

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Daniel José Damas Follador

Desenvolvimento de um controle cinemático
robusto para um manipulador robótico
desenvolvido para cirurgia NOTES

São Paulo

2011

ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Daniel José Damas Follador

Desenvolvimento de um Controle Cinemático
Robusto para um Manipulador Robótico
Desenvolvido para Cirurgia NOTES

Trabalho de Conclusão de Curso apresentado à Escola
Politécnica da Universidade de São Paulo para a obtenção
do título de Bacharel em Engenharia Mecatrônica

Área de Concentração: Cirurgia Robótica

Orientador: Prof. Dr. Tarcísio Antônio Hess-Coelho

Co-Orientador: Prof. Dr. Pietro Cerveri

São Paulo

2011

FICHA CATALOGRÁFICA

Follador, Daniel José Damas

Desenvolvimento de um controle cinemático robusto para um manipulador robótico desenvolvido para cirurgia NOTES / D.J.D. Follador. -- São Paulo, 2012.

75 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos.

1.Robótica 2.Cirurgia 3.Controle cartesiano 4.Singularidades cinemáticas I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos II.t.

Agradecimentos

À Aline, por sua paciência e apoio incondicional.

Aos meus orientadores Pietro e Tarcisio, por sua atenção e disponibilidade.

Aos meus pais, Maria José e Wilson, por seu amor, compreensão e por tornarem possíveis todas as minhas realizações.

Abstract

Driven by the surgeon's desire for less-invasive procedures, a new technique called NOTES (Natural Orifice Transluminal Endoscopic Surgery) has emerged. The present instrumentation used for this type of surgery is adapted from Endoscopy and is not perfectly adequate for many reasons: lack of dexterity, lack of tactile feedback, fewer degrees of freedom than desired. Those problems were the motivation for a Italian research group to develop a robotic manipulator with several advantages over the current technology. The snake-like mechanical design of the robot contains three kinematic singularities whose physical interpretation were one of the objectives of this work. The second main goal of this work was to design an adequate kinematic control algorithm for avoiding singularities by using geometric redundancy from other degrees of freedom. For that matter, four different trajectories were tested with four different algorithms and the results were evaluated in terms of tracking error in the Cartesian space and in the joint space.

Resumo

Impulsionado pelo desejo do cirurgião para procedimentos menos invasivos, uma nova técnica chamada NOTES (Natural Orifice Transluminal Endoscopic Surgery) emergiu. A instrumentação utilizada para apresentar este tipo de cirurgia é adaptado da Endoscopia e não é perfeitamente adequado por várias razões: falta de destreza, falta de feedback tátil, menos graus de liberdade do que o desejado. Esses problemas foram a motivação para um grupo de pesquisa italiana para desenvolver um manipulador robótico com várias vantagens sobre a tecnologia atual. O projeto mecânico em formato de cobra do robô contém três singularidades cinemáticas cujas interpretações físicas foram um dos objetivos deste trabalho. O segundo objetivo principal deste trabalho foi desenvolver um algoritmo de controle cinemático adequado para evitar singularidades usando a redundância geométrica fornecida por outros graus de liberdade. Para essa matéria, quatro trajetórias diferentes foram testadas com quatro diferentes algoritmos e os resultados foram avaliados em termos de erro de rastreamento no espaço cartesiano e no espaço das juntas.

Sumário

1. Introdução	9
1.1. Cirurgia Minimamente Invasiva.....	10
1.2. Cirurgia Laparoscópica	10
1.3. Cirurgia Robótica	12
1.4. NOTES	14
2. Projeto NOTESNAIL	17
2.1. Motivação	17
2.2. Detalhamento Construtivo	19
2.3. Cinemática direta do NOTESnail.....	22
3. Estudo das Singularidades Cinemáticas.....	24
3.1. Teoria Geral das Singularidades.....	24
3.2. Estudo das singularidades NOTESnail	29
3.2.1. Modelando o espaço de trabalho com geometria analítica	31
3.2.2. Configurações atingíveis e medida de redundância	34
3.2.3. Configurações singulares NOTESnail	36
4. Algoritmos de Controle em Robótica	40
4.1. Computed torque control	42
4.2. Visual servoing control	42
4.3. Controle Cartesiano	44
4.3.1. Método da Transposição da Jacobiana	47
4.3.2. Método da Jacobiana pseudo-invertida	47
4.3.3. Método dos mínimos quadrados amortecidos.....	48
4.3.4. Método da Decomposição em Valores Singulares	49
4.3.5. Método de Otimização	50

5. Resultados Experimentais	51
5.1. Trajetórias	51
5.1.1. Pick and Pull	52
5.1.2. Trajetória de teste de singularidade do tipo I	53
5.1.3. Trajetória de teste de singularidade do tipo II	54
5.1.4. Simulação da Sutura	55
5.2. Modelos Simulink	57
5.2.1. Método da Transposta	58
5.2.2. Método da Pseudoinversa	58
5.2.3. Método dos mínimos quadrados amortecidos.....	59
5.2.4. Método SVD	59
5.2.5. Optimization method.....	60
5.3. Resultados gráficos	61
6. Discussão e Conclusão.....	63
6.1. Trajetórias	63
6.2. Método da Transposta	65
6.3. Método dos mínimos quadrados amortecidos.....	66
6.4. Singular Value Decomposition	67
6.5. Método Pseudoinversa	67
6.6. Algoritmo de Otimização	67
6.7. Considerações finais.....	68
7. Referências	69
8. Apêndice A.....	73

1. Introdução

Os primeiros indícios de um procedimento cirúrgico ocorreu no início da pré-história. O registro de uma trepanação (abertura de um ou mais buracos no crânio para aliviar a pressão intracraniana) foi encontrado em pinturas rupestres e, mais tarde nos registros históricos (CAPASSO, 2002). Desde então, a cirurgia tem evoluído ao longo da história, novas técnicas e instrumentos têm sido desenvolvidos e muito se aprendeu sobre a anatomia, fisiologia, risco de infecção e contaminação, culminando hoje cerca de 20 milhões de americanos operados anualmente (ROAN, 2005).

Podemos classificar uma operação de acordo com alguns critérios: (AMERICAN COLLEGE OF SURGEONS, 2007)

- Com base na **urgência**, de acordo com o risco para a sobrevivência do paciente, a cirurgia é considerada eletiva, urgentes ou opcional;
- Com base no **efeito**, uma cirurgia exploratória pode ser (para confirmar o diagnóstico) ou terapia;
- Com base no **tipo** de procedimento, a cirurgia pode variar de acordo com o padrão de intervenção (por exemplo, a amputação, cirurgia plástica, transplantes, etc);
- Com base na **localização anatômica**, tomando como exemplos a cirurgia cardiovascular, ortopedia e gastrointestinal.
- Por tipo de **equipamento** utilizado, pode envolver ferramentas clássicas (como um bisturi e pinças), as ferramentas mais modernas, como laser ou mesmo instrumentos mecatrônicos, como Intuitive Surgical do robô da Vinci (INTUITIVE SURGICAL, 2010);
- Em grau de invasão, a cirurgia pode ser:
 - aberto, o que envolverá a abertura de vários tecidos do cirurgião para atingir a meta desejada;

- cirurgia minimamente invasiva, que se tornam possíveis através dos tribunais inferiores, tais como na cirurgia laparoscópica, angioplastia e cirurgia NOTES

1.1. Cirurgia Minimamente Invasiva

A cirurgia minimamente invasiva é um procedimento que envolve um conjunto diferente de ferramentas para o mesmo objetivo da cirurgia aberta, mas com muito menos danos aos tecidos biológicos. É importante ressaltar que toda a instrumentação para esse tipo de cirurgia é especialmente concebido e submetido à constante inovação.

Em comparação, cirurgias minimamente invasivas têm algumas vantagens sobre a cirurgia aberta. Um estudo do departamento de cirurgia no Hospital Henry Ford em Detroit, nos EUA (VELANOVICH, 2000) mostraram que, além de reduzir a dor para os pacientes e garantir mais rápido retorno à função normal, a cirurgia minimamente invasiva (MIS), também resulta em melhor qualidade de vida em comparação com técnicas abertas.

1.2. Cirurgia Laparoscópica

Laparoscópica ou "minimamente invasiva" A cirurgia é uma técnica especializada para realizar a cirurgia. No passado, esta técnica era comumente usado para cirurgia ginecológica e para a cirurgia da vesícula biliar. Nos últimos 10 anos, o uso desta técnica tem se expandido em cirurgia intestinal.

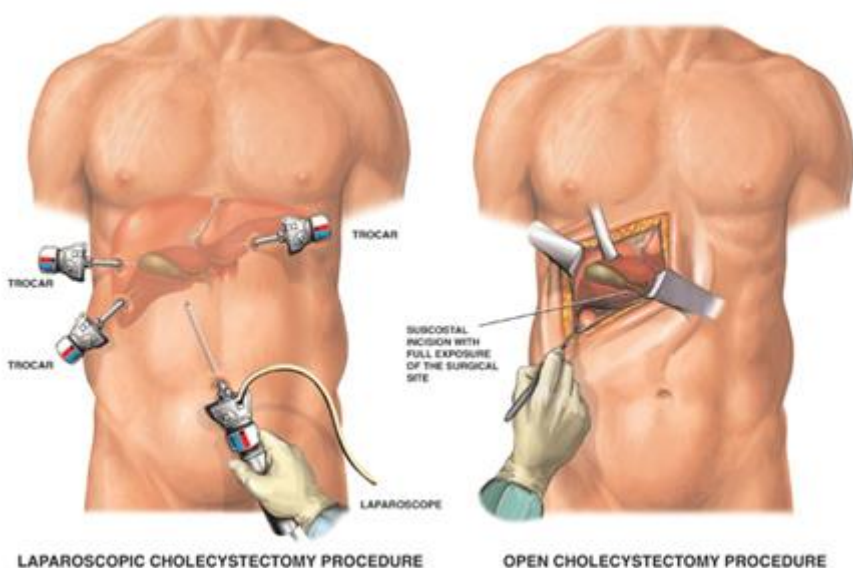


Figura 1 - Laparoscopia vs. cirurgia aberta

Conforme mostrado na Figura 1, na tradicional cirurgia "aberta" o cirurgião utiliza uma incisão única para entrar no abdômen. A cirurgia laparoscópica usa várias incisões de 0,5 a 1 cm. Cada incisão é chamada de "porta". Em cada porta um instrumento tubular conhecida como um trocar está inserido. Instrumentos especializados e uma câmera especial conhecido como um laparoscópio é passada através do trocadores durante o procedimento. No início do procedimento, o abdome é inflado com gás de dióxido de carbono para fornecer um espaço de trabalho e visualização para o cirurgião. O laparoscópio transmite imagens da cavidade abdominal para monitores de alta resolução de vídeo na sala de cirurgia. Durante a operação o cirurgião relógios imagens detalhadas do abdômen no monitor. Este sistema permite ao cirurgião realizar as mesmas operações que a cirurgia tradicional, mas com mínimas incisões. Em certas situações, o cirurgião pode optar por usar um tipo especial de porta que é grande o suficiente para inserir uma mão. Quando uma porta de mão é usada a técnica cirúrgica é chamado de "mão assistida" laparoscopia. A incisão necessária para a porta de mão é maior do que a outras incisões laparoscópicas, mas é geralmente menor do que a incisão necessária para a cirurgia tradicional. Em

comparação com a cirurgia aberta tradicional, os pacientes muitas vezes experimentam menos dor, uma recuperação mais curta e menos cicatrizes com a cirurgia laparoscópica. (AMERICAN COLLEGE OF SURGEONS, 2007)

1.3. Cirurgia Robótica

A cirurgia robótica é uma técnica em que um cirurgião realiza a cirurgia usando um computador que controla remotamente instrumentos muito pequenos ligados a um robô.

Este procedimento é feito sob anestesia geral. O cirurgião se senta em uma estação de computador por perto e dirige os movimentos de um robô. Pequenos instrumentos estão ligados a braços do robô. O cirurgião insere primeiro destes instrumentos no corpo do paciente através de pequenas cortes cirúrgicos. Sob a direção do cirurgião, o robô imita os movimentos da mão do médico para realizar o procedimento utilizando os instrumentos minúsculos, como mostrado na Figura 2



Figura 2 - Exemplo de um sistema de cirurgia robótica

Um tubo fino com uma câmera presa à extremidade do endoscópio permite ao cirurgião visão altamente ampliada imagens tridimensionais em um monitor em tempo real. A cirurgia robótica é um tipo de procedimento que é semelhante à cirurgia laparoscópica.

Também pode ser realizada através de cortes cirúrgicos menores quando comparados à cirurgia aberta tradicional. Os movimentos pequenos e precisos que são possíveis com este tipo de cirurgia dar-lhe algumas vantagens sobre o padrão de técnicas endoscópicas. (OLEYNIKOV, 2008)

Às vezes, a laparoscopia robótica assistida pode permitir que um cirurgião para realize um procedimento menos invasivo que era apenas possível com cirurgia aberta. Uma vez que é colocado no abdômen, um braço robótico é de mais fácil utilização para o cirurgião do que os instrumentos em cirurgia endoscópica. O robô reduz os movimentos do médico (por exemplo, movendo-se 1 / 2 polegada para cada 1 polegada os movimentos do cirurgião), o que reduz alguns dos tremores nas mãos e alguns movimentos que poderiam fazer a cirurgia menos precisa. Além disso, os instrumentos robóticos podem acessar áreas mais difíceis de alcançar através de pequenos cortes cirúrgicos em relação à cirurgia aberta e laparoscópica tradicional.

Durante a cirurgia robótica, o cirurgião pode ver mais facilmente a área a ser operada. O cirurgião também está em uma posição muito mais confortável e pode se mover de uma forma mais natural do que durante a endoscopia. A cirurgia robótica pode ser utilizada para uma série de procedimentos diferentes, incluindo: (OLEYNIKOV, 2008)

- Revascularização
- Remoção da vesícula biliar
- Substituição do quadril
- Histerectomia
- Remoção de rim
- Transplante de rim
- Reparo da valva mitral
- Prostatectomia radical
- Ligadura

Levando-se em conta a tecnologia atual, a cirurgia robótica não pode ser usado para alguns procedimentos complexos. Por exemplo, não é apropriado para certos tipos de cirurgia cardíaca que requerem uma maior capacidade de mover os instrumentos no tórax do paciente.

1.4. NOTES

NOTES (acrônimo para *Natural Orifice Transluminal Endoscopic Surgery*, ou em português: Cirurgia Endoscópica Transluminal através dos Orifícios Naturais) foi impulsionado pelo desejo do cirurgião para procedimentos menos invasivos e um acesso cada vez menor, a técnica é baseada no conceito de que a cavidade peritoneal pode ser acessada através de orifícios naturais. NOTES designa um procedimento cirúrgico que utiliza um ou mais orifícios naturais do corpo do paciente com a intenção de punção de vísceras ocas, a fim de penetrar uma cavidade do corpo que seria inacessível de outra forma. (KALLOO, SINGH, *et al.*, 2004)

Orifícios naturais geralmente incluem a boca (atingindo o estômago), anus (atingindo o cólon), vagina (atingindo o útero) e da uretra (atingindo a bexiga). Teoricamente as vantagens de um procedimento NOTES sobre a abordagem laparoscópica são resultantes da ausência de uma incisão externa abdominal: um procedimento menos invasivo que permite a minimização da anestesia e analgesia; redução da dor pós-operatória da parede abdominal, da infecção da ferida e da formação de hérnia e aderências. (BOWMAN, 2006)

Devido ao seu novo conceito e o processo em curso na medicina, a conotação e classificação da NOTES ainda não são definitivos e são, por vezes controversos. De acordo com o estado atual dos estudos NOTES, seus procedimentos são principalmente divididos em duas categorias, NOTES "puro" e NOTES "híbrido". (GEOFFREY, TIMOTHY, *et al.*, 2008)

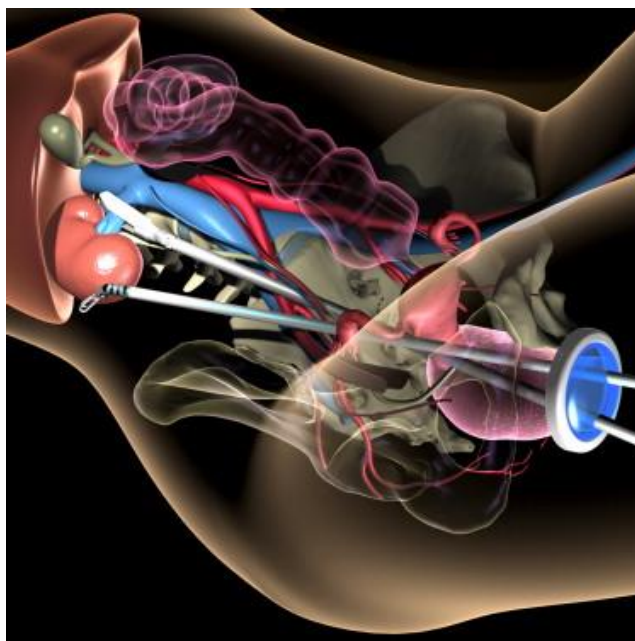


Figura 3 - Abordagem NOTES "puro" com acesso transvaginal

O "puro" NOTES refere-se a um procedimento que é concluído sem quaisquer portas transabdominais, incluindo o umbigo, como mostrado na Figura 3. O NOTES "híbrido" refere-se a um "misto" de tecnologias usando instrumentação transabdominal para facilitar o procedimento NOTES.

Para analisar a instrumentação adequada para NOTES, é válido para olhar para trás, alguns procedimentos que têm sido feito até agora. (KALLOO, SINGH, *et al.*, 2004) descreveram uma colecistectomia em um modelo suíno utilizando uma série de acessórios endoscópicos, tais como a agulha faca, sphincterotome e balões de dilatação. O acesso transgástrica foi complexo em termos de identificação, manipulação e ressecção da vesícula biliar.

(PAI, FONG, *et al.*, 2006) por outro lado, executaram um acesso transcolônico porque acreditavam que iria permitir uma melhor visualização e estabilidade endoscópica já que a orientação em relação aos órgãos do abdômen superior seria frontal.

(HABER, CROUZET, *et al.*, 2008) apresentaram uma experiência inicial de NOTES robótico usando o sistema cirúrgico Da Vinci™. Dez suínos do sexo feminino foram submetidos a 10 pyeloplasties, 10 nefrectomias parciais e 10 nefrectomias radicais com sucesso

utilizando uma abordagem híbrida (incisões umbilicais e transvaginais). Os dados intra-operatórios mostraram pequeno tempo operatório e pouca perda de sangue, mas algumas limitações foram percebidas: houveram cinco episódios de conflitos entre os instrumentos (que representam 17% da amostra), 3 episódios de contatos indesejados entre endoscópio e instrumentos (9%) e em 3 episódios (9%), o instrumento não pode alcançar o rim.

Em suma, NOTES é uma técnica cirúrgica muito promissora, com várias aplicações, incluindo a cirurgia oncológica (RIEDER e SWANSTROM, 2011). Muitos pesquisadores (médicos ou engenheiros biomédicos) estão à procura de tecnologias e práticas para melhorar este tipo de cirurgia e melhorar os resultados do paciente.

2. Projeto NOTESNAIL

2.1. Motivação

Embora NOTES seja uma técnica muito promissora, alguns estudos têm sido realizados sobre a qualidade da manipulabilidade da instrumentação atualmente em uso e suas limitações. Uma das principais preocupações, existentes até mesmo na cirurgia laparoscópica, é a coordenação olho-mão, que limita a triangulação em três dimensões. Essa questão é acentuada em NOTES dado que a câmera e os instrumentos estão no mesmo eixo, como mostrado na Figura 4.

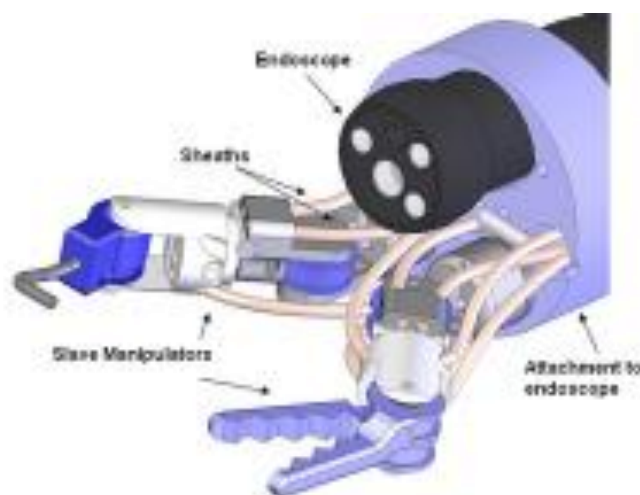


Figura 4 - Instrumentação NOTES proposta em Cingapura

Outro grande obstáculo para o cirurgião é a flexibilidade de um endoscópio. Por um lado, o instrumento tem de ser flexível, a fim de chegar ao local correto de operação. Por outro lado, o instrumento tem que ser rígido, a fim de executar as forças e pressões adequadas para a tarefa dada. (BOWMAN, 2006)

(CERVERI, 2008) sugere que os instrumentos operativos não podem ser simplesmente adaptados de cirurgia endoscópica ou laparoscópica por causa de seu significativamente diferentes condições de trabalho:

- O cirurgião não pode girar a parte externa do instrumento, a fim de orientar a ferramenta dentro do corpo do paciente;

- A primeira parte do instrumento (parte proximal), que não entra na cavidade peritoneal, deve atingir a incisão de entrada que pode ser muito longe da abertura externa do orifício natural;
- A segunda parte do instrumento (parte dispositiva, entrando na cavidade peritoneal) precisa de vários graus de liberdade para chegar à região operatória, mais do que em instrumentos endoscópicos, onde 4 graus de liberdade são obtidos apenas girando a parte externa do instrumento fora do corpo do paciente;
- A transmissão mecânica de movimento à parte dispositiva do instrumento a partir de uma alça fora do orifício não é viável devido ao tamanho e geometria da peça proximal do instrumento.
- A perda de feedback de força também representa um problema. Ela está relacionada à cirurgia minimamente invasiva em geral e é particularmente relevante com NOTES porque o controlador do instrumento é significativamente distante da região operatória.

A necessidade de superar essas limitações sugere que haveria muitos benefícios na construção de determinados instrumentos cirúrgicos com alta destreza. E esta foi a motivação para uma equipe de pesquisa na Itália composta por estudantes e professores de quatro grandes universidades: Università degli studi di Genova, Politecnico di Milano, Università degli studi di Bergamo e Università degli Studi de L'Aquila. O projeto de pesquisa tem dois conceitos inovadores: arquitetura "snake" e atuação rigidez variável. Seu primeiro objetivo principal é construir um protótipo em escala real com dois diferentes módulos distais: uma pinça VS e uma micro-câmera. O esquema de todo o sistema é mostrado na Figura 5.

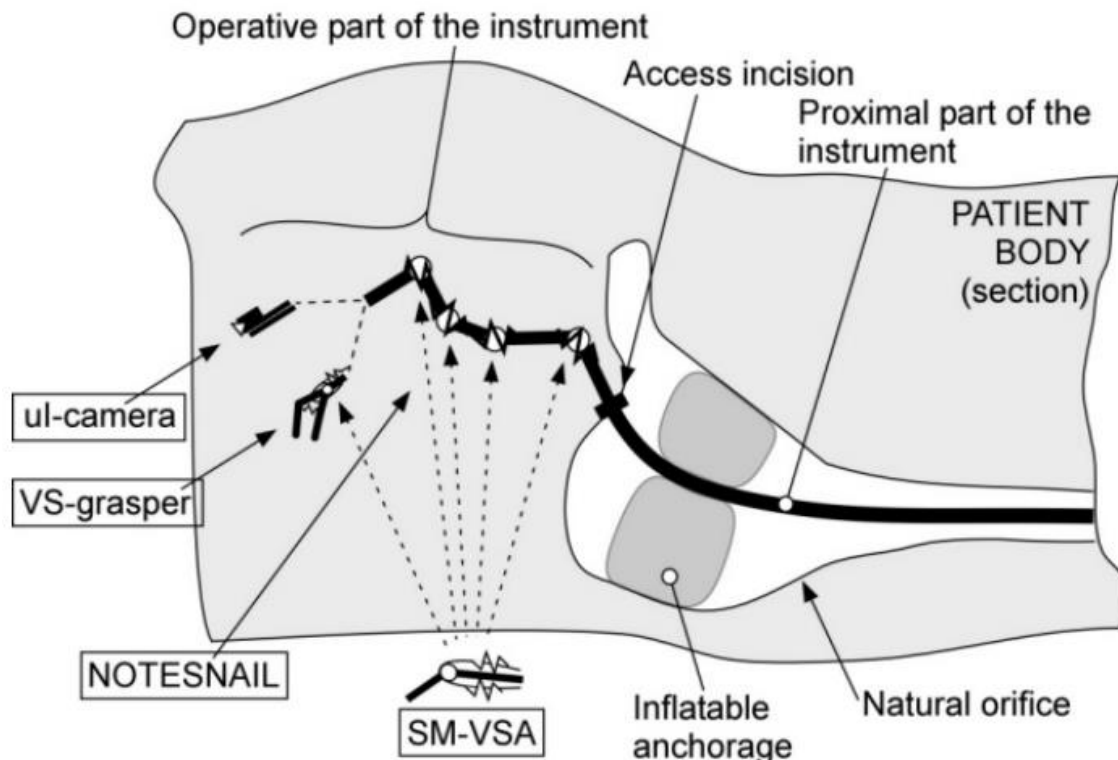


Figura 5 - Esquema do projeto NOTESnail

Um segundo grande objetivo é construir um esquema de controle que permita feedbacks posição e força, além de evitar colisões, o que melhoraria a segurança do paciente e cirurgião. O último objectivo é melhorar a qualidade da visualização do espaço operacional.

Deste ponto em diante, este robô especial será chamado NOTESnail.

2.2. Detalhamento Construtivo

O projeto construtivo do NOTESnail foi feito após uma série de requisitos técnicos que são comuns a muitas aplicações cirúrgicas, além de uma série de exigências dos nossos uso específico:

- Tem que ser inserível em um orifício de 10 mm de diâmetro;
- Capacidade de manobra suficiente para mover-se ao redor de um alvo;
- Ser remotamente controlável através de um console remoto;
- Baixa dissipação de calor e baixo consumo de energia
- Iluminação autónoma
- Atuação eletromecânica

- Sensor para codificação de posição e para interação mecânica com objetos ao redor
- Esterilizável
- Esquemas de controle por visual servoing e force servoing

Trabalhos anteriores feitos pelos italianos trouxeram um conceito modular, com cada módulo independente possuindo dois micro-motores movendo um conjunto de dobra (mostrado na Figura 6) e um conjunto de torção (mostrado na Figura 7). Uma representação computador aided design do robô final de montagem é mostrado na Figura 8.

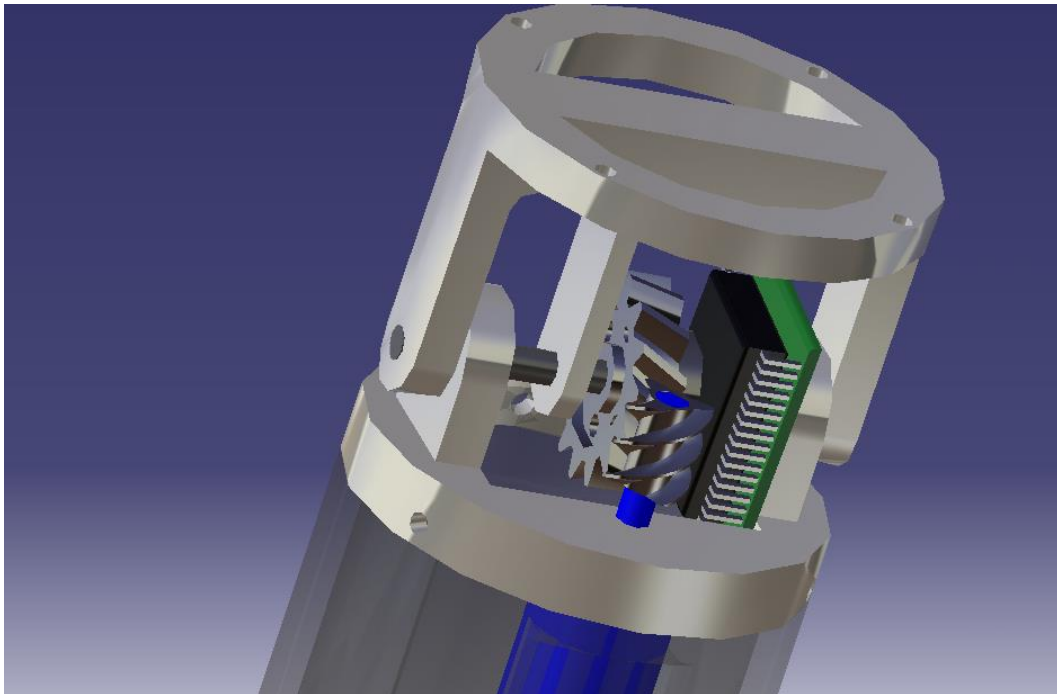


Figura 6 - Junta de flexão

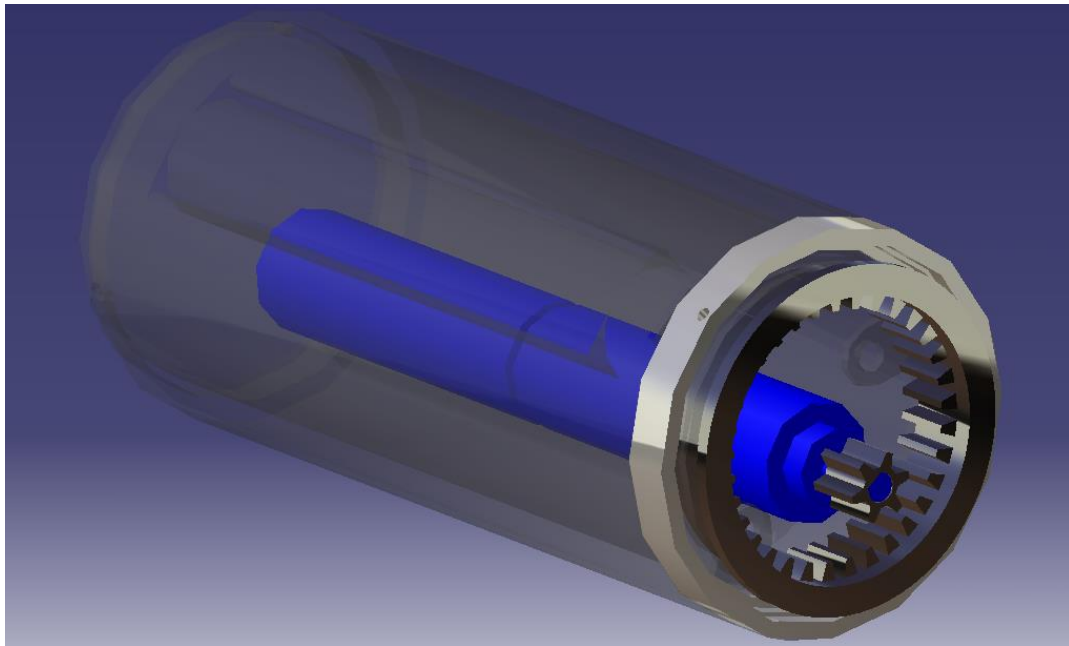


Figura 7 - Junta de torção

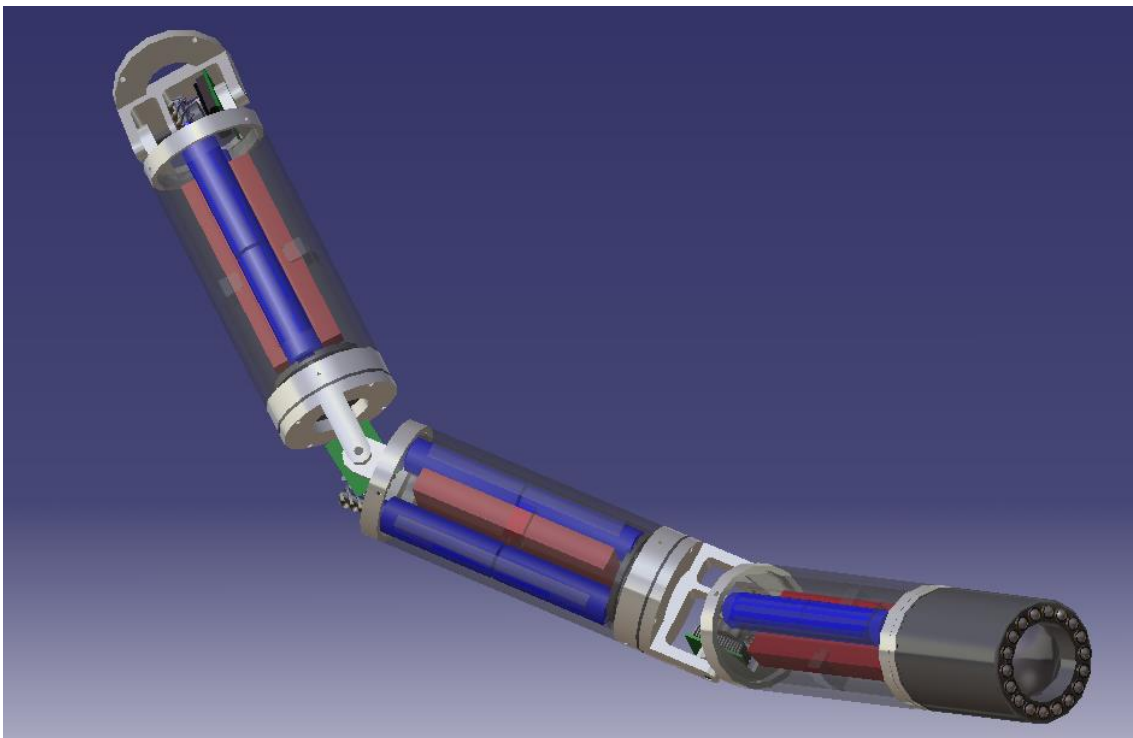


Figura 8 - Representação final do NOTESnail

Vale a pena perceber as razões para a escolha do número de módulos independentes. Três módulos devem ser suficientes para fornecer seis graus de liberdade, o que será suficiente para o cirurgião chegar a qualquer local da operação com a orientação correta. Um maior número de módulos faria o algoritmo de controle mais complicado (o que equivale a dizer que ele terá um maior custo computacional).

Algumas das características que são mais importantes para simular o comportamento cinemático e dinâmico de NOTESnail são apresentados na Tabela 1. Centro de massa e momento de inércia foram calculados usando o computador de software de desenho auxiliado e pode ser encontrada no Apêndice A.

Tabela 1 - Propriedades físicas do NOTESnail

Comprimento do módulo	3.5 cm
Alcance da junta de flexão	De -0.5π rad a 0.5π rad
Alcance da junta de torção	De -0.75π rad a 0.75π rad
Massa total do módulo	48 gramas
Massa da camera	30 gramas
Inércia do motor	$9.5e^{-10} \text{ kg/m}^2$
Atrito viscoso	$8e^{-7} \text{ Pa.s}$

2.3. Cinemática direta do NOTESnail

O problema da cinemática direta é a preocupação com a relação entre as articulações individuais do manipulador robô e a posição e orientação da ferramenta ou do efetuador. Para a computação das equações da cinemática direta do NOTESnail, a convenção Danevit-Hartenberg foi usada (SPONG, HUTCHINSON e HUTCHINSON, 2004). A Figura 9 exibe uma representação simplificada do NOTESnail e a Tabela 2 exibe os parâmetros de Denavit-Hartenberg.

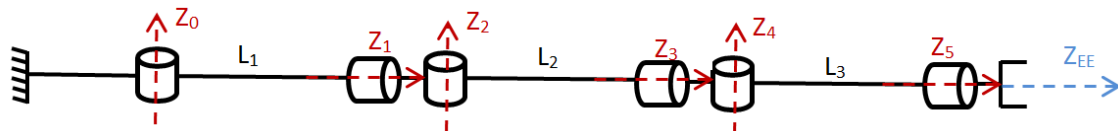


Figura 9 - Representação esquemática do NOTESnail

Tabela 2 - Parâmetros de Denavit-Hartenberg

D-H	a	α	d	θ
1	0	$-\pi/2$	0	Θ_1
2	0	$\pi/2$	L_1	Θ_2
3	0	$-\pi/2$	0	Θ_3

4	0	$\pi/2$	L_2	Θ_4
5	0	$-\pi/2$	0	Θ_5
EE	0	0	L_3	Θ_6

Os parâmetros Denavit-Hartenberg possibilitaram a construção de uma matriz de transformação homogênea (dimensão 4-por-4) que determina a posição e orientação do efetuador de acordo ao sistema de coordenadas fixas na base. Usando essas equações, variando os ângulos conjunta ao longo de suas faixas, é possível mapear área de trabalho do robô, mostrada na Figura 10.

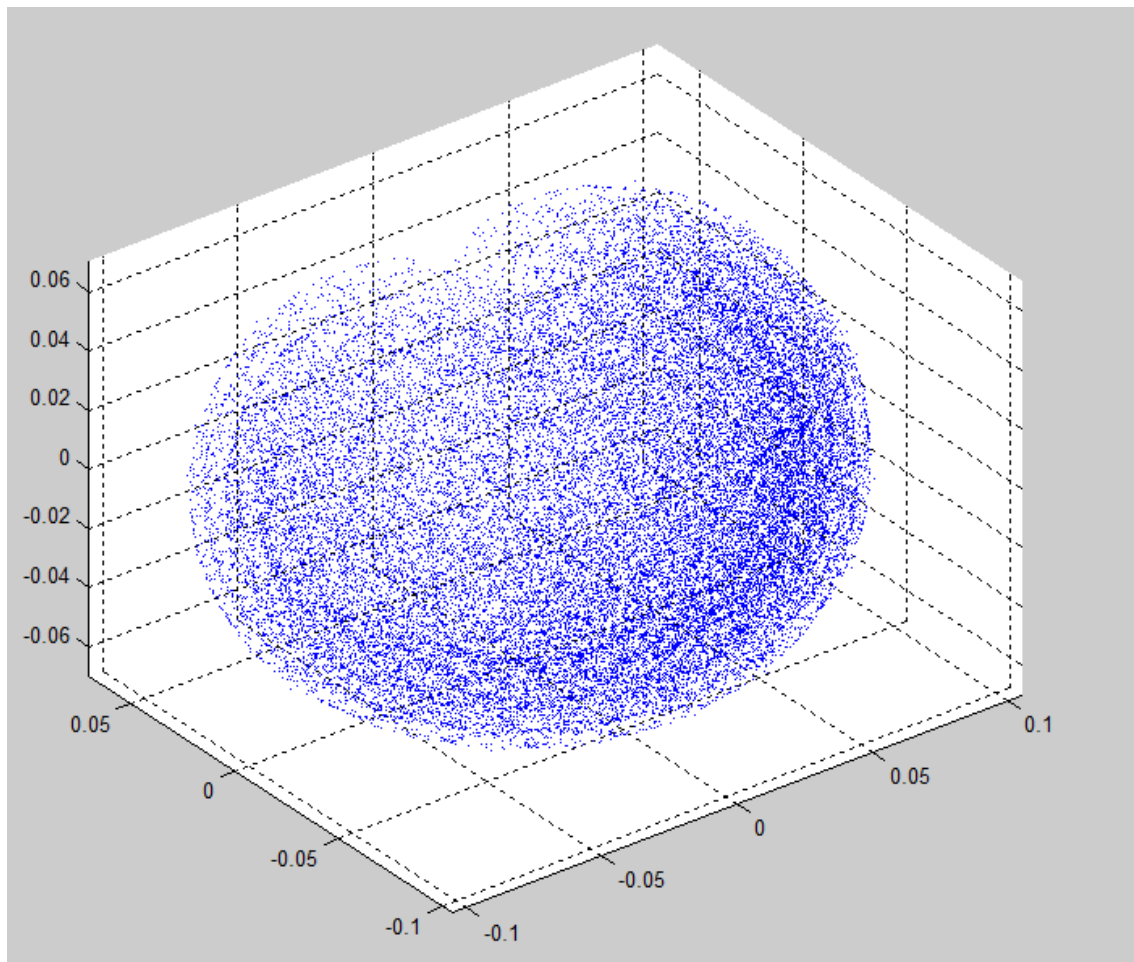


Figura 10 - Espaço de trabalho do NOTESnail

3. Estudo das Singularidades Cinemáticas

Para entender melhor o significado da singularidade de um robô, é necessário entender a cinemática da velocidade e o Jacobiano do manipulador. Matematicamente, as equações de cinemática direta definem uma função entre o espaço de posições e orientações cartesianas e o espaço de posições comuns. As relações de velocidade são, então, determinadas pela Jacobiana desta função. (SPONG, HUTCHINSON e HUTCHINSON, 2004)

Esta matriz Jacobiana tem uma regra essencial na análise e controle de movimento do robô: desde o planejamento e execução trajetória para a transformação das forças e torques a partir do fim-efetoras para as articulações do manipulador. Outra propriedade importante de uma matriz Jacobiana (mais precisamente, seu determinante) é a detecção de singularidades.

3.1. Teoria Geral das Singularidades

Dado um robô com n juntas, o jacobiano $J(q)$ de dimensões $6\text{-por-}n$ define um mapa entre o vetor $\dot{q}$ de velocidades das juntas e o vetor $\dot{X}$ de velocidades do efetuador. No limite infinitesimal, isto define uma transformação linear dada pela equação [1].

$$dX = J(q)dq \quad [1]$$

Um conjunto de coordenadas de junta que faz com que o posto da matriz Jacobiana diminua é chamado de configuração singular ou singularidade. Sua importância é dada porque: (SPONG, HUTCHINSON e HUTCHINSON, 2004)

- Singularidades representam configurações a partir do qual certas direções de movimento pode ser impossível.
- Nas singularidades, pequenas efetuador velocidades podem corresponder a velocidades muito grandes nas juntas;

- Nas singularidades, pequenas efetuator forças e torques podem corresponder a muito grande torques nas juntas
- Singularidades podem corresponder aos limites do espaço de trabalho do manipulador
- Ao redor das singularidades não existirá uma solução única para o problema da cinemática inversa. Nesses casos pode haver nenhuma solução ou pode haver uma infinidade de soluções.

(YOSHIKAWA, 1985) define um valor escalar chamado de "medida de manipulabilidade", que é definida pela equação 2. Essa medida é concedida por um manipulador de certos em uma determinada configuração. Alguns autores (OEMOTO e ANG JR., 2007) usam essa medida para distinguir diferentes comportamentos de seu algoritmo de controle.

$$w = \sqrt{\det(J(\theta)J^T(\theta))} \quad [2]$$

Muitos métodos têm sido propostos para lidar com singularidades e eles foram divididos por (OEMOTO e ANG JR., 2007) em duas categorias principais. Na primeira categoria, uma estratégia de controle uniforme é adotado em todo o espaço de trabalho inteiro, a construção de uma função contínua que introduz uma ligeira alteração com a especificação do espaço tarefa ou seu mapeamento para o espaço das juntas do manipulador. Isto geralmente resulta em uma estratégia de controle estável, onde o efetuator evita a configuração singular. O segundo envolve uma divisão do espaço de trabalho onde um algoritmo de controle diferente é aplicado para a região ao redor das singularidades.

Quando o manipulador está em uma configuração singular, movimentos e forças ao longo da direção singular não são controláveis. Se a tarefa inclui um movimento ao longo dessa direção singular, ela pode ser alcançada usando o movimento espaço nulo (o que corresponde a minimizar uma função potencial correspondente ao objetivo da tarefa).

(OEMOTO e ANG JR., 2007) divide singularidades em duas categorias de acordo com o efeito que o movimento espaço nulo tem sobre eles. Singularidades do tipo I (como mostrado na Figura 11) são aquelas onde o movimento cria espaço nulo do efetuador movimento na direção singular e provoca o fim-efetoras para escapar da região singular através desta direção. Singularidades do tipo II (como mostrado na Figura 12) são aquelas onde o movimento espaço nulo afeta apenas o movimento interno das juntas, e muda a direção singular, sem afetar o movimento/forças do efetuador.

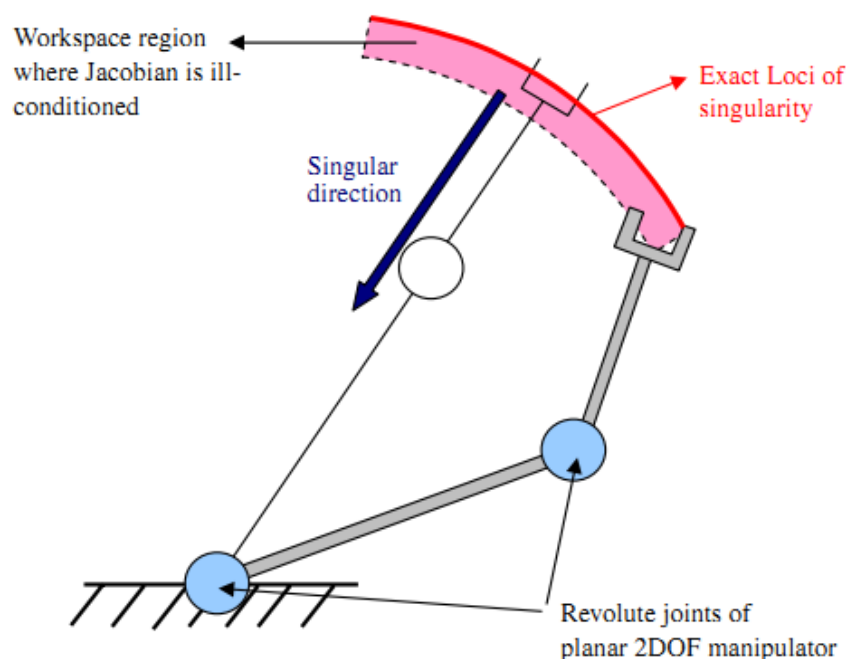


Figura 11 - Exemplo de singularidade do tipo I

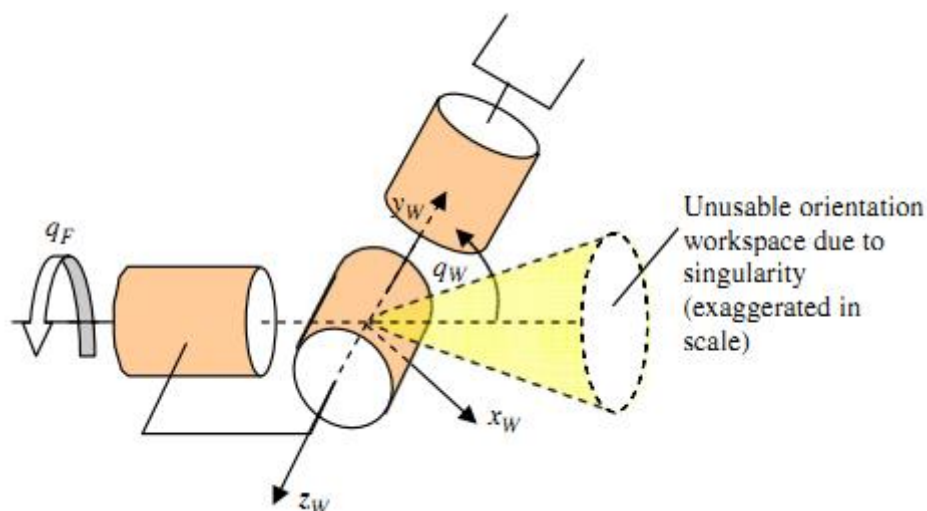


Figura 12 - Exemplo de singularidade do tipo II

(OEMOTO e ANG JR., 2007) implementaram seu algoritmo robusto nas singularidades no exemplo mais famoso entre os livros didáticos: o Puma560, mostrada na sua configuração "descanso" na Figura 13. Sua Jacobiana pode ser dividida em duas partes, correspondendo a "braço" e "punho". Analisando que determinante da matriz, os autores foram capazes de encontrar três diferentes configurações singulares correspondente ao "elbow-lock", "wrist-lock" e "head-lock".

O "bloqueio do cotovelo" é um exemplo de uma singularidade do tipo I causado pelo alinhamento de dois eixos das juntas (como mostra a Figura 14a). Apesar de que a terceira junta é a causa da redução da medida de manipulabilidade, é através do seu movimento que o robô pode "dobrar fora" da configuração singular (como mostrado na Figura 14b).

O "lock cabeça" é um exemplo de uma singularidade do tipo II que acontece quando o ponto de pulso é exatamente acima do primeiro eixo das juntas (ver Figura 15). Um movimento no espaço nulo pode girar a direção singular, permitindo que a tarefa a ser completada sem alterar a posição do efetuador ou orientação (ver Figura 15).

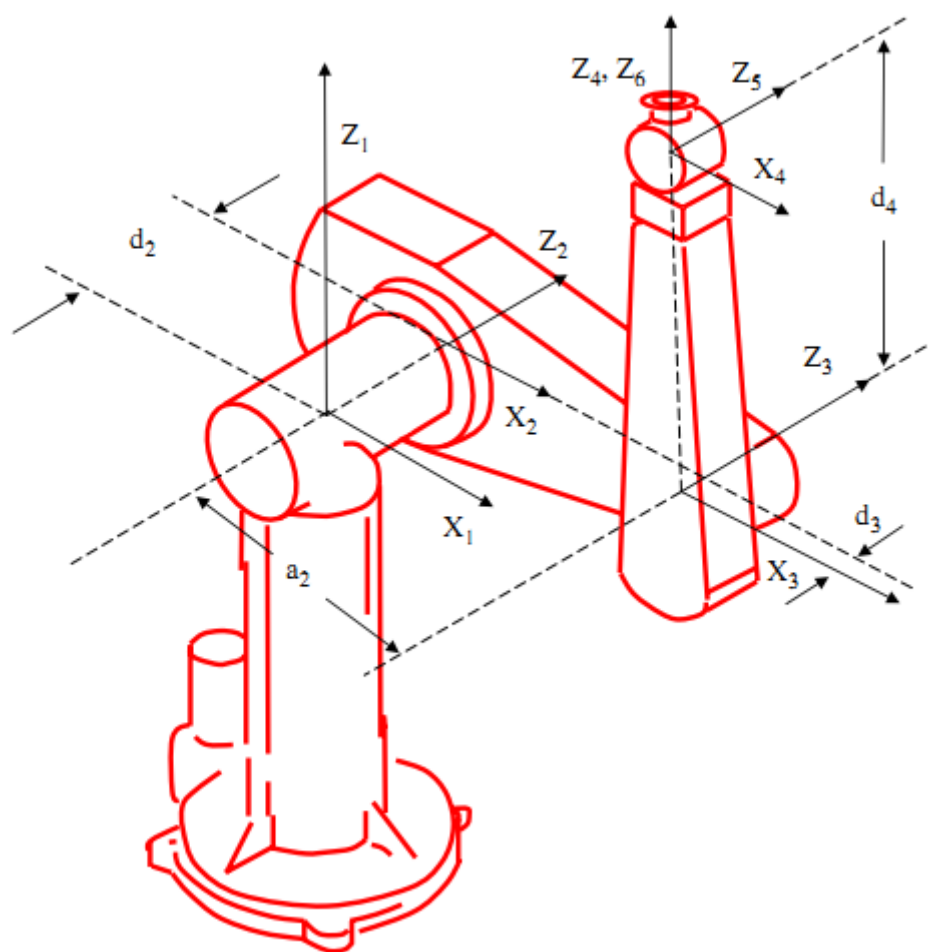


Figura 13 - Puma560 na posição de "descanso"

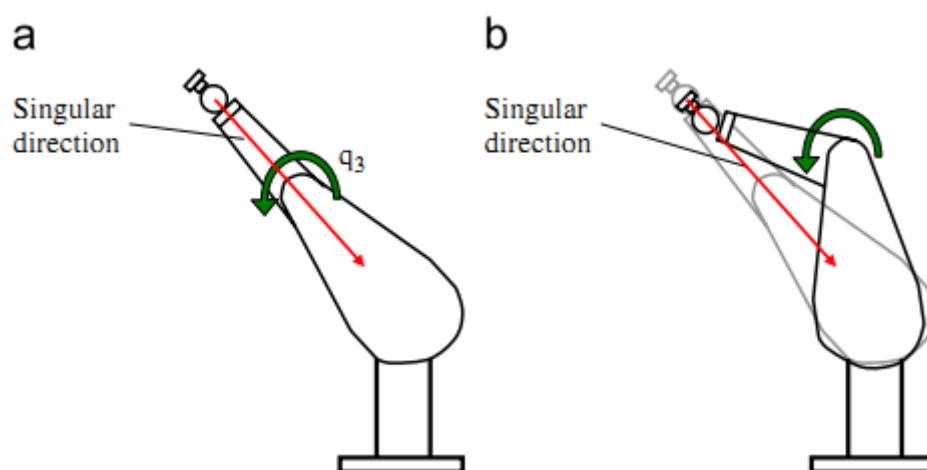


Figura 14 - Puma560 em "elbow lock"

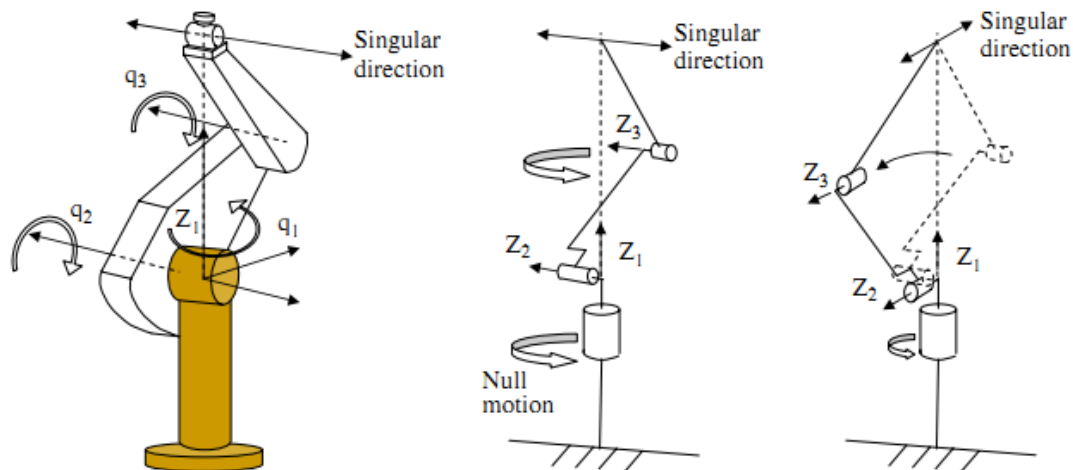


Figura 15 - Puma560 em "head-lock"

3.2. Estudo das singularidades NOTESnail

O robô Puma560 tem uma configuração muito importante especial que lhe permite dissociar "braço" e "pulso". Esse procedimento se torna muito útil dado que a distância entre as articulações no punho são muito menores do que as distâncias no braço.

Portanto, a arquitetura Puma560 implica que o posicionamento será uma tarefa principalmente realizada pelas três primeiras articulações e orientação será feito principalmente pelas três últimas articulações, que combinadas são chamadas de "junta esférica". A junta esférica também tem a propriedade especial que todos os eixos de rotação se cruzam em um ponto comum.

Já que o NOTESnail tem o mesmo número de juntas, e todas do mesmo tipo (de rotação), vale a hipótese de uma abordagem semelhante da posição e orientação dissociadas. Entre as 4ª e 5ª articulações, a distância é desprezível, mas a distância entre a 5ª e a 6ª juntas corresponde a um terço do comprimento do robô total. Isso significa que a posição e orientamento serão tarefas executadas por todas as seis articulações.

Deste ponto em diante, é conveniente fazer a distinção de orientação e "mira". Usando a convenção de Denavit-Hartenberg, é recomendado que o eixo z no sistema de coordenadas do efetuador

seja igual a direção de "ataque", que no nosso caso também é igual ao sentido último módulo, como mostrado na Figura 15. Por conseguinte, para o propósito desta tese, a "mira" será definida como o eixo z do efetuador, expresso no sistema de coordenadas da base. Orientação será dada pela matriz de rotação do efetuador, expressas no quadro de coordenadas base.

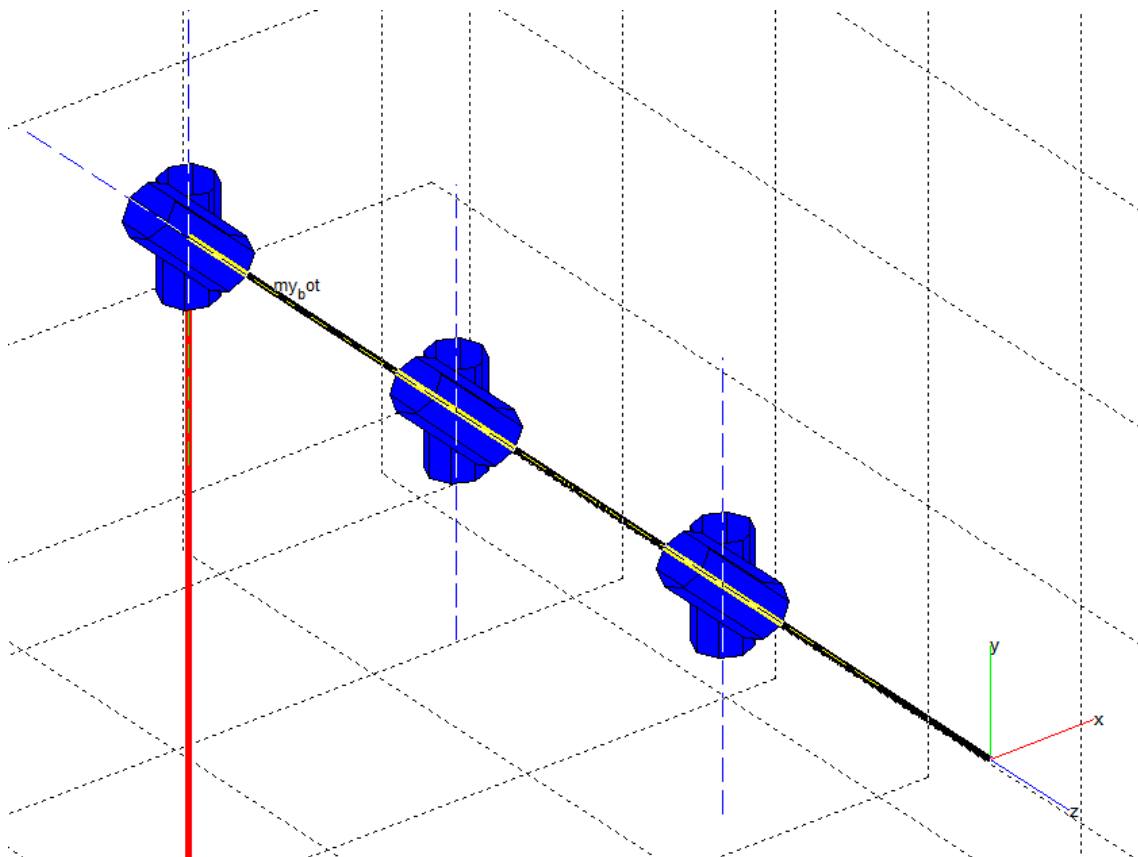


Figura 16 - Representação do NOTESnail em MATLAB

Analisando as transformações homogêneas entre os sistemas de coordenadas entre a sexta articulação e o efetuador (mostrado na equação 3), pode-se inferir que a junta 6 não desempenha nenhum papel no posicionamento ou na "mira" do efetuador. A última junta de revolução é o recurso mais fácil para o cirurgião orientar corretamente uma pinça, por exemplo.

$$A_6^e = \begin{bmatrix} \cos \theta_6 & -\sin \theta_6 & 0 & 0 \\ \sin \theta_6 & \cos \theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad [3]$$

Supõe-se que a amplitude de movimento da 6ª articulação será capaz de fornecer todas as orientações necessárias para o cirurgião, em um dado objetivo, portanto, esta última junta será ignorada na análise das seções posteriores

3.2.1. Modelando o espaço de trabalho com geometria analítica

Cada um dos módulos NOTESnail tem duas articulações: um mecanismo de flexão na parte proximal e um mecanismo de torção na parte distal. Nesta seção, dois diferentes espaços de trabalho serão modelados: como se o efetuator fosse colocado no final do módulo 1 e como se fosse no final do módulo 2.

Estes modelos serão representados por uma equação de geometria analítica e uma representação gráfica. Ambos os modelos serão úteis para verificar se uma determinada configuração é alcançável, avaliar singularidades, construir algoritmos de otimização e muito mais, como mostrado nas próximas seções.

O primeiro espaço de trabalho, correspondente aos efeitos da 1ª e 2ª articulações é mostrado na Figura 17, e a equação pode ser aproximada por:

$$x^2 + y^2 = d^2 \quad [4]$$

Onde d é o comprimento do modulo e a equação 4 existe somente se $y \leq 0$.

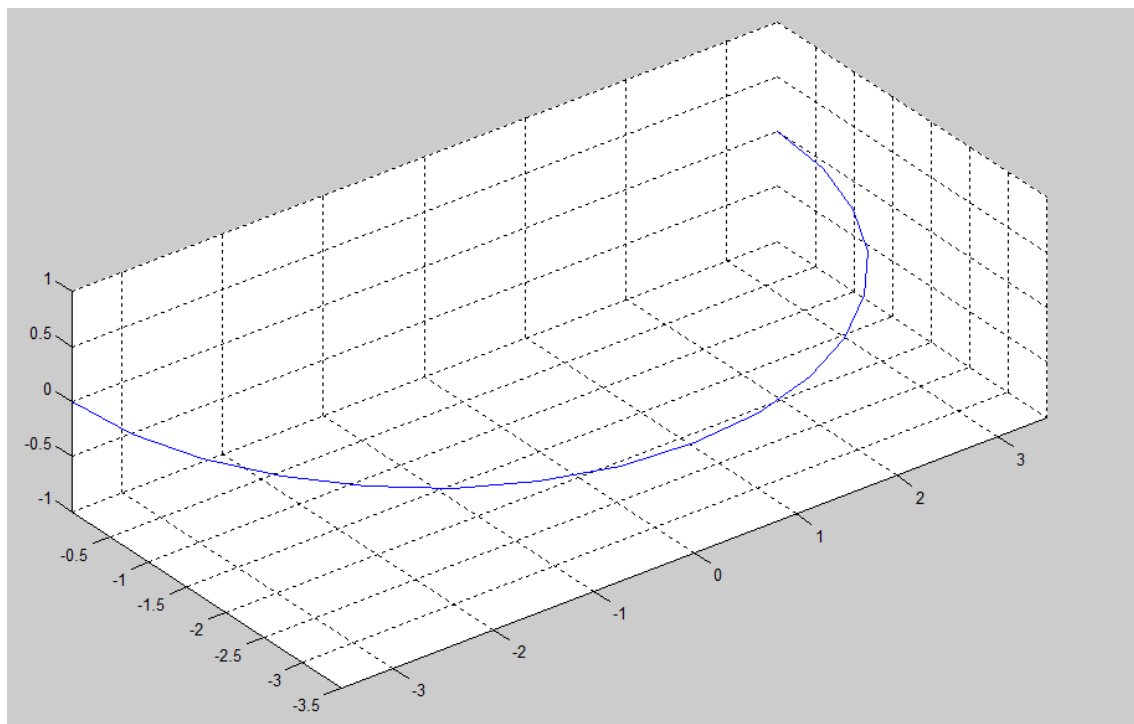


Figura 17 - Espaço de trabalho do primeiro módulo

O espaço de trabalho segundo é um sólido. Inicialmente, a combinação das ações segunda e terceira articulações sobre o segundo módulo iria construir uma superfície em forma de meia esfera, centrada em um ponto no espaço de trabalho do primeiro módulo (dada pelo ângulo da 1ª junta), como mostrado na Figura 18. Se o ângulo 1 conjunto é variada dentro de sua escala, o resultado final é mostrado na Figura 19 e Figura 20.

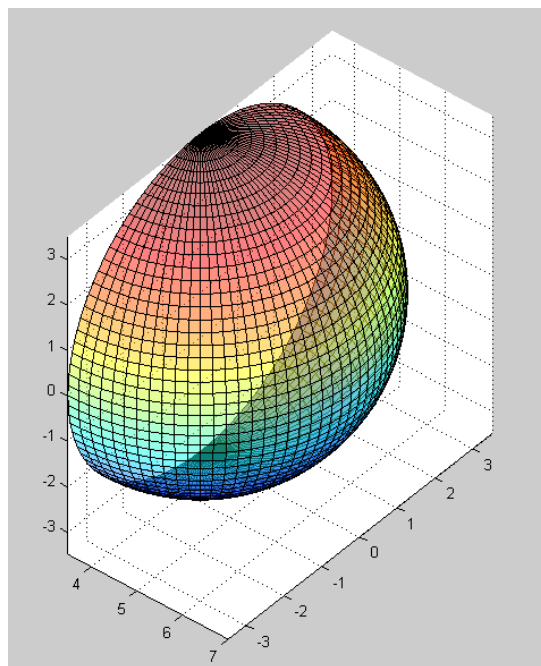


Figura 18 - Ação da 2ª e 3ª juntas quando a 1ª é mantida nula

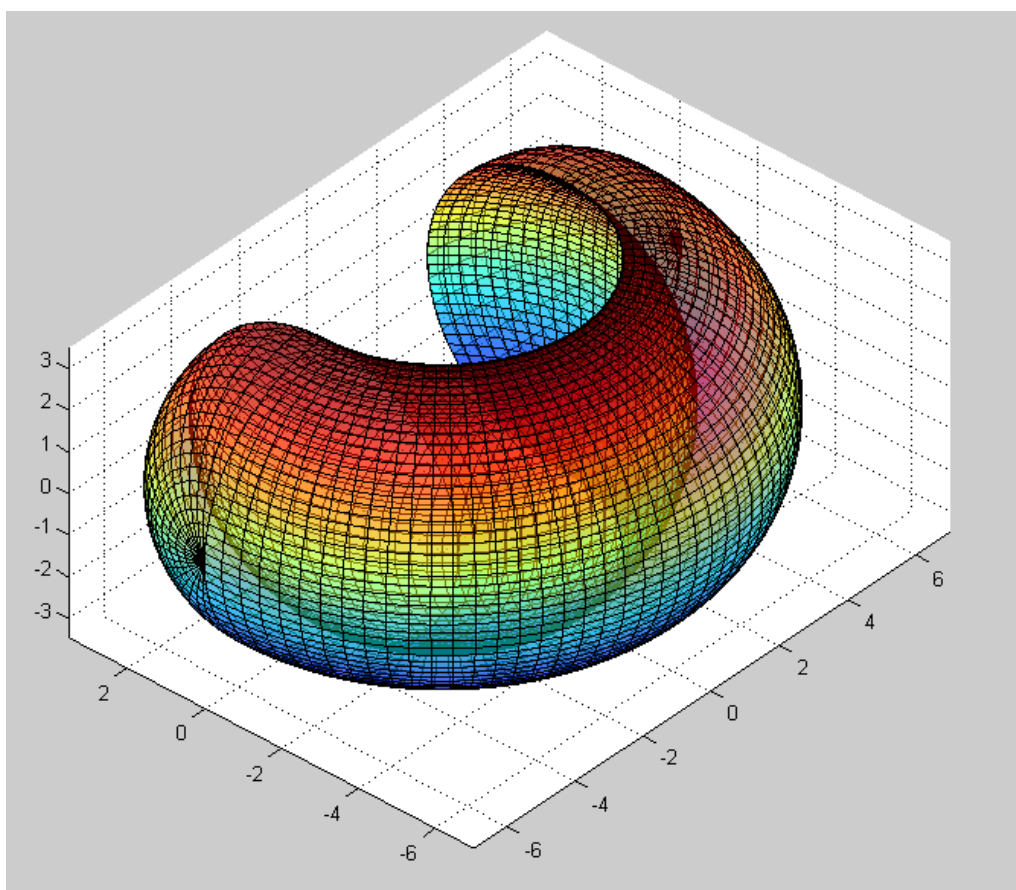


Figura 19 - Espaço de trabalho a dois módulos

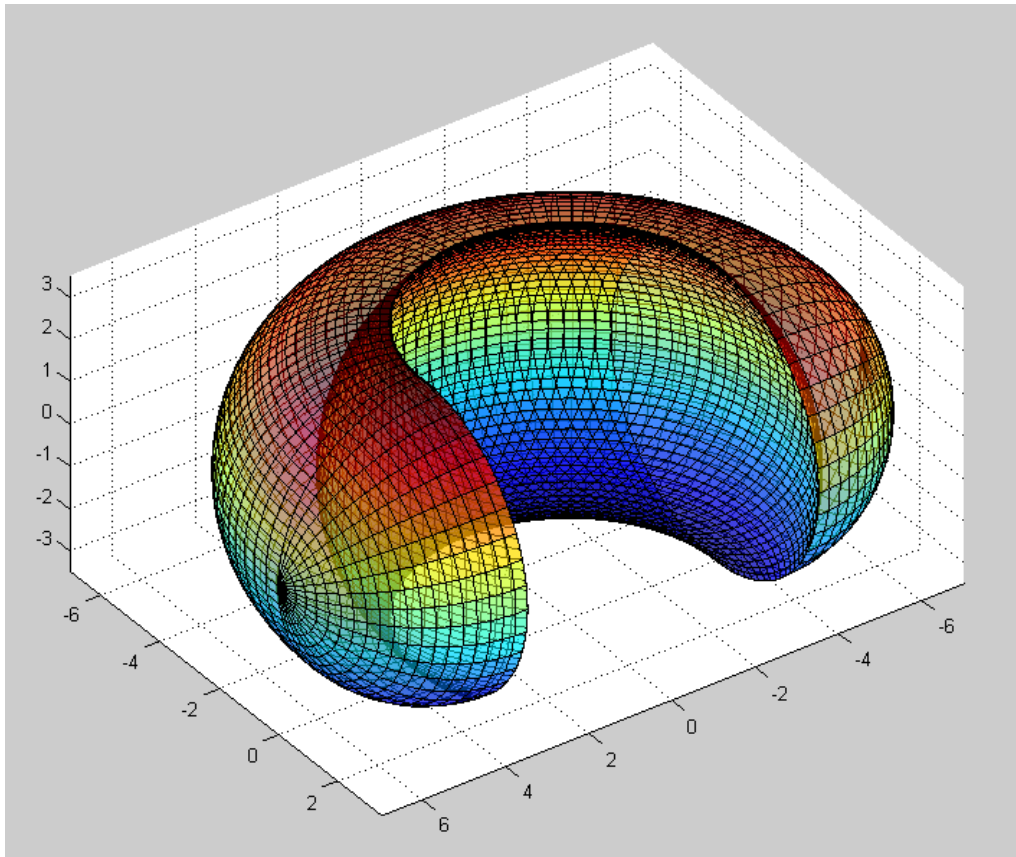


Figura 20 - Espaço de trabalho a dois módulos

O sólido pode ser aproximado por um toro (aproximação válida apenas para $y \leq d$ e $x^2 + y^2 \geq d^2$), mas sua equação geral pode ser dada por [5]

$$x^2 + y^2 + z^2 = 2d\sqrt{x^2 + y^2} \quad [5]$$

3.2.2. Configurações atingíveis e medida de redundância

Como apontado nas seções anteriores, esta análise será baseada no posicionamento e “mira”, ignorando os efeitos da articulação 6.

Se o efetuador está ancorado em um determinado ponto no espaço, os vetores representando todas as miras possíveis somados iriam construir uma esfera centrada naquele determinado ponto (representado na Figura 21 pelo ponto amarelo). Tendo em conta que o vetor com a “mira” tem a mesma direção do eixo longitudinal do módulo, podemos definir o raio da esfera igual ao comprimento do módulo. Como resultado, o modelo geométrico é relevante uma superfície na forma de uma esfera centrada no ponto final desejado

do efetuator com raio igual ao comprimento do módulo (representado na Figura 21 pela esfera vermelha). O modelo geométrico é equivalente à equação 6.

$$(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 = d^2 \quad [6]$$

Onde (x_0, y_0, z_0) são as coordenadas cartesianas do alvo e d é o comprimento do módulo.

Conclusão 1: Se a superfície esférica intersepta o toro sólido, então (x_0, y_0, z_0) é alcançável.

Conclusão 2: A interseção entre a superfície da esfera e o sólido toro-like é uma calota esférica.

Conclusão 3: A área de calota esférica corresponde ao número de possíveis configurações do NOTESnail (ou seja, todas as possíveis “miras”) com o posicionamento desejado (como mostrado na Figura 21). Esse número será chamado **medida de redundância**.

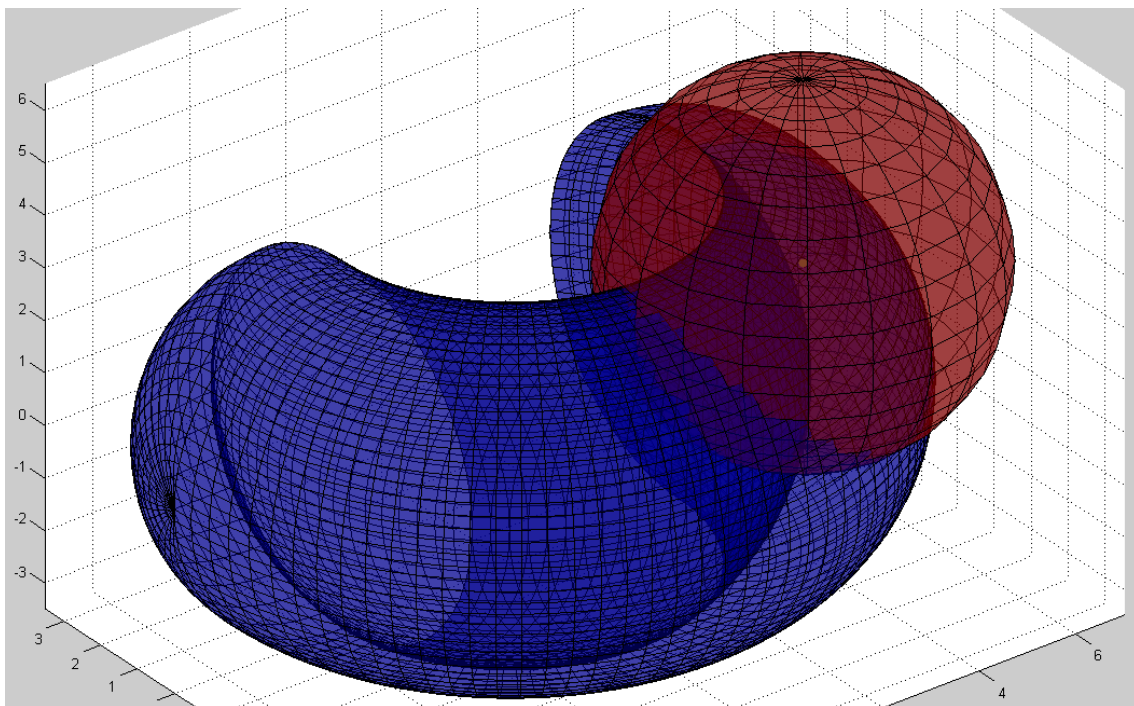


Figura 21 - Ponto atingível (em amarelo)

3.2.3. Configurações singulares NOTESnail

As configurações singulares, teoria introduzida na seção 3.1, foram encontradas utilizando a *screw theory*.

3.2.3.1. 2ª junta – singularidade do tipo I

A 2ª articulação sendo igual a $+\pi/2$ ou $-\pi/2$ representa uma singularidade do tipo I, porque representa a fronteira do espaço de trabalho a 2 módulos. O último módulo é uma parte importante na determinação da “mira” do efetuador e o trabalho de posicionamento se torna uma tarefa dos dois primeiros módulos.

Para analisar os movimentos possíveis quando esta configuração singular tenha sido atingida, o melhor sistema de coordenadas a ser utilizado é o mostrado na Figura 22. Por causa da interpretação de fronteira, é claro que qualquer movimento que é tangente ao espaço de trabalho 2-módulo seria possível (representado por $\hat{\theta}$ e $\hat{\phi}$). Na direção negativa de $\hat{r}$, qualquer movimento ainda é permitido, mas a direção positiva de $\hat{r}$ é impossível, assim como qualquer outra trajetória que contém o vetor como um componente parcial.

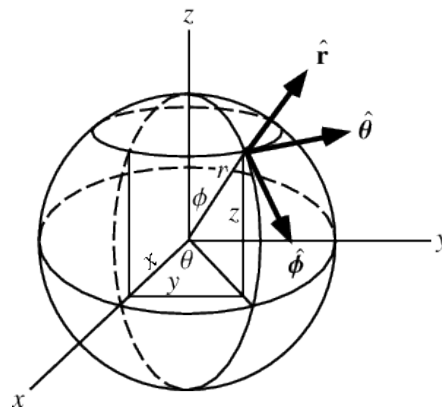


Figura 22 - Coordenadas esféricas

3.2.3.2. 3ª junta – singularidade do tipo I

A singularidade da 3ª junta também representa uma singularidade de fronteira, como mostrado na Figura 23, mas fisicamente, mas também representa a anulação dos efeitos da articulação 2, dada a arquitetura mecânica mostrada na Figura 9. É importante notar que a

singularidade da 3ª junta está contida na singularidade da 2ª junta, tanto no sentido físico quanto no matemático, o que significa que um algoritmo para evitar singularidades para a articulação 2 será também eficaz para a articulação 3.

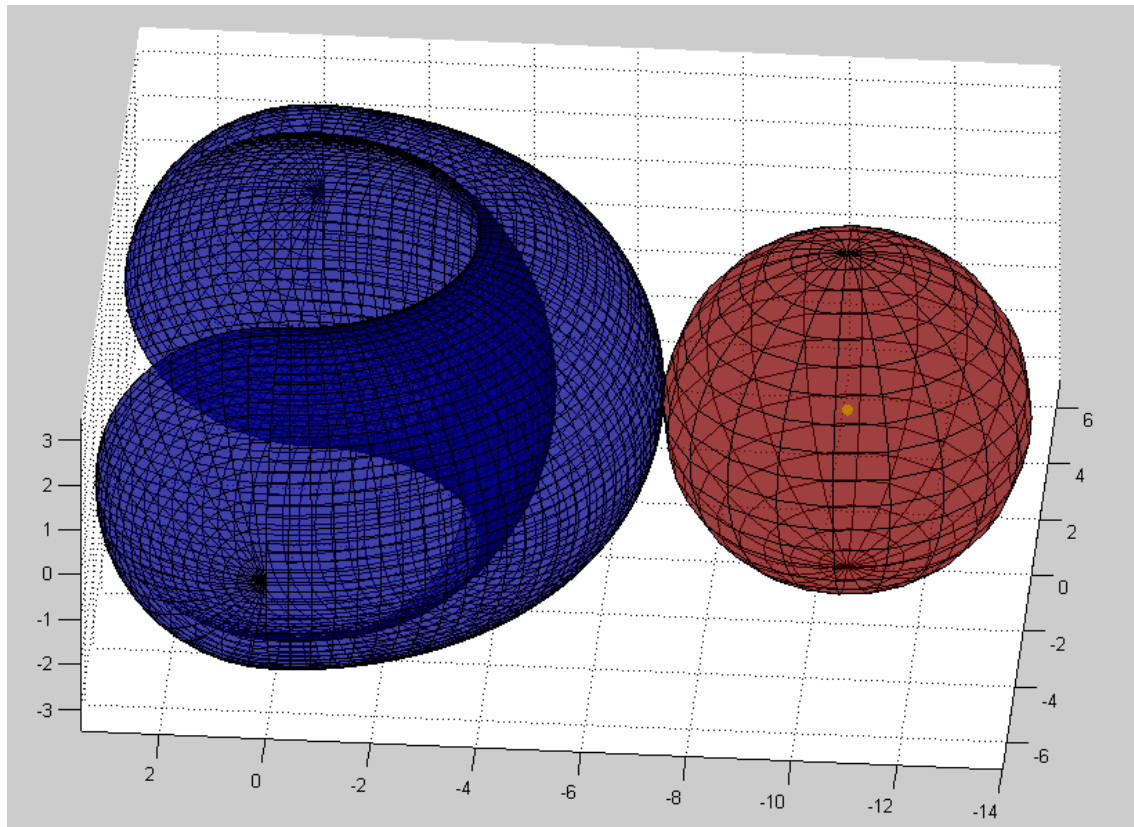


Figura 23 - Singularidade da 3ª junta

3.2.3.3. 5ª junta – singularidade do tipo II

A singularidade da 5ª junta, dada por ($\theta_5 = 0$) representa o alinhamento físico do módulo 2 e 3 e é a única entre as estudadas até agora que é perfeitamente evitável usando uma configuração alternativa, como mostrado na Figura 24. Isto implica que ao aproximar uma configuração singular, o algoritmo de controle desejado criaria um movimento interno para atingir a configuração alternativa sem passar pela singularidade, como mostrado na Figura 24.

A partir deste ponto, alguns problemas surgiram:

- Dados os limites das juntas do robô, nem todas as configurações singulares têm uma configuração não-singular simétrica, como o mostrado na Figura 25
- Dependendo do ângulo de rotação que o robô é introduzido no corpo do paciente, os limites das de rotação podem representar um problema para o movimento interno.
- Ter uma configuração não simétrica singular exigiria o sistema de controle a ser coordenado por um sistema de navegação, o que seria complexo e independente do usuário. Este sistema de navegação teria que ser capaz de medir a posição e orientação do efetuador, a fim de inserir os comandos para criar o movimento interno capaz de evitar a singularidade. Esse sistema não é difícil de ser projetado, mas sua necessidade deve ser bem avaliada de acordo com o comportamento dos algoritmos de cinemática inversa.

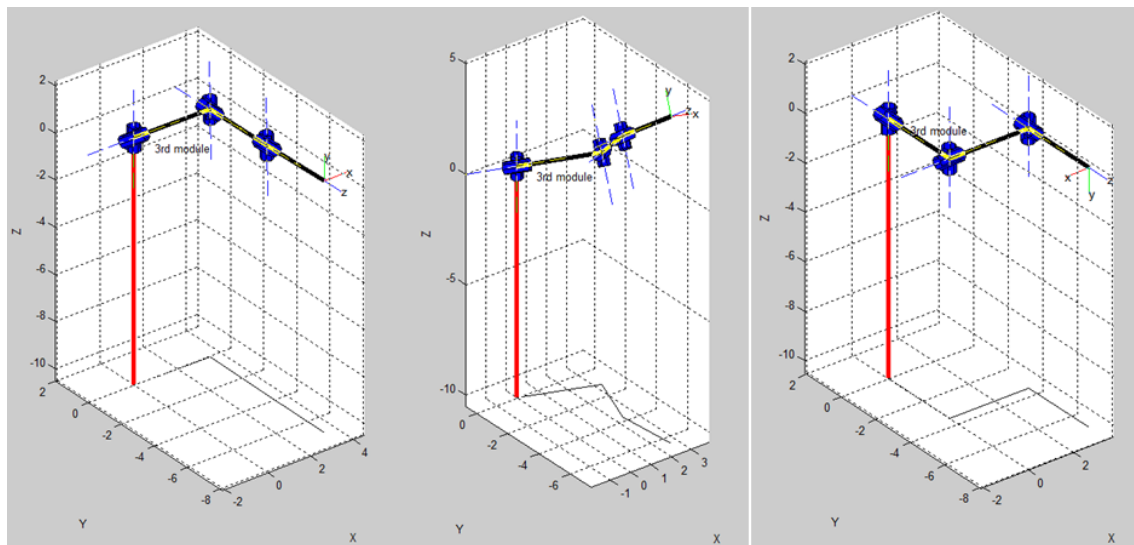


Figura 24 - Movimento de uma configuração singular para a sua alternativa simétrica

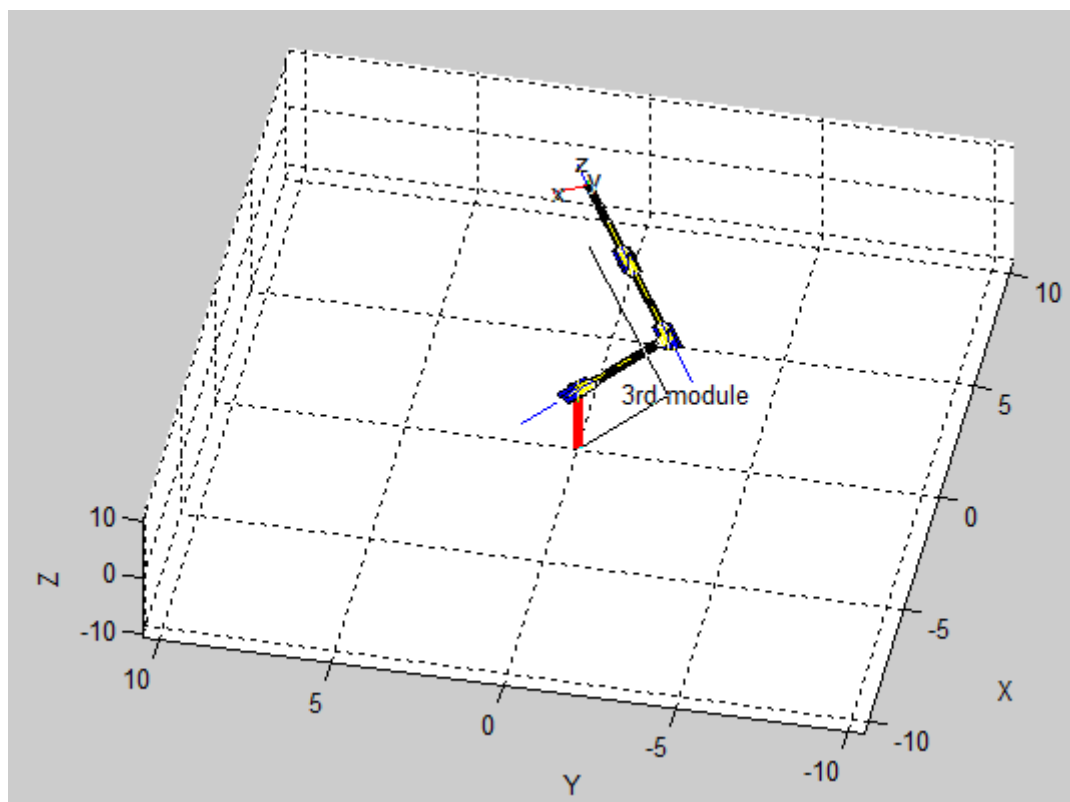


Figura 25 - Configuração singular sem configuração não-singular alcançável

4. Algoritmos de Controle em Robótica

A partir da teoria geral de controle, um controlador manipula as entradas de sistema dinâmico, a fim de obter uma saída desejada chamado de "referência". Um sistema de controle em malha aberta não mede as saídas, a fim de alterar o controle. Por outro lado, um sistema de controle de malha fechada tem sensores cujas medidas são utilizadas para conduzir mudanças dinâmicas no sistema global. (OGATA, 2001)

Um sistema de controle de malha fechada é geralmente representado em um diagrama de blocos, como mostrado na Figura 26. Os objetivos mais comuns de um algoritmo de controle são:

- Robustez à perturbação;
- Melhorar o desempenho em torno de modelos incertos;
- Estabilizar o sistema;

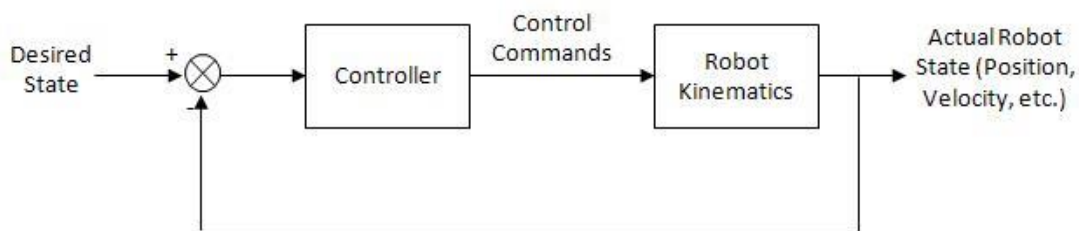


Figura 26 - Esquema a blocos de um sistema a malha fechada

Existem muitas técnicas de controle e metodologias que podem ser aplicados ao controle de manipuladores. O particular método de controle escolhido, bem como a maneira como é aplicado pode ter um impacto significativo sobre o desempenho do manipulador e, conseqüentemente, sobre a gama de suas possíveis aplicações. Por exemplo, rastreamento de caminho contínuo exige uma arquitetura de controle diferente do que faz ponto a ponto de controle. (SPONG, HUTCHINSON e HUTCHINSON, 2004)

Além disso, o projeto mecânico do manipulador em si vai influenciar o tipo de esquema de controle necessário. Por exemplo, os problemas encontrados com o controle de um manipulador cartesiano são

fundamentalmente diferentes daqueles encontrados com um manipulador tipo "cotovelo". Nas seções 4.1 a 4.3.5, alguns esquemas de controle (ou seus componentes importantes) são apresentados.

Uma vez que cada esquema de controle é baseado em um valor de referência (dado como entrada), um paradigma importante que precisa ser explorado é o planejamento de trajetória versus entrada em tempo de um humano-controlador. O primeiro é principalmente usado em muitos robôs industriais e autônomos, já que suas tarefas são programáveis e serão executadas por diversas vezes.

No planejamento de trajetória (ou geração), o problema básico é mover o manipulador de uma configuração inicial para uma configuração final (talvez através de alguma outra configuração). A trajetória será um conjunto de posições, velocidades e acelerações para cada grau de liberdade. O constrange pode ser espacial, temporal e/ou sobre a lisura.

As soluções para o problema de geração de trajetória pode estar no espaço comum ou no espaço cartesiano.

Prover uma solução no espaço articular tem a vantagem de evitar a singularidade e menor custo computacional, mas a solução pode não seguir uma linha reta. Por outro lado, uma solução no espaço cartesiano tem problemas de descontinuidade, como por exemplo, muitas configurações inalcançáveis e singulares ao longo do caminho. Muitos autores têm proposto soluções para o planejamento de trajetória que são boas, rápidas e confiáveis usando interpolação matemática ou algoritmos genéticos (ABO-HAMMOUR, MIRZA, *et al.*, 2002). Algumas destas soluções incluem o desvio de singularidades e/ou obstáculos, muito útil para tarefas pré-programadas.

O problema é que o cirurgião controlando o NOTESnail dará as entradas em tempo real para o sistema de controle no espaço cartesiano, que é o espaço intuitivo para os seres humanos. Além disso, tanto a referência quanto o "sensor de feedback" serão o

próprio cirurgião, portanto, o controle eletrônico será necessário apenas para estabilizar o sistema

4.1. Computed torque control

Esta é uma técnica de controle dinâmico clássica onde o modelo dinâmico do corpo rígido é invertido para calcular a demanda de torque do robô com base nas atuais velocidades e ângulos das juntas e nas acelerações desejadas (PAUL, 1981)

No caso do kit de ferramentas de simulação utilizada neste trabalho, a dinâmica inversa de NOTESnail é calculado usando o algoritmo recursivo de Newton-Euler. Como os parâmetros dinâmicos não são exatamente conhecidos, o autor do toolkit recomenda a introdução de uma perturbação. (CORKE, 2008)

No entanto, os algoritmos de *computed torque control* têm como entrada uma trajetória (ou a posição desejada) dada no espaço articular, o que é indesejável para o nosso usuário. Para mover a partir do espaço cartesiano (user-friendly) para o espaço articular, tanto a cinemática inversa quanto a inversa da função Jacobiana são obrigatórios.

4.2. Visual servoing control

Visual servoing é o conceito de que a visão da máquina pode fornecer controle de posição em malha fechada para o efetuator de um robô. A tarefa é usar a informação visual para controlar a pose do efetuator em relação a um objeto-alvo ou um conjunto de características-alvo. (HUTCHINSON, HAGER e CORKE, 1996)

Servovisão depende de uma série de outras disciplinas, tais como: transformações de coordenadas, velocidade de um objeto rígido; modelos de projeção da câmera; características da imagem e da imagem do espaço parâmetro características e configuração da câmera (montada no efetuator ou fixa na área de trabalho)

Há uma taxonomia, introduzido por (SANDERSON e WEISS, 1980), que categoriza os sistemas servo visual em quatro esquemas diferentes (mostrados nas Figura 27, Figura 28, Figura 29 e Figura 30), respondendo a duas perguntas:

- É a estrutura de controle hierárquica, com o sistema visual fornecendo set-points como entrada para o controlador do robô no nível das juntas, ou se o controlador visual calcular diretamente as entradas no ível das juntas?
- É o sinal de erro definida em coordenadas 3D (espaço tarefa), ou diretamente em termos de características de imagem?

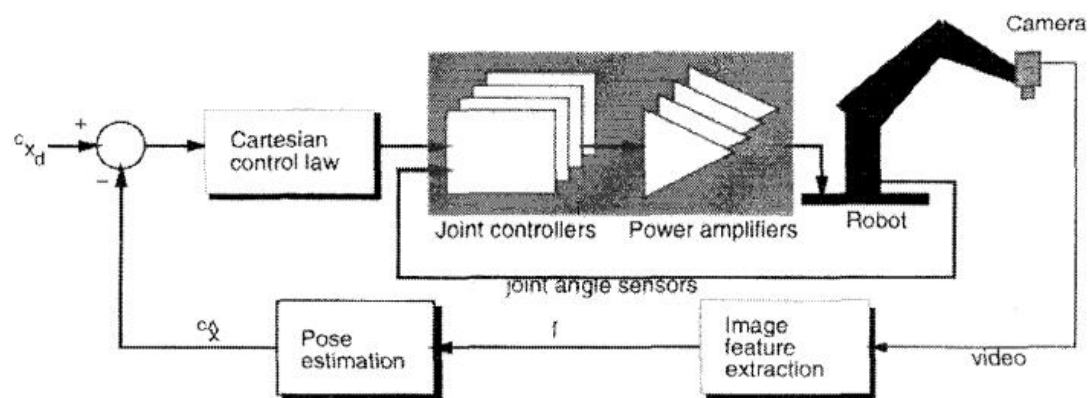


Figura 27 - Estrutura dinâmica look-and-move baseada na posição

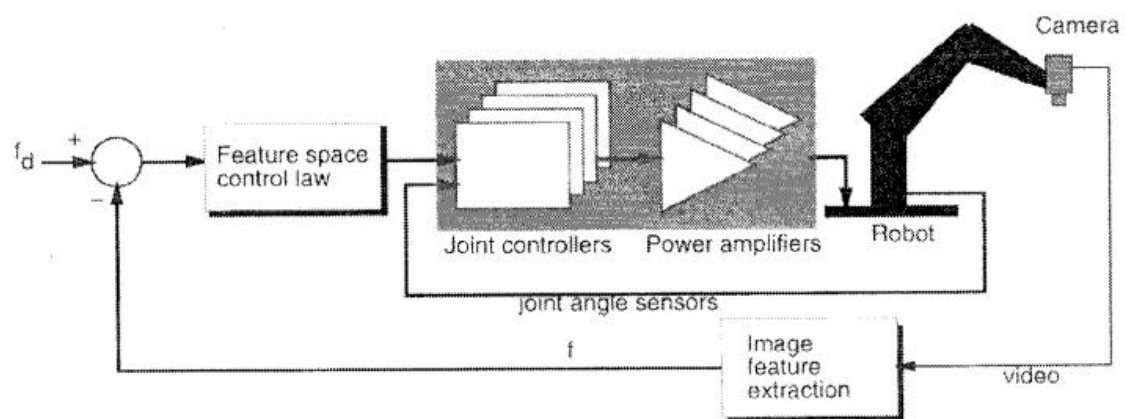


Figura 28 - Estrutura dinâmica look-and-move baseada na imagem

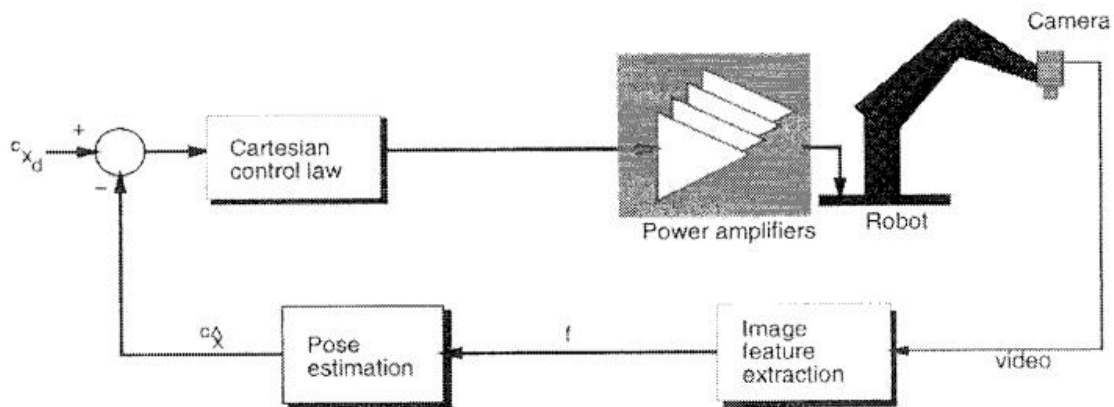


Figura 29 - Visual servo baseado na posição

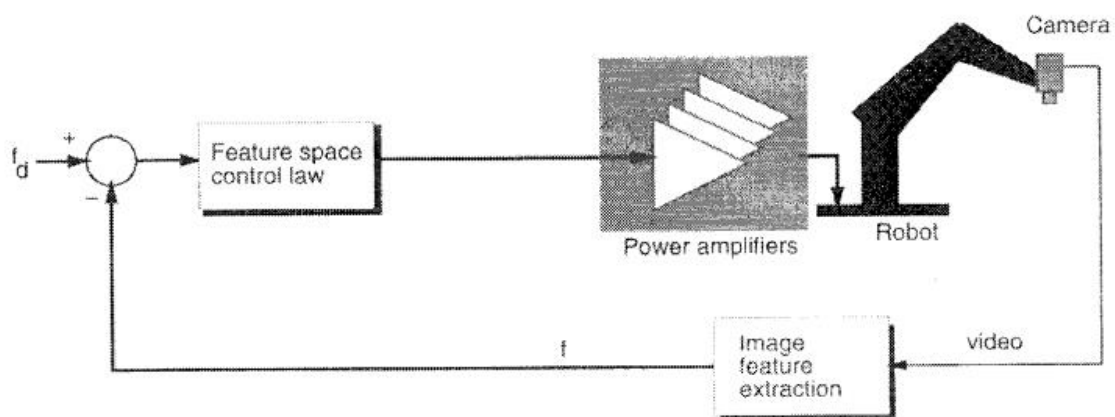


Figura 30 - Visual servo baseado na imagem

Hoje, as aplicações desta tecnologia permanecem limitadas devido aos elevados custos da tecnologia e as habilidades necessárias para construir um sistema robótico integrado e visualmente controlado (HUTCHINSON, HAGER e CORKE, 1996), mas ainda é uma ferramenta muito útil para NOTESnail.

4.3. Controle Cartesiano

Um objetivo importante no controle de um manipulador robótico é garantir que o efetuador siga uma trajetória desejada no espaço cartesiano (tarefa), próxima tanto quanto possível. Este é um problema difícil porque, na descrição da dinâmica do manipulador, os torques de controle são definidos em termos de variáveis de junta e não em variáveis do efetuador (MISRA, PATEL e BALAFOUTIS, 1988)

Controle do manipulador é, portanto, normalmente realizada no espaço das juntas. A trajetória desejada no espaço tarefa é transformada em trajetória no espaço articular em um número suficiente de pontos a fim de garantir que qualquer erro no rastreamento da trajetória espaço articular não causará que o efetuador se desvie significativamente da trajetória desejada no espaço tarefa. No entanto, uma vez que a relação entre as variáveis de junta e do efetuador é, em geral, altamente não-linear, é possível que um pequeno desvio em uma trajetória espaço articular possa causar um desvio relativamente grande na trajetória correspondente do espaço-tarefa.

O esquema básico para um controle cartesiano utiliza:

- A **referência**: uma trajetória dada em transformações homogêneas do efetuador;
- O **erro** é a diferença entre a referência e a posição real (dada pelo encoders de ângulos das juntas além de um algoritmo de cinemática direta);
- Este erro é transformado em um **vetor diferencial** (6×1) equivalente a um vetor coluna de velocidades cartesianas;
- Esse vetor é multiplicado pelo **inverso do Jacobiano** do manipulador. O resultado é um vetor coluna de velocidades das juntas;
- Se este vetor de velocidades das juntas é integrado, a configuração atual conjunto é alcançada e retornada ao sistema, conforme apresentado na Figura 31.

4.3.1. Método da Transposição da Jacobiana

A idéia básica é muito simples: use a transposta de J em vez da inversa de J . Ou seja, vamos definir $\Delta\theta$ igual a:

$$\Delta\theta = \alpha J^T \cdot e; \quad [7]$$

para algum escalar α apropriado. Agora, é claro, a transposição do Jacobiano não é o mesmo que o inverso, no entanto, é possível justificar o uso da transposição em termos de forças virtuais. (BALESTRINO, DE MARIA e SCIAVICCO, 1984)

4.3.2. Método da Jacobiana pseudo-invertida

O método da pseudoinverse define o valor $\Delta\theta$ como:

$$\Delta\theta = J^* \cdot e; \quad [8]$$

Onde a matriz $(n \times m)$ J^* é a pseudoinverse de J , também chamada a inversa de Moore-Penrose. Esta inversão pode ser usada para todas as matrizes, mesmo aquelas que não têm posto completo. A pseudoinverse dá a melhor solução possível para a equação $J \cdot \Delta\theta = e$ no sentido dos mínimos quadrados.

(BUSS, 2004) afirma que:

Seja $\Delta\theta$ definido pela equação [8]. Primeiramente, suponha-se que e esteja dentro da escala de J . Neste caso, $J \cdot \Delta\theta = e$; além disso, $\Delta\theta$ é o único vetor de menor magnitude que satisfaz $J \cdot \Delta\theta = e$. Posteriormente, suponha que e não está na escala de J . Nesse caso, $J \cdot \Delta\theta = e$ é impossível. No entanto, $\Delta\theta$ tem a propriedade de minimizar a magnitude da diferença $J \cdot \Delta\theta - e$. Além disso, $\Delta\theta$ é o único vetor de menor magnitude que minimiza $\|J \cdot \Delta\theta - e\|$.

A pseudoinversa não está imune a problemas de estabilidade nas vizinhanças de singularidades. Se a configuração é exatamente a uma singularidade, então o método pseudoinversa vai se comportar bem e não vai tentar um movimento em uma direção impossível. No entanto, se a configuração está perto de uma singularidade, então o método pseudoinversa vai levar a mudanças muito grandes nos

ângulos articulares, mesmo para pequenos movimentos na posição de destino.

A pseudoinverse tem a propriedade adicional de que a matriz $(I - J^*J)$ executa uma projeção sobre o espaço nulo de J . Logo, para todo e qualquer vetor $\boldsymbol{\varphi}$, $J.(I - J^*.J).\boldsymbol{\varphi} = 0$. Isto significa que pode-se definir $\Delta\boldsymbol{\theta}$ de acordo com a equação [9].

$$\Delta\boldsymbol{\theta} = J^*.\mathbf{e} + (I - J^*.J) \boldsymbol{\varphi} \quad [9]$$

Para qualquer vetor $\boldsymbol{\varphi}$ ainda pode-se obter um valor para $\Delta\boldsymbol{\theta}$ que minimize o valor de $J.\Delta\boldsymbol{\theta} - \mathbf{e}$. Ao escolher valores especiais para $\boldsymbol{\varphi}$, pode-se atingir objetivos secundários além do seguimento da trajetória desejada. Por exemplo, $\boldsymbol{\varphi}$ pode ser definido para retornar os ângulos das juntas para a posição de “descanso”, o que pode ajudar a evitar configurações singulares. (GIRARD e MACIEJEWSKI, 1985)

Vários autores têm utilizado o método de espaço nulo para ajudar a evitar configurações singulares através da maximização da medida manipulabilidade (YOSHIKAWA, 1985). Autores como (MACIEJEWSKI e KLEIN, 1985) utilizaram o método espaço nulo para evitar obstáculos.

4.3.3. Método dos mínimos quadrados amortecidos

O método dos mínimos quadrados amortecidos pode ser justificado em teoria como o seguinte: Ao invés de encontrar o mínimo vetor $\Delta\boldsymbol{\theta}$ que dá a melhor solução para a equação $\mathbf{e} = J \Delta\boldsymbol{\theta}$, pode-se encontrar o valor de $\Delta\boldsymbol{\theta}$ que minimiza a quantidade definida pela equação [10]

$$||J.\Delta\boldsymbol{\theta} - \mathbf{e}||^2 + \lambda^2 ||\Delta\boldsymbol{\theta}||^2, \quad [10]$$

Onde $\lambda \in \mathbb{R}$ é uma constant de amortecimento não-nula. Utilizando a decomposição em valores singulares, pode-se mostrar que $J^T.J + \lambda^2.I$ não é singular. Logo, a solução dos mínimos quadrados amortecidos é igual a equação [11].

$$\Delta\theta = (J^T.J + \lambda^2.I)^{-1}.J^T.\vec{e} \quad [11]$$

Agora, $J^T.J$ é uma matriz de dimensão n -por- n , onde n é o número de graus de liberdade. É fácil demonstrar que $(J^T.J + \lambda^2.I)^{-1}.J^T = J^T.(J.J^T + \lambda^2.I)^{-1}$. Então,

$$\Delta\theta = J^T.(J.J^T + \lambda^2.I)^{-1}.\vec{e} \quad [12]$$

A vantagem da equação [12] sobre a equação [11] é que a inversão da matriz é executada em uma matriz de dimensões m -por- m ao invés de uma matriz de dimensões n -por- n (onde m é o número de graus de liberdade e n é o número de juntas, que é normalmente maior).

4.3.4. Método da Decomposição em Valores Singulares

Seja A uma matriz real m -por- n com $m \geq n$:

$$A = U\Sigma V^T \quad [13]$$

Onde:

$$U^T.U = V^T.V = V.V^T = I_n \text{ e } \Sigma = \text{diag}(\sigma_1, \dots, \sigma_n).$$

A matriz U consiste de n autovetores orthonormalizados associados com os n maiores autovalores de $A.A^T$, e a matriz V é composto dos autovalores orthonormalizados de $A^T.A$. Os elementos da diagonal de Σ são as raízes não negativo quadrados dos autovalores de $A^T.A$, pois eles são chamados de valores singulares. Supõe-se que:

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_n \geq 0$$

Logo, se o posto de $A = r$, $\sigma_{r+1} = \sigma_{r+2} = \dots = \sigma_n = 0$. A decomposição apresentada na equação [13] é chamada *singular value decomposition* (SVD). (GOLUB e REINSCH, 1970)

Uma das aplicações do método SVD é o cálculo da pseudoinversa X , que satisfaz as seguintes quatro propriedades:

- $A.X.A = A$;
- $X.A.X = X$;
- $(A.X)^T = A.X$;
- $(X.A)^T = X.A$.

A única solução é definida por A^+ . É fácil verificar que $A = U.\Sigma.V^T$, logo, $A^+ = V.\Sigma^+.U^T$ onde $\Sigma^+ = \text{diag}(\sigma_i^+)$ e

$$\sigma_i^+ = \frac{1}{\sigma_i} \text{ para } \sigma_i > 0$$

$$\sigma_i^+ = 0 \text{ para } \sigma_i = 0$$

4.3.5. Método de Otimização

Existem muitos algoritmos de otimização disponíveis para o cálculo da cinemática inversa. Para o propósito deste trabalho, foi escolhida a minimização da função não-linear multivariável constrangida. As variáveis a serem otimizadas são os ângulos de articulações e a função a ser minimizada é:

$$f(q) = \text{abs}(\text{norm}(tr - \text{fkine}(\text{robot}, q)))$$

Onde "abs" calcula o valor absoluto dentro dos parênteses, "norm" calcula a norma da matriz dentro dos parênteses e "fkine" calcula a cinemática direta do "robot", enquanto na configuração definida por "q".

Em comparação com os métodos de cinemática inversa que usam qualquer variação da matriz Jacobiana, o método de otimização tem a vantagem de respeitar os limites articulares, mas a desvantagem de maior custo computacional. Ambos os métodos têm problemas de perto singularidades, dependendo da escolha da estimativa inicial.

5. Resultados Experimentais

Ao lidar com aplicações cirúrgicas de robôs, especialmente tendo uma entrada de controladores humanos para o movimento do manipulador, é muito importante ter um controle de movimento robusto nas singularidades, a fim de evitar erros de rastreamento, bloqueio ou qualquer outro comportamento indesejado que colocaria em risco a vida de um paciente durante uma cirurgia.

Para uma melhor escolha de um algoritmo de controle de movimento, uma série de testes foi realizada a fim de analisar o comportamento de cada método apresentado nas seções 4.3.1 a 4.3.5. O protocolo do teste envolve uma série de etapas desde a construção de uma trajetória até a coleta dos resultados, que serão apresentados nas seções seguintes.

5.1. Trajetórias

As trajetórias foram escolhidas em duas perspectivas diferentes: na primeira, deve-se tentar imitar os movimentos possíveis que um cirurgião executaria dentro da sala de operação; na segunda, o comportamento do robô em condições de singularidade extrema têm de ser testados, mesmo que essas situações não se apliquem em uma situação do mundo real.

Como mencionado nas seções anteriores, as trajetórias simuladas têm de ser definidas no espaço cartesiano, já que esses são os insumos naturais comando para o controlador humano. Mas, para se certificar de que as trajetórias são compostas apenas de configurações acessíveis, todas elas são testadas através de um algoritmo de otimização.

Este algoritmo de teste usa a função mesmo objetivo da cinemática inversa na seção 4.3.5, mas retorna um array (com variáveis de tempo) de coordenadas no espaço das juntas e um array (com variáveis de tempo) dos valores da função objetivo em cada

determinado conjunto de coordenadas. Em outros termos, dá uma medida do erro durante a trajetória.

Todas as trajetórias de teste a seguir têm um rastreamento de erro inferior a 10^{-5} metros e a norma da velocidade linear máxima é inferior a 0,5 cm/s.

5.1.1. Pick and Pull

A trajetória pick-and-pull é uma trajetória muito comum na sala de cirurgia. Literalmente centenas de vezes o cirurgião tem de pegar um pedaço de tecido ou linha de sutura e puxa-la em outra direção. A trajetória proposta segue uma linha reta ao longo da direção positiva do eixo z.

A orientação é mantida paralelamente ao plano X-Z com uma rotação de 45 graus negativos em torno do eixo Y, como mostrado na Figura 32. A distância total percorrida é de 2 centímetros, a partir de [3; -5; -4,5]. A trajetória em coordenadas de junta é mostrada na Figura 33:

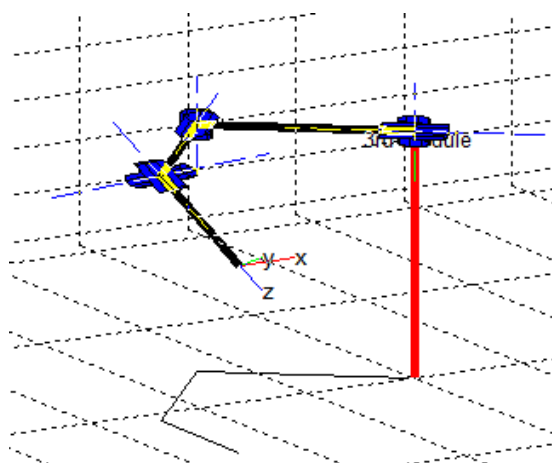


Figura 32 - Configuração final da trajetória 1

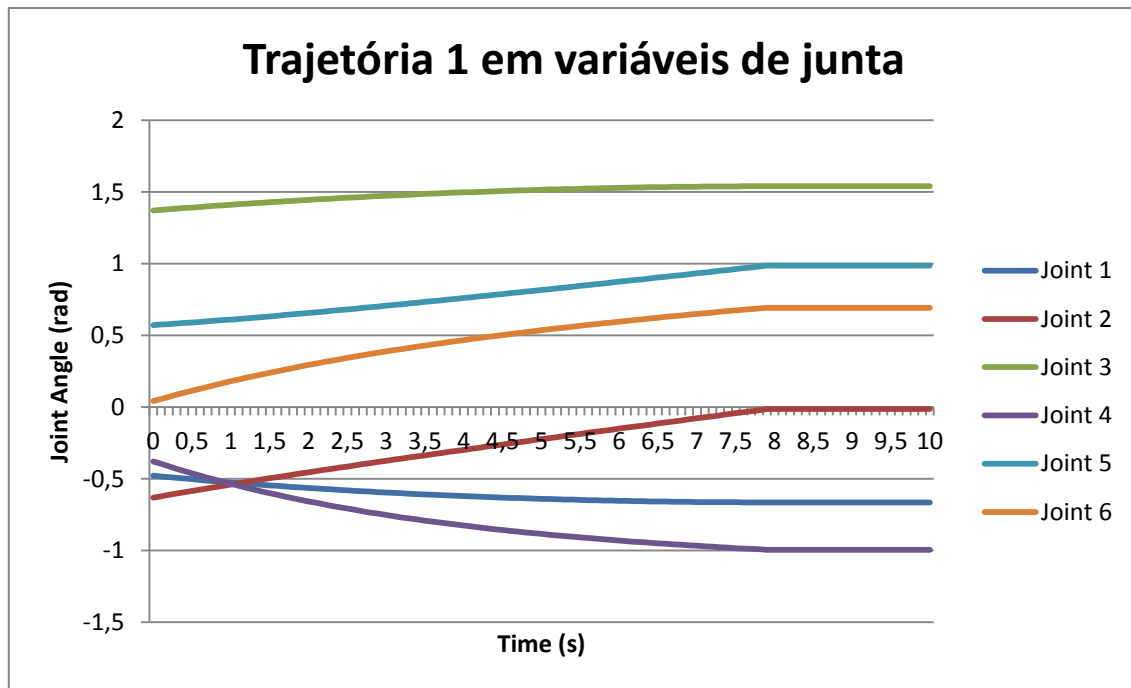


Figura 33 - Trajetória 1 em coordenadas de junta

5.1.2. Trajetória de teste de singularidade do tipo I

Esta é uma trajetória que não é comparável às aplicações do mundo real, mas é importante como um teste de condição extrema. Nesta situação, o ponto de partida não era uma rota de coordenadas cartesianas: o procedimento era para bloquear os ângulos das articulações 1, 3, 4, 5 e 6, enquanto variando a articulação 2 dentro de sua escala completa.

Isto faz com que NOTESnail passe por uma singularidade do tipo I duas vezes: quando q_2 é igual a $-\pi/2$ e $\pi/2$, como mostrado na Figura 34.

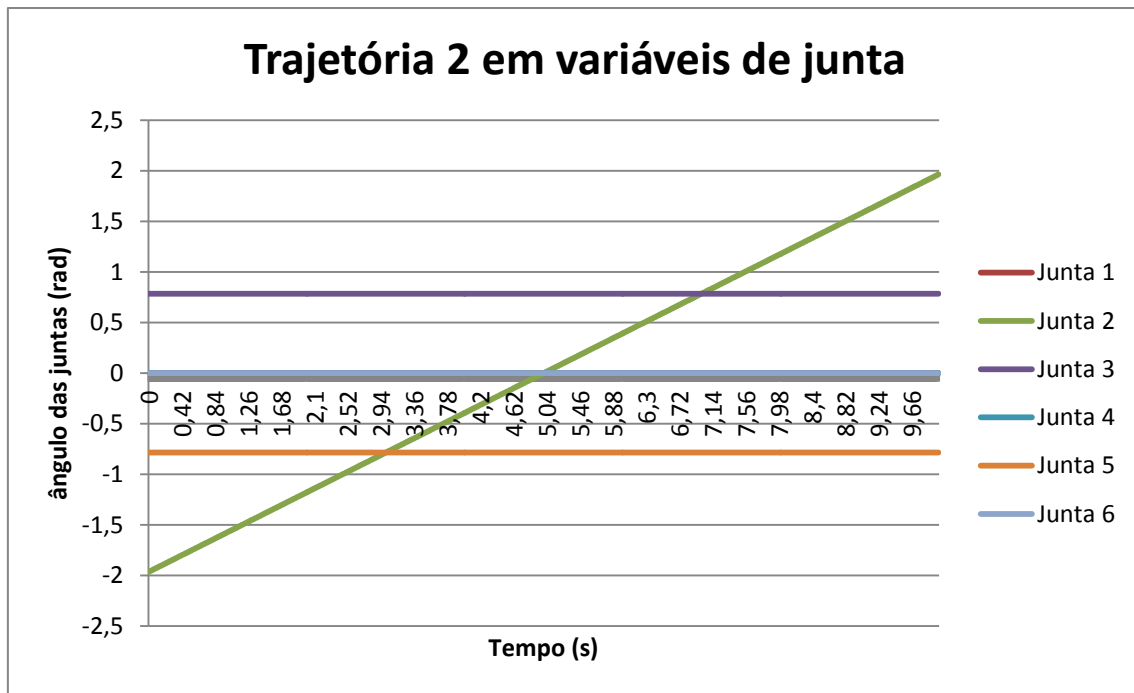


Figura 34 - Trajetória 2 em variáveis de junta

5.1.3. Trajetória de teste de singularidade do tipo II

Essa trajetória segue o mesmo princípio da trajetória 5.1.2 a respeito da aplicação no mundo real, mas em vez de variar a segunda junta, a quinta junta é a ser estudada utilizando o procedimento exatamente equivalente. A visualização das variáveis de junta é mostrada na Figura 35 (note que algumas coordenadas das juntas não podem ser visualizadas uma vez que todos têm o mesmo valor "0").

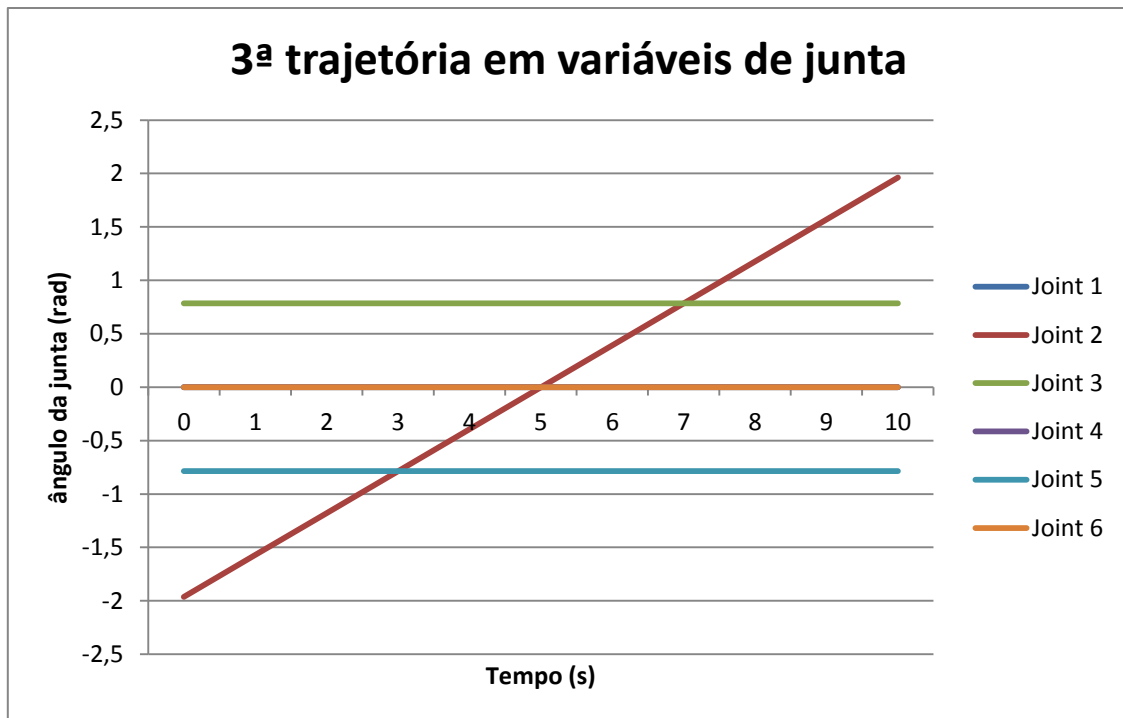


Figura 35 - 3ª trajetória em variáveis das juntas

5.1.4. Simulação da Sutura

Outra trajetória amplamente utilizada na sala de cirurgia é a sutura, como mostrado na Figura 36. Os movimentos mostrados nas seções A, C, D, E e F são alcançados com o movimento "pick and pull" exibido anteriormente. O movimento mais complicado é mostrado na Figura 40-B, quando o cirurgião faz uma trajetória semelhante a uma hélice, a fim de "enrolar" a linha de sutura ao redor do corpo da pinça.

Esse movimento pode ser matematicamente aproximado com uma hélice em torno eixo z do efetuador. De acordo com a prática da cirurgia comum, em vez de manter a orientação constante durante a trajetória completa, ela será usada para ajudar o movimento circular. Juntas 1, 2 e 3 serão utilizadas para o movimento linear ao longo do eixo z do último módulo. As articulações 4 e 5 serão usadas para o movimento circular da hélice.

As trajetórias no espaço de articulação e no espaço cartesiano são mostrados na Figura 37 e Figura 38 respectivamente.

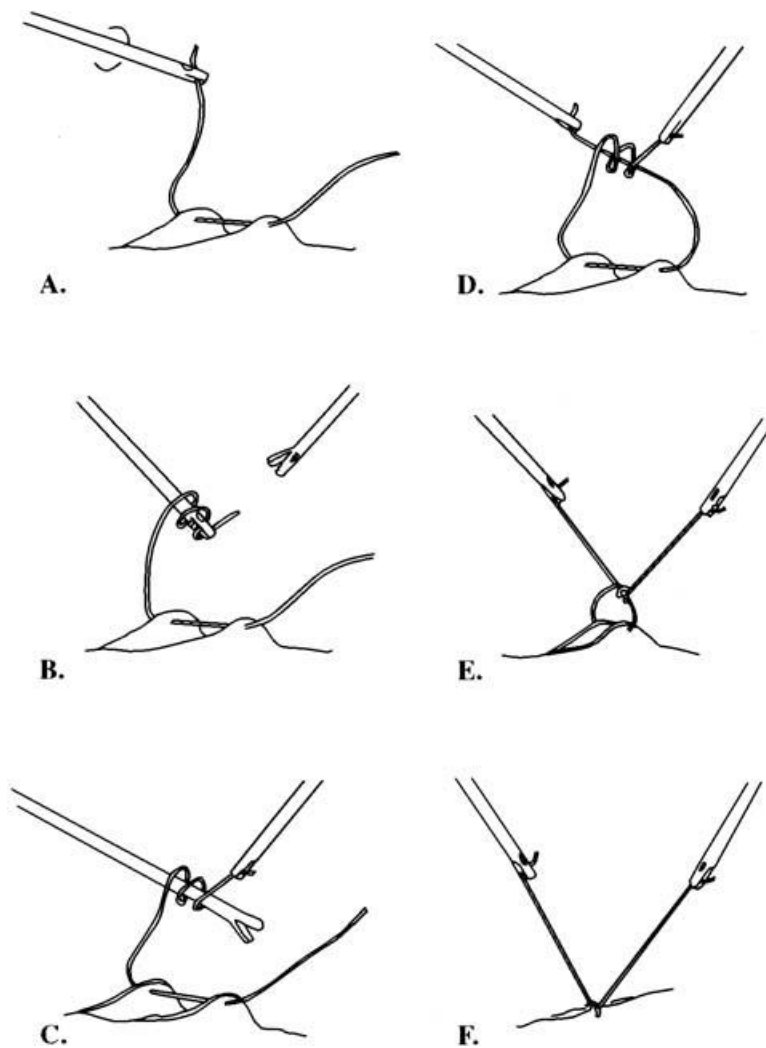


Figura 36 - Movimento de sutura laparoscópica

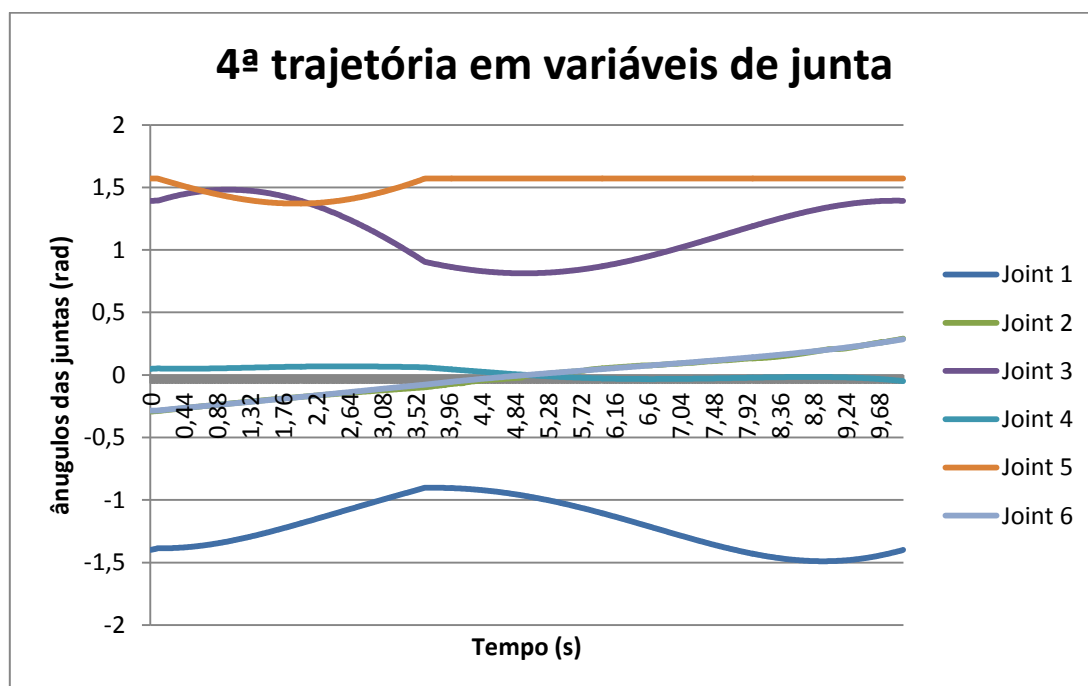


Figura 37 - 4ª trajetória em variáveis de junta

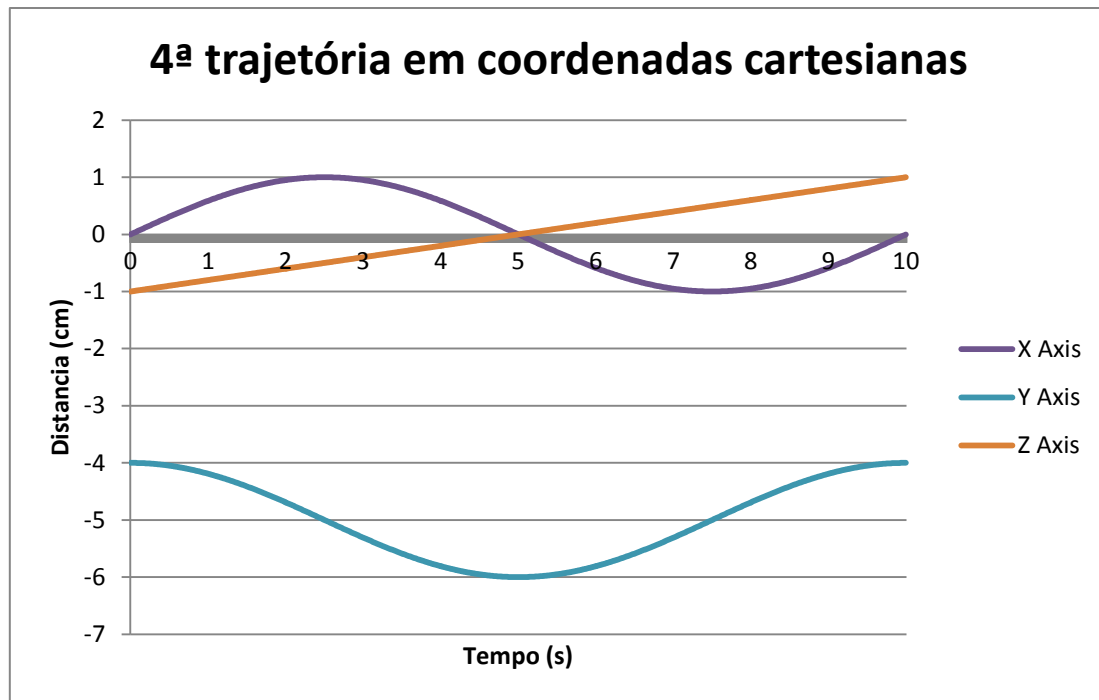


Figura 38 - 4ª trajetória em coordenadas cartesianas

5.2. Modelos Simulink

Simulink, desenvolvido pela Mathworks é uma ferramenta para modelagem, simulação e análise de sistemas dinâmicos. A sua interface principal é baseado em diagramas de blocos, tornando fácil a mudança arquiteturas, parâmetros e etc. Essa foi a principal razão para escolhê-lo para o propósito deste trabalho.

Em todos os modelos, os blocos são os seguintes:

- trajetória da cartesiano é obtido a partir de um arquivo MATLAB (resultado de um arquivo de "construção de trajetória");
- a diferença entre a posição atual ea posição desejada é calculado na forma de um vetor diferencial (6x1);
- este vetor diferencial cartesiano é transformado em um vetor do espaço articular diferencial de acordo com um dos métodos;
- A operação proporcional integrativa é realizada. O ganho pode ser alterado conforme o desejado;
- A cinemática direta opera as variáveis de volta para o espaço cartesiano;

- O bloco "plot" fornece ao usuário uma visualização do movimento.

5.2.1. Método da Transposta

O método de transposição é auto-explicativo. As variáveis de juntas são tomadas em cada passo para o cálculo do Jacobiano do manipulador, que será transposto para multiplicar o vetor diferencial cartesiano.

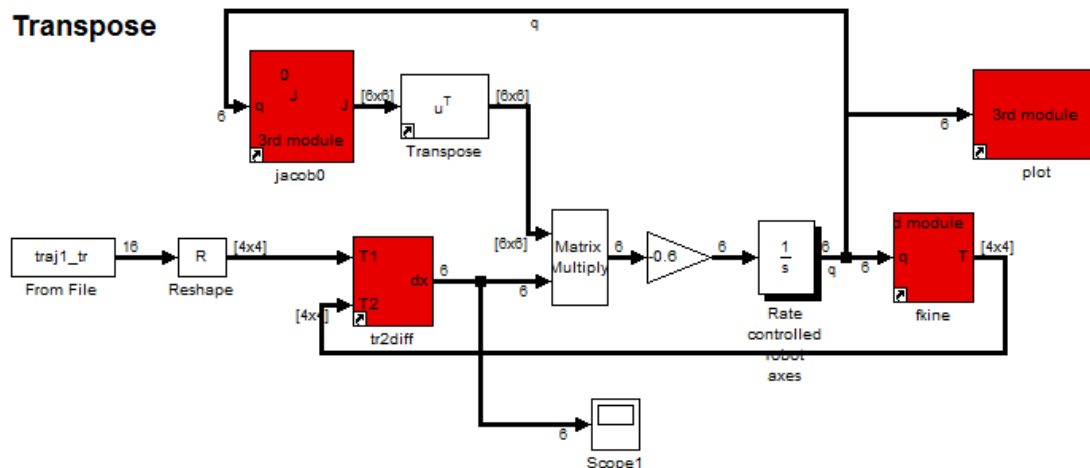


Figura 39 - Modelo Simulink do método da transposta

5.2.2. Método da Pseudoinversa

O método da pseudoinversa introduz um número maior de blocos, a fim de executar as operações descritas na seção 4.3.2. O bloco chamado "Subsystem1" na Figura 44 tem uma entrada determinada pelo usuário chamado "vetor espaço nulo", a fim de realizar tal operação.

Pseudoinverse - 1st traj

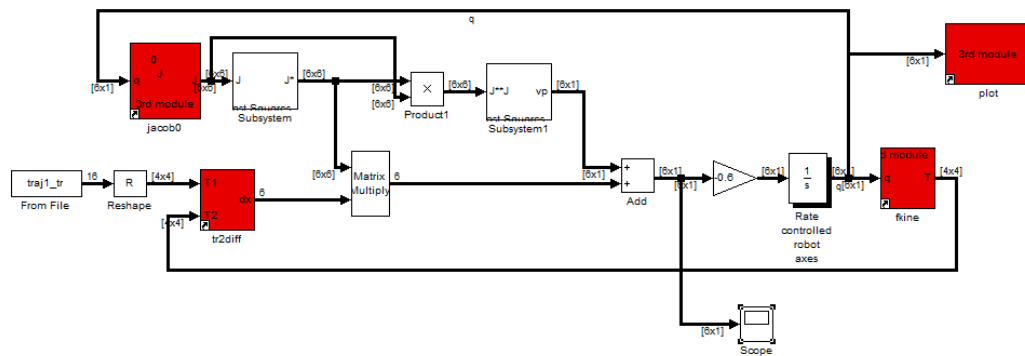


Figura 40 - Modelo simulink para o método da pseudoinversa

5.2.3. Método dos mínimos quadrados amortecidos

Este modelo Simulink também é auto-explicativo. As operações matemáticas propostas na seção 4.3.3 são realizadas com uma entrada definido pelo usuário para a constante de amortecimento.

Damped Least Squares

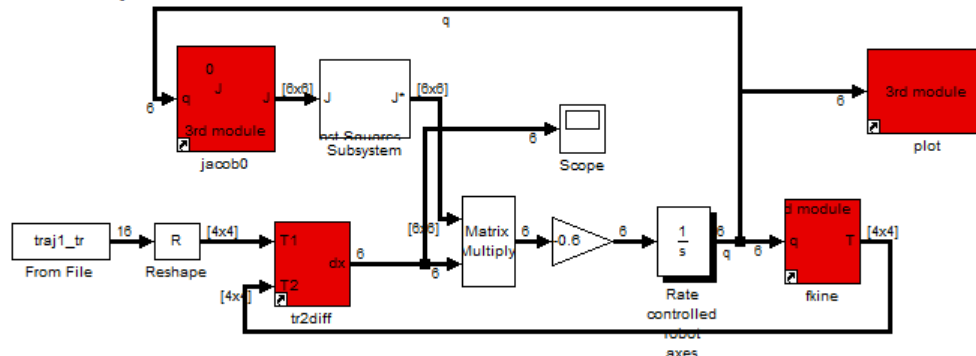


Figura 41 - Modelo Simulink para o método dos mínimos quadrados amortecidos

5.2.4. Método SVD

Para o método SVD, uma função MATLAB foi criada, ao invés de combinar vários blocos Simulink®, tornando o modelo mais claro em geral. Esta função pode ser encontrada no Apêndice A

Optimization

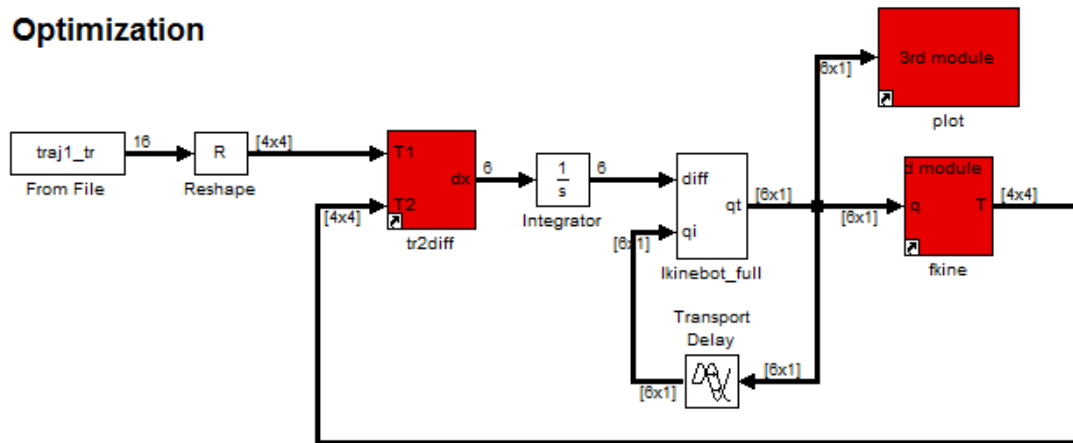
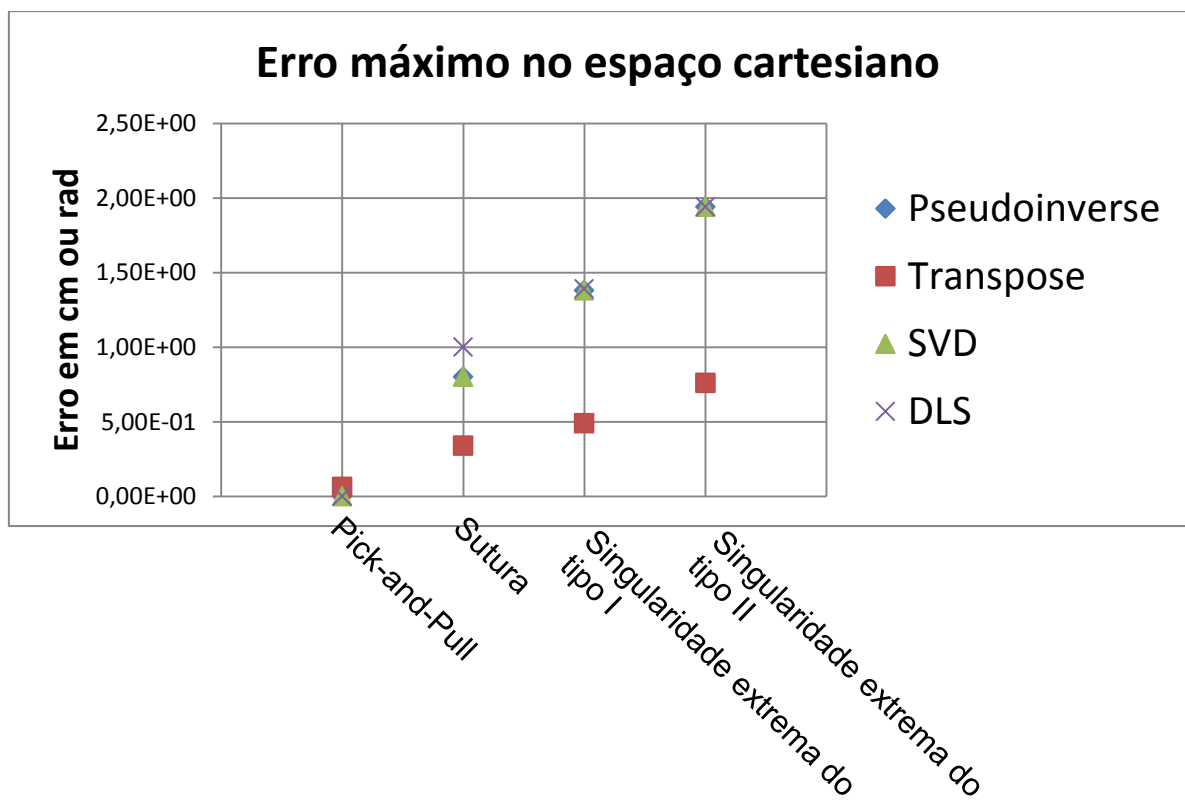
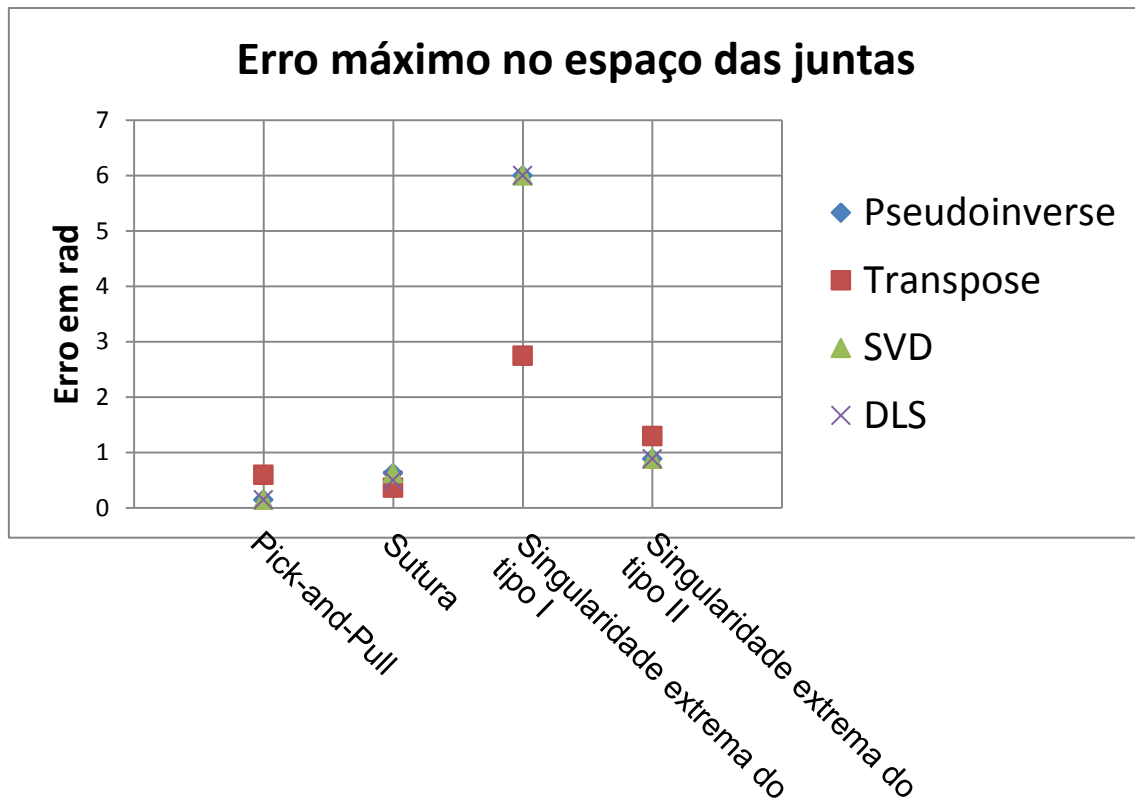


Figura 43 - Modelo Simulink para o método de otimização

5.3. Resultados gráficos





6. Discussão e Conclusão

O objetivo principal deste trabalho foi testar algoritmos de controle cinemáticas em trajetórias cirurgia do mundo real. O algoritmo deve se comportar bem quando na configuração singular seja seguindo a trajetória original no espaço das juntas ou usando outros graus de liberdade para seguir a trajetória no espaço cartesiano.

Como analisado no ponto 3.2.3, as únicas configurações singulares verdadeiramente evitáveis sem alterar a posição ou orientação são as causadas pela anulação do ângulo da 5ª junta. Algumas dessas configurações tem uma configuração simétrica não-singular, e pode-se considerar o movimento entre uma configuração quase-singular e sua simétrica.

A primeira preocupação seria se o movimento interno das juntas afetaria os tecidos circundantes. O que não deve ser um problema desde o local da operação é inflado com dióxido de carbono abrindo espaço suficiente para o manipulador de se mover.

A segunda preocupação é sobre o pequeno movimento do efetuator que é inevitável durante o movimento interno das juntas. Na pior das hipóteses, se o cirurgião está realizando uma sutura ou outro procedimento delicado, será que um deslocamento de 1 cm irá afetar sua técnica ou resultado do paciente?

6.1. Trajetórias

Durante a fase de testes deste trabalho, muitas trajetórias diferentes foram consideradas, a fim de avaliar o desempenho dos algoritmos de controle cinemático. As ferramentas computacionais disponíveis para a visualização 3-D em tempo real do NOTESnail requerem uma entrada de espaço das juntas, o que torna difícil a elaboração de trajetórias baseadas na intuição.

Por outro lado, a abordagem à moda antiga para desenhar trajetórias revelaram importantes limitações cinemáticas no projeto do NOTESnail com 3 módulos:

- Se a abordagem ao projetar uma trajetória inclui manter uma orientação fixa ao mover o efetuator ao longo das coordenadas cartesianas, o espaço de trabalho revelou-se muito limitada. Como mostrado na seção 3.2.2, dada a posição do efetuator e orientação, é possível recuperar a posição da 4ª e 5ª articulações e a configuração é alcançável se e somente se:
 - A posição das articulações 4 e 5 está dentro do espaço de trabalho a 2 módulos mostrada na Figura 19;
 - Os ângulos das juntas 4 e 5 conjuntas necessárias para posicionar o último módulo estão dentro do intervalo permitido.

Estas duas condições tornam o número de trajetórias possíveis com orientação fixa muito limitada quando se considera o tamanho do espaço de trabalho com 3 módulos.

- Conforme mostrado na Figura 18, o intervalo $[-3.\pi/4 ; 3.\pi/4]$ para as juntas de rotação combinada com o intervalo $[-\pi/2 ; \pi/2]$ para a junta de flexão parece ser mais do que suficiente para a construção de uma semi-esfera capaz de atingir qualquer ponto desejado, mas quando se considera estes pontos em uma trajetória, alguns problemas surgem:
 - Considerando-se o movimento de sutura, como mostrado na Figura 36, quando feito com instrumentos comuns de laparoscopia com 4 graus de liberdade. O movimento circular feito em torno da linha de sutura quando analisado em variáveis das juntas (como mostrado na Figura 44) daria um valor contínuo aumento (de 0 a $2.n.\pi$, onde n é o número de rotações de sutura), que não é compatível com limitações dos ângulos de junta do NOTESnail

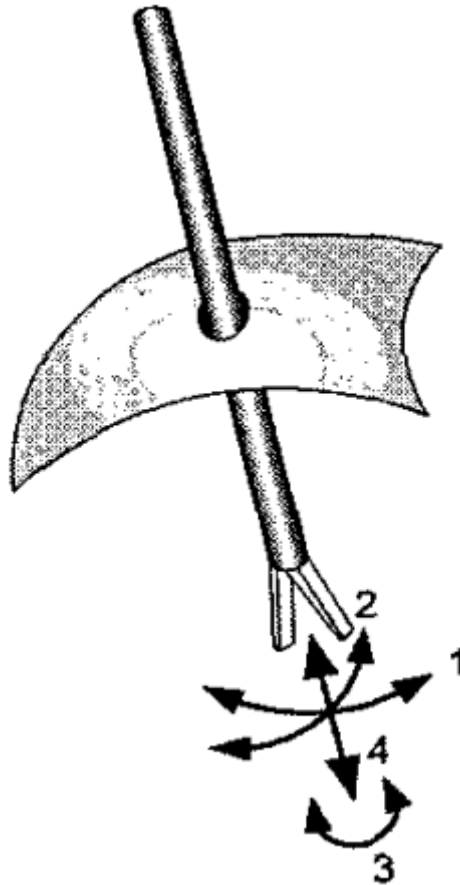


Figura 44 - Variáveis de junta em um instrumento laparoscópico

Comparando todos os algoritmos de cinemática inversa, algumas considerações devem ser apontadas:

6.2. Método da Transposta

O método de transposição tem o pior comportamento entre todas as soluções: ele tem uma convergência lenta e instabilidade a cada passo (resultando em um gráfico comb-like)

No espaço cartesiano, o método de transposição tem uma das piores performances, mas por outro lado, mantém uma distância mais curta para a trajetória esperada no espaço das juntas quando se considera uma trajetória rica em singularidades.

Por outro lado, o fato de seguir a trajetória esperada no espaço das juntas implica que não há evasão das singularidades usando outros graus de liberdade, que foi um dos principais objetivos deste trabalho.

6.3. Método dos mínimos quadrados amortecidos

O método dos mínimos quadrados amortecidos tem performances diversas no espaço cartesiano e no espaço das juntas de acordo com a constante de amortecimento:

- No caso das trajetórias aplicáveis no mundo real, o método DLS teve a mesma ótima performance que os métodos pseudoinverse e SVD (tanto no espaço cartesiano quanto no espaço das juntas). No caso de uma trajetória do mundo real com configurações inatingíveis, o método DLS tem desempenho superior quando comparado aos métodos anteriormente citados.
- Em condições de singularidade extrema, o método DLS tem um desempenho equivalente aos métodos pseudoinverse e SVD ao se analisar o espaço cartesiano e melhor desempenho ao se analisar o espaço das juntas.
- A constante de amortecimento afeta a performance geral quando é variada entre os valores de $[-1;1]$. Fora deste intervalo, o desempenho se torna saturado nos valores extremos. O gráfico qualitativo do desempenho vs. valor da constante de amortecimento tem o formato de uma gaussiana com o valor máximo centrado em 0, como mostrado na Figura 45

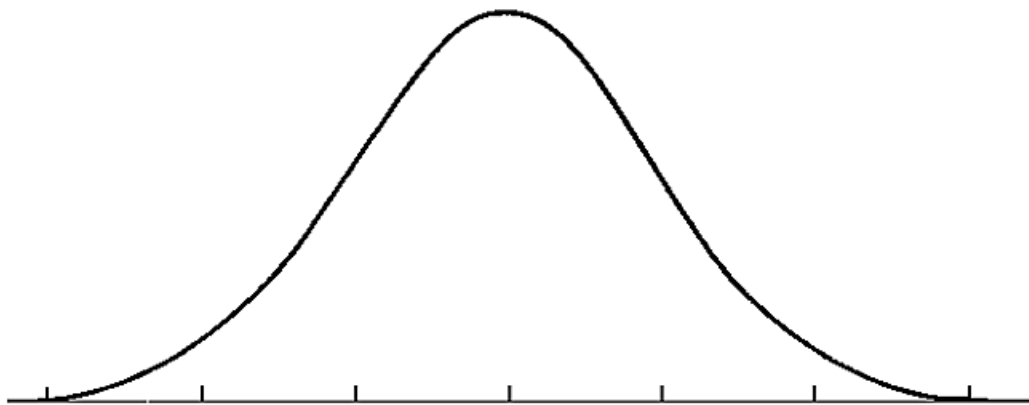


Figura 45 - Valor da constante de amortecimento vs. performance geral

6.4. Singular Value Decomposition

Assim como o método da pseudoinversa, o método SVD é bastante robusto e estável. Ambos métodos tem a performance mais próxima ao esperado, no sentido de evitar singularidades utilizando-se da redundância nos graus de liberdade mantendo-se fiel à trajetória esperada. Sem maiores considerações a serem feitas.

6.5. Método Pseudoinversa

O método da pseudoinverse, assim como o método SVD, tem o comportamento mais próximo ao esperado pelo objetivo deste trabalho. Ambos têm a mesma performance quando o vetor de projeção no espaço nulo tem valor zero, o que aponta a vantagem deste método sobre o anterior.

Ao manipular o vetor de projeção no espaço nulo de movimentos, pode-se evitar configurações singulares para criar movimento interno, o que era o objetivo principal deste trabalho. A melhor manipulação do vetor espaço nulo pode ser objetivo de futuros trabalhos. A única manipulação experimentada nesta tese é apresentada na equação:

$$null_{space} = [0, \frac{\frac{q_2}{\|q_2\|}}{\frac{\pi}{2} - \|q_2\|}, \frac{1}{q_3}, 0, \frac{1}{q_5}, 0] \quad [13]$$

Esta manipulação foi pensada para manter o NOTESnail o mais distante possível de suas configurações singulares. Os problemas surgiram pois tais configurações eram inatingíveis e inatravessáveis e com os atuais 3 módulos e correspondentes 6 graus de liberdade, nem sempre era possível criar movimento interno das juntas para alcançar outras regiões do espaço de trabalho, tornando-o ainda mais reduzido.

6.6. Algoritmo de Otimização

O algoritmo de otimização foi muito bem sucedido no planejamento de trajetórias e na execução da cinemática inversa, mas se mostrou impraticável na execução em tempo real. Os resolutores do Simulink

não progrediam no passo temporal de execução, além de bloquear o sistema operative.

6.7. Considerações finais

Dadas todas as considerações sobre as configurações singulares no sistema NOTESnail e o comportamento dos algoritmos de cinemática inversa, o autor deste trabalho sugere que a prevenção singularidade se beneficiaria com o aumento do número de módulos de construção. O aumento do número de módulos para 4 irá fornecer o sistema com 8 graus de liberdade, 2 a mais do que o necessário para chegar a qualquer ponto com qualquer orientação, o que fornecerá a qualquer algoritmo de cinemática inversa a redundância necessária para atingir o objetivo inicial de evitar a singularidades.

Considerando os dados coletados, a melhor opção para o controle cinemático é o método da pseudoinversa. Este algoritmo é o que melhor responde ao objetivo principal, que era usar outros graus de liberdade para superar a simples inversão da matriz Jacobiana de posto incompleto, assim como se comporta perfeitamente ao longo de trajetórias livres de singularidade.

A possibilidade de explorar vetores de projeção no espaço nulo dá ao método pseudoinversa uma vantagem sobre a Decomposição de Valor Singular e pode ser uma ferramenta poderosa quando o manipulador é forçado em uma singularidade de fronteira. O erro de rastreamento no espaço cartesiano é aceitável, especialmente porque não existem trajetórias geradas por computador, mas um controle visual humano.

Finalmente, este trabalho mostrou a importância de analisar o significado físico de cada singularidade e qual algoritmo de controle cinemático é mais apropriado para compensar um design de 3 módulos que, apesar de muito adequado para a aplicação em cirurgia NOTES, é um projeto muito limitado em termos de trajetórias possíveis.

7. Referências

ABO-HAMMOUR, Z. et al. Cartesian path generation of robot manipulators using continuous genetic algorithms. **Robotics and Autonomous Systems**, 41, n. 4, 31 December 2002. 179-223.

AMERICAN COLLEGE OF SURGEONS. A Guide to Surgical Specialists. **American College of Surgeons**, 2007. Disponível em: <http://www.facs.org/public_info/yourhealth/guide.html>. Acesso em: 26 out. 2011.

AMERICAN SOCIETY OF COLON & RECTAL SURGEONS. Laparoscopic Surgery - What Is It? **Patients & Public Information**, 2008. Disponível em: <http://www.fascrs.org/patients/treatments_and_screenings/laparoscopic_surgery/>. Acesso em: 26 out. 2011.

BALESTRINO, A.; DE MARIA, G.; SCIAVICCO, L. **Robust Control of Robotic Manipulators**. Proceedings of the 9th IFAC World Congress. [S.l.]: [s.n.]. 1984. p. 2435-2440.

BOWMAN, D. E. ASGE/SAGES Working Group on Natural Orifice Translumenal Endoscopic Surgery: White Paper October 2005. **Gastrointestinal Endoscopy**, 63, n. 2, fev. 2006. 199-203.

BUSS, S. R. **Introduction to Inverse Kinematics with Jacobian Transpose, Pseudoinverse and Damped Least Squares methods**. University of California. San Diego, p. 1-19. 2004.

CAPASSO, L. **Principi di storia della patologia umana**: corso di storia della medicina per gli studenti della Facoltà di medicina e chirurgia e della Facoltà di scienze infermieristiche. Roma: SEU, 2002.

CERVERI, P. **Progetto di Ricerca - Modello A**. MINISTERO DELL'ISTRUZIONE DELL'UNIVERSITÀ E DELLA RICERCA. Roma, p. 1-24. 2008.

CORKE, P. Robotics Toolbox for Matlab. **Peter Corke**, December 2008. Disponível em: <<http://www.petercorke.com>>. Acesso em: 15 nov. 2011.

GEOFFREY, B. et al. Nomenclature of Natural Orifice Translumenal Endoscopic Surgery (NOTES) and Laparoendoscopic Single-Site Surgery (LESS) Procedures in Urology. **Journal of Endourology**, 22, n. 11, nov. 2008. 2575-2582.

GIRARD, M.; MACIEJEWSKI, A. A. Computational modeling for the computer animation of legged figures. **Computer Graphics**, 19, 1985. 263-270.

GOLUB, G.; REINSCH, C. Singular value decomposition and least squares solutions. **Numerische Mathematik**, 14, n. 5, 29 abr. 1970. 403-420.

HABER, G.-P. et al. Robotic NOTES (Natural Orifice Translumenal Endoscopic Surgery) in Reconstructive Urology: Initial Laboratory Experience. **Urology**, 71, n. 6, June 2008. 996-1000.

HUTCHINSON, S.; HAGER, G. D.; CORKE, P. I. A tutorial on visual servo control. **IEEE Transactions on robotics and automation**, 12, n. 5, October 1996. 651-67.

INTUITIVE SURGICAL. The da Vinci Surgical System. **Intuitive Surgical**, 2010. Disponível em: <http://www.intuitivesurgical.com/products/davinci_surgical_system/>. Acesso em: 26 out. 2011.

KALLOO, A. N. et al. Flexible transgastric peritoneoscopy: a novel approach to diagnostic and therapeutic interventions in the peritoneal cavity. **Gastrointestinal Endoscopy**, 60, n. 1, jul. 2004. 114-117.

LOULMET, D. et al. Endoscopic coronary artery bypass grafting with the aid of robotic assisted instruments. **Journal of Thoracic Cardiovascular Surgery**, 1999. 118:4-10.

MACIEJEWSKI, A. A.; KLEIN, C. A. Obstacle avoidance for kinematically redundant manipulators in dynamically varying

environments. **International Journal of Robotic Research**, 4, 1985. 109-117.

MISRA, P.; PATEL, R. V.; BALAFOUTIS, C. A. **Robust Control of Robot Manipulators in Cartesian Space**. American Control Conference. [S.l.]: [s.n.]. 1988. p. 1351-1356.

OEMOTO, D.; ANG JR., M. Singularity robust algorithm in serial manipulators. **Robotics and Computer-Integrated Manufacturing**, 25, 12 September 2007. 122-134.

OGATA, K. **Modern Control Engineering**. 4. ed. [S.l.]: Prentice Hall, 2001.

OLEYNIKOV, D. Robotic Surgery. **Surgical Clinics of North America**, 88, n. 5, out. 2008. 1121-1130.

OYAMA, E. et al. **Inverse kinematics learning by modular architecture neural networks with performance prediction networks**. Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on. [S.l.]: [s.n.]. 2001. p. 1006-1012.

PAI, R. et al. Transcolonic endoscopic cholecystectomy: a NOTES survival study in a porcine model. **Gastrointestinal Endoscopy**, 64, n. 3, Setembro 2006. 428-434.

PAUL, R. P. **Robot Manipulators: Mathematics, Programming, and Control**. Cambridge: The MIT Press, 1981. 197-217 p.

RAO, G. V.; REDDY, N.; BANERJEE, R. NOTES: Human Experience. **Gastrointestinal Endoscopy Clinics of North America**, 18, n. 2, abr. 2008. 361-370.

RIEDER, E.; SWANSTROM, L. Advances in cancer surgery: Natural orifice surgery (NOTES) for oncological diseases. **Surgical Oncology**, 20, n. 3, September 2011. 211-218.

ROAN, S. Anesthesia's effects may linger after patient goes home. **Pittsburgh Post-Gazette**, 2005. Disponível em: <<http://www.post-gazette.com/pg/05201/540318.stm>>. Acesso em: 25 out. 2011.

SANDERSON, A. C.; WEISS, L. E. Image-based visual servo control using relational graph error signals. **Proc. IEEE**, 1980. 1074-1077.

SPONG, M.; HUTCHINSON, S.; HUTCHINSON, M. **Robot Dynamics and Control**. 1. ed. [S.l.]: Wiley, 2004. 95-124 p.

TOLANI, D.; GOSWAMI, A.; BADLER, N. Real-Time Inverse Kinematics Techniques for Anthropomorphic Limbs. **Graphical Models**, 62, 2000. 353-388.

VELANOVICH, V. Laparoscopic vs open surgery. **Surgical Endoscopy**, New York, 01 jan. 2000. 16-21.

WAMPLER, C. W. Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators. **Journal of Dynamic Systems, Measurement, and Control**, 110, 1988. 31-38.

WANG, L.-C. T.; CHEN, C. C. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. **Robotics and Automation, IEEE Transactions on**, 7, n. 4, August 1991. 489-499.

WHITNEY, D. E. Resolved motion rate control of manipulators and human prostheses. **IEEE Transactions on Man-Machine Systems**, 10, n. 2, June 1969. 47-53.

YOSHIKAWA, T. Dynamic manipulability of robot manipulators. **Journal of Robotic Systems**, 2, 1985. 113-124.

8. Apêndice A

8.1. Código MATLAB para a construção do objeto “robot”

```
clear L

d=3.5;

L{1} = link([pi/2  0 0 0], 'standard');
L{2} = link([-pi/2 0 0 d], 'standard');
L{3} = link([pi/2  0 0 0], 'standard');
L{4} = link([-pi/2 0 0 d], 'standard');
L{5} = link([pi/2  0 0 0], 'standard');
L{6} = link([0      0 0 d], 'standard');

L{1}.qlim = [-pi/2  pi/2];
L{2}.qlim = [-pi    pi];
L{3}.qlim = [-pi/2  pi/2];
L{4}.qlim = [-pi    pi];
L{5}.qlim = [-pi/2  pi/2];
L{6}.qlim = [-pi    pi];

L{1}.m = 0.033;
L{2}.m = 0.015;
L{3}.m = 0.033;
L{4}.m = 0.015;
L{5}.m = 0.033;
L{6}.m = 0.015+0.030; % with microcamera

% Center of Mass in respect to the origin of each link
L{1}.r = [ 0.000179  0.000243  0.025109 ];
L{2}.r = [ 0          0.017786  0.000012 ];
L{3}.r = [ 0.000179  0.000243  0.025109 ];
L{4}.r = [ 0          0.017786  0.000012 ];
L{5}.r = [ 0.000179  0.000243  0.025109 ];
L{6}.r = [ 0          0.017786-0.0177  0.000012 ];

% Momento of inertia (from CAD)
L{1}.I = [ 7.543e-006  2.337e-006  8.587e-006  2.147e-6  -
3.194e-7  8.8e-7];
L{2}.I = [ 9.645e-007  1.696e-006  1.693e-006  1.44e-7  -
1.311e-7  -1.522e-7];
L{3}.I = [ 7.543e-006  2.337e-006  8.587e-006  2.147e-6  -
3.194e-7  8.8e-7];
L{4}.I = [ 9.645e-007  1.696e-006  1.693e-006  1.44e-7  -
1.311e-7  -1.522e-7];
L{5}.I = [ 7.543e-006  2.337e-006  8.587e-006  2.147e-6  -
3.194e-7  8.8e-7];
L{6}.I = [ 9.645e-007  1.696e-006  1.693e-006  1.44e-7  -
1.311e-7  -1.522e-7];

% Motor inertia (from datasheet)
L{1}.Jm = 9.5e-010;
L{2}.Jm = 9.5e-010;
L{3}.Jm = 9.5e-010;
L{4}.Jm = 9.5e-010;
L{5}.Jm = 9.5e-010;
L{6}.Jm = 9.5e-010;
```

```

% Motor reduction
L{1}.G = 1024*14/3;
L{2}.G = 1024*33/6;
L{3}.G = 1024*14/3;
L{4}.G = 1024*33/6;
L{5}.G = 1024*14/3;
L{6}.G = 1024*33/6;

% viscous friction (motor referenced) (from datasheet)
L{1}.B = 8e-7;
L{2}.B = 8e-7;
L{3}.B = 8e-7;
L{4}.B = 8e-7;
L{5}.B = 8e-7;
L{6}.B = 8e-7;

bot3 = robot(L);

bot3.name = '3rd module';

bot3.manuf = 'PoliMi';

clear L d;

```

8.2. Código MATLAB para otimização

```

function [qt yout] = ikinebot_full (robot,tr,qi)

liminf = [-pi/2 -3*pi/4 -pi/2 -3*pi/4 -pi/2 -3*pi/4];
limsup = [pi/2 3*pi/4 pi/2 3*pi/4 pi/2 3*pi/4];
f = @(x)full_matrix_op(x,robot,tr);
[qt yout] = fmincon(f,qi,[],[],[],[],liminf,limsup);

end

function y = full_matrix_op(x,robot,tr)

y = abs(norm(tr-fkine(robot,x)));

end

```

8.3. Código MATLAB para a SVD

```

function y = svd_inv(J)

[U,S,V] = svd(J);

y = inv(U*S*V');

end

```

8.4. Código MATLAB para a construção da MESH do espaço de trabalho a 2 módulos

```
d = 3.5;
n = 50;
% calota interna 1
theta = (-n:2:n)/n*pi/2;
phi = (-n:2:n)'/n*pi/2;
cosphi = cos(phi);
dsinphi = d*sin(phi);
dcostheta = d*cos(theta);
dsintheta = d*sin(theta);

x = cosphi*dcostheta + ones(n+1,1)*dsintheta;
y = cosphi*dsintheta - ones(n+1,1)*dcostheta;
z = dsinphi*ones(1,n+1);
c = ones(n+1);

surf(x,y,z,c)

hold on

% calota interna 2
theta = (-n:2:n)/n*pi/2;
phi = (-n:2:n)'/n*pi/2;
cosphi = cos(phi);
dsinphi = d*sin(phi);
dcostheta = d*cos(theta);
dsintheta = d*sin(theta);

x = -cosphi*dcostheta + ones(n+1,1)*dsintheta;
y = -cosphi*dsintheta - ones(n+1,1)*dcostheta;
z = -dsinphi*ones(1,n+1);
c = ones(n+1);

surf(x,y,z,c)

%calota externa
th1 = (-n:2:n)/n*pi/2;
th2 = (-n:2:n)'/n*pi/2;
costh2 = cos(th2);
dsinth2 = d*sin(th2);
dcosth1 = d*cos(th1);
dsinth1 = d*sin(th1);

xa = costh2*dsinth1 + ones(n+1,1)*dsinth1;
ya = - costh2*dcosth1 - ones(n+1,1)*dcosth1;
za = dsinth2*ones(1,n+1);
c = ones(n+1);

surf(xa,ya,za,c)

n = 20;

% borda 1
th2 = (-n:2:n)/n*pi/2;
th3 = (-n:1:0)'/n*pi/2;
```

```

dcosth3 = d*cos(th3);
dsinth3 = d*sin(th3);

x = -dcosth3*ones(1,n+1) -d*ones(n+1);
y = -dsinth3*cos(th2);
z = dsinth3*sin(th2);
c = ones(n+1);

surf(x,y,z,c)

% borda 2
th2 = (-n:2:n)/n*pi/2;
th3 = (0:1:n)'/n*pi/2;
dcosth3 = d*cos(th3);
dsinth3 = d*sin(th3);

x = dcosth3*ones(1,n+1) +d*ones(n+1);
y = dsinth3*cos(th2);
z = dsinth3*sin(th2);
c = ones(n+1);

surf(x,y,z,c)

x0 = 0;
y0 = -10.5;
z0 = 0;

scatter3(x0,y0,z0,50,'filled','y');

x0 = x0*ones(n+1);
y0 = y0*ones(n+1);
z0 = z0*ones(n+1);
r0 = 3.5;

% -pi <= theta <= pi is a row vector.
% -pi/2 <= phi <= pi/2 is a column vector.
theta = (-n:2:n)/n*pi;
phi = (-n:2:n)'/n*pi/2;
cosphi = cos(phi); cosphi(1) = 0; cosphi(n+1) = 0;
sintheta = sin(theta); sintheta(1) = 0; sintheta(n+1) = 0;

x = x0 + r0*cosphi*cos(theta);
y = y0 + r0*cosphi*sintheta;
z = z0 + r0*sin(phi)*ones(1,n+1);
c = 2*ones(n+1);

surf(x,y,z,c)

alpha(0.5)
axis equal
hold off
clear all

```