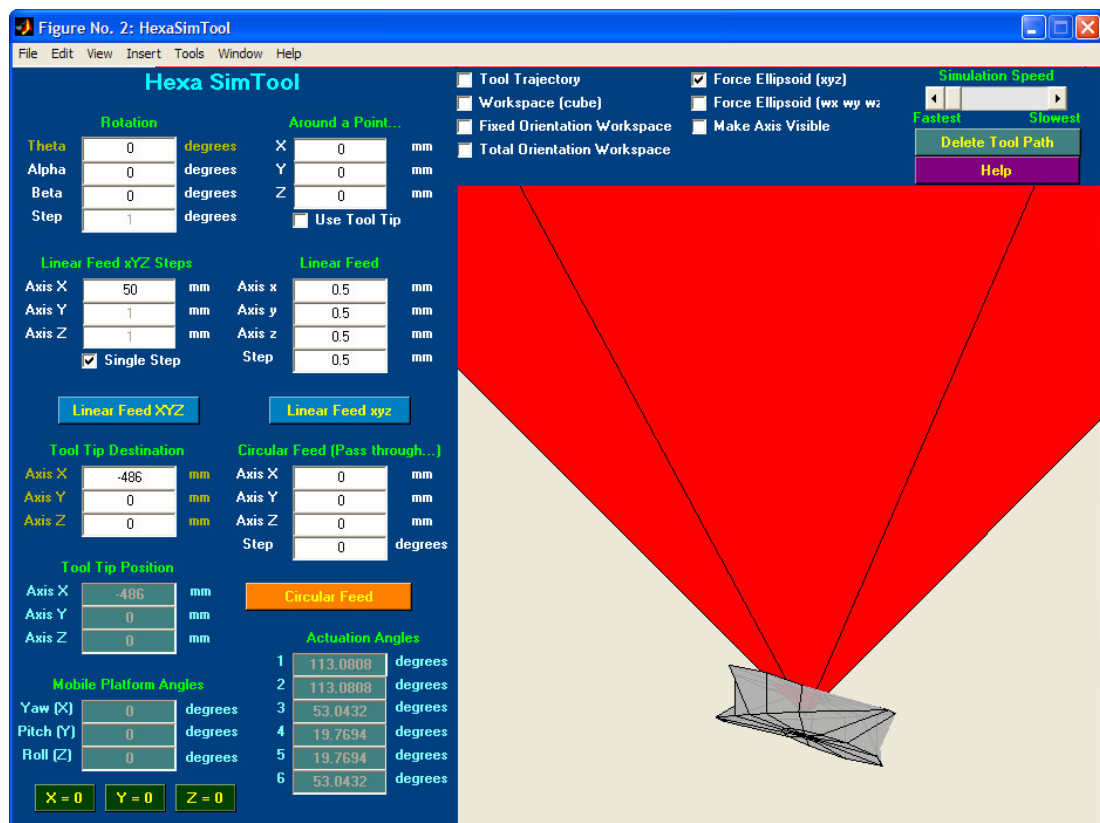


Figura 65: Elipsóide de forças em  $X = -486$  mm.

## 9.5. Cinemática Direta: Análise de Posições

Esta ferramenta calcula a cinemática direta da Hexa e compara os resultados com o modelo da cinemática inversa, com o único propósito de demonstrar o método iterativo que utiliza a matriz Jacobiano. Também não foi uma preocupação o cálculo das velocidades e acelerações, que poderão ser calculadas por derivação numérica. Para os pontos que faltarem como o centro das juntas que unem os ligamentos passivos e ativos, podem ser utilizadas as equações da cinemática inversa, já que os dados da plataforma móvel já serão obtidos por este método.

A ferramenta não possui interface gráfica, apenas textual no ambiente do Matlab. Utilizam-se os mesmos parâmetros do modelo da Hexa e, como exemplo, tem-se o caso onde a ponta da ferramenta parte da origem em direção ao ponto  $X,Y,Z = [10 \ 10 \ 10]$ . As entradas do algoritmo são:

Qual o ponto de partida? [x y z]: [0 0 0]

Qual o ponto de destino? [x y z]: [10 10 10]

Qual o numero de iteracoes? [n]: 100

E os resultados da simulação são:

**RESULTADO 1:** Com o Jacobiano Inalterado ao longo da movimentação

O ponto final estimado pelas iterações foi o ponto (os três últimos são a orientação da ferramenta): [10,2697 10,2172 9,8138 0,0004 0,0002 0,0000]

O ponto final real era o ponto: [10 10 10]

Distancia entre os dois pontos: 0,39 mm

**RESULTADO 2:** Com o Jacobiano Alterado ao longo da movimentação

O ponto final estimado pelas iterações foi o ponto (os três últimos são a orientação da ferramenta): [10,0027 10,0022 9,9983 0,0000 0,0000 0,0000]

O ponto final real era o ponto: [10 10 10]

Distancia entre os dois pontos: 0,0039 mm

Obviamente, com as correções contínuas do Jacobiano (aproximadamente uma a cada 0,17 mm), o resultado fica bastante preciso se comparado ao caso onde não se corrige o Jacobiano. Deve-se, no entanto, ter cuidado com o custo computacional do método, que, embora ainda muitas vezes inferior à solução através do sistema não linear de equações geométricas acopladas, ainda é alto. Pode-se, entretanto, utilizar uma relação custo benefício, ou seja, corrigir a matriz Jacobiano apenas depois de algumas iterações.

Quando se aumenta o número de passos, tem-se uma diminuição do erro, contudo até certo ponto. Embora não se tenham feito estudos neste assunto, sabe-se que caso seja utilizado um número muito grande de passos o erro numérico começa a ser maior que o erro da aproximação e então o algoritmo irá divergir.

A seguir, segue outro exemplo onde a plataforma movimenta-se da origem para o ponto  $X, Y, Z = [200 \ 150 \ -150]$ , que também pertence ao volume de trabalho. Pela distância percorrida ocorre uma atualização a cada 0,16 mm de movimento da plataforma móvel se o número de iterações for de 1800.

Qual o ponto de partida?  $[x \ y \ z] : [0 \ 0 \ 0]$

Qual o ponto de destino?  $[x \ y \ z] : [200 \ 150 \ -150]$

Qual o numero de iteracoes?  $[n] : 1800$

E os resultados da simulação são:

**RESULTADO 1:** Com o Jacobiano Inalterado ao longo da movimentação

O ponto final estimado pelas iterações foi o ponto (os três últimos são a orientação da ferramenta):  $[152,2450 \ 95,5045 \ -216,2331 \ 0,2164 \ -0,1916 \ 0,024310]$

O ponto final real era o ponto:  $[200 \ 150 \ -150]$

Distancia entre os dois pontos: 98 mm

**RESULTADO 2:** Com o Jacobiano Alterado ao longo da movimentação

O ponto final estimado pelas iterações foi o ponto (os três últimos são a orientação da ferramenta):  $[199,9882 \ 149,9597 \ -150,0728 \ 0,0002 \ -0,0002 \ 0,0001]$

O ponto final real era o ponto:  $[200 \ 150 \ -150]$

Distancia entre os dois pontos: 0,084 mm



Se o número de iterações for 4 vezes maior sendo igual a 7200, ocorre uma atualização do Jacobiano a cada 0,04 mm de movimento. O resultado com as correções também fica quatro vezes melhor (a relação é linear):

**RESULTADO 2:** Com o Jacobiano Alterado ao longo da movimentação

O ponto final estimado pelas iterações foi o ponto (os três últimos são a orientação da ferramenta): [199,9970 149,9899 -150,0182 0,0000 -0,0001 0,0000]

O ponto final real era o ponto: [200 150 -150]

Distancia entre os dois pontos: 0,021 mm

Os cálculos no Matlab do último exemplo de 7200 iterações, incluindo o preparo da matriz de posições dos atuadores a ser seguida pelo algoritmo, levaram menos de 20 segundos num computador Centrino® 2 GHz com 2 GB de RAM. O tempo poderia ser significativamente reduzido com a utilização de computadores dedicados e linguagens compiladas como C/C++. A Fig. 66 ilustra a posição do ponto [200 150 -150] no volume de trabalho.

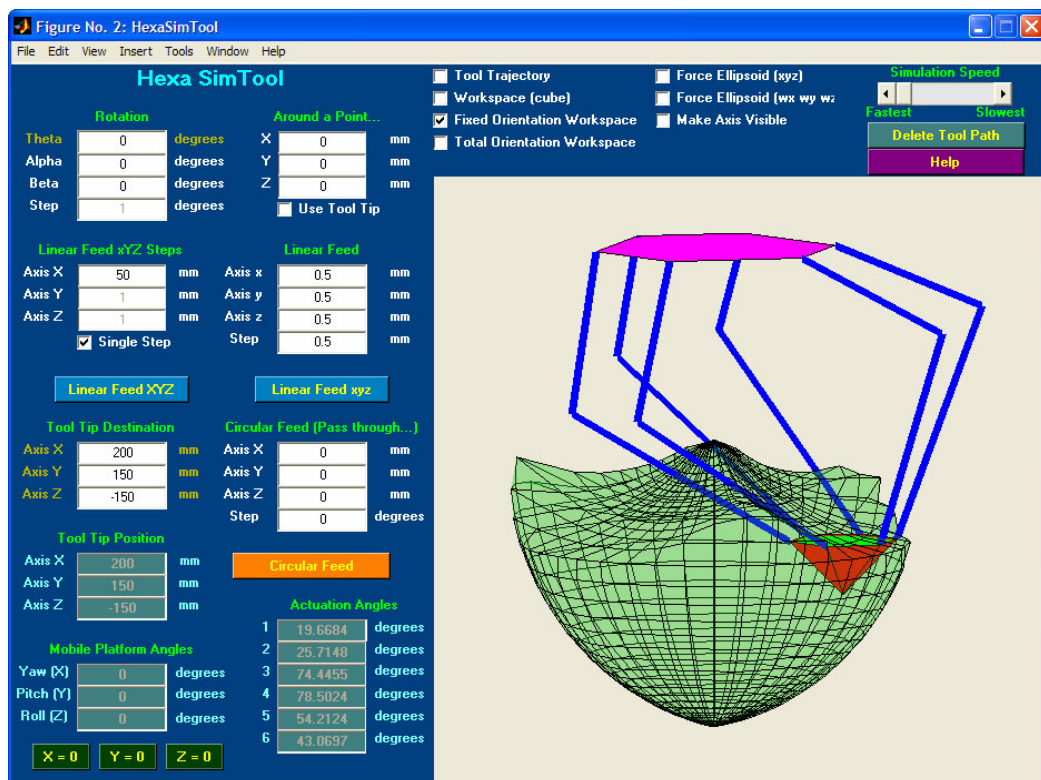


Figura 66: Ponto  $X, Y, Z = [250 \ 150 \ -150]$  no volume de trabalho de orientação fixa.

## 9.6. Análise Dinâmica

Seja um sistema de coordenadas solidário a cada ligamento atuado ou passivo. A matriz de inércia para um ligamento de comprimento  $L$ , de raio  $r$  e de massa  $m$  é dada por (eixo  $x$  é coincidente com o ligamento e seu centro está no baricentro):

$$\mathbf{I}_{\text{ligamentos}} = \begin{bmatrix} \frac{1}{2}m \cdot r^2 & 0 & 0 \\ 0 & \frac{1}{12}m \cdot (3r^2 + L^2) & 0 \\ 0 & 0 & \frac{1}{12}m \cdot (3r^2 + L^2) \end{bmatrix} \quad (144)$$

No caso da plataforma móvel, ela pode ser aproximada por um cone de raio  $t$  (lado do hexágono) e altura  $h_{\text{tool}}$ . Se o centro do sistema solidário à plataforma for coincidente com o baricentro, tem-se que a matriz de inércia da plataforma é dada por (eixo  $z$  passa ao longo do cone):

$$\mathbf{I}_{\text{plataforma}} = \begin{bmatrix} \frac{3}{20}m \cdot \left( t^2 + \frac{1}{4}h_{\text{tool}}^2 \right) & 0 & 0 \\ 0 & \frac{3}{20}m \cdot \left( t^2 + \frac{1}{4}h_{\text{tool}}^2 \right) & 0 \\ 0 & 0 & \frac{3}{10}m \cdot t^2 \end{bmatrix} \quad (145)$$

Utilizam-se os mesmos dados geométricos do modelo apresentado anteriormente. Considera-se também para os ligamentos atuados uma massa igual a 0,3 kg e raio de 15 mm. Para os ligamentos passivos utilizou-se massa de 0,5 kg e raio de 15 mm. Para a plataforma móvel,  $t = h_{\text{tool}} = 100$  mm e a massa igual a 5 kg.

A Fig. 67 ilustra a ferramenta de análise dinâmica desenvolvida. Existem apenas dois botões: o que calcula as saídas e o que apaga todos os campos. Os campos texto já possuem suas funções listadas. As figuras de 68 a 72 apresentam os resultados da Fig. 67.



Este caso simula uma operação onde se deseja, por exemplo, apertar um parafuso em um furo inclinado com relação à vertical. A Hexa está terminando a operação e já começa a frear. Os dados numéricos são apenas ilustrativos e não possuem a intenção de serem idênticos a algum caso real. A plataforma está com uma rotação de  $45^\circ$  no eixo  $X$ , sentido positivo, e o ponto de interesse da ferramenta,  $P_c$ , está no centro do sistema (coordenadas iguais a zero em todos os eixos). Os dados dinâmicos são:

- Velocidade da plataforma móvel: 1 m/s no eixo  $z$  da ferramenta, ou seja, um vetor em  $m/s$  no sistema global equivalente a  $[0 \ 0,71 \ -0,71]$ ;
- Aceleração da plataforma móvel: -1G no eixo  $z$  da ferramenta, ou seja, um vetor em  $m/s^2$  no sistema global equivalente a  $[0 \ -6,9 \ 6,9]$ ;
- Velocidades e acelerações angulares nulas;
- Forças externas nulas e torque no eixo  $z$  da ferramenta, sentido positivo, de 1 Nm, ou seja, um vetor em  $Nm$  no sistema global equivalente a  $[0 \ 0,71 \ -0,71]$ .

As matrizes numéricas de inércia dos ligamentos atuados e passivos e da plataforma móvel são dadas, respectivamente, por:

$$\mathbf{I}_{ij} = \begin{bmatrix} 3,375^{-5} & 0 & 0 \\ 0 & 0,0023 & 0 \\ 0 & 0 & 0,0023 \end{bmatrix} \quad (146)$$

$$\mathbf{I}_{hj} = \begin{bmatrix} 5,625^{-5} & 0 & 0 \\ 0 & 0,0104 & 0 \\ 0 & 0 & 0,0104 \end{bmatrix} \quad (147)$$

$$\mathbf{I}_{plat} = \begin{bmatrix} 0,0094 & 0 & 0 \\ 0 & 0,0094 & 0 \\ 0 & 0 & 0,0150 \end{bmatrix} \quad (148)$$

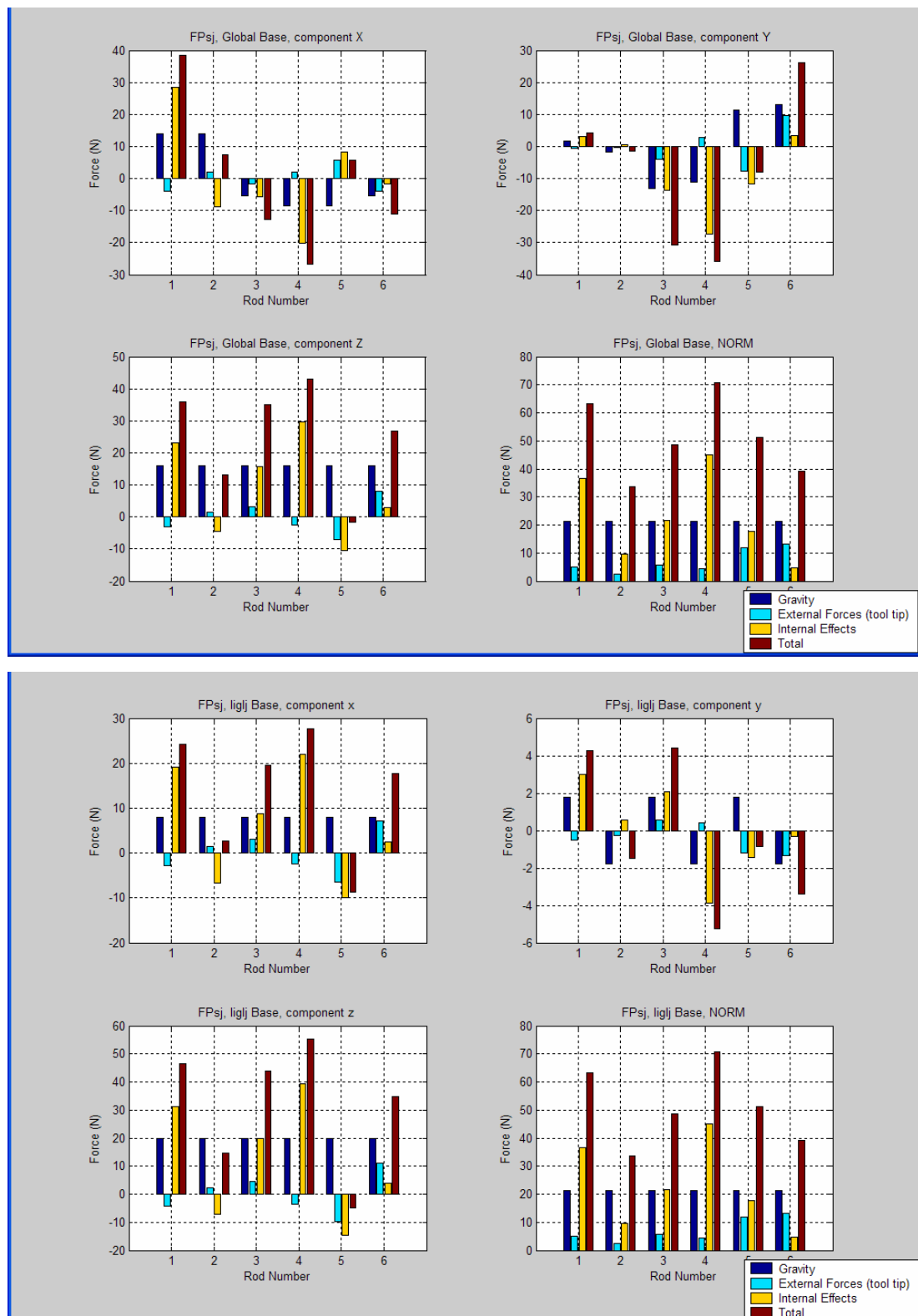
Para efeito do cálculo do torque nos motores foi considerado uma relação de transmissão arbitrária de 1:10.



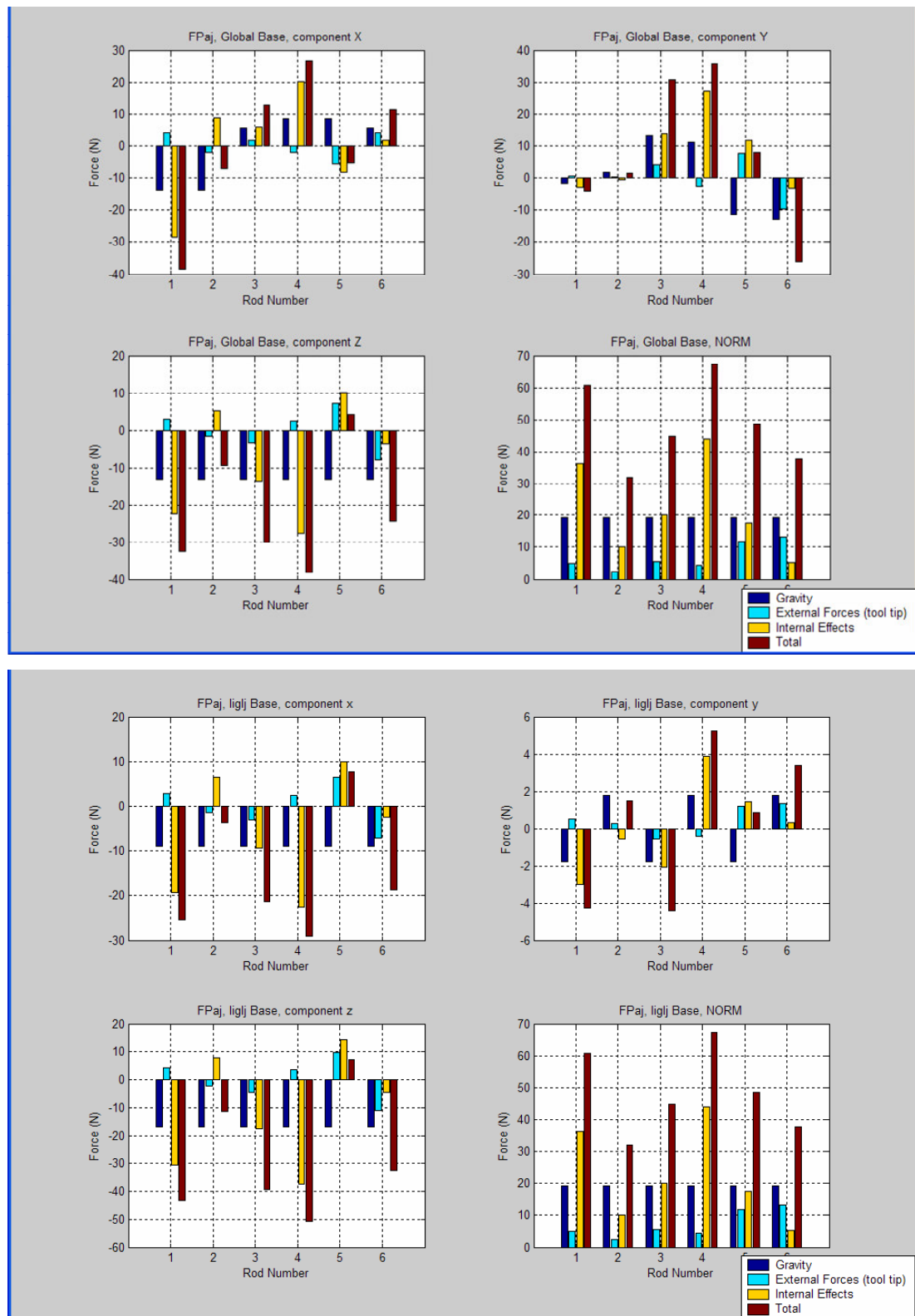
Figura 67: Ferramenta de análise dinâmica.

Algumas observações interessantes podem ser feitas sobre os resultados obtidos neste caso:

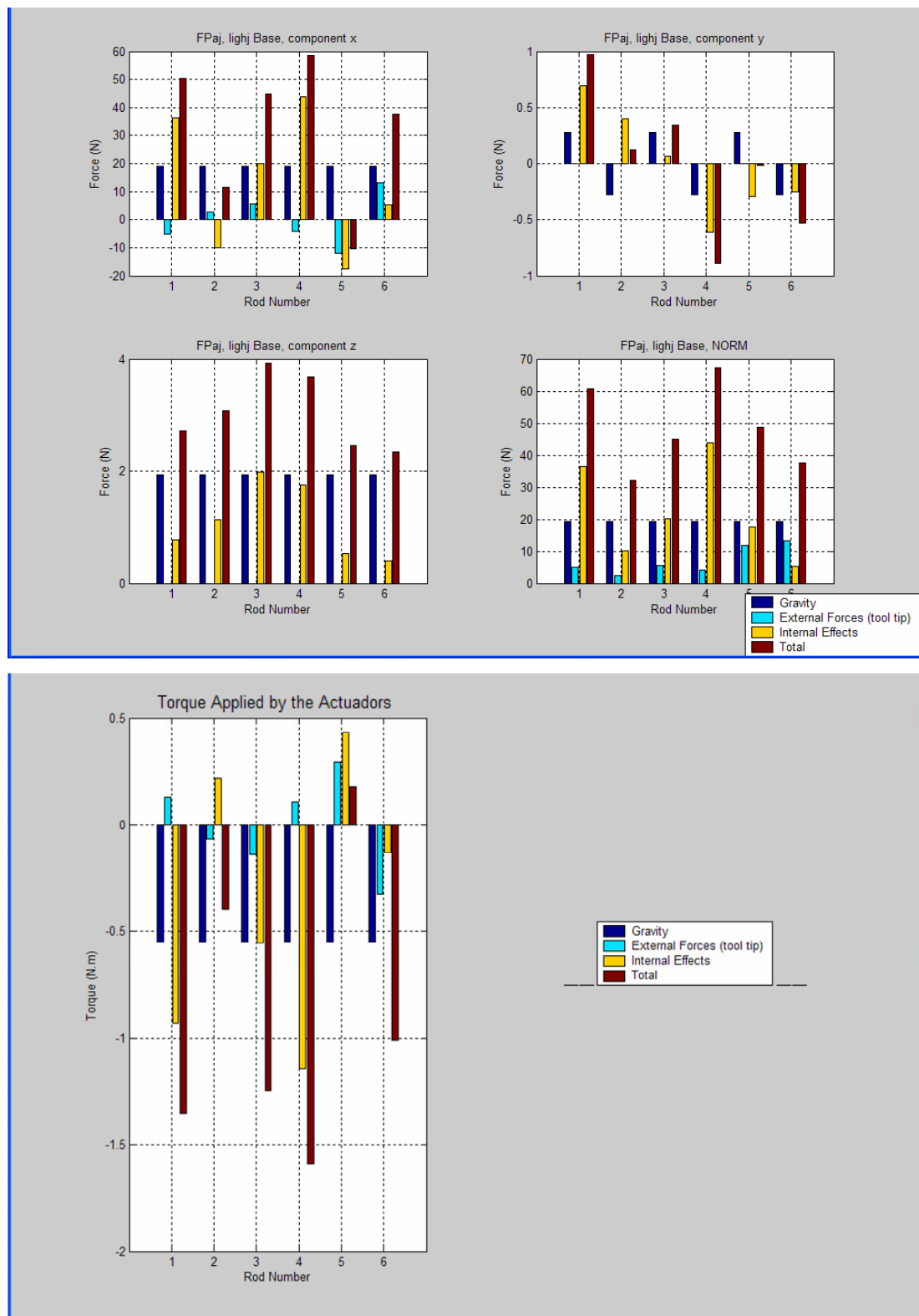
- Na Fig. 72, quando aos momentos em  $P_{s_j}$  vistos na base dos ligamentos atuados, observa-se na componente  $x$  que existe um resíduo numérico da ordem de  $10^{-4}$ . Para o modelo ideal deste estudo, os valores deveriam ser zero (não existe torção nos ligamentos);
- Os torques máximos nos atuadores foram da ordem de 1,5 Nm;
- As forças máximas no modelo foram da ordem de 70 N e os momentos máximos da ordem de 20 Nm.



**Figura 68:** Forças nos pontos da base fixa em duas bases – global (acima) e dos ligamentos atuados (abaixo).



**Figura 69:** Forças nos centros da articulação entre ligamentos atuados e passivos em duas bases diferentes, sendo acima a global e abaixo a dos ligamentos atuados.



**Figura 70:** Forças nos centros da articulação entre ligamentos atuados e passivos na base dos ligamentos passivos (acima) e torque aplicado pelos atuadores (abaixo).

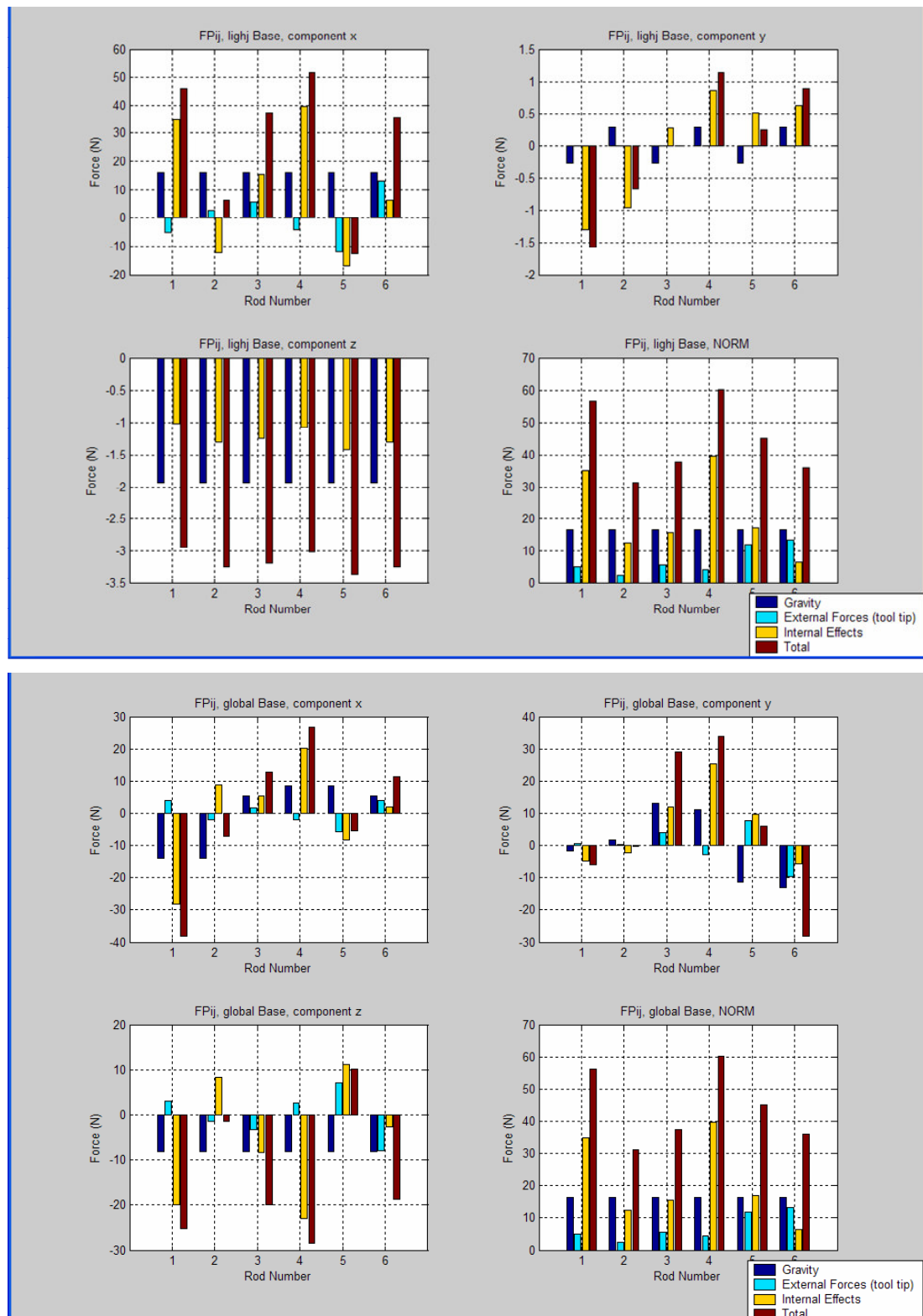
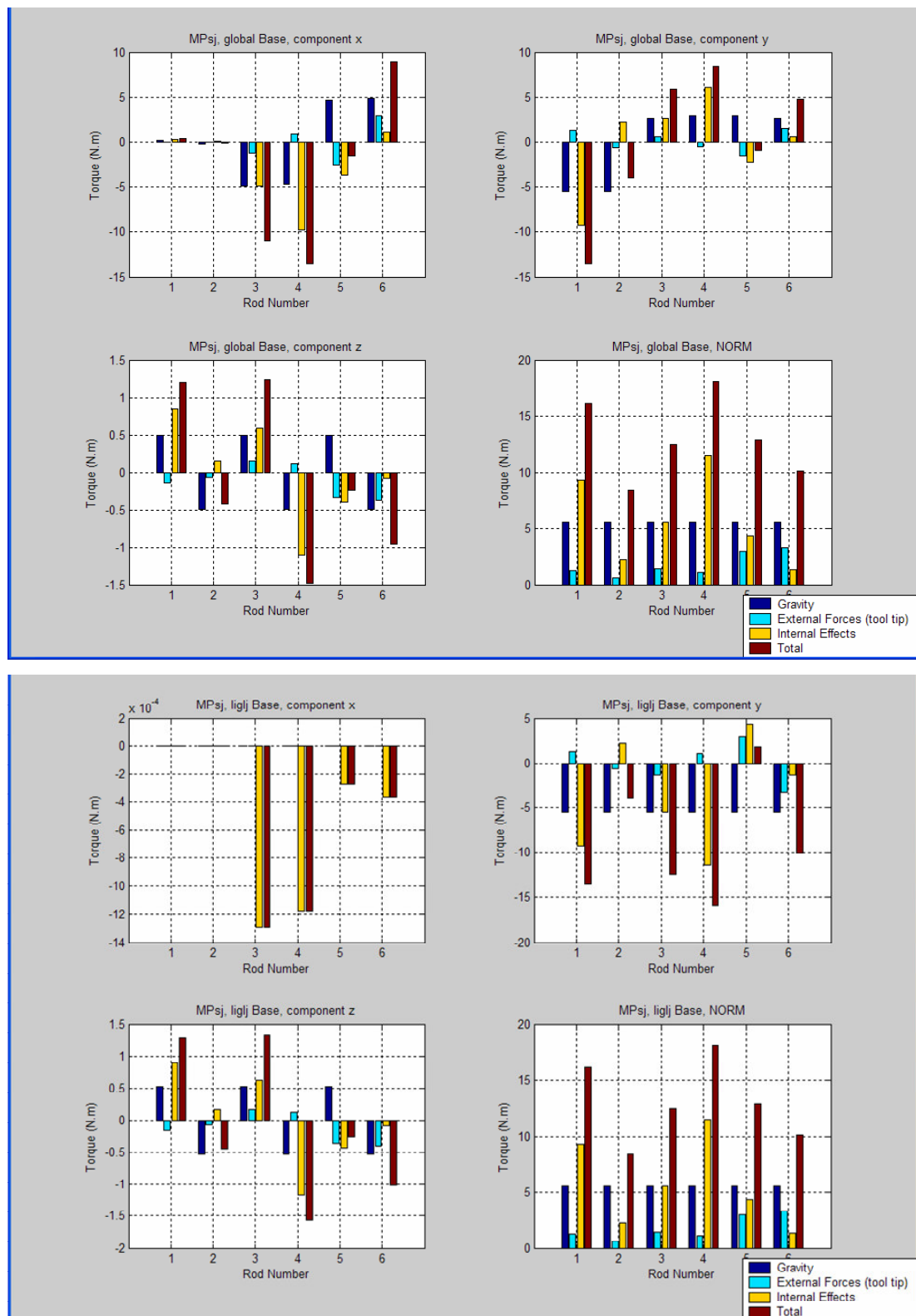


Figura 71: Forças nos pontos da plataforma móvel em duas bases – global (acima) e dos ligamentos passivos (abaixo).



**Figura 72:** Momentos nos pontos da base fixa em duas bases - global (acima) e dos ligamentos atuados (abaixo).

## 9.7. Otimização

A interface do programa de otimização, apresentada na Fig. 73, permite que o usuário defina todas as restrições laterais de cada variável de projeto, escolha o tipo de função objetivo, níveis de detalhe das análises e condições extras de parada que são baseadas no número máximo de iterações e de avaliações da função objetivo.

Conforme demonstrado no Capítulo 8, existem duas funções objetivo. Cada uma delas é utilizada ponto a ponto dentro do espaço de trabalho desejado pelo usuário. O grau de detalhamento das análises se dá através de dois campos:

- Um deles considera o número de pontos da malha de avaliações. Por exemplo, se for utilizado um número de dez pontos por dimensão, tem-se um total de  $10 \times 10 \times 10 = 1.000$  pontos para avaliar;
- O outro parâmetro é o número de divisões angulares. Esse parâmetro determina o número de orientações avaliadas em cada ponto. Por exemplo, se for utilizado um número de três divisões angulares para um ângulo máximo de  $45^\circ$  para com a vertical, a máquina será testada a  $0^\circ$ ,  $22,5^\circ$  e  $45^\circ$ . Além disso, esse número determina quantas serão as direções na qual a máquina irá se inclinar com relação a vertical. No caso é utilizado um número igual a duas vezes o número de divisões. Assim, cada um dos 1.000 pontos seria avaliado um total de  $3 \times 6 = 18$  vezes. O cálculo da função objetivo integral envolveria 18.000 avaliações de desempenho para então se chegar a uma média e somar o pior caso.

Após escolher todos os parâmetros o usuário aperta o botão e inicia a otimização. Ao término do algoritmo uma janela com uma lista de opções (vide a Fig. 74) aparece e podem-se escolher detalhes adicionais da solução, como a matriz Hessiana do ponto ótimo e multiplicadores de Lagrange. A mesma figura ainda mostra um aviso que aparece quando o usuário tenta fechar a janela da resposta encontrada (uma salvaguarda para que não se percam as informações finais, o que é importante pois a execução da rotina é lenta).



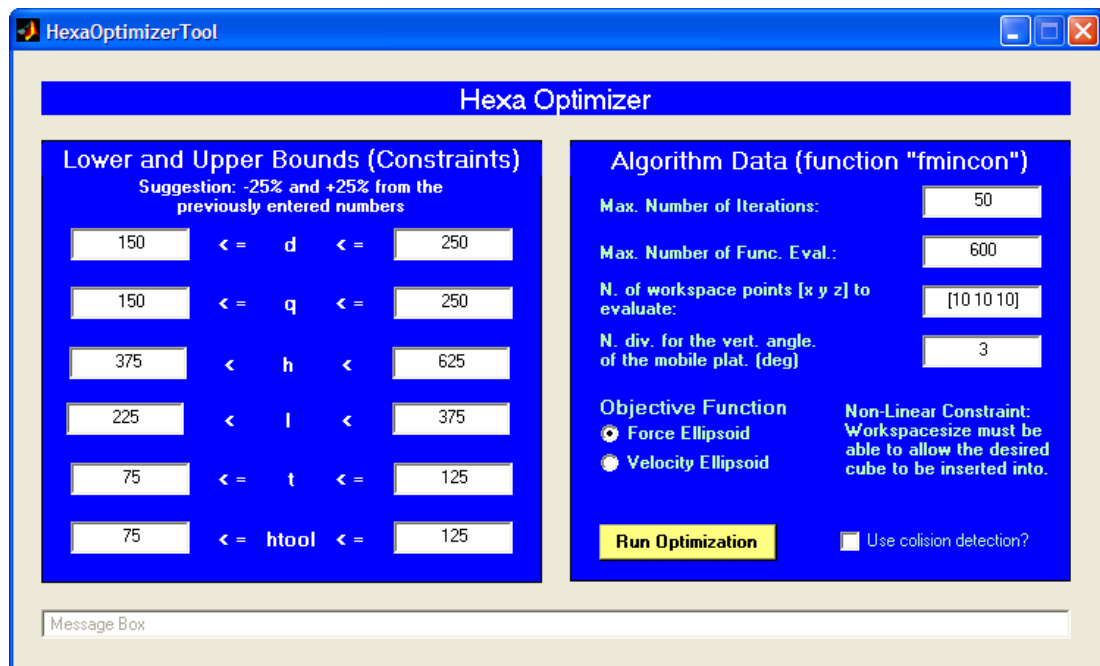


Figura 73: Ferramenta de Otimização.

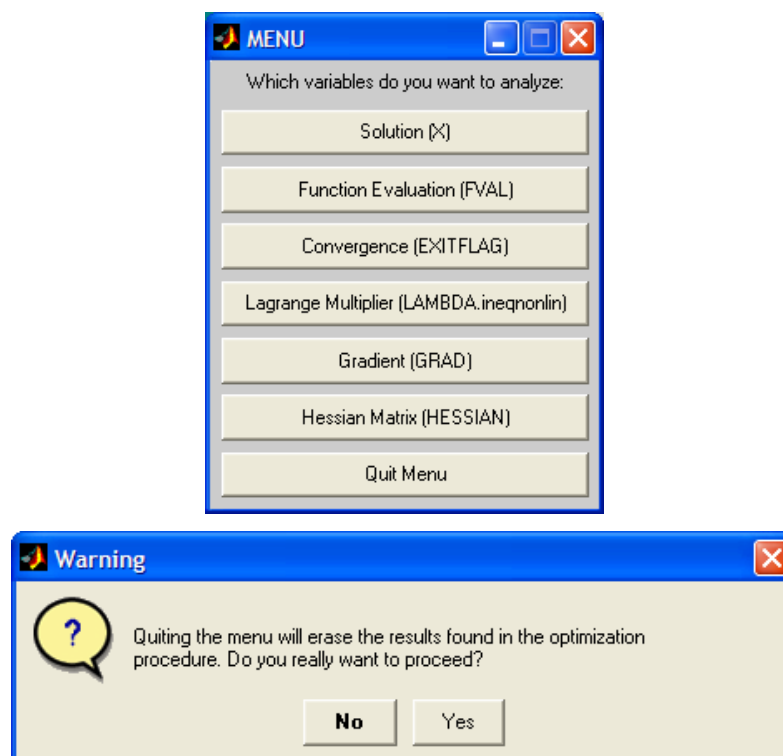


Figura 74: Acima a lista que aparece após a execução de uma otimização. Abaixo uma janela de aviso que aparece quando se deseja fechar a lista de opções.

### 9.7.1. Otimização Através do Elipsóide de Forças

Como chutes iniciais foram utilizados os parâmetros geométricos da máquina utilizada nos exemplos anteriores. Esses parâmetros para o chute inicial, bem como o tamanho do volume de trabalho desejado, são importados da janela inicial da ferramenta da Fig. 48. Os parâmetros e suas restrições laterais (em mm) são:

- $d = 200$        $150 \leq d \leq 250$  ;
- $q = 200$        $150 \leq q \leq 250$  ;
- $h = 500$        $375 \leq h \leq 625$  ;
- $l = 300$        $225 \leq l \leq 375$  ;
- $t = 100$        $75 \leq t \leq 125$  ;
- $h_{tool} = 100$        $75 \leq d \leq 125$  .

O tamanho para o volume de trabalho desejado foi um cubo de 200 mm de lado (200 mm X 200 mm X 200 mm).

O número de pontos avaliados na malha foi de 10 por eixo, totalizando 1.000 pontos avaliados. O número de divisões angulares escolhido foi 3. Conforme já mencionado, o algoritmo de colisões é extremamente lento de ser avaliado e esta restrição pouco influenciou o modelo da Hexa, sendo assim ele não foi utilizado.

O número máximo de iterações e avaliações da função objetivo utilizados foram, respectivamente, 50 e 600. Num Centrino® 2 GHz com 2 GB de memória RAM a execução levou pouco mais de quatro horas.

O algoritmo atingiu convergência. No software essa verificação é dada pela variável *EXITFLAG* que, em sendo maior que zero, informa que o programa garantiu a convergência. O valor fornecido pelo programa foi 1.

Existem outros indicadores da convergência do algoritmo como o valor da derivada que é pequeno na última iteração. Além disso, todos os autovalores da matriz Hessina foram maiores que zero, indicando um mínimo local (obviamente não se pode ter certeza que o mínimo encontrado é o mínimo global dentro do domínio definido pelas restrições). Para ilustrar passo a passo a convergência do método é apresentada a seqüência de iterações:

Iter	Function Count	f(x)	Max Constraint	Step-size	Directional Derivative
1	21	9.80E+06	-24.61	0.0156	1.77E+10
2	37	5.23E+06	-24.51	0.00391	-7.18E+08
3	52	5.08E+06	-24.52	0.00781	1.33E+08
4	63	4.72E+06	-22.25	0.125	-2.08E+07
5	77	3.56E+06	-21.9	0.0156	3.37E+07
6	93	2.97E+06	-21.84	0.00391	-1.33E+08
7	105	2.96E+06	-20.66	0.0625	-3.27E+07
8	121	2.84E+06	-20.58	0.00391	-4.09E+07
9	137	2.79E+06	-20.69	0.00391	8.74E+07
10	152	2.77E+06	-20.92	0.00781	3.17E+06
11	167	2.77E+06	-21.03	0.00781	1.34E+06
12	182	2.75E+06	-20.86	0.00781	-1.54E+06
13	192	2.71E+06	-20.82	0.25	5.77E+05
14	200	2.63E+06	-21.04	1	5.86E+04
15	215	2.63E+06	-21.04	0.00781	1.41E+05
16	223	2.63E+06	-21.04	1	-922
17	232	2.63E+06	-21.01	0.5	1.96E+04
18	240	2.62E+06	-21.02	1	-1.38E+04
19	263	2.62E+06	-21.02	3.05E-05	-2.39E+07
20	282	2.62E+06	-21.02	0.000488	-1.03E+04
21	298	2.62E+06	-21.02	0.00391	-1.47E+04
22	327	2.62E+06	-21.02	4.77E-07	-1.47E+06
23	340	2.62E+06	-21.02	0.0313	-708
24	356	2.62E+06	-21.02	0.00391	3.11E+03
25	366	2.62E+06	-21.02	0.25	2.56E+03
26	375	2.62E+06	-21.02	0.5	58
27	383	2.62E+06	-21.02	1	-91.7
28	410	2.62E+06	-21.02	1.91E-06	1.82E+06
29	420	2.62E+06	-21.02	0.25	6.27

**Tabela 8:** Iterações da otimização através do conceito do elipsóide de forças.

A solução encontrada para esta otimização, respectivamente, para as variáveis  $d$ ,  $q$ ,  $h$ ,  $l$ ,  $t$  e  $h_{tool}$ , foi [195.06 196.98 473.81 288.42 103.83 96.02]. Nenhuma restrição está ativa nesse ponto (restrição lateral ou do volume de trabalho).

Finalmente também se percebe que o volume de trabalho para a solução ótima é cerca de 10% maior que o volume desejado (volume útil).

### 9.7.2. Otimização Através do Elipsóide de Velocidades

Foram utilizados os mesmos parâmetros da otimização com o elipsóide de forças, bem como a mesma discretização da malha de pontos e o mesmo tamanho do volume de trabalho desejado. A seqüência de iterações foi:

Iter	Function count	f(x)	Max Constraint	Step-size	Directional Derivative
1	16	3554.02	-12.5	0.5	-6.79E+06
2	26	890.877	-9.375	0.25	3.66E+04
3	40	889.292	-9.593	0.0156	-1.22E+05
4	59	873.799	-9.6	0.000488	1.02E+05
5	75	861.692	-9.622	0.00391	-4.11E+03
6	94	860.189	-9.625	0.000488	-4.22E+03
7	115	859.788	-9.625	0.000122	-0.183

**Tabela 9:** Iterações da otimização através do conceito do elipsóide de velocidades.

A solução encontrada para esta otimização, respectivamente, para as variáveis  $d$ ,  $q$ ,  $h$ ,  $l$ ,  $t$  e  $h_{tool}$ , foi [211.44 231.52 574.74 347.10 96.60 115.37].

Ocorreram sete iterações e a função objetivo foi calculada 115 vezes. Ao longo das iterações observa-se uma melhora significativa da função objetivo, indicando que se está mais próximo de uma transformação isotrópica ao longo de todo o volume de trabalho ou pelo menos os piores casos foram eliminados.

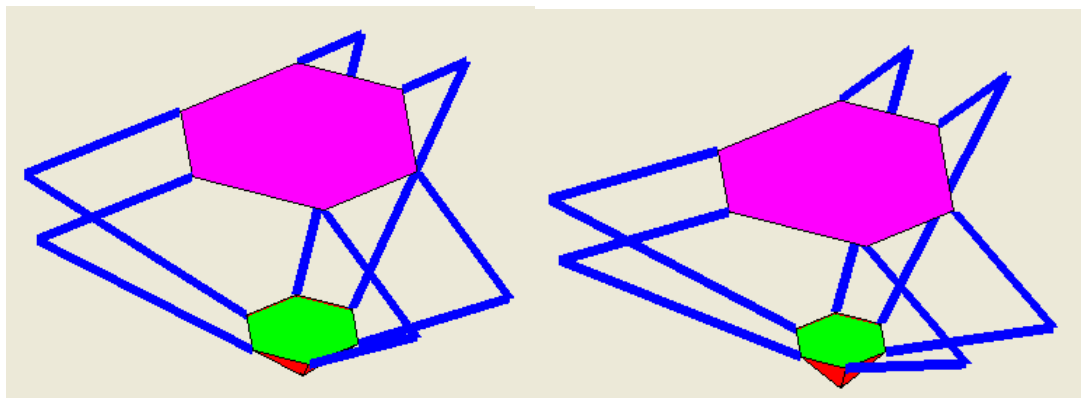
A convergência foi atingida e os mesmos comentários sobre gradiente e Hessiana podem ser feitos.

O volume de trabalho alcançável pela máquina ficou quase 10 mm maior que o desejado (cerca de 5% maior em volume). Também não existem restrições ativas.

### 9.7.3. Comentários das Soluções Encontradas e Observações

Quando se compara a melhoria do valor das funções objetivo ao longo das iterações, ambas as abordagens produziram valores semelhantes (cerca de 75% de melhoria se forem comparadas a solução inicial e a última iteração).

Contudo, a convergência através do elipsóide de forças foi muito mais lenta. Como os pontos analisados eram próximos e parte do objetivo de ambos era semelhante (transformação isotrópica), deduz-se que a magnitude dos valores da função objetivo para o elipsóide de forças deve-se principalmente ao termo responsável pelo tamanho do menor eixo do elipsóide. Realmente em alguns pontos, especialmente na borda do volume de trabalho com a plataforma inclinada, podem existir elipsóides com eixos muito pequenos. A Fig. 75 ilustra ambas as soluções encontradas com as duas abordagens.



**Figura 75:** À esquerda a solução da otimização através do elipsóide de forças e à direita a solução para o elipsóide de velocidades.

Para o caso do elipsóide de forças, vai-se ilustrar a solução em dois pontos:  $[0, 0, 0]$  e  $[100, 0, 0]$ . Em cada um a razão entre os tamanhos dos eixos melhorou de 5% a 7% e a avaliação do ponto pelo conceito da função objetivo melhorou de 10% a 15% (vide Tabela 10).

Infelizmente não se teve acesso a dados de outros protótipos de arquitetura Hexa para que se pudesse comparar os resultados de cada um com o resultado deste trabalho, assim procurou-se observar a evolução da solução a partir do chute inicial.

Pontos	Chute Inicial			Solução Encontrada		
	<i>Menor Eixo (A)</i>	<i>Maior Eixo (B)</i>	<i>Razão (A/B)</i>	<i>Menor Eixo (C)</i>	<i>Maior Eixo (D)</i>	<i>Razão (C/D)</i>
$[0, 0, 0]$	0.1620	5.9138	<b>36.5049</b>	0.1715	5.9720	<b>34.8222</b>
$[100, 0, 0]$	0.1148	5.9927	<b>52.2012</b>	0.1248	6.0873	<b>48.7764</b>

Pontos	Chute Inicial			Solução Encontrada		
	<i>1+1/(eixo mínimo)</i>	<i>Razão (A/B)</i>	<i>Avaliação do ponto</i>	<i>1+1/(eixo mínimo)</i>	<i>Razão (A/B)</i>	<i>Avaliação do ponto</i>
$[0, 0, 0]$	7.1728	36.5049	<b>261.8423</b>	6.8309	34.8222	<b>237.8670</b>
$[100, 0, 0]$	9.7108	52.2012	<b>506.9154</b>	9.0128	48.7764	<b>439.6120</b>

**Tabela 10:** Avaliação em dois pontos do conceito da função objetivo do elipsóide de forças.

Vale ainda uma observação a respeito das funções objetivo escolhidas e do processo de otimização. As funções, para a parte responsável por uma transformação isotrópica, utilizam uma razão entre os comprimentos dos eixos do elipsóide e, por conseguinte, dos autovalores da matriz característica. Sendo assim não se pode garantir que as funções objetivo sejam contínuas, pois se em algum momento houver uma inversão de autovalores ocorre um ponto de descontinuidade (**Silva, 2003**). Além disso, os pontos analisados não necessariamente são os mesmos em todas as iterações, pois a geometria da máquina é alterada em cada passo, alterando o volume de trabalho viável.

Embora o algoritmo tenha tido que lidar com os problemas citados, ocorreu convergência por ambos os métodos, demonstrando assim a viabilidade da abordagem.

## 10. CONCLUSÕES E PRÓXIMOS PASSOS

Este trabalho foi guiado por dois objetivos principais quanto ao robô escolhido de arquitetura paralela para aplicações industriais (a Hexa):

- O estudo, a modelagem cinemática e dinâmica do robô;
- A otimização do robô.

Foi desenvolvido um modelo completo da cinemática inversa da máquina, seguido da construção de um simulador virtual dotado de algumas funções simples capazes de demonstrar os métodos utilizados e o modelo desenvolvido.

Depois o estudo seguiu com a obtenção da matriz Jacobiano e o estudo das implicações desta matriz para o modelo da máquina. Assuntos como os elipsóides de forças e de velocidades, a rigidez e as singularidades foram abordados e discutidos. A análise da equação que relaciona a velocidade dos atuadores com a velocidade da plataforma móvel através da matriz Jacobiano também permitiu que se chegasse num método numérico para resolver a cinemática direta, evitando a complexa formulação através da dedução e solução numérica de sistemas de equações não lineares baseados nas restrições geométricas da estrutura do robô (como foi feito para a cinemática inversa).

Para a análise dinâmica foi deduzido um modelo baseado na formulação de Newton – Euler, onde a preocupação principal foi o detalhe de informações, em detrimento de um baixo custo computacional. O modelo, resolvido na ferramenta de dinâmica construída, é capaz de calcular todas as forças e momentos em diversos pontos de interesse da máquina, possuindo como dados de entrada as características da estrutura do robô (massas, dimensões, inércia dos elementos) e a posição, a orientação e as condições dinâmicas da plataforma móvel.

Na otimização, buscou-se uma formulação baseada na análise das equações da matriz Jacobiano, conforme fora sugerido por diversos pesquisadores. Também a fim de demonstrar a abordagem proposta, uma ferramenta fora construída. Embora tenha sido realizada a análise da dinâmica inversa da máquina, otimizações baseadas no modelo dinâmico não foram executadas, ficando como sugestão para trabalhos futuros.

## 10.1. Conclusões

O modelo da cinemática inversa do robô, baseado em sistemas de equações sobre as restrições geométricas da máquina, foi resolvido sem maiores complicações, sendo de execução e formulação muito simples, fato comum à grande maioria dos robôs de arquitetura paralela.

O simulador virtual foi totalmente baseado em elementos simples, como retas e polígonos, sendo estes últimos também servindo para o desenho de círculos (computacionalmente, os círculos são entendidos como polígonos de muitos lados). Computacionalmente, o simulador cumpriu as funções de se ter um baixo custo de execução e de ser de fácil utilização; dois pontos muito importantes quando se considera o usuário do programa.

Capaz de apresentar a máquina em diferentes posições e orientações e em movimento (embora sem considerações mais complexas como acelerações), o simulador também permitiu a observação e o estudo de alguns casos do elipsóide de forças e pode, com pequenas alterações adicionais, comportar o elipsóide de velocidades em seu programa. Na observação do elipsóide, também foi possível perceber para algumas trajetórias da máquina as mudanças até mesmo bruscas do tamanho e forma do elipsóide o que indica uma queda / aumento de desempenho.

Partindo-se dessas observações e das sugestões de outros pesquisadores, utilizou-se o modelo dos elipsóides como uma abordagem à construção de algoritmos de otimização que pudessem levar a um desenho ótimo da geometria da máquina, respeitando um volume de trabalho mínimo. Simulações com as funções propostas indicam que o caminho escolhido é viável. Utilizou-se o conceito de ótimo-pareto equilibrando a média do valor da função e o pior caso encontrado e para ambas as funções objetivo escolhidas alcançou-se convergência.

Uma das utilidades do modelo de otimização (além da melhoria de desempenho) está na restrição do volume de trabalho, já que ainda é bastante difícil se conceber as dimensões de um robô para um dado tamanho e forma de volume de trabalho desejado.

Finalmente, para a análise dinâmica, o modelo ficou aparentemente desvinculado no estudo por não ter sido utilizado na otimização. Contudo, sua



utilidade também está na análise da solução proposta pelo algoritmo de otimização que embora não obtenha convergência através da análise dinâmica pode se beneficiar de alguns testes sobre as respostas produzidas.

A abordagem utilizada para o modelo da cinemática direta fica como uma sugestão eficiente computacionalmente para que possa ser implementada em algoritmos de controle, outro campo que pode se beneficiar dos estudos apresentados nesta dissertação.

De forma geral, as ferramentas desenvolvidas possibilitaram um entendimento profundo do funcionamento da Hexa tanto cinematicamente quanto dinamicamente. Esses estudos possibilitarão, dentre outros, o desenho de algoritmos de controle da máquina e a investigação das forças mecânicas para o projeto da estrutura e partes móveis.

## 10.2. Sugestão de trabalhos futuros

Encerra-se aqui o escopo deste trabalho de Mestrado. Contudo, para trabalhos futuros que desejem ter como base alguns itens discutidos aqui, fica como sugestão algumas áreas:

- **Na Cinemática Direta:** investigação analítica da cinemática e construção de algoritmos mais eficientes computacionalmente e com menor erro. Atualmente, este campo tem sido foco de muitos trabalhos;
- **Na Obtenção do Volume de Trabalho:** detalhamento adicional do algoritmo, dando a capacidade de se saber o porquê do seu limite (ou borda). Conhecendo-se as principais causas, é possível estudar formas de aumentar o volume através da diminuição de tais restrições. Esse estudo já foi sugerido e executado por **Bianchi et al (2000)** que construiu uma ferramenta virtual que desenha o volume e indica com cores o motivo de ter ocorrido uma borda;
- **Na Dinâmica:** investigar a análise dinâmica direta e refinar o modelo da dinâmica inversa proposto aqui, simplificando-o para operar em tempo real e inseri-lo em algoritmos de controle. Também podem ser estudados outros métodos como o princípio do trabalho virtual ou a formulação de Lagrange;

- **Na Otimização:** investigar outras possibilidades, como descrito por **Brogårdh (2002)**, que apresenta a metodologia de projeto de produtos da ABB. Esta é uma área bastante complexa, não apenas pela otimização em si, mas também porque demanda a preparação de modelos complexos;
- **Na Síntese:** este é uma área que ainda é muito nova e pouco explorada na robótica, principalmente com robôs de arquiteturas mais complexas como os de arquitetura paralela. Praticamente quaisquer trabalhos aqui caminhariam no estado – da – arte em robótica computacional.
- **No Projeto Mecânico:** o desenho, a fabricação e a montagem das articulações e componentes principais do robô, tendo como ápice a construção física da estrutura;
- **No Projeto do Controlador:** o desenho e a implementação do controlador, incluindo a abordagem do controle da máquina e a eletrônica de comando.

## 11. REFERÊNCIAS BIBLIOGRÁFICAS

- Angeles, 2002.** Jorge Angeles, “The Qualitative Synthesis of Parallel Manipulators”, Workshop on Fundamental Issues and Future Directions on Parallel Mechanisms and Manipulators, Outubro, 2002.
- Bianchi et al, 2000.** Giacomo Bianchi, Irene Fassi, Lorenzo Molinari Tosatti, “A Virtual Prototyping Environment for Parallel Kinematic Machine Analysis and Design”, 2000.
- Bidault et al, 2001.** Florence Bidault, Chin-Pun Teng, Jorge Angeles, “Structural Optimization of a Spherical Parallel Manipulator Using a Two-Level Approach”, ASME Design Engineering Technical Conferences, 2001.
- Bonev e Gosselin, 2000.** Ilian A. Bonev, Clément M. Gosselin, “A Geometric Algorithm for the Computation of the Constant-Orientation Workspace of 6-RUS Parallel Manipulators”, ASME 2000, Design Engineering Technical Conferences and Computers and Information in Engineering Conference, Baltimore, Maryland, Setembro, 2000.
- Bonev e Ryu, 1999.** Ilian A. Bonev, Jeha Ryu, “A new Method for Solving the Direct Kinematics of General 6-6 Stewart Platforms using three Linear extra Sensors”, Mechanism and Machine Theory, Elsevier Science, 1999.
- Borel, 1908.** E. Borel, “Mémoire sur les déplacements à trajectoire sphériques”, Mémoires presentes par divers savants, 1908.
- Bricard, 1906.** R. Bricard, “Mémoire sur les déplacements à trajectoire sphériques”, Journal de l'École Polytechnique, 1906.
- Brogårdh, 2002.** Torgny Brogårdh, “PKM Research – Important Issues, as seen from a Product Development Perspective at ABB Robotics”, Proceedings of the Workshop on Fundamental Issues and Future Research Directions for Parallel Mechanisms and Manipulators, Quebec, Canadá, Outubro, 2002.
- Bruyninckx, 1997.** Herman Bruyninckx, “The 321-HEXA: A Fully-Parallel Manipulator with Closed-Form Position and Velocity Kinematics”, IEEE International Conference on Robotics and Automation, Albuquerque, New Mexico, April, 1997.

- Bruyninckx, 2005.** Herman Bruyninckx, “The Robotics WEBook”, Chapter Parallel Robots, Agosto de 2005 (www.roble.info).
- Bruzzone et al., 2002.** L. E. Bruzzone, R. M. Molfino, R. P. Razzoli, “Modelling and Design of a Parallel Robot for Laser-Cutting Applications” Proc. of the IASTED International Conference Modelling, Identification and Control (MIC2002), Innsbruck, Austria, February 18-21, 2002.
- Cabral, 2003.** Eduardo L. L. Cabral, “Apostila de Robótica – Capítulo 2: Preliminares Matemáticos”, EPUSP, 2003.
- Chablat e Wenger, 2003.** Damien Chablat, Philippe Wenger, “Architecture Optimization of a 3-DOF Translational Parallel Mechanism for Machining Applications, the Orthoglide”, IEEE Transactions on Robotics and Automation, Vol. 19, No 3, Junho, 2003.
- Chan e Ebert-Uphoff, 2001.** Vincent Kent Chan, Imme Ebert-Uphoff, “Investigation of the Deficiencies of Parallel Manipulators in Singular Configurations Through the Jacobian Nullspace”, Georgia Institute of Technology, 2001.
- Codourey e Burdet, 1997.** A. Codourey, E. Burdet, “A body oriented method for finding a linear form of the dynamic equations of fully parallel robots”, In IEEE International Conference on Robotics and Automation, pages 1612-1618, Albuquerque, April, 21-28, 1997.
- Conti et al., 1997.** Joseph P. Conti, Charles M. Clinton, Guangming Zhang, “Workspace Variation of a Hexapod Machine Tool”, 1997.
- Dafaoui et al, 1994.** M. Dafaoui, Y. Amirat, J. Pontnau, C. François, “Manipulateur parallèle à six degrés de liberté: modeles et volume de travail”. Revue d’Automatique et de productique appliqué, 7(2): 195-220, 1994.
- Dasgupta e Mruthyunjaya, 1998a.** Bhaskar Dasgupta e T. S. Mruthyunjaya, “Closed-Form Dynamic Equations of the General Stewart Platform through the Newton-Euler Approach”, Mech. Mach. Theory, Vol. 33, No. 7, pp. 993-1012, 1998.
- Dasgupta e Mruthyunjaya, 1998b.** Bhaskar Dasgupta e T. S. Mruthyunjaya, “A Newton-Euler Formulation for the Inverse Dynamics of the Stewart Platform Manipulator”, Mech. Mach. Theory, Vol. 33, No. 8 pp. 1135-1152, 1998.

- Do e Yang, 1988.** W. Q. D. Do, D. C. H. Yang, “Inverse dynamic analysis and simulation of a platform type of robot”, *Journal of Robotics Systems*, 5(3): 209-227, 1988.
- D’Souza, 2003.** Ajay D’Souza, “Modeling and Kinematic Analysis of a 6 D.O.F MOOG Motion Base using Virtual Prototyping Tools”, *Mechanical and Aerospace Eng. Dep., State University of New York at Buffalo*, 2003.
- Fioretti, 1994.** A. Fioretti, “Implementation oriented kinematics analysis of a six dof parallel robotic platform”, In 4<sup>th</sup> IFAC Symposium of Robot Control, pp. 43-50, Capri, September, 1994.
- Freund, 2004.** Robert M. Freund, “Optimality Conditions for Constrained Optimization Problems”, *Massachusetts Institute of Technology*, fevereiro, 2004 (*vide link da internet ao final da bibliografia*).
- Geng e Haynes, 1991.** Z. Geng e L. Haynes, “Neural Network Solution for the Forward Kinematics Problem of a Stewart Platform”, *International Conference on Robotics and Automation*, IEEE, 1991.
- Gosselin e Angeles, 1990.** C. Gosselin, J. Angeles, “Singularity analysis of closed-loop chains”, *IEEE Trans. on Robotics and Automation*, 6(3): 281-290, June 1990.
- Gough, 1956.** V. E. Gough, “Contribution to discussion of papers on research in automobile stability, control and tire performance”, 1956-1957, *Proc. Auto Div. Inst. Mech. Eng.*
- Gough, 1962.** V. E. Gough, S.G. Whitehall, “Universal tire test machine”, In *Proceedings 9<sup>th</sup> Int. Technical Congress F.I.S.I.T.A.*, volume 117, páginas 117-135, Maio de 1962.
- Han et al, 1994.** Kilryong Han, WanKyun Chung, Y. Youm, “Local Structurization for the Forward Kinematics of Parallel Manipulators Using Extra Sensor Data”, *Pohang University of Science and Technology (POSTECH)*, Korea, 1994.
- Hannan e Walker, 1999.** M. Hannan and I.D. Walker, “A Novel Elephant's Trunk Robot”, *Proc. IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Atlanta, GA, Setembro 1999, pp 410-415.

- Hao et al, 1996.** Zou Hao, Wang Qiyi, Li Qunming, Zhang Bo, “The Kinematics and Workspace Analysis of A Parallel Manipulator for Manufacturing”, Proceedings of the IEEE International Conference on Industrial Technology, 1996.
- Honegger et al., 1997.** M. Honegger, A. Codourey, E. Burdet, “Adaptive control of the hexaglide, a 6 dof parallel manipulator”, In IEEE International Conference on Robotics and Automation, pages 543-548, Albuquerque, April, 21-28, 1997.
- Honegger et al., 2000.** M. Honegger, R. Brega, G. Schweitzer, “Application of a nonlinear adaptive controller to a 6 dof parallel manipulator”, Proceedings of the IEEE International Conference on Robotics and Automation, April, 2000, pages 1930-1935.
- Innocenti e Parenti-Castelli, 19XX.** Innocenti C., Parenti-Castelli V., “Exhaustive enumeration of fully parallel kinematic chains”, Dynamic System and Control, Vol. 55-2, pp. 1135-1141.
- Ji, 1993.** Zhiming Ji, “Study of the Effect of Leg Inertia in Stewart Platforms”, ICRA, pp 121-126, Atlanta, USA, 1993.
- Kim et al., 1999.** J. Kim, F. C. Park, J. M. Lee, “A new machine tool capable of five face machining”, Seoul National University, Korea, 1999.
- Kim e Tsai, 2002.** Han Sung Kim, Lung-Wei Tsai, “Design Optimization of a Cartesian Parallel Manipulator”, University of California, July, 2002.
- Kosuge et al, 1993.** Kazuhiro Kosuge, Koji Takeo, Toshio Fukuda, Katsumi Kai, Tomo Mizuno, Hiroaki Tomimatsu, “Computation of Parallel Link Manipulator Dynamics”, Int. Conf. on Indus. Electronics, Control and Instrumentation (IECON), pp 1672-1677, Hawaii, Novembro, 1993.
- Knapp e Cobet, 2000.** Wolfgang Knapp, Matthias Cobet, “The IWF Hexaglide: A new concept for high speed machining”, IWF, Zürich, November, 2000.
- Kreffft et al, 2002.** Mathias Kreffft, Jürgen Hessenbach, Mathias Frindt, Friedrich Wahl, “Robotic Systems for Handling and Assembly – High Dynamic Parallel Structures with Adaptronic Components, Outubro, 2003.
- Last et al., 2005.** Philip Last, Christoph Budde, Jürgen Hesselbach, “Self – Calibration of the HEXA Parallel Structure”, IEEE International Conference

on Automation Science and Engineering, Edmonton, Canada, August, 1<sup>st</sup> and 2<sup>nd</sup>, 2005.

- Lazarevic, 2000.** Zoran Lazarevic, “Feasibility of a Stewart Platform with Fixed Actuators as a Platform for CABG Surgery Device”, Master’s Thesis, Columbia University, Department of Bioengineering, 2000.
- Leal, 2005.** Rafael Della Giustina Leal, “Impactos Sociais e Econômicos da Robotização: Estudo de Caso do Projeto Roboturb”, Dissertação de Mestrado, Universidade Federal de Santa Catarina, Engenharia Elétrica, 2005.
- Lebret et al., 1993.** G. Lebret, K. Liu, F. Lewis, “Dynamic Analysis and Control of a Stewart Platform Manipulator”, Journal of Robotic Systems, pp 629-655, Julho, 1993.
- Leger, 1999.** Chris Leger, “Automated Synthesis and Optimization of Robot Configurations: an Evolutionary Approach”, Doctorate Thesis, Carnegie Mellon University, 1999.
- Liu et al., 1993.** Kai Liu, John M. Fitzgerald, Frank L. Lewis, “Kinematic Analysis of a Stewart Platform Manipulator”, IEEE Transactions on Industrial Electronics, Vol. 40, No. 2, Abril, 1993.
- Masory et al., 1993.** O. Masory, J. Wang, H. Zhuang, “On the Accuracy of a Stewart Platform Part II: Kinematic Compensation and Calibration”, IEEE International Conference on Robotics and Automation, pp 725-731, Maio, 1993.
- Masory e Wang, 1993.** O. Masory, J. Wang, “Workspace Evaluation of Stewart Platforms”, Advanced Robotics, 9(4):443-461, 1995.
- Maurine et al., 1999.** Patrick Maurine, Koyu Abe, Masaru Uchiyama, “Towards More Accurate Parallel Robots”, IMEKO-XV World Congress, Osaka, Japan, Volume X (1999/6/13-18), 73-80.
- Merlet, 2000.** Jean-Pierre Merlet, “*Parallel Robots*”, Kluwer Academic Publishers, Solid Mechanics and its applications, 2000.
- Morizono et al., 2003.** Tetsuya Morizono, Yoji Yamada, Yoji Umetani, Takahira Yamamoto, Tetsuji Yoshida, Shigeru Aoki, “Design of a New Exoskeletal Mechanism for a Shoulder Joint of Wearable Robots: The Wearable HEXA

- Mechanism”, IEEE International Conference on Robotics and Automation, Taipei, Taiwan, September, 14<sup>th</sup> – 19<sup>th</sup>, 2003.
- Nguyen et al., 1991.** Charles C. Nguyen, Zhen-Lei Zhou, Sami S. Antrazi, C. E. Campbell Jr., “Efficient Computation of Forward Kinematics and Jacobian Matrix of a Stewart Platform Base Manipulator”, IEEE, 1991.
- Company e Pierrot, 1999.** Olivier Company, François Pierrot, “A New 3T-1R Parallel Robot”, In ICAR, Tokyo, November, 1999.
- Pierrot et al., 1990.** F. Pierrot, M. Uchiyama, P. Dauchez, A. Fournier, “A New Design of a 6-DOF Parallel Robot”, J. of Robotics and Mechatronics, Vol. 2, No. 4, pp. 308-315, 1990.
- Pierrot et al., 1991a.** F. Pierrot, A. Fournier, P. Dauchez, “Towards a Fully-Parallel 6 DOF Robot for High-Speed Applications”, IEEE International Conference on Robotics and Automation, Sacramento, California, Estados Unidos, Abril, 1991.
- Pierrot et al., 1991b.** F. Pierrot, P. Dauchez, A. Fournier, “Hexa : A Fast six-DOF Fully-Parallel Robot”, IEEE, 1991.
- Pierrot e Company, 1999.** François Pierrot, Olivier Company, “H4: A New Family of 4-dof Parallel Robots”, IEEE International Conference on Advanced Intelligent Mechatronics, September, 1999.
- Queiroz et al., 2005.** Eduardo Martins de Queiroz, Carlos Cezar Bier, Alexandre Campos, Raul Guenther, Joen Maass, “Direct Singularity Avoidance Strategy for the Hexa Parallel Robot”, 18<sup>th</sup> International Congress of Mechanical Engineering, November, 6<sup>th</sup> – 11<sup>th</sup>, Ouro Preto, Minas Gerais, Brazil, 2005.
- Rosenman, 1997.** M. Rosenman, “The Generation of Form Using an Evolutionary Approach”, In D. Dasgupta and Z. Michalewicz (eds.) Evolutionary Algorithms in Engineering Applications, pages 135–154. Springer-Verlag, 1997.
- Ryu et al., 1998.** S. J. Ryu, J. W. Kim, J. C. Hwang, C. Park, H. S. Cho, K. Lee, Y. Lee, U. Cornel, F. C. Park, J. Kim, “Eclipse: An Overactuated Parallel Mechanism for Rapid Machining”, Seul National University, 1998.
- Sato et al., 2001.** D. Sato, Y. Ishii, T. Shitashimizu, D. Kim, M. Uchiyama, “3D Graphics-Based Off-line Task Teaching for a Force-Controlled High-Speed



- Parallel Robot”, IEEE International Symposium on Assembly and Task Planning Soft Research Park, Fukuoka, Japan, May 28-29, 2001.
- Silva, 2003.** Emílio Carlos Nelli Silva, “Apostila de PMR5215 – Otimização Aplicada ao Projeto de Sistemas Mecânicos”, Escola Politécnica da USP, Departamento de Engenharia Mecatrônica e Sistemas Mecânicos, 2003.
- Sima’an et al, 1998.** N. Sima’an, D. Glozman, M. Shoham, “Design Considerations of New Six Degrees-of-Freedom Parallel Robots”, IEEE International Conference on Robotics and Automation, 1998.
- Smith e Nguyen, 1991.** Walter F. Smith, III, Charles C. Nguyen, “On the Mechanical Design of a Stewart Platform Based Robotic End-Effector”, IEEE, 1991.
- Stewart, 1965.** D. Stewart, “A platform with 6 degrees of freedom”, Proc. of the institution of Mechanical Engineers, 180 (Part 1, 15): 371-386, 1965.
- Stankóczy, 1999.** Zoltán Stankóczy, “Development of a new Parallel Kinematics Machine Tool”, Periodica Polytechnica Ser. Mech. Eng. Vol. 43, No 2, pp. 147-154, 1999.
- Tartari e Marchiori, 2003.** Sylvio Celso Tartari Filho, Ricardo Marchiori, “Projeto Mecânico de um Robô de Arquitetura Paralela do tipo Hexaglide para Usinagem”, Dissertação de Conclusão de Curso de Graduação, EPUSP, 2003.
- Tsai, 1998.** Lung-Wen Tsai, “Solving the Inverse Dynamics of Parallel Manipulators by the Principle of Virtual Work”, ASME Design Engineering Technical Conferences, 1998.
- Tsai, 1999.** Lung-Wen Tsai, “Robot Analysis: The Mechanics of Serial and Parallel Manipulators”, John Wiley & Sons, 1999.
- Tsuzuki, 2003.** Marcos Tsuzuki, “Apostila de Introdução a CAD/CAM”, EPUSP, Departamento de Engenharia Mecatrônica, 2003.
- Uchiyama et al., 1994.** Masaru Uchiyama, Kazuyuki Masukawa, Tetsuya Sadotomo, “Experiment on Dynamic Control of a Hexa-Type Parallel Robot”, World Automation Congress (WAC), Hawaii, Agosto 14-17, 1994.
- Uchiyama et al., 1996.** M. Uchiyama, T. Miwa, D. N. Nenchev, “A Very Fast Parallel Robot to be Applied to Dexterous Motion”, WAC, Montpellier, França, Maio 28-30, 1996.

- Yang e Chen, 2001.** Guilin Yang, I-Ming Chen, “Singularity Analysis of Three-Legged Parallel Robots Based on Passive-Joints Velocities”, IEEE Transactions on Robotics and Automation, Vol. 17, No. 4, Agosto, 2001.
- Zanganeh et al., 1997.** K. E. Zanganeh, R. Sinatra, J. Angeles, “Kinematics and dynamics of a six degrees-of-freedom parallel manipulator with revolute legs”, Robotica, 15(4):385-394, July-August, 1997.
- Zhang e Song, 1991.** Chang-de Zhang, Shin-Min Song, “Forward Kinematics of a Class of Parallel (Stewart) Platforms with Closed-Form Solutions”, International Conference on Robotics and Automation, IEEE, 1991.
- Zlatanov et al., 1992.** D. Zlatanov, M. Q. Dai, R. G. Fenton, B. Benhabib, “Mechanical Design and Kinematics Analysis of a three legged six degrees-of-freedom parallel manipulator”, 22<sup>nd</sup> Biennial Mechanisms Conference, volume DE-45, pages 529-536, Scottsdale, September, 13-16, 1992.

### ***Páginas da Internet***

(quando a página apresentava a data da última atualização, esta foi adotada, caso contrário, utilizou-se o ano em que se acessou a página):

**ABB, 2005:**

<http://www.abb.com>

**Ansett Flight Simulator Centre, 2004:**

<http://www.ansettsimulators.com>

**François Pierrot, 2005:**

<http://www.lirmm.fr/~pierrot>

**FANUC, 2005:**

<http://www.fanuc.com>

**Ingersoll, 2004:**

<http://www.ingersoll.com>

**IPMP, 2002. Innovative Parallel Mechanism Platforms:**

<http://www.nrl-eclipse.org/index.asp>

**IFM, 2005 (Institut für Maschinenbau):**

<http://www.maschinenbau.hs-magdeburg.de>

Para o link específico da Hexa, deve-se acrescentar ao link anterior:

[/personal/bargfrede/fue/parallel/ws\\_parallel.html](/personal/bargfrede/fue/parallel/ws_parallel.html)

**IWF, 2004 (Institut für Werkzeugmaschinen und Fertigungstechnik):**

<http://www.iwf.ing.tu-bs.de/f+e/fawm/SFB562.html>

Ou também pode ser utilizado o link específico da Hexa:

<http://www.tu-braunschweig.de/sfb562/galerie/hexa2?lang=de>

**KAIST (Korea Advanced Institute of Technology), 2004:**

<http://cwllab.kaist.ac.kr/Research/simulator/main.php>

**Karel Chapek Website, 2004:**

<http://capek.misto.cz/english/index.html>

**LIRMM, 2004:**

<http://www.lirmm.fr/xml/fr/lirmm.html>

**LSRO, 2005 (École Polytechnique Fédérale de Lausanne, Institut de Production et Robotique, Laboratoire de Systèmes Robotiques):**

<http://lsro.epfl.ch/prd/applications.php>

**METU (Middle East Technical University), 2005:**

<http://www.metu.edu.tr>

**Parallemic, 2002:**

<http://www.parallemic.org>

**PI, 2005:**

<http://www.physikinstrumente.de>

Link do modelo Hexapod M-850:

<http://www.physikinstrumente.de/products/prdetail.php?secid=7-16>

**Robotics Online, 2004 e 2005:**

<http://www.roboticsonline.com>

**RRG (Robotics Research Group), University of Texas, 2005:**

<http://www.robotics.utexas.edu/rrg>

**Space Machines Laboratory, 2004:**

<http://www.space.mech.tohoku.ac.jp/index-e.html>

**Notas de Aula de Robert M. Freund, Fevereiro, 2004, MIT:**

[http://ocw.mit.edu/NR/rdonlyres/Sloan-School-of-Management/15-084JSpring2004/7240EF84-B20D-419F-B1C0-2DAF3277F5C4/0/lec6\\_constr\\_opt.pdf](http://ocw.mit.edu/NR/rdonlyres/Sloan-School-of-Management/15-084JSpring2004/7240EF84-B20D-419F-B1C0-2DAF3277F5C4/0/lec6_constr_opt.pdf)

## APÊNDICE A: DEDUÇÕES COMPLEMENTARES DA ANÁLISE CINEMÁTICA INVERSA DA HEXA

Este item demonstra como as equações da cinemática inversa podem ser escritas na forma  $a = b \cdot \cos \theta + c \cdot \sin \theta$  e também apresenta uma forma analítica alternativa para a solução de equações deste tipo, caso não se deseje utilizar o manipulador simbólico do Matlab ou de outro software similar.

Como notação, o item utiliza as coordenadas dos pontos  $P_{ij} = [x_{P_{ij}}, y_{P_{ij}}, z_{P_{ij}}]$ . O mesmo vale para  $P_{sj}$ .

Para que se demonstrem as deduções para se chegar às equações do tipo  $a = b \cdot \cos \theta + c \cdot \sin \theta$  é feita uma análise de uma das equações, ficando as deduções das outras subentendidas, já que são análogas. Para a terceira equação de distância dos ligamentos passivos, tem-se:

$$h_3^2 = (x_{Ps_3} - x_{Pi_3} - l_3 \cos 60^\circ \cos \theta_3)^2 + (y_{Ps_3} - y_{Pi_3} - l_3 \sin 60^\circ \cos \theta_3)^2 + (z_{Ps_3} - z_{Pi_3} - l_3 \sin \theta_3)^2 \quad (149)$$

Abrindo todos os quadrados e agrupando convenientemente os termos, a equação anterior pode ser escrita como:

$$h_3^2 = (x_{Ps_3} - x_{Pi_3})^2 - 2l_3(x_{Ps_3} - x_{Pi_3})\cos 60^\circ \cos \theta_3 + l_3^2 \cos^2 60^\circ \cos^2 \theta_3 + (y_{Ps_3} - y_{Pi_3})^2 - 2l_3(y_{Ps_3} - y_{Pi_3})\sin 60^\circ \cos \theta_3 + l_3^2 \sin^2 60^\circ \cos^2 \theta_3 + (z_{Ps_3} - z_{Pi_3})^2 - 2l_3(z_{Ps_3} - z_{Pi_3})\sin \theta_3 + l_3^2 \sin^2 \theta_3 \quad (150)$$

$$h_3^2 - (x_{Ps_3} - x_{Pi_3})^2 - (y_{Ps_3} - y_{Pi_3})^2 - (z_{Ps_3} - z_{Pi_3})^2 = -2l_3(x_{Ps_3} - x_{Pi_3})\cos 60^\circ \cos \theta_3 - 2l_3(y_{Ps_3} - y_{Pi_3})\sin 60^\circ \cos \theta_3 - 2l_3(z_{Ps_3} - z_{Pi_3})\sin \theta_3 + l_3^2 [(\sin^2 60^\circ + \cos^2 60^\circ)\cos^2 \theta_3 + \sin^2 \theta_3] \quad (151)$$

Utilizando da geometria a relação que diz que o quadrado do co-seno de um ângulo somado ao quadrado do seno do mesmo ângulo vale uma unidade, o termo em colchetes vale uma unidade. Desta forma, tem-se a equação:

$$h_3^2 - (x_{Ps_3} - x_{Pi_3})^2 - (y_{Ps_3} - y_{Pi_3})^2 - (z_{Ps_3} - z_{Pi_3})^2 - l_3^2 = -2l_3(z_{Ps_3} - z_{Pi_3})\sin\theta_3 - [2l_3(x_{Ps_3} - x_{Pi_3})\cos 60^\circ + 2l_3(y_{Ps_3} - y_{Pi_3})\sin 60^\circ]\cos\theta_3 \quad (152)$$

Dividindo a equação pelo comprimento do ligamento atuado  $l_3$  e substituindo os valores do seno e do co-seno de  $60^\circ$ , chega-se à forma  $a = b \cdot \cos\theta + c \cdot \sin\theta$  proposta no texto, onde, para esta cadeia cinemática:

$$a = \frac{h_3^2 - (x_{Ps_3} - x_{Pi_3})^2 - (y_{Ps_3} - y_{Pi_3})^2 - (z_{Ps_3} - z_{Pi_3})^2 - l_3^2}{l_3}$$

$$b = -[\sqrt{3}(y_{Ps_3} - y_{Pi_3}) + (x_{Ps_3} - x_{Pi_3})]$$

$$c = -2(z_{Ps_3} - z_{Pi_3}) \quad (153)$$

Caso não se deseje utilizar o manipulador simbólico do Matlab, podem ser utilizadas duas relações:

$$\sin x = \frac{2 \tan \frac{x}{2}}{1 + \tan^2 \frac{x}{2}} \quad \cos x = \frac{1 - \tan^2 \frac{x}{2}}{1 + \tan^2 \frac{x}{2}} \quad (154)$$

Dessa maneira, a equação  $a = b \cdot \cos\theta + c \cdot \sin\theta$  pode ser escrita em função apenas da tangente do meio arco:

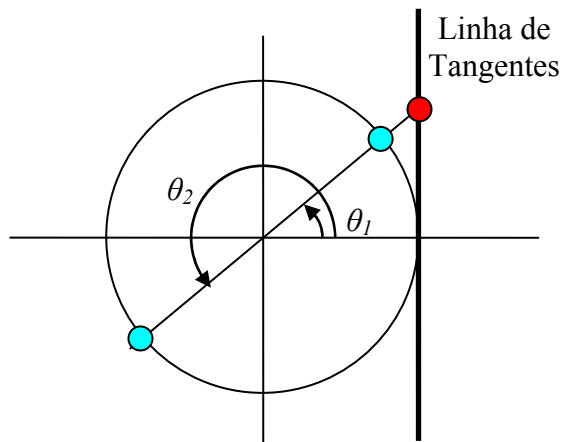
$$a \left(1 + \tan^2 \frac{\theta}{2}\right) = b \left(1 - \tan^2 \frac{\theta}{2}\right) + c \left(2 \tan \frac{\theta}{2}\right) \quad (155)$$

$$(a + b)\tan^2 \frac{\theta}{2} - 2c \tan \frac{\theta}{2} + (a - b) = 0 \quad (156)$$

$$\tan \frac{\theta}{2} = \frac{4c^2 \pm \sqrt{4(a+b)(a-b)}}{2(a+b)} \quad (157)$$

$$\theta = 2 \arctan \left( \frac{4c^2 \pm \sqrt{4(a+b)(a-b)}}{2(a+b)} \right) \quad (158)$$

Vale apenas lembrar que utilizando esta abordagem e considerando um domínio de zero a 360°, quatro soluções são possíveis para a eq. (158), sendo duas oriundas da equação de segundo grau e o fato de se ter um arco tangente que multiplica o número de soluções possíveis por dois. Dessa forma, deve-se ter o cuidado adicional de se analisar as respostas nos diferentes quadrantes dos ângulos de solução. Qual das duas soluções abaixo é a correta para o valor da tangente citado?  $\theta_1$  ou  $\theta_2$ ? O algoritmo terá que considerar esta decisão.



**Figura 76:** Possíveis ângulos de solução de uma tangente.

## APÊNDICE B: DEDUÇÃO DA EQUAÇÃO GERAL DE UM ELIPSÓIDE DE DIMENSÃO N

Este item apresenta a dedução da equação geral do elipsóide partindo dos princípios de álgebra linear. Não se encontrou esta definição em livros ou trabalhos, contudo é de vital importância para o capítulo de análise da matriz Jacobiano.

Na álgebra linear é comum se utilizar expressões da forma  $a_1x_1 + a_2x_2 + \dots + a_nx_n$ , onde  $a_1, a_2, \dots, a_n$  são constantes fixas e  $x_1, x_2, \dots, x_n$  são variáveis, o que resulta numa **forma linear de  $R^n$**  ( $R$  é o universo dos números reais), formas estas que representam hiperplanos. Contudo, para a equação do elipsóide (ou qualquer outra cônica) se deseja focar nas **formas quadráticas de  $R^n$** , ou seja, expressões da forma  $a_1x_1^2 + a_2x_2^2 + \dots + a_nx_n^2 +$  termos  $a_kx_ix_j$ , onde  $i$  e  $j$  variam de 1 a  $n$ .

No espaço  $R^2$  define-se uma seção cônica como sendo a curva oriunda do corte de um cone circular reto por um plano. Essas curvas podem ser de três tipos: elipse, parábola ou hipérbole (lembrando que a circunferência é um caso particular da elipse onde os seus centros focais são coincidentes). As formas quadráticas de  $R^2$  surgem naturalmente do estudo das cônicas e estas possuem como equação (agrupados os termos  $xy$  e  $yx$ ):

$$ax^2 + 2bxy + cy^2 + dx + ey + f = 0 \quad (159)$$

As variáveis do espaço  $R^2$  são  $x$  e  $y$ , sendo  $a, b, c, d, e$  e  $f$  constantes. Se os termos lineares forem zero ( $d = e = 0$ ) tem-se a forma de *cônica central* ou *cônica reduzida*, resultando numa equação que só não define a parábola. Essa equação é a equação da forma quadrática do espaço  $R^2$  igualada a uma constante igual a  $-f$ .

$$ax^2 + 2bxy + cy^2 = -f \quad (160)$$

Se, além disso, o termo misto também valer zero, ou seja,  $b = 0$ , então se tem uma *cônica central em posição canônica*, pois os eixos principais da cônica estarão

em  $x$  e  $y$  e ela estará centrada no ponto  $[0, 0]$ . Costuma-se também dividir a equação por  $-f$  resultando em:

$$a'x^2 + c'y^2 = 1 \quad (161)$$

Por último, se ambas as constantes são positivas, a cônica representa uma elipse ou circunferência, se uma constante é negativa e a outra positiva, a cônica representa uma hipérbole. Nesses casos a cônica não é degenerada.

Os mesmos conceitos de cônicas podem ser expandidos para múltiplas dimensões. No espaço  $R^n$  a circunferência se torna uma esfera n-dimensional, a elipse um elipsóide n-dimensional e assim por diante. Todos oriundos do corte de um cone circular reto de múltiplas dimensões por um *hiperplano*.

Para o caso de uma dimensão igual a três, a forma quadrática pode ser escrita conforme a expressão a seguir, onde os termos  $x_i x_j$  e  $x_j x_i$  foram agrupados:

$$a_1 x_1^2 + a_2 x_2^2 + a_3 x_3^2 + 2a_4 x_1 x_2 + 2a_5 x_1 x_3 + 2a_6 x_2 x_3 \quad (162)$$

Se a forma quadrática for escrita na forma matricial, chega-se à seguinte expressão (novamente, o conceito pode ser expandido para  $n$  dimensões):

$$\begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \begin{bmatrix} a_1 & a_4 & a_5 \\ a_4 & a_2 & a_6 \\ a_5 & a_6 & a_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \mathbf{x}^T \mathbf{A} \mathbf{x} \quad (163)$$

A matriz  $\mathbf{A}$  é simétrica pela própria definição da forma, o que também a torna válida para os casos apresentados neste trabalho.

Prosseguindo com a dedução, utiliza-se agora uma transformação tal que a matriz característica se torne diagonal, o que dá um direcionamento a uma transformação para uma cônica central em posição canônica. Desta forma todos os termos mistos deixam de existir e apenas os termos quadráticos aparecem. A transformação deve ser através da mudança de variáveis  $\mathbf{x} = \mathbf{P}\mathbf{y}$ , o que resulta em  $\mathbf{x}^T \mathbf{A} \mathbf{x} = \mathbf{y}^T (\mathbf{P}^T \mathbf{A} \mathbf{P}) \mathbf{y}$ . Para que ocorra uma diagonalização, a matriz  $\mathbf{P}$  deve ser



invertível e a matriz quadrada  $\mathbf{A}$  deve possuir autovalores linearmente independentes. Neste caso,  $\mathbf{P}$  é a matriz de autovetores de  $\mathbf{A}$ .

Para demonstrar esta última afirmação, propõe-se que exista uma matriz  $\mathbf{P}$  tal que a transformação  $\mathbf{P}^{-1}\mathbf{A}\mathbf{P}$  resulte numa matriz diagonal  $\mathbf{D}$ . Sejam  $\mathbf{P}$  e  $\mathbf{D}$  definidas como as seguintes matrizes:

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix} = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_n] \quad \mathbf{D} = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \quad (164)$$

Assim, a demonstração de  $\mathbf{D} = \mathbf{P}^{-1}\mathbf{A}\mathbf{P}$  ou  $\mathbf{P}\mathbf{D} = \mathbf{A}\mathbf{P}$  é dada por:

$$\mathbf{P}\mathbf{D} = [\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_n] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} = [\lambda_1\mathbf{p}_1 \quad \lambda_2\mathbf{p}_2 \quad \cdots \quad \lambda_n\mathbf{p}_n] \quad (165)$$

$$\mathbf{A}\mathbf{P} = \mathbf{A}[\mathbf{p}_1 \quad \mathbf{p}_2 \quad \cdots \quad \mathbf{p}_n] = [\mathbf{A}\mathbf{p}_1 \quad \mathbf{A}\mathbf{p}_2 \quad \cdots \quad \mathbf{A}\mathbf{p}_n] \quad (166)$$

$$[\lambda_1\mathbf{p}_1 \quad \lambda_2\mathbf{p}_2 \quad \cdots \quad \lambda_n\mathbf{p}_n] = [\mathbf{A}\mathbf{p}_1 \quad \mathbf{A}\mathbf{p}_2 \quad \cdots \quad \mathbf{A}\mathbf{p}_n] \quad (167)$$

$$\lambda_1\mathbf{p}_1 = \mathbf{A}\mathbf{p}_1, \lambda_2\mathbf{p}_2 = \mathbf{A}\mathbf{p}_2, \dots, \lambda_n\mathbf{p}_n = \mathbf{A}\mathbf{p}_n \quad (168)$$

Portanto, para  $\mathbf{P}\mathbf{D} = \mathbf{A}\mathbf{P}$  chega-se ao fato de que  $\mathbf{P}$  é a matriz de autovetores de  $\mathbf{A}$  e  $\mathbf{D}$  é a matriz diagonal dos autovalores de  $\mathbf{A}$ . Esta, na verdade, é a prova de cálculo de autovalores e autovetores, onde vale que ( $\mathbf{I}$  é a matriz identidade de mesma dimensão que  $\mathbf{A}$ ):

$$\mathbf{p}_i(\mathbf{A} - \lambda_i\mathbf{I}) = 0 \quad (169)$$

Faz-se apenas a ressalva que em sendo  $\mathbf{P}$  uma matriz ortogonal,  $\mathbf{P}^{-1} = \mathbf{P}^T$ . Esta relação onde a transposta é igual a inversa em matrizes ortogonais não é demonstrada aqui, mas pode ser estudada na maioria dos livros de álgebra linear. Voltando à dedução do elipsóide, tem-se que:

$$\mathbf{x}^T \mathbf{A} \mathbf{x} = \mathbf{y}^T (\mathbf{P}^T \mathbf{A} \mathbf{P}) \mathbf{y} = [y_1 \quad y_2 \quad \cdots \quad y_n] \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (170)$$

$$\mathbf{y}^T (\mathbf{P}^T \mathbf{A} \mathbf{P}) \mathbf{y} = \lambda_1 y_1^2 + \lambda_2 y_2^2 + \cdots + \lambda_n y_n^2 \quad (171)$$

$$\mathbf{y}^T (\mathbf{P}^T \mathbf{A} \mathbf{P}) \mathbf{y} = \frac{y_1^2}{\left(\frac{1}{\sqrt{\lambda_1}}\right)^2} + \frac{y_2^2}{\left(\frac{1}{\sqrt{\lambda_2}}\right)^2} + \cdots + \frac{y_n^2}{\left(\frac{1}{\sqrt{\lambda_n}}\right)^2} \quad (172)$$

Se a última equação for escrita agora como uma forma cônica central em posição canônica, tem-se que ( $k$  é uma constante):

$$\frac{y_1^2}{\left(\frac{1}{\sqrt{\lambda_1}}\right)^2} + \frac{y_2^2}{\left(\frac{1}{\sqrt{\lambda_2}}\right)^2} + \cdots + \frac{y_n^2}{\left(\frac{1}{\sqrt{\lambda_n}}\right)^2} = k \quad (173)$$

$$\frac{y_1^2}{\left(\sqrt{\frac{k}{\lambda_1}}\right)^2} + \frac{y_2^2}{\left(\sqrt{\frac{k}{\lambda_2}}\right)^2} + \cdots + \frac{y_n^2}{\left(\sqrt{\frac{k}{\lambda_n}}\right)^2} = 1 \quad (174)$$

Vale ressaltar que para o caso deste trabalho, as matrizes características representadas por  $\mathbf{A}$  são positivas definidas ou positivas semi-definidas. Dessa forma todos os autovalores são positivos ou iguais a zero o que implica na impossibilidade

de se terem termos negativos, ou seja, não se pode definir outra cônica que não um elipsóide de  $n$  dimensões (ou uma esfera se todos os autovalores forem iguais).

Agora se retorna ao espaço  $R^n$  original onde as variáveis são dadas por  $x_1, x_2, \dots, x_n$ . Foi definido que  $\mathbf{x} = \mathbf{P} \mathbf{y}$ , então vale a inversa onde  $\mathbf{y} = \mathbf{P}^{-1} \mathbf{x}$ . Se os autovalores de  $\mathbf{A}$  formam uma base ortogonal (na verdade *ortonormal*, já que possuem também módulo unitário), então  $\mathbf{y} = \mathbf{P}^T \mathbf{x}$ . Dessa forma:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} p_{11} & p_{21} & \cdots & p_{n1} \\ p_{12} & p_{22} & \cdots & p_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ p_{1n} & p_{2n} & \cdots & p_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (175)$$

Também é bom lembrar que cada coluna de  $\mathbf{P}$  é um autovetor de  $\mathbf{A}$  e cada linha de  $\mathbf{P}^T$  é um autovetor de  $\mathbf{A}$ . Substituindo a transformação inversa de  $\mathbf{y}$  para  $\mathbf{x}$  na forma quadrática do elipsóide central canônico, tem-se que:

$$\frac{\left( \sum_{i=1}^n p_{i1} x_i \right)^2}{\left( \sqrt{\frac{k}{\lambda_1}} \right)^2} + \frac{\left( \sum_{i=1}^n p_{i2} x_i \right)^2}{\left( \sqrt{\frac{k}{\lambda_2}} \right)^2} + \cdots + \frac{\left( \sum_{i=1}^n p_{in} x_i \right)^2}{\left( \sqrt{\frac{k}{\lambda_n}} \right)^2} = 1 \quad (176)$$

$$\sum_{j=1}^n \left[ \left( \sqrt{\frac{\lambda_j}{k}} \right)^2 \left( \sum_{i=1}^n p_{ij} x_i \right)^2 \right] = 1 \quad (177)$$

Da mesma maneira que nas elipses do espaço  $R^2$ , os elipsóides do espaço  $R^n$  também possuem eixos principais cujos tamanhos valem  $2\sqrt{\frac{k}{\lambda_i}}$ , onde  $i = 1, 2, \dots, n$ .

Encerram-se assim, as deduções para se chegar à equação geral do elipsóide de  $n$  dimensões, utilizada no texto da dissertação.

## APÊNDICE C: FUNDAMENTOS MATEMÁTICOS PARA O SIMULADOR VIRTUAL

Este item contempla a teoria matemática envolvida na construção do simulador virtual. São tratados os tópicos: transformação de coordenadas, coordenadas homogêneas, translações e rotações tridimensionais em torno de um eixo qualquer, ângulos de *roll – pitch – yaw* e interpolações lineares e circulares. Como fontes bibliográficas recomendam-se **Cabral (2003)**, **Tsai (1999)** e **Tsuzuki (2003)**.

### Transformação de Coordenadas

Sejam dois sistemas de eixos coordenados:  $O\text{-}XYZ$  e  $O'\text{-}X'Y'Z'$  e um ponto  $P$  cujas coordenadas nos sistemas são respectivamente  $[P_x, P_y, P_z]$  e  $[P_{x'}, P_{y'}, P_{z'}]$ . Assim, entende-se por transformação de coordenadas como sendo as operações que fazem por relacionar as coordenadas de  $P$  nos dois sistemas de coordenadas. Geralmente são realizadas por uma simples multiplicação de matrizes.

Seja o vetor que aponta para  $O'$  partindo de  $O$  definido por  $\mathbf{x}_0$ . Sejam também os versores dos sistemas de coordenadas apresentados  $\mathbf{i}, \mathbf{j}, \mathbf{k}$  e  $\mathbf{i}', \mathbf{j}', \mathbf{k}'$ , onde estes três últimos são os versores de  $O'\text{-}X'Y'Z'$  escritos em coordenadas de  $O\text{-}XYZ$ .

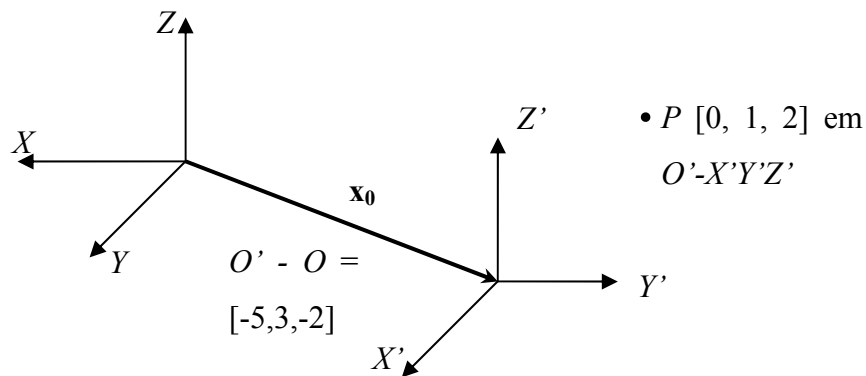
A partir do que se discutiu, a seguinte equação pode ser escrita:

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \mathbf{x}_0 + \begin{bmatrix} \uparrow & \uparrow & \uparrow \\ \mathbf{i}' & \mathbf{j}' & \mathbf{k}' \\ \downarrow & \downarrow & \downarrow \end{bmatrix} \cdot \begin{bmatrix} P_{x'} \\ P_{y'} \\ P_{z'} \end{bmatrix} \quad (178)$$

A matriz formada pelos versores de  $O'\text{-}X'Y'Z'$  é chamada de Matriz de Transformação de Coordenadas. A notação das matrizes procederá da seguinte forma:  ${}^b\mathbf{M}_a$ , ou seja, a matriz  $\mathbf{M}$  que escreve pontos cujas coordenadas estão em  $a$  nas coordenadas  $b$ . Em outras palavras, quando a matriz é multiplicada por um ponto cujas coordenadas estão escritas em  $a$  tem-se como resultado o mesmo ponto com suas coordenadas escritas em  $b$ .

Vale lembrar também que essas matrizes são ortonormais, ou seja, elas possuem como propriedade matemática o fato de que suas transpostas são iguais a suas inversas.

Como o que foi escrito se torna um pouco difícil de entender, apresenta-se um exemplo numérico a seguir:



**Figura 77:** Exemplo numérico para uma transformação de coordenadas.

Para que se tenham as coordenadas de  $P$  escritas no sistema  $O-XYZ$ , segue-se o que foi ilustrado anteriormente. Os versores  $\mathbf{i}'$ ,  $\mathbf{j}'$  e  $\mathbf{k}'$  do sistema  $O'-X'Y'Z'$  escritos no sistema  $O-XYZ$  são entendidos como:  $\mathbf{i}' = [0 \ 1 \ 0]^T$ ,  $\mathbf{j}' = [-1 \ 0 \ 0]^T$  e, por último,  $\mathbf{k}' = [0 \ 0 \ 1]^T$ . Assim:

$$\begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} = \begin{bmatrix} -5 \\ 3 \\ -2 \end{bmatrix} + \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 2 \end{bmatrix} = \begin{bmatrix} -5 \\ 3 \\ -2 \end{bmatrix} + \begin{bmatrix} -1 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} -6 \\ 3 \\ 0 \end{bmatrix} \quad (179)$$

A matriz de transformação nesse caso é  ${}^{O-XYZ}\mathbf{M}_{O'-X'Y'Z'}$ . Atentando para a mesma, também é possível realizar o caminho inverso do que foi apresentado e testar a propriedade de ortonormalidade.

Continuando ainda no tema, vão-se explorar casos onde os sistemas possuem origens coincidentes, mas sofreram rotação. Seja agora  $\mathbf{P}$  o vetor que aponta para  $P$  partindo de  $O$  ou  $O'$ , já que são coincidentes.  $\mathbf{P}$  está escrito em  $O-XYZ$ . Seja também  $\mathbf{P}'$  o mesmo vetor, mas escrito em  $O'-X'Y'Z'$ . Tem-se que:

$$\begin{bmatrix} P_X \\ P_Y \\ P_Z \end{bmatrix} = \begin{bmatrix} \mathbf{P} \cdot \mathbf{i} \\ \mathbf{P} \cdot \mathbf{j} \\ \mathbf{P} \cdot \mathbf{k} \end{bmatrix} = \begin{bmatrix} \mathbf{P}' \cdot \mathbf{i} \\ \mathbf{P}' \cdot \mathbf{j} \\ \mathbf{P}' \cdot \mathbf{k} \end{bmatrix} = \begin{bmatrix} \mathbf{i}' \cdot \mathbf{i} & \mathbf{j}' \cdot \mathbf{i} & \mathbf{k}' \cdot \mathbf{i} \\ \mathbf{i}' \cdot \mathbf{j} & \mathbf{j}' \cdot \mathbf{j} & \mathbf{k}' \cdot \mathbf{j} \\ \mathbf{i}' \cdot \mathbf{k} & \mathbf{j}' \cdot \mathbf{k} & \mathbf{k}' \cdot \mathbf{k} \end{bmatrix} \cdot \begin{bmatrix} P_{X'} \\ P_{Y'} \\ P_{Z'} \end{bmatrix} \quad (180)$$

### Coordenadas Homogêneas

Quando se trabalha com representações computacionais 3D de objetos é comum utilizar coordenadas homogêneas. Esse tipo de coordenada representa um ponto  $[X \ Y \ Z]$  no espaço real através de quatro elementos  $[x' \ y' \ z' \ w]$ , tal que:

$$[X \ Y \ Z]^T = \begin{bmatrix} x' & y' & z' \\ w & w & w \end{bmatrix}^T \quad (181)$$

A representação homogênea permite que alguns pontos no espaço muito distantes sejam representados facilmente como, por exemplo, um ponto no infinito no eixo  $X$ . Ele pode ser representado por  $x' = 1$  e  $w = 0$ . Outra propriedade interessante é que existe uma combinação infinita de elementos homogêneos que podem representar o mesmo ponto no espaço real.

Esse tipo de abordagem também facilita a execução de operações básicas como translações e rotações, operações estas fundamentais a um simulador.

### Translação Tridimensional

A operação de translação é aquela de mover um ponto ou objeto linearmente no espaço até um outro ponto definido. Seja um ponto  $[X \ Y \ Z]$  qualquer no espaço que se deseja transladar para o ponto  $[X' \ Y' \ Z']$ . As distâncias a serem percorridas são dadas por  $[X' \ Y' \ Z'] - [X \ Y \ Z] = [t_x \ t_y \ t_z]$ . Em coordenadas homogêneas essa tarefa se dá através de uma simples multiplicação de matrizes:

$$\begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} x'+t_x \\ y'+t_y \\ z'+t_z \\ 1 \end{bmatrix} \quad (182)$$

### Rotação Tridimensional Simples

Por rotação tridimensional entende-se o movimento circular que um ponto faz em torno de um eixo. A trajetória executada será perpendicular a esse eixo. O termo rotação simples é dado para aqueles casos particulares onde os próprios eixos  $X$ ,  $Y$  ou  $Z$  são aqueles onde ao seu redor será feita a rotação. Analogamente à translação, busca-se uma matriz de transformação.

Seja então um exemplo onde um ponto pertencente a  $X$  tem como eixo de rotação  $Z$ . O ângulo da rotação vale  $\beta$  e o raio de rotação vale  $r$ . O raio é igual à coordenada  $X$  do ponto que sofrerá rotação. Agora partindo do novo ponto obtido, faz-se uma nova rotação idêntica e de ângulo  $\alpha$ . Tendo-se as coordenadas iniciais e finais do ponto para com a nova rotação dadas respectivamente por  $[X, Y]$  e  $[X', Y']$ , pode-se escrever as seguintes equações:

$$\begin{aligned} X' &= r \cdot \cos(\alpha + \beta) = r \cdot \cos \beta \cos \alpha - r \cdot \sin \beta \sin \alpha \\ Y' &= r \cdot \sin(\alpha + \beta) = r \cdot \cos \beta \sin \alpha + r \cdot \sin \beta \cos \alpha \end{aligned} \quad (183)$$

Lembrando que:

$$\begin{aligned} X &= r \cdot \cos \beta \\ Y &= r \cdot \sin \beta \end{aligned} \quad (184)$$

Chega-se, finalmente, à formula genérica de rotações em torno de  $Z$ :

$$\begin{aligned} X' &= X \cdot \cos \alpha - Y \cdot \sin \alpha \\ Y' &= X \cdot \sin \alpha + Y \cdot \cos \alpha \end{aligned} \quad (185)$$

Agora se expande o conceito para as três dimensões e utilizam-se matrizes para representar estas rotações:

$$\mathbf{R}_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (186)$$

$$\mathbf{R}_{y,\alpha} = \begin{bmatrix} \cos \alpha & 0 & \sin \alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \alpha & 0 & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (187)$$

$$\mathbf{R}_{z,\alpha} = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (188)$$

### **Rotação em Torno de um Eixo Qualquer (Rotação 3D Genérica)**

Uma maneira de se executar uma rotação em torno de um eixo qualquer é se valer de uma sucessão de operações mais simples, como aquelas apresentadas anteriormente. Seja o vetor  $\mathbf{n}$  de módulo unitário definido no sistema de coordenadas  $O-XYZ$ . O vetor é definido pelos ângulos  $\alpha$  e  $\beta$ , onde o primeiro refere-se ao ângulo que o eixo  $X$  faz com o plano  $Z\mathbf{n}$  e o segundo refere-se ao ângulo entre  $Z$  e  $\mathbf{n}$ . Tem-se então o caso onde se deseja realizar uma rotação de um ponto em torno deste vetor (eixo) de um ângulo de valor  $\theta$ . Por último, tem-se que o “centro da rotação” é o ponto  $C$  de coordenadas  $[c_x \ c_y \ c_z]$ , ponto este por onde passa o vetor  $\mathbf{n}$  (já que numa rotação 3D um ponto jamais pode ser o centro e sim um eixo). Necessita-se então de uma matriz de rotação  $\mathbf{R}_{\mathbf{n},\theta}$  definida a seguir:

$$\mathbf{R}_{\mathbf{n},\theta} = \mathbf{T}_C \mathbf{R}_{z,\alpha} \mathbf{R}_{y,\beta} \mathbf{R}_{z,\theta} \mathbf{R}_{y,-\beta} \mathbf{R}_{z,-\alpha} \mathbf{T}_{-C} \quad (189)$$



Onde  $\mathbf{R}_{n,\theta}$  segue a seguinte linha de raciocínio:

1. Deve-se transladar o centro de rotação para a origem do sistema ( $\mathbf{T}_C$ )
2. As rotações representadas pelos ângulos  $\alpha$  e  $\beta$  são eliminadas ( $\mathbf{R}_{Y,-\beta}\mathbf{R}_{Z,-\alpha}$ )
3. Executa-se uma simples rotação do ponto ao redor do eixo Z de  $\theta$  ( $\mathbf{R}_{Z,\theta}$ )
4. Devolvem-se ao ponto as suas características originais ( $\mathbf{T}_C\mathbf{R}_{Z,\alpha}\mathbf{R}_{Y,\beta}$ )

$$\mathbf{T}_C = \begin{bmatrix} 1 & 0 & 0 & c_x \\ 0 & 1 & 0 & c_y \\ 0 & 0 & 1 & c_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{T}_{-C} = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 1 & -c_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (190)$$

$$\mathbf{R}_{Z,\alpha} = \begin{bmatrix} \cos \alpha & -\sin \alpha & 0 & 0 \\ \sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{R}_{Z,-\alpha} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 & 0 \\ -\sin \alpha & \cos \alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (191)$$

$$\mathbf{R}_{Y,\beta} = \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{e} \quad \mathbf{R}_{Y,-\beta} = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (192)$$

$$\mathbf{R}_{Z,\theta} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (193)$$

Segue uma ilustração dos ângulos e dos vetores mencionados nesse caso de uma rotação tridimensional genérica.

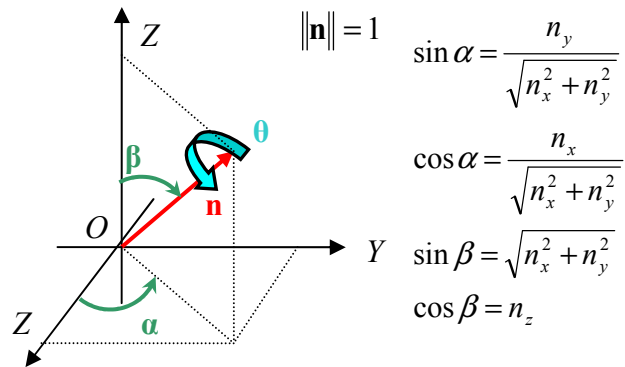


Figura 78: Figura ilustrativa da rotação tridimensional em torno de um eixo qualquer.

### Ângulos de Roll – Pitch – Yaw

Uma outra maneira de se descrever a orientação de corpos rígidos utiliza um conjunto independente de três ângulos, chamados de *Roll – Pitch – Yaw*. A sequência de operações executadas corresponde à multiplicação de matrizes apresentada a seguir.

$$\mathbf{R}_{\phi, \theta, \varphi} = \mathbf{R}_{Z, \varphi} \cdot \mathbf{R}_{Y, \theta} \cdot \mathbf{R}_{X, \phi} \quad (194)$$

$$\mathbf{R}_{\phi, \theta, \varphi} = \begin{bmatrix} \cos \phi & -\sin \phi & 0 & 0 \\ \sin \phi & \cos \phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \varphi & -\sin \varphi & 0 \\ 0 & \sin \varphi & \cos \varphi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (195)$$

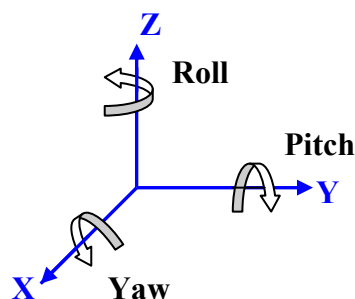


Figura 79: Ângulos de Roll – Pitch – Yaw.

### Interpolação Linear e Avanço Linear

A tarefa mais simples que se imagina num simulador é que o ponto de interesse possa seguir uma linha reta. Seja o ponto de destino  $P_d = [x_d, y_d, z_d]$ , o ponto de partida  $P_p = [x_p, y_p, z_p]$  e o passo  $p_{il}$ . Assim, tem-se que:

- O vetor que indica a direção e sentido de movimentos é dado por  $P_d - P_p$ ;
- O número de passos é igual a  $\|P_d - P_p\| / p_{il}$  e todos os passos valem  $p_{il}$ , exceto o último passo, que pode ser  $\leq p_{il}$ ;
- A cada iteração a posição do ponto de interesse terá o valor de:

$$P_p + \frac{P_d - P_p}{\|P_d - P_p\|} \cdot p_{il} \cdot n_{iteração}, \text{ onde } n_{iteração} \text{ varia de 1 até o número de passos.}$$

É importante lembrar que um algoritmo desses para um simulador é diferente do que um para o sistema de controle. Não trataremos aqui de algoritmos de Bresenhan que lidam com a resolução da máquina (Tsuzuki, 2003).

No caso do avanço linear, utiliza-se o mesmo princípio, contudo, se utiliza um sistema de coordenadas diferente do global. Utiliza-se um preso à ferramenta e solidário à plataforma móvel.

### Interpolação Circular

O algoritmo de interpolação circular, que também está presente em praticamente todas as máquinas do gênero, deve ser totalmente genérico, assim o arco de círculo pode ser traçado em qualquer plano e qualquer tamanho. Para cumprir tal tarefa é preciso definir alguns parâmetros: ponto de partida  $P_p$ , ponto de destino  $P_d$ , ponto pelo qual o círculo deve passar  $P_m = [x_m, y_m, z_m]$  e passo  $p_{ic}$ .

Se os três pontos requeridos não estão alinhados (apenas três pontos não alinhados são capazes de definir um único plano) e nenhum deles é coincidente com outro, existe um único ponto  $P_{cc} = [x_{cc}, y_{cc}, z_{cc}]$  que é o centro desta circunferência. Para encontrar  $P_{cc}$ , deve-se simplesmente relacionar a distância de qualquer um dos pontos de referência ao centro da circunferência, que é o raio  $r$ :

$$r = \|P_p - P_{cc}\| = \|P_m - P_{cc}\| = \|P_d - P_{cc}\| \quad (196)$$

Subtraindo uma equação das outras na seqüência de equações anterior, chega-se ao sistema de equações detalhado a seguir:

$$\begin{bmatrix} 2(x_m - x_p) & 2(y_m - y_p) & 2(z_m - z_p) \\ 2(x_d - x_p) & 2(y_d - y_p) & 2(z_d - z_p) \\ 2(x_m - x_d) & 2(y_m - y_d) & 2(z_m - z_d) \end{bmatrix} \cdot \begin{bmatrix} x_{cc} \\ y_{cc} \\ z_{cc} \end{bmatrix} = \begin{bmatrix} \|P_m\|^2 - \|P_p\|^2 \\ \|P_d\|^2 - \|P_p\|^2 \\ \|P_m\|^2 - \|P_d\|^2 \end{bmatrix} \quad (197)$$

A solução do sistema resulta na obtenção do centro da circunferência e, por conseguinte, o raio da mesma. Para que se possa então desenhar a circunferência, segue-se a seguinte linha de raciocínio:

- Definem-se os vetores  $\mathbf{V}_p = P_p - P_{cc}$ ,  $\mathbf{V}_d = P_d - P_{cc}$  e  $\mathbf{V}_m = P_m - P_{cc}$ ;
- Calcula-se o eixo normal à circunferência, dado por  $\mathbf{N} = \mathbf{V}_p \times \mathbf{V}_d$ . Caso o vetor seja nulo, ou seja,  $\mathbf{V}_p$  e  $\mathbf{V}_d$  formam um ângulo de  $180^\circ$ , utiliza-se  $\mathbf{N} = \mathbf{V}_p \times \mathbf{V}_m$ ;
- Verifica-se se a circunferência excedeu  $\frac{1}{2}$  de uma volta completa, pois se isso ocorreu,  $\mathbf{N} = -\mathbf{V}_p \times \mathbf{V}_d$ . Essa verificação é complexa, pois cálculos de álgebra linear como produto interno entre vetores, só são capazes de caracterizar ângulos de até  $180^\circ$ . No entanto, existe uma rotina que pode ser seguida:
  - Calculam-se os ângulos entre  $\mathbf{V}_d$  e  $\mathbf{V}_p$  e entre  $\mathbf{V}_m$  e  $\mathbf{V}_p$ ;
  - Utilizando esses ângulos, verifica-se para a direção do vetor normal qual o sentido que produz as localizações reais de  $P_d$  e  $P_m$  (utilizar a matriz  $\mathbf{R}_{n,0}$ ).
- Quanto ao passo, deve-se lembrar que no simulador a circunferência é aproximada por um conjunto de retas. O simulador utiliza como passo um valor em graus que indica quanto do arco é desenhado a cada iteração (este pequeno arco é aproximado por uma reta), mas pode-se utilizar uma outra aproximação. Assim o novo passo angular em radianos será  $p_{tc-\text{radianos}} = L_{\text{arco}} / R$ .

- Utilizando o vetor normal já caracterizado e o passo definido, traça-se a circunferência.

O único caso não tratado no algoritmo é o caso de uma circunferência completa, no entanto, é bastante fácil. Define-se um arco qualquer e o executa. Após fazê-lo o ponto antigo de destino torna-se o de partida e o antigo ponto de partida o de destino. O centro e o vetor normal já estão definidos, então se gira o ponto  $P_m$  com relação ao vetor  $(P_{cc} - P_p)$  com um ângulo de  $180^\circ$  e executam-se novamente as últimas etapas da rotina.

## APÊNDICE D: FUNDAMENTOS TEÓRICOS PARA OTIMIZAÇÃO

Este item apresenta alguns dos fundamentos teóricos básicos da otimização, como os elementos necessários e tipos de algoritmos.

### D.1. Elementos Principais de uma Otimização

Este item tem por objetivo descrever brevemente os componentes básicos de uma otimização. Para uma referência mais completa, pode-se consultar **Silva (2003)**, texto do qual foi baseado grande parte deste descritivo. Em todos os problemas de otimização estão presentes quatro elementos principais:

- Variáveis de Projeto
- Função Objetivo
- Restrições
- Domínio

#### Variáveis de Projeto

Variáveis de projeto são os parâmetros do problema que podem ser alterados para se chegar a uma solução ótima. Por exemplo, num robô os comprimentos dos seus ligamentos podem ser considerados variáveis de projeto.

Existe também uma classificação para as variáveis que podem ser *contínuas* ou *discretas*. As contínuas podem assumir qualquer valor. As discretas podem assumir apenas alguns. Toma-se então como exemplo um robô que para projetá-lo, dentre outras variáveis, pode-se alterar o comprimento dos ligamentos e o material (retratado pelo coeficiente de elasticidade). O comprimento dos ligamentos pode, a princípio, ter qualquer valor, contudo o material fará parte de uma tabela de parâmetros, não podendo ser diferente daqueles definidos.

Dentre as variáveis contínuas existe ainda uma classificação: *variáveis de parâmetro distribuído* e *de parâmetro discreto*. Um exemplo simples que demonstra

o conceito é o caso da seção transversal de uma viga engastada. A seção, na realidade, pode variar de maneira contínua à medida que se aproxima ou se afasta do ponto de engastamento. Todavia, utilizando um software de elementos finitos, essa função contínua será discretizada ao longo do comprimento da viga. A essa última dá-se o nome de *variável contínua do tipo parâmetro discreto*.

### **Função Objetivo**

Essa é a medida de eficiência. Função das variáveis de projeto, é responsável por quantificar o que realmente se quer otimizar. É muito importante definir corretamente a função objetivo, pois o sucesso ou o fracasso de uma otimização está diretamente ligado a ela.

Dentre as maiores dificuldades de se preparar uma função objetivo estão os casos onde se deve quantificar algo que é intangível através de uma função matemática. Por exemplo, quantificar um desenho de um automóvel, que levará em consideração além do desempenho (coeficiente de arrasto e outros), fatores como estética, aceitação do público e inovação. Este caso também é, na verdade, um caso de função *multiobjetivo*, pois são desejáveis mais de um objetivo.

Na solução de funções *multiobjetivo* costuma-se empregar uma das seguintes técnicas: (1) um dos objetivos, no caso o mais significativo, é mantido na função e os outros são transformados em restrições; (2) utiliza-se o conceito de Ótimo-Pareto, onde são mantidos os objetivos, mas são lhes dados pesos diferentes de forma a compor 100% do objetivo global. Alguns cuidados devem ser tomados para que cada objetivo tenha a mesma ordem de grandeza (o que exige geralmente normalizações).

Contudo, mesmo em casos puramente matemáticos, ainda surgem algumas questões: como se pode definir o desempenho de um robô? Sua velocidade máxima de operação? Número de singularidades? A relação carga máxima pela potência de atuação? O que se deve fazer para alcançar este resultado? Minimizar a força nos atuadores? Estas perguntas são de difícil resposta, mas que devem se colocadas em uma equação que é função de uma ou mais variáveis de projeto.

Deve-se ter em mente que a forma óbvia de se escrever um problema nem sempre é a melhor. Ao invés de se tentar otimizar, por exemplo, o módulo de uma

função, pode ser mais interessante otimizar a função ao quadrado, o que também facilita o algoritmo que pode ter a sua disposição as fórmulas analíticas da derivada e da derivada segunda da função (importância essa que será apresentada mais adiante).

### Restrições

Restrições são limitações impostas ao problema de otimização. Existem três tipos de restrições: *igualdades*, *desigualdades* e *restrições laterais*. Elas também podem ser *locais* ou *globais* e *lineares* ou *não-lineares*.

Sendo as variáveis de projeto dadas pelo conjunto  $x = \{x_1, x_2, x_3, \dots, x_m\}$ , tem-se que as:

- Igualdades são funções do tipo  $h_k(x) = 0$ , com  $k = 1, 2, \dots, n$  *igualdades*;
- Desigualdades são funções do tipo  $g_j(x) \geq 0$ , com  $j = 1, 2, \dots, n$  *desigualdades*;
- Restrições laterais são limites do tipo  $x_{\min i} \leq x_i \leq x_{\max i}$ , com  $i = 1, 2, 3, \dots, m$ .

Quanto ao fato de uma restrição ser local ou global, tem-se que:

- *Restrições locais* são aquelas que se aplicam apenas à parte do domínio. Por exemplo, para uma dada porção do espaço (ou mesmo apenas um ponto), a plataforma móvel deve poder girar 45° ou mais em qualquer direção.
- *Restrições globais* são restrições que se aplicam à estrutura como um todo. Por exemplo, a rigidez do robô não deve ser menor que certo valor exigido.

### Domínio

Existe uma divisão do conceito domínio: *viável* e *inviável*. O domínio viável é o espaço (multidimensional) de busca para encontrar a(s) resposta(s) ótima(s), sendo o espaço cercado por restrições e que as respeita. O domínio inviável é aquele que está fora do espaço definido pelas restrições, impossibilitando a validade de qualquer resposta.



## D.2. Métodos de Solução de Problemas de Otimização

Dentre os muitos métodos de solução de problemas de otimização, existem três mais comuns: gráficos, analíticos e numéricos.

Os métodos gráficos permitem a solução de problemas com até duas variáveis e consistem na definição da função objetivo, das restrições e do domínio viável para obtenção da resposta. Mesmo problemas simples podem não ser passíveis de solução por este método, o que o destina mais ao aprendizado que a aplicações práticas.

Métodos analíticos como o cálculo diferencial e o variacional também permitem a solução apenas de problemas simples, contudo podem servir para realizar estudos ou análises conceituais e validar soluções de algoritmos numéricos.

Por último, existem os métodos numéricos, que serão aqueles utilizados aqui. Existem dois tipos: específicos e gerais. Os específicos geralmente exigem uma formulação empírica para cada problema, o que os torna pouco práticos. Nos métodos gerais, ainda existe uma divisão: *métodos baseados em programação matemática* e *métodos probabilísticos*. A diferença principal entre esses dois últimos métodos é o fato de que o primeiro busca essencialmente um mínimo (ou máximo) local que poderá ser ou não o mínimo global e o segundo foge destas prisões. A Tabela 11 apresenta algumas diferenças entre esses métodos.

Existem muitos métodos baseados em programação matemática e alguns se destacam: Simplex e Karmakar da AT&T (ambos de programação linear), direções conjugadas de Powell, Steepest Descent, Newton, Quasi-Newton (métodos para problemas não-lineares sem restrições), dos Gradientes Reduzidos, das Direções Admissíveis, da Penalização Exterior, Programação Quadrática Seqüencial - PQS ou SQP em inglês (métodos para problemas não lineares com restrições), e muitos outros.

Dentre os métodos probabilísticos mais conhecidos estão o *Simulated Annealing* e os *Algoritmos Genéticos*, ambos baseados em fenômenos naturais. O primeiro faz uma analogia ao processo de recozimento e opera como se perturbasse a posição de um átomo da estrutura do material calculando a variação de energia do sistema. A cada iteração reduz-se a temperatura. O segundo baseia-se na teoria da evolução de Darwin onde os espécimes mais adequados têm maiores chances de

sobreviver e são utilizados operadores genéticos tais como *crossover* e *mutação* a cada geração.

Itens	Métodos de Prog. Matemática	Métodos Probabilísticos
Chances de encontrar o mínimo ou máximo global	<b>Menores Chaces</b>	<b>Maiores Chances</b>
Garantias de que o ponto encontrado é mínimo ou máximo global	<b>Não há</b>	<b>Não há</b>
Permite lidar com variáveis discretas	<b>Não</b>	<b>Sim</b>
Custo computacional	<b>Mais baixo</b>	<b>Em geral, muito mais alto</b>

**Tabela 11:** Comparação entre os métodos gerais de otimização.

### D.3. Conceitos Matemáticos para Otimização

Este item descreve alguns dos conceitos matemáticos básicos da otimização que permitirão o entendimento do algoritmo aplicado para a Hexa e a interpretação dos seus resultados.

#### Conceitos do Cálculo Diferencial

Seja uma função contínua  $f(x)$ , onde  $x$  é a sua variável. Sabe-se do cálculo diferencial que um ponto  $x^*$  de mínimo, de inflexão ou de máximo ocorre quando é verdadeira a sentença:

$$\left. \frac{df}{dx} \right|_{x^*} = 0 \tag{198}$$

A diferenciação entre os tipos de ponto é determinada, respectivamente, segundo as sentenças:

$$\left. \frac{d^2 f}{dx^2} \right|_{x^*} > 0 \rightarrow \text{ponto de mínimo} \quad (199)$$

$$\left. \frac{d^2 f}{dx^2} \right|_{x^*} = 0 \rightarrow \text{ponto de inflexão (ou de sela)} \quad (200)$$

$$\left. \frac{d^2 f}{dx^2} \right|_{x^*} < 0 \rightarrow \text{ponto de máximo} \quad (201)$$

O conceito pode ser expandido para um espaço multivariável. Para avaliar o tipo de ponto, utilizam-se as derivadas de segunda ordem na forma de uma matriz **H**, que é chamada matriz Hessiana:

$$\left. \frac{\partial f}{\partial x_1} \right|_{x^*} = \left. \frac{\partial f}{\partial x_2} \right|_{x^*} = \dots = \left. \frac{\partial f}{\partial x_n} \right|_{x^*} = 0 \quad (202)$$

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \dots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \dots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \quad (203)$$

Sobre a matriz Hessiana avaliada no ponto  $\mathbf{x}^*$ , alguns comentários importantes são apresentados:

- Se ela é positiva – definida, ou seja, todos os seus autovalores são maiores que zero, tem-se um ponto de mínimo;
- Se ela é negativa – definida, ou seja, todos os seus autovalores são menores que zero, tem-se um ponto de máximo;

- Se ela é positiva – semidefinida, ou seja, todos os seus autovalores são maiores ou iguais a zero, faz-se necessário uma análise com derivadas de alta ordem para que se comprove a condição de um ponto de mínimo;
- Se ela é negativa – semidefinida, ou seja, todos os seus autovalores são menores ou iguais a zero, faz-se necessário uma análise com derivadas de alta ordem para que se comprove a condição de um ponto de máximo;
- Se ela for indefinida (nenhum dos casos anteriores), o ponto é de sela. O ponto de sela é equivalente ao ponto de inflexão (como por exemplo, em  $x^3$  que possui um ponto de inflexão em  $x = 0$ ) para curvas multidimensionais.

### Multiplificadores de Lagrange

Quando problemas de otimização com restrições são resolvidos, as variáveis de projeto não mais ficam independentes umas das outras. Desta forma as relações de cálculo diferencial que descrevem um ponto de mínimo ou de máximo (todas as derivadas parciais nulas no ponto) não são mais necessariamente verdades. Assim, precisa-se de um método que possibilite a solução deste novo problema.

Quanto às restrições laterais, não existe nenhum problema, já que apenas limitam os tamanhos máximos e / ou mínimos das variáveis, sem criar relações de interdependência. Para o caso das igualdades, tendo-se um problema com  $n$  variáveis de projeto e  $n_i$  restrições de igualdade, onde  $n$  deve ser maior que  $n_i$  para que se tenha um problema de otimização<sup>††</sup>: Minimizar a função  $f(\mathbf{x})$  tal que seja respeitada a restrição de igualdade  $r_j(\mathbf{x}) = 0$ , onde  $r_j$  é uma função genérica e  $j = 1, 2, \dots, n_i$ .

Pode-se escrever que tanto a derivada de  $f$  como de  $r_j$  valem zero no ponto ótimo (mínimo).

$$df = \left. \frac{\partial f}{\partial x_1} \right|_{\mathbf{x}^*} dx_1 + \left. \frac{\partial f}{\partial x_2} \right|_{\mathbf{x}^*} dx_2 + \dots + \left. \frac{\partial f}{\partial x_n} \right|_{\mathbf{x}^*} dx_n = 0 \quad (204)$$

---

<sup>††</sup> Caso  $n$  e  $n_i$  sejam iguais, o problema pode ter apenas um ponto de solução e se  $n_i$  for maior e as restrições formarem um sistema de equações independentes, não há solução.

$$dr_j = \left. \frac{\partial r_j}{\partial x_1} \right|_{\mathbf{x}^*} dx_1 + \left. \frac{\partial r_j}{\partial x_2} \right|_{\mathbf{x}^*} dx_2 + \dots + \left. \frac{\partial r_j}{\partial x_n} \right|_{\mathbf{x}^*} dx_n = 0 \quad (205)$$

Multiplicando a derivada das restrições por uma constante  $\lambda$  (que é uma incógnita a mais, chamada de Multiplicador de Lagrange) e somando o resultado à derivada da função  $f$ , tem-se (para uma restrição):

$$\begin{aligned} df + \lambda dr &= \\ &= \left( \left. \frac{\partial f}{\partial x_1} \right|_{\mathbf{x}^*} + \lambda \left. \frac{\partial r}{\partial x_1} \right|_{\mathbf{x}^*} \right) dx_1 + \left( \left. \frac{\partial f}{\partial x_2} \right|_{\mathbf{x}^*} + \lambda \left. \frac{\partial r}{\partial x_2} \right|_{\mathbf{x}^*} \right) dx_2 + \dots + \left( \left. \frac{\partial f}{\partial x_n} \right|_{\mathbf{x}^*} + \lambda \left. \frac{\partial r}{\partial x_n} \right|_{\mathbf{x}^*} \right) dx_n = 0 \end{aligned} \quad (206)$$

Esta função é chamada de Lagrangeano do problema (símbolo  $L$ ) e pode ser resolvida como um problema de otimização sem restrições. Seguem também as derivadas do Lagrangeano com relação às variáveis de projeto no ponto ótimo (que deve valer zero) e aos multiplicadores de Lagrange (que retorna a restrição).

$$L(\mathbf{x}, \lambda_j) = f(\mathbf{x}) + \sum_{j=1}^{n_i} \lambda_j r_j \quad (207)$$

$$\left. \frac{\partial L}{\partial x_k} \right|_{\mathbf{x}^*} = 0 \quad k = 1, 2, \dots, n \quad (208)$$

$$r_j(\mathbf{x}) = \frac{\partial L}{\partial \lambda_j} = 0 \quad j = 1, 2, \dots, n_i \quad (209)$$

Quando se trabalha com inequações como restrições, deve-se utilizar uma função auxiliar, mas o conceito do Lagrangeano e dos multiplicadores é o mesmo. Considera-se que existem  $m$  restrições do tipo:  $s_p(\mathbf{x}) \leq 0$  ( $p = 1, 2, \dots, m$ ). Essas restrições são transformadas em restrições de igualdade através da utilização de funções auxiliares como  $t_p$  (apresentada ao quadrado para que se possa trabalhar com suas derivadas). De forma análoga, também são escritas algumas relações:

$$s_p(\mathbf{x}) - t_p(\mathbf{x})^2 = 0 \quad (210)$$

O Lagrangeano é dado por:

$$L(\mathbf{x}, \lambda_p, t_p) = f(\mathbf{x}) + \sum_{p=1}^m \lambda_p (s_p(\mathbf{x}) - t_p(\mathbf{x})^2) \quad (211)$$

Novamente escrevem-se as suas derivadas com relação às variáveis de projeto no ponto ótimo e com relação aos multiplicadores de Lagrange:

$$\left. \frac{\partial L}{\partial x_k} \right|_{\mathbf{x}^*} = 0 \quad k = 1, 2, \dots, n \quad (212)$$

$$\frac{\partial L}{\partial \lambda_p} = s_p - t_p^2 = 0 \quad p = 1, 2, \dots, m \quad (213)$$

A derivada do Lagrangeano com relação à função auxiliar  $t_p$  é dada segundo a relação:

$$\frac{\partial L}{\partial t_p} = -2\lambda_p t_p = 0 \quad p = 1, 2, \dots, m \quad (214)$$

Se  $\lambda_p t_p = 0$ , então  $\lambda_p s_p = 0$ , já que  $s_p(\mathbf{x}) - t_p(\mathbf{x})^2 = 0$ . Essa relação é chamada de *condição de estacionaridade* e diz que se o multiplicador de Lagrange é zero, então a restrição é diferente de zero, ou seja, a restrição não está ativa. Caso o multiplicador não valha zero, então a restrição deve valer zero, estando assim ativa.

Desta forma, o Lagrangeano com todas as restrições, será a combinação dos termos das eq. (167) e eq. (171).

$$L(\mathbf{x}, \lambda_p, t_p) = f(\mathbf{x}) + \sum_{j=1}^{n_r} \lambda_j r_j + \sum_{p=1}^m \lambda_p (s_p(\mathbf{x}) - t_p(\mathbf{x})^2) \quad (215)$$

### Condições KKT de Optimalidade

As condições KKT<sup>††</sup> permitem verificar se um ponto realmente é um ótimo local (não há, exceto em casos muito específicos, como confirmar se um ponto é ótimo global). A demonstração matemática completa destas condições pode ser encontrada em **Freund, 2004**.

Quando se considera o Lagrangeano descrito no item anterior no ponto ótimo de mínimo  $\mathbf{x}^*$ , pode-se chegar a algumas condições; condições estas chamadas de KKT. Elas afirmam que:

- Se o ponto ótimo é viável, todas as restrições são válidas, ou seja, as restrições de igualdade valem zero e as inequações são menores ou iguais a zero;
- O gradiente do Lagrangeano no ponto ótimo vale zero (condição de estacionaridade). Desta forma, os multiplicadores de Lagrange devem ser maiores ou iguais a zero para que (a) seja verdade e o gradiente valha zero;
- Se o multiplicador de Lagrange de uma restrição vale zero, ela estará inativa (condição de complementaridade). Caso ele seja maior que zero ela estará ativa;

Existem ainda casos onde as condições KKT são necessárias, mas não suficientes. Nesses casos (quando o número de variáveis de projeto é diferente do número de restrições ativas), deve ser verificada a Hessiana do Lagrangeano. Valem as mesmas relações apresentadas anteriormente para a Hessiana (positiva definida indica um mínimo, negativa definida um máximo, etc).

---

<sup>††</sup> Karush-Kuhn-Tucker