

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
ESCOLA DE ENGENHARIA - CURSO DE ENGENHARIA MECÂNICA  
TRABALHO DE CONCLUSÃO DE CURSO

MODELAGEM CINEMÁTICA E DINÂMICA DE UM MANIPULADOR TIPO  
SNAKE ARM

por

Gustavo Comerlato Rodrigues

Monografia apresentada ao Departamento de Engenharia Mecânica da Escola de Engenharia da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para obtenção do diploma de Engenheiro Mecânico.

Porto Alegre, Setembro de 2023

## DADOS INTERNACIONAIS DE CATALOGAÇÃO

Rodrigues, Gustavo Comerlato  
Modelagem cinemática e dinâmica de um manipulador  
tipo Snake Arm / Gustavo Comerlato Rodrigues. -- 2023.  
35 f.  
Orientador: Rafael Antônio Comparsi Laranja.

Coorientador: Walter Fetter Lages.

Trabalho de conclusão de curso (Graduação) --  
Universidade Federal do Rio Grande do Sul, Escola de  
Engenharia, Curso de Engenharia Mecânica, Porto  
Alegre, BR-RS, 2023.

1. Robôs Snake Arm. 2. Robôs contínuos. 3. Dinâmica  
de robôs. 4. Robot Operating System. 5. Modelagem de  
Sistemas. I. Laranja, Rafael Antônio Comparsi, orient.  
II. Lages, Walter Fetter, coorient. III. Título.

Gustavo Comerlato Rodrigues

MODELAGEM CINEMÁTICA E DINÂMICA DE UM MANIPULADOR TIPO  
SNAKE ARM

ESTA MONOGRAFIA FOI JULGADA ADEQUADA COMO PARTE DOS  
REQUISITOS PARA A OBTENÇÃO DO TÍTULO DE  
**ENGENHEIRO MECÂNICO**  
APROVADA EM SUA FORMA FINAL PELA BANCA EXAMINADORA DO  
CURSO DE ENGENHARIA MECÂNICA

Prof. Dr. Ignácio Iturrioz  
Coordenador do Curso de Engenharia Mecânica

Área de Concentração: Processos de Fabricação

Orientador: Prof. Dr. Rafael Antônio Comparsi Laranja

Coorientador: Prof. Dr. Walter Fetter Lages

Comissão de Avaliação:

Prof. Dr. Rafael Antônio Comparsi Laranja

Prof. Dr. Eduardo André Perondi

Prof. Dr. Mário Roland Sobczyk Sobrinho

Porto Alegre, Setembro de 2023

A todos aqueles que acreditam que um mundo melhor não só é possível, mas necessário.

## AGRADECIMENTOS

Agradeço a minha família por todo o seu apoio ao longo de minha vida. Devo tudo a meus pais, Pedro Paulo e Andréa. Obrigado por tudo e por nunca desistirem.

Agradeço a todos os amigos que marcaram a minha vida na graduação na UFRGS. Agradeço a Antônio, Ângelo, Bernardo, Bianca, Boniatti, Borsoi, Carol, Eduardo, Giliane, Heberton, Hilgert, Josh, Kevin, Kristian, Larissa, Marcus, Menegazzo, Paulo, Valter e outros.

Agradeço aos (ex-)colegas da STIHL pela camaradagem durante o período de estágio e auxílio com minhas decisões profissionais; em particular a meus (ex-)chefes Gunar e Guilherme.

Agradeço a todos os professores que incentivaram a persistência e despertaram a minha curiosidade intelectual ao longo de minha trajetória acadêmica. Agradeço em particular ao professor Casas, por toda a sua ajuda, paciência, disposição e amizade ao longo destes anos.

Agradeço aos professores Laranja e Walter por toda a ajuda e conselhos concedidos durante o trabalho, bem como a infinita paciência para sanar as minhas dúvidas.

Agradeço ao LAMECC e ao técnico Henrique Werkhausen pela disposição do robô *Snake Arm* como objeto de estudo neste trabalho e por toda a ajuda com o mesmo. Agradeço também ao professor Renato “Renatinho” Ventura e seu aluno Rafael O. Gomes pelo empréstimo dos IMUs utilizados neste trabalho.

*“I think that one of these days,” he said, “you’re going to have to find out where you want to go. And then you’ve got to start going there. But immediately. You can’t afford to lose a minute. Not you.”*

*J.D. Salinger*

Comerlato Rodrigues, Gustavo. **Modelagem Cinemática e Dinâmica de um manipulador tipo Snake Arm**. 2023. 35 páginas. Monografia de Trabalho de Conclusão do Curso em Engenharia Mecânica – Curso de Engenharia Mecânica, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2023.

## RESUMO

Neste trabalho foram desenvolvidos modelos cinemáticos e dinâmicos para um manipulador Snake Arm com juntas esféricas atuado por quatro cabos. O modelo cinemático foi desenvolvido considerando a similaridade com robôs contínuos trabalhando sob a hipótese de curvatura constante por seção, junto com a cinemática diferencial inversa de primeira ordem. O modelo dinâmico foi construído via mecânica Lagrangeana para o caso planar com atuação de dois cabos, incluindo atrito, rigidez nas juntas e a propagação de tensões nos cabos. O modelo cinemático foi implementado em um pacote para o ROS 2, enquanto o modelo dinâmico foi implementado no MATLAB na forma de espaço de estados. Como resultados, o modelo cinemático foi utilizado para geração de trajetórias no espaço dos cabos, enquanto o modelo dinâmico foi comparado com a dinâmica de um protótipo de uma seção sob ação da gravidade, com identificação de parâmetros via um algoritmo de otimização heurística. Finalmente, os dois modelos foram comparados experimentalmente com relação à ângulos de juntas e posição previstas para flexão no plano. Os resultados mostram que o modelo dinâmico é capaz de reproduzir mais fielmente a flexão no plano, apesar de captar apenas parcialmente o comportamento dinâmico sob gravidade.

**PALAVRAS-CHAVE:** Robôs Snake Arm, robôs contínuos, dinâmica de robôs, Robot Operating System, modelagem de sistemas.

Comerlato Rodrigues, Gustavo. **Kinematic and dynamic modelling of a Snake Arm manipulator**. 2023. 35 pages. Mechanical Engineering End of Course Monography – Mechanical Engineering degree, The Federal University of Rio Grande do Sul, Porto Alegre, 2023.

## ABSTRACT

This work presents the development of kinematic and dynamic models for a Snake Arm manipulator with spherical joints driven by four cables. The kinematic model was developed considering the similarity between continuum robots with the constant curvature section hypothesis, including the first order differential inverse kinematics. The dynamical model was developed via Lagrangian mechanics for the in-plane bending case with two cable actuators, including friction effects, joint stiffnesses and tension propagation along the cables. The kinematic model was implemented in a ROS 2 package, while the dynamic model was implemented in MATLAB in state space form. As results, the kinematic model was used to generate trajectories in the cable space, while the dynamic model was compared with experimental results from a one section prototype under gravity, with parameter identification through a heuristic optimization algorithm. Finally, the two models were experimentally compared with relation to predicted joint angles and positions for in-plane bending. The results show that the dynamic model is able to reproduce more adequately the in-plane bending, although it can only partially predict the dynamic behavior under gravity.

**KEYWORDS:** Snake Arm robots, continuum robots, robot dynamics, Robot Operating System, systems modelling

## NOMENCLATURA

### Símbolos

$a$	Distância dos pontos de fixação dos cabos ao centro do elo	[m]
$b$	Comprimento de uma barra do módulo	[m]
$c$	Ponto de vixação do cabo	[m]
$c_\theta, s_\theta$	Cosseno e seno de um dado ângulo $\theta$	
$D, P$	Índices dos <i>frames</i> Distal e Proximal de uma seção ou módulo	
$\mathbf{B}$	Matriz de ponderação das tensões	
$\mathbf{C}$	Matriz com termos de Coriolis	[Nms/rad]
$F_x, F_y$	Forças em $x$ e $y$ nos elos	[N]
$\mathbf{F}$	Vetor de torques devido ao atrito	[Nm]
$\mathcal{F}$	<i>Wrench</i>	[N;Nm]
$FK_{cart}$	Mapa cinemático direto de seções para espaço cartesiano global	
$FK_{gen}$	Mapa cinemático direto de uma seção	
$FK_{spe}$	Mapa cinemático direto de um módulo	
$\mathbf{G}$	Vetor de torques equivalentes nas juntas devido à gravidade	[Nm]
$I_z$	Inércia torsional de um elo no centro de massa	[kgm <sup>2</sup> ]
$\mathbf{I}$	Matriz identidade	
$IK_{cart}$	Mapa cinemático inverso do espaço cartesiano global para seções	
$IK_{gen}$	Mapa cinemático inverso de uma seção	
$IK_{spe}$	Mapa cinemático inverso de um módulo	
$\mathbf{J}$	Matriz jacobiana analítica entre dois espaços	
$\mathbf{J}_s$	Matriz de jacobianos seriais para os elos	
$k_v$	Inclinação da tangente hiperbólica para suavização na origem	[s/rad]

$K$	Rigidez torcional equivalente de uma junta	[Nm/rad]
$\mathbf{K}$	Matriz de rigidez torcional equivalente das juntas	[Nm/rad]
$\ell$	Comprimento de cabo	[m]
$m_i$	Massa de um elo	[kg]
$\mathbf{M}$	Matriz de inércia equivalente no espaço das juntas	[Nms <sup>2</sup> /rad]
$n_c$	Número de cabos	
$n_j$	Número de juntas de uma seção	
$n_t$	Número de instantes de tempo amostrados de uma repetição	
$\mathbf{q}_i$	Espaço de atuadores de uma seção	
$\mathbf{q}_g$	Espaço global de atuadores	
$s$	Comprimento de arco sob curvatura constante	[m]
$S$	Magnitude do efeito de Stribeck	
$T$	Transformação homogênea entre <i>frames</i> ; Tensão nos cabos	
$u$	Componente X do eixo de rotação da seção	[m]
$\mathbf{u}$	Tensões de entrada nos cabos no modelo dinâmico	[N]
$v$	Componente Y do eixo de rotação da seção	[m]
$v_d$	Velocidade de Stribeck	[rad/s]
$v$	Componente linear de um <i>twist</i>	[m/s]
$\mathcal{V}^s$	<i>Twist</i>	[m/s ; rad/s]
$w$	Eixo de rotação da seção	[m]
$\mathbf{W}$	Jacobiano de forças de um módulo	
$\mathbf{x}_i$	Espaço cartesiano em coordenadas locais da seção	
$\mathbf{X}_g$	Espaço cartesiano em coordenadas da base	

## Símbolos gregos

$\alpha_i$	Azimute do $i$ -ésimo cabo com relação ao eixo $X$	[rad]
$\zeta_i$	Espaço de configuração da $i$ -ésima seção	
$\theta_i$	Ângulo de flexão de uma seção	[rad]
$\theta_m$	Ângulo de flexão de uma junta medido	[rad]
$\mu_c$	Coeficiente de atrito dos cabos com os discos espaçadores	
$\kappa_i$	Curvatura de um módulo ou seção	[m <sup>-1</sup> ]
$\tau_c$	Torque devido a atrito de Coulomb	[Nm]
$\tau_v$	Torque de atrito viscoso por velocidade angular	[Nms/rad]
$\sigma_i$	Ângulo de torção de um módulo ou seção	[rad]
$\varphi_i$	Ângulo azimute de um módulo ou seção	[rad]
$\omega$	Componente rotacional de um <i>twist</i>	[rad/s]

## Abreviaturas e acrônimos

DDS	<i>Data distribution service</i>
FABRIK	<i>Forwards And Backwards Reaching Inverse Kinematics</i>
IMU	<i>Inertial Motion Unit</i>
LAMECC	Laboratório de mecatrônica e controle
NRMSE	<i>Normalized Root Mean Square Error</i>
OROCOS	<i>Open Robot Control Software Kinematics and Dynamics</i>
KDL	<i>Library</i>
ROS	<i>Robot Operating System</i>
SMA/SMP	<i>Shame Memory Alloy/Shape Memory Polymer</i>
SPS	Cadeia cinemática com juntas esférica-prismática-esférica
URDF	<i>Universal Robot Description Format</i>

## LISTA DE FIGURAS

Figura 1 - Dois exemplos de robôs contínuos tipo <i>Snake Arm</i> .....	1
Figura 2 - Espaços para descrição cinemática de um manipulador <i>Snake Arm</i> . ....	3
Figura 3 – Nomenclatura e sistemas de coordenadas utilizados para o robô neste trabalho. ....	4
Figura 4 – Representação cinemática do módulo de uma seção. ....	6
Figura 5 – Componentes do modelo estático de atrito e sua suavização.....	8
Figura 6 - Tensões devido a cabos e <i>wrench</i> $\mathcal{F}$ equivalente atuando em um elo. ....	9
Figura 7 - <i>Setup</i> do robô com IMUs montados em placas de madeira.....	11
Figura 8 – À esquerda: Comprimentos totais dos cabos. À direita: Comparação entre velocidades dos cabos.....	13
Figura 9 - Posição angular prevista VS medida das juntas para a repetição 1. ....	14
Figura 10 – Comparação entre ângulos e posições de juntas previstos.....	14
Figura 11 – Conteúdos do meta-pacote <code>continuum</code> desenvolvido. ....	18
Figura 12 – Cadeia cinemática de uma seção gerada pelo pacote <code>continuum_description</code> e sua representação no RViz como <i>frames</i> .....	19
Figura 13 – Grafo de computação do simulador cinemático.....	20
Figura 14 – Visualização de um manipulador de 3 seções no RViz. ....	20
Figura 15 – Relações de cinemática diferencial via jacobianos. ....	21

## SUMÁRIO

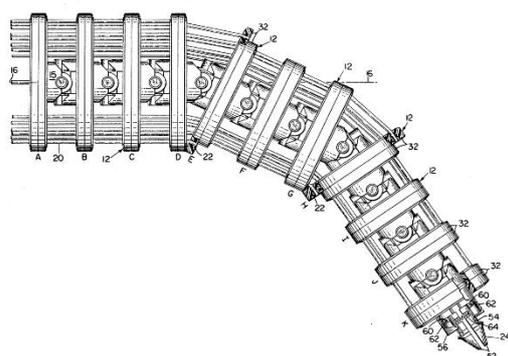
NOMENCLATURA.....	ix
1. INTRODUÇÃO.....	1
2. FUNDAMENTAÇÃO TEÓRICA E MODELAGEM.....	2
2.1. ESTRUTURA DE ROBÔS SNAKE-ARM.....	3
2.2. MODELO CINEMÁTICO.....	3
2.2.1. CINEMÁTICA GENÉRICA DE UMA SEÇÃO.....	3
2.2.2. CINEMÁTICA ESPECÍFICA DE UM MÓDULO.....	5
2.2.3. ACOPLAMENTO ENTRE SEÇÕES.....	6
2.2.4. CINEMÁTICA DIFERENCIAL.....	7
2.3. MODELO DINÂMICO.....	7
3. DESENVOLVIMENTO.....	10
3.1. SIMULADOR CINEMÁTICO.....	10
3.2. SIMULADOR DINÂMICO.....	10
3.3. CASO DE ESTUDO: DINÂMICA DE UM MANIPULADOR PLANAR.....	11
4. RESULTADOS.....	12
4.1. CINEMÁTICA DIFERENCIAL INVERSA.....	12
4.2. RESPOSTA DINÂMICA À GRAVIDADE.....	13
4.3. CURVATURA PREVISTA PARA FLEXÃO.....	14
5. CONCLUSÕES.....	15
REFERÊNCIAS BIBLIOGRÁFICAS.....	16
APÊNDICE A – Implementação do simulador cinemático em ROS 2.....	18
APÊNDICE B – Relações de cinemática diferencial.....	21

## 1. INTRODUÇÃO

A área de robótica, tanto em termos de problemas de estudo quanto em aplicações, tem se generalizado consideravelmente desde suas origens industriais na segunda metade do Século XX. Muitas disciplinas tiveram uma relação simbiótica com o estudo, desde dinâmica e controle até tópicos como visão computacional e topologia, dentre outras. Isso ocorreu tanto pelo seu uso na formalização e solução de problemas enfrentados na robótica quanto pelo enriquecimento dessas áreas ao ter esta como objeto de estudo. Devido a esta simbiose, atualmente não é incomum ouvirmos falar de robôs sendo utilizados em operações fora das fábricas, sejam essas no ramo de serviços, como na construção, operações de busca e resgate ou auxílio biomédico, quanto em ambientes não estruturados, como em meios urbanos, aquáticos ou fortemente confinados (SICILIANO; KHATIB, 2016).

Em particular, os manipuladores hiper-redundantes pertencem a uma classe de robôs caracterizada por apresentarem um número de graus de liberdade muito maior que a dimensão do seu espaço de trabalho, sendo esta 6 para o grupo euclidiano de movimentos de corpos rígidos. Este número alto de graus de liberdade por sua vez se traduz em uma redundância cinemática no espaço de trabalho do manipulador que pode ser utilizada para tarefas como evitar obstáculos ou minimização de alguma função de controle (MURRAY; LI; SASTRY, 1994).

Dentre os manipuladores hiper-redundantes existe uma classe de robôs denominados *Snake Arm* ("Braço de cobra", em português). Dois exemplos de robôs deste tipo podem ser vistos na Figura 1 (a) e (b), a seguir. A sua estrutura é tipicamente composta de uma "coluna vertebral" (do inglês *backbone*) conectando uma série de elos rígidos por juntas revolutas (cada junta com tipicamente 2 ou 3 eixos de rotação, como por exemplo em uma junta esférica) em sequência. Tipicamente, cada elo está conectado a um disco espaçador (as "vértebras" da coluna) que podem servir para roteamento de cabos ou troca de ferramentas cirúrgicas (RUSSO *et al.*, 2023). A estrutura dos manipuladores *Snake Arm* é também em muitos casos dividida entre diversas "seções", que são conjuntos de vários "módulos" atuados simultaneamente. Em manipuladores atuados por cabos, por exemplo, os cabos são roteados da base do robô, passando pelas vértebras, e afixados na vértebra localizada na extremidade da seção.



(a) *Tensor Arm*. Fonte: (ANDERSON; HORN, 1968)



(b) *OCR LaserSnake*. Fonte: <https://www.bbc.com/news/uk-england-bristol-21940753>.

Figura 1 - Dois exemplos de robôs contínuos tipo *Snake Arm*.

A atuação destes robôs pode ocorrer por várias formas diferentes. Dentre elas pode-se citar motores elétricos coordenando diretamente as juntas; transmissão via cabos; músculos pneumáticos; SMPs ou SMAs (*Shape Memory Polymers e Alloys*, respectivamente). Como muitos destes manipuladores são projetados para uso em espaços confinados, é desejável que

os seus atuadores fiquem na base para diminuir o volume ocupado pelo braço em si. Uma solução popular para isto é o emprego de um sistema baseado em tendões fixados e controlados na base, que transmitem as forças para cada uma das seções destes robôs (RAO *et al.*, 2021).

A estratégia de atuação por tendões na base, apesar de seu baixo custo, leva a algumas dificuldades adicionais em termos do controle do robô. Como exemplo, pode-se citar o atrito dos cabos com os discos espaçadores do robô, a inabilidade dos cabos resistirem a forças de compressão, e a inércia dos cabos. Outra característica presente em muitas arquiteturas de robôs do tipo *Snake Arm* é a sub-atuação; ou seja, eles apresentam mais graus de liberdade que atuadores (TEDRAKE, 2023), devido ao número e disposição física dos cabos ao longo das seções. Tendo em vista estes aspectos, a síntese de controladores para tais robôs enfrenta uma série de complicações adicionais às tipicamente enfrentadas em robôs "clássicos".

Devido à complexidade de mecanismos deste tipo, o uso de ferramentas computacionais e modelos matemáticos é de grande auxílio durante as fases de projeto tanto da estrutura destes robôs quanto do seu controle. O uso de simuladores e ferramentas computacionais na área de robótica é de longa data (LYNCH; PARK, 2017; SICILIANO; KHATIB, 2016). Seu uso é multifacetado, sendo utilizados para fins tais quais programação *offline*, desenvolvimento de algoritmos e controladores, e estudos de protótipos dos robôs em si, dentre outros. Dentre estas, vale mencionar a plataforma ROS 2, evolução do conhecido **ROS** (*Robot Operating System*, "Sistema Operacional de Robôs", em tradução livre) (QUIGLEY *et al.*, 2009), uma plataforma livre e de código aberto que permite a interconexão de diversos módulos, algoritmos, robôs e controladores em tempo real via **DDS** (*Data Distribution Service*). Já modelos matemáticos para robôs são essenciais não só para entender o comportamento físico destes sistemas, mas também para o projeto de controladores de mais alta performance (como é o caso de movimentos com altas acelerações, em que se requer conhecer a dinâmica do sistema (SICILIANO; KHATIB, 2016)) e ainda para evidenciar possibilidades de melhoria em seu projeto. Desta forma fica evidenciado o grande interesse no desenvolvimento desses modelos, tanto por motivos particulares a estes robôs quanto em um contexto mais geral.

Dessarte, este trabalho objetiva a aplicação e implementação de modelos para a cinemática de posição e diferencial de robôs *Snake Arm* multiseção em três dimensões, e o desenvolvimento e implementação de um modelo para a dinâmica de um manipulador *Snake Arm* planar. A verificação destes modelos foi feita tanto computacionalmente quanto experimentalmente com um robô protótipo de uma seção. O restante do trabalho é estruturado da seguinte maneira: na segunda seção é apresentada a fundamentação teórica e o trabalho de modelagem presente na literatura para a cinemática de robôs multiseção, seguido do desenvolvimento de um modelo para a dinâmica de um manipulador planar de uma seção; a terceira seção apresenta os detalhes de implementação computacional destes modelos nos ambientes ROS 2 e MATLAB, bem como o robô protótipo utilizado como caso de estudo deste trabalho; na quarta seção são apresentados e comentados os resultados dos modelos, utilizando o robô protótipo como validação experimental; e finalmente a quinta seção contém as conclusões deste trabalho e para trabalhos futuros.

## 2. FUNDAMENTAÇÃO TEÓRICA E MODELAGEM

Nesta seção serão apresentadas as principais considerações sobre a estrutura dos robôs *Snake Arm*, além dos modelos cinemáticos de múltiplas seções e um modelo dinâmico de uma seção, com aplicação para flexão no plano.

## 2.1. ESTRUTURA DE ROBÔS SNAKE-ARM

Uma maneira comum de modelar robôs *Snake Arm* é via a descrição de sua cadeia como uma cadeia aberta híbrida; ou seja, o robô é modelado como uma série de  $n_s$  seções atuadas por cabos conectadas em série, sendo que cada seção é composta de vários módulos individuais, que são individualmente cadeias cinemáticas paralelas. Este tipo de abordagem é conveniente pois permite desacoplar parcialmente as cinemáticas da seção das cinemáticas dos módulos. Seguindo (WEBSTER; JONES, 2010), há a cinemática *genérica* direta e inversa das seções,  $FK_{gen}$  e  $IK_{gen}$ , que independe da atuação utilizada, e as cinemáticas *específicas* direta e inversa dos módulos,  $FK_{spe}$  e  $IK_{spe}$ , assim nomeadas por dependerem do tipo de atuação empregado. Além disso, os efeitos de acoplamento entre seções atuadas por cabos (denominados *tangle* e *untangle*) também devem ser considerados para a interface do robô com o mundo e com os atuadores dos cabos. O resultado é que um robô *Snake Arm* multiseção pode ser entendido cinematicamente com seções individuais em paralelo, acopladas na descrição cartesiana global e no comprimento total dos cabos. Mais detalhes podem ser vistos na Figura 2.

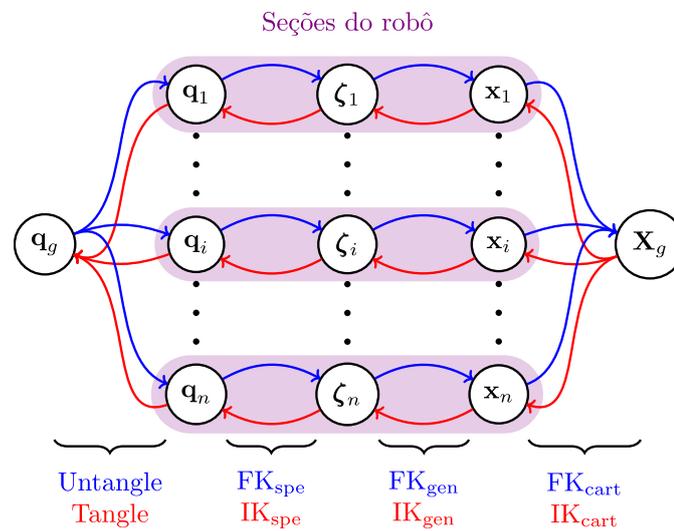


Figura 2 - Espaços para descrição cinemática de um manipulador *Snake Arm* considerados neste trabalho

A Figura 2 ilustra esquematicamente os mapas entre os diferentes espaços de variáveis considerados para a modelagem cinemática neste trabalho, com as variáveis referentes às seções do robô circuladas em cor violeta. Neste diagrama, os espaços dos cabos são denotados pela letra  $\mathbf{q}$ , os espaços de configuração pela letra  $\boldsymbol{\zeta}$ , e o espaço cartesiano referente à cada seção pela letra  $\mathbf{x}$ . Em particular,  $\mathbf{q}_g$  denota os comprimentos totais dos cabos e  $\mathbf{X}_g$  o espaço cartesiano em coordenadas globais centradas na base do robô.

## 2.2. MODELO CINEMÁTICO

Nesta subseção é apresentada a descrição cinemática comumente utilizada para modelar robôs contínuos, e adaptada para a descrição de um robô *Snake Arm*.

### 2.2.1. CINEMÁTICA GENÉRICA DE UMA SEÇÃO

A  $i$ -ésima seção de um robô *Snake Arm* possui um número total de  $n_c$  cabos, podendo estes ser divididos entre cabos  $n_{c,p_i}$  *proximais atuantes* (subscrito  $p$ ), que são conectados a esta seção e atuam nela diretamente, e  $n_{c,d_i}$  cabos *distais passantes* por ela, que afetam de uma forma

secundária a seção. Para a construção do modelo cinemático, uma abordagem comum na literatura (HANNAN; WALKER, 2003; NEPPALLI *et al.*, 2009; WEBSTER; JONES, 2010) é a consideração de que as seções se dobram com curvatura constante, como arcos de círculo. Esta consideração é uma simplificação que ignora os efeitos como gravidade e atrito dos cabos e juntas no robô, mas que serve como aproximação eficiente para construir os mapas  $FK_{gen}$  e  $IK_{gen}$ . Ela foi adotada na modelagem cinemática neste trabalho devido à sua simplicidade.

Pela hipótese da curvatura constante existem duas principais parametrizações de uma seção em voga na literatura consultada (FRAZELLE, 2021; WEBSTER; JONES, 2010). A mais comum é a abordagem originalmente apresentada em (JONES; WALKER, 2006), que considera 3 parâmetros,  $\kappa$  (curvatura),  $\varphi$  (azimute) e  $s$  (comprimento de arco) para um dado ângulo de flexão total  $\theta$ , pertencentes ao espaço de configuração  $\zeta$  da seção. Devido à presença de singularidade na configuração reta ( $\kappa = 0$ ), recentemente a abordagem de (ALLEN *et al.*, 2020) foi proposta como uma alternativa. Nela, são utilizados os parâmetros  $u$  e  $v$ , componentes  $x$  e  $y$  do vetor de rotação  $w$  no sistema da base, ao invés de  $\kappa$  e  $\varphi$ . A Figura 3 apresenta estas parametrizações bem como a nomenclatura utilizada neste trabalho, enquanto as equações (1), (2) e (3) são utilizadas para realizar a transformação entre elas, onde  $\text{atan2}(y, x)$  é a função arcotangente de 2 argumentos.

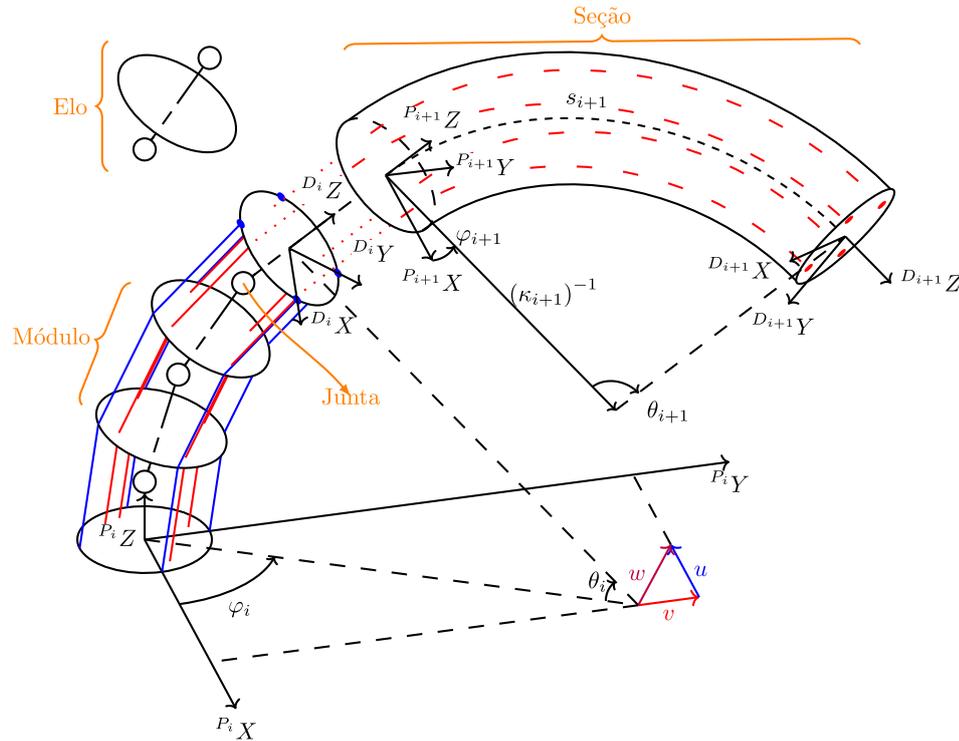


Figura 3 – Nomenclatura e sistemas de coordenadas utilizados. As coordenadas  $(\phi, \kappa, s)$  definem a parametrização usual de Jones, enquanto  $(u, v, s)$  formam a parametrização de Allen. Cabos azuis são cabos proximais atuantes, enquanto cabos vermelhos são cabos distais passantes.

$$\theta = s\kappa = \sqrt{u^2 + v^2} \quad (1)$$

$$\varphi = \text{atan2}(-u, v) \quad (2)$$

$$(u, v) = (-\theta \sin \varphi, \theta \cos \varphi) \quad (3)$$

De acordo com Allen *et al.*, (2020), a matriz de transformação homogênea entre o sistema distal da  $i$ -ésima seção para a sua base  ${}^{P_i}T_{D_i}$  é escrita pela Eq. (4), onde  $\gamma = (\cos(\theta) - 1)/\theta^2$

e  $\xi = \sin(\theta) / \theta$ , fornecendo assim o mapa  $FK_{gen}$ . Já o mapa cinemático inverso de uma seção  $IK_{gen}$  para  $\varphi_i, \kappa_i$  e  $\theta_i$  dado um ponto  ${}^{P_i}(x, y, z)$  é expresso pela Equação (5).

$$\mathbf{x}_i = FK_{gen}(\boldsymbol{\zeta}_i) = \begin{cases} {}^{P_i}T_{D_i} = \begin{bmatrix} \xi_i v_i^2 + 1 & -\gamma_i u_i v_i & \xi_i v_i & -s_i v_i \gamma_i \\ -\gamma_i u_i v_i & \gamma_i u_i^2 + 1 & -\xi_i u_i & s_i u_i \gamma_i \\ -\xi_i v_i & \xi_i u_i & \cos \theta_i & \xi_i s \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{cases} \quad (4)$$

$$\boldsymbol{\zeta}_i = IK_{gen}(\mathbf{x}_i) = \begin{cases} \varphi_i = \text{atan2}({}^{P_i}y, {}^{P_i}x) \\ \kappa_i = \frac{2\sqrt{{}^{P_i}x^2 + {}^{P_i}y^2}}{{}^{P_i}x^2 + {}^{P_i}y^2 + {}^{P_i}z^2} \\ \theta_i = \begin{cases} \text{acos}\left(1 - \kappa_i \sqrt{{}^{P_i}x^2 + {}^{P_i}y^2}\right), & {}^{P_i}z > 0 \\ 2\pi - \text{acos}\left(1 - \kappa_i \sqrt{{}^{P_i}x^2 + {}^{P_i}y^2}\right), & {}^{P_i}z \leq 0 \end{cases} \end{cases} \quad (5)$$

O uso de 3 parâmetros de configuração de forma independente leva a casos de extensão/contração da seção. Como isso não ocorre para robôs de elos discretos, é possível definir apenas dois graus de liberdade para a cinemática de uma seção, a partir do qual o terceiro pode ser calculado. Neste trabalho, foram escolhidos  $\varphi$  e  $\theta$  como parâmetros de entrada; a partir de  $\theta$ , pode-se calcular  $\kappa$  e  $s$  pelas Equações (6) e (7), respectivamente.

$$s_i(\theta_i) = b_i \theta_i / \tan \frac{\theta_i}{2n_{j,i}} \quad (6)$$

$$\kappa_i(\theta_i) = \frac{\theta_i}{s_i} = \tan \frac{\theta_i}{2n_{j,i}} / b_i \quad (7)$$

### 2.2.2. CINEMÁTICA ESPECÍFICA DE UM MÓDULO

A cinemática específica dos módulos pode ser obtida pela consideração destes como mecanismos paralelos similares à plataforma de Stewart, seguindo (GHOSAL, 2006; MUSTAFA, 2008). A Figura 4 apresenta uma vista representativa da estrutura de um destes módulos. Nesse modelo eles são formados por duas plataformas conectadas via uma coluna central com junta esférica passiva, atuada indiretamente pelos cabos. Quando tensionados, os cabos podem ser representados como pernas tipo **SPS** (i.e., cadeias cinemáticas com juntas Esférica-Prismática-Esférica) com atuação apenas da junta prismática, como realizado em (OKOLI *et al.*, 2019).

A cinemática inversa  $IK_{spe}$  de um destes módulos (isto é, comprimento do  $j$ -ésimo cabo da  $i$ -ésima seção  ${}^i\ell_j$  em função dos ângulos da junta esférica) é obtida pela aplicação de uma equação de fechamento de laço aplicada para cada um dos cabos no módulo, e é dada na Eq. (8), onde  $\mathbf{c}_j$  é o ponto de fixação do  $j$ -ésimo cabo. Em contrapartida, o procedimento para obter  $FK_{spe}$  é mais envolvido; este mapa é apresentado na Eq. (9), como em (WEBSTER; JONES, 2010).

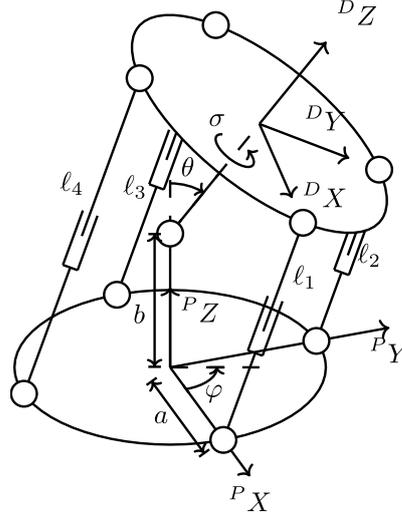


Figura 4 – Representação cinemática do módulo de uma seção com cabos tensionados como juntas prismáticas.

$$\mathbf{q}_i = IK_{spe}(\boldsymbol{\zeta}_i) = {}^i\ell_j = \| {}^iP_iT_{D_i}(\boldsymbol{\zeta}_i)\mathbf{c}_j - \mathbf{c}_j \|, \quad j = 1 \dots n_c \quad (8)$$

$$\boldsymbol{\zeta}_i = FK_{spe}(\mathbf{q}_i) = \begin{cases} \varphi_i = \text{atan2}({}^i\ell_4 - {}^i\ell_2, {}^i\ell_3 - {}^i\ell_1) \\ \kappa_i = \frac{({}^i\ell_1 - 3 {}^i\ell_2 + {}^i\ell_3 + {}^i\ell_4) \sqrt{({}^i\ell_4 - {}^i\ell_2)^2 + ({}^i\ell_3 - {}^i\ell_1)^2}}{a_i ({}^i\ell_1 + {}^i\ell_2 + {}^i\ell_3 + {}^i\ell_4) ({}^i\ell_4 - {}^i\ell_2)} \\ s_i = \frac{2n_{j,i}}{\kappa_i} \text{atan2}(\kappa_i b_i) \\ \theta_i = \kappa_i s_i \end{cases} \quad (9)$$

Nota-se que com esta configuração de cabos não haverá atuação do ângulo de rotação  $\sigma$  da junta esférica central no caso estático; portanto,  $\sigma = 0$  para o modelo cinemático. Ademais, como os cabos só podem aplicar trações nos módulos, pequenas rotações em  ${}^DZ$  são contra-atuadas pelos cabos, que tenderão a ficar retos quando tracionados. Apesar disto, a rigor  $\sigma$  deve estar presente em modelos dinâmicos, já que o peso próprio do robô levará a um torque em  $\sigma$  equivalente nas juntas e rotações fora do plano de dobra.

### 2.2.3. ACOPLAMENTO ENTRE SEÇÕES

A solução completa dos mapas cinemáticos entre  $\mathbf{q}_g$  e  $\mathbf{X}_g$  envolve encontrar os mapas  $FK_{cart}$  e  $IK_{cart}$ , além dos mapas entre  $\mathbf{q}_g$  e  $\mathbf{q}_i$ . Considerando estrutura da cadeia cinemática aberta deste robô, a solução de  $FK_{cart}$  é direta; caso os frames  $D_i = P_{i+1}$ , pode-se realizar a multiplicação direta das matrizes de transformação homogênea das seções, como realizado na Eq. (10), para encontrar a posição e orientação do sistema de coordenadas distal com relação à base do manipulador.

$$FK_{cart}(\boldsymbol{\zeta}_1, \dots, \boldsymbol{\zeta}_n) = {}^{P_1}T_{D_1}(\boldsymbol{\zeta}_1) \dots {}^{P_i}T_{D_i}(\boldsymbol{\zeta}_i) \dots {}^{P_n}T_{D_n}(\boldsymbol{\zeta}_n) \quad (10)$$

A solução de cinemática inversa para um manipulador *Snake Arm* de múltiplas seções, em contrapartida, é de difícil obtenção, sendo que não foi encontrada expressão analítica na literatura consultada para seções inextensíveis. Dessa forma, neste trabalho foi implementado um *solver* heurístico descrito por (ZHANG *et al.*, 2018), baseado no FABRIK, originalmente

proposto em (ARISTIDOU; LASENBY, 2011). A ideia dos *solvers* do tipo FABRIK é tentar iterativamente primeiro alcançar o *frame* da base partindo-se do *frame* desejado no espaço cartesiano, e então tentar alcançar o *frame* desejado partindo-se da base, atualizando os valores das juntas em cada etapa. Mais detalhes podem ser vistos nas publicações citadas.

Com relação aos mapas entre  $\mathbf{q}_g$  e  $\mathbf{q}_i$ , esses descrevem essencialmente como será o acoplamento cinemático entre cabos das seções. Devido ao roteamento e *offset* angular dos cabos proximais entre seções, o controle cinemático de um manipulador via extensão/relaxação de cabos implica em uma atuação simultânea de todas as seções do manipulador. Isto já é visível em um robô planar com 2 seções, pois ao se puxar um cabo proximal da primeira seção (mais próxima à base) os comprimentos de todos os cabos que passarem por esta seção serão afetados. Na literatura consultada, esses efeitos são referidos como *entrelaçamento/desentrelaçamento* dos cabos (do inglês *Tangle/Untangle*) e podem ser resolvidos via algoritmos específicos para tal, conforme descrito em (JONES; WALKER, 2006). Esses algoritmos foram implementados no simulador cinemático desenvolvido.

#### 2.2.4. CINEMÁTICA DIFERENCIAL

Uma vez obtidos os mapas cinemáticos instantâneos entre os diferentes espaços do atuador, é possível obter as relações locais entre as velocidades desses espaços pelas matrizes jacobianas dos mapas cinemáticos. Para tal, basta derivar cada um desses mapas com relação às suas variáveis de entrada para se obter a matriz jacobiana  $\mathbf{J}_{x_i q_i}$  entre esses espaços e aplicar a regra da cadeia, resultando na Equação (11). O desenvolvimento e obtenção dos jacobianos para os mapas cinemáticos se encontram descritos com mais detalhes no Apêndice B.

$${}^{P_i}\mathbf{x}_i = FK_{gen}(FK_{spe}(\mathbf{q}_i)) \Rightarrow {}^{P_i}\dot{\mathbf{x}}_i = \mathbf{J}_{x_i q_i} \dot{\mathbf{q}}_i = \mathbf{J}_{x_i \zeta_i} \mathbf{J}_{\zeta_i q_i} \dot{\mathbf{q}}_i \quad (11)$$

#### 2.3. MODELO DINÂMICO

O modelo dinâmico de um manipulador de cadeia aberta pode ser obtido de várias formas, mas ele pode ser entendido mais claramente quando expresso em forma matricial, seguindo a abordagem de mecânica Lagrangeana (GHOSAL, 2006; MURRAY; LI; SASTRY, 1994). Considerando o uso em simulação, optou-se pelo uso de um modelo formulado em espaço de juntas, em contrapartida ao modelo cinemático anteriormente apresentado, sem fazer hipótese sobre a curvatura de uma seção. Tendo em vista o robô caso de estudo deste trabalho, propõe-se representar a sua dinâmica em espaço de juntas pela Equação (12).

$$\mathbf{M}(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + \mathbf{C}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + \mathbf{K}(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \mathbf{G}(\boldsymbol{\theta}) + \mathbf{F}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{J}_S^T(\boldsymbol{\theta})\mathbf{W}(\boldsymbol{\theta})\mathbf{B}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\mathbf{u} \quad (12)$$

O lado esquerdo da expressão contém a matriz de inércia generalizada do manipulador  $\mathbf{M}$ , a matriz com efeitos de forças centrípetas e Coriolis  $\mathbf{C}$ , e um vetor de esforços nas juntas devido à gravidade  $\mathbf{G}$ , termos usuais na dinâmica de manipuladores. Junto a estes foram adicionados  $\mathbf{K}$ , uma matriz com as rigidezes das juntas entre elos, e  $\mathbf{F}$ , um vetor de esforços (devido ao atrito) refletidos nas juntas. Os termos  $\mathbf{M}$ ,  $\mathbf{C}$  e  $\mathbf{G}$  foram obtidos conforme descritos em (MURRAY; LI; SASTRY, 1994); já  $\mathbf{K}$  é uma matriz diagonal (refletindo apenas as elasticidades das molas do robô), com termos aqui considerados constantes, e  $\mathbf{F}$  dependerá do modelo de atrito adotado. No trabalho adotou-se um modelo estático com componentes de Coulomb, viscoso e efeito Stribeck suavizado na origem com uma função tangente hiperbólica, conforme descrito em (EGLE, 2021), que expressa o torque em uma junta rotativa pela Eq.

(13), ilustrada na Figura 5. Nesta equação  $\tau_c$  é o torque de atrito de Coulomb,  $\tau_\mu$  o coeficiente de atrito viscoso por velocidade de rotação,  $S$  o tamanho relativo do efeito Stribeck,  $v_d$  a velocidade de Stribeck, e  $k_v$  a inclinação da suavização via função tangente hiperbólica.

$$\mathbf{F}(\theta, \dot{\theta})_i = \tau_\mu \dot{\theta}_i + \tau_c \tanh(k_v \dot{\theta}_i) \left( 1 + S \exp\left(-\frac{\dot{\theta}_i}{v_d}\right)^2 \right) \quad (13)$$

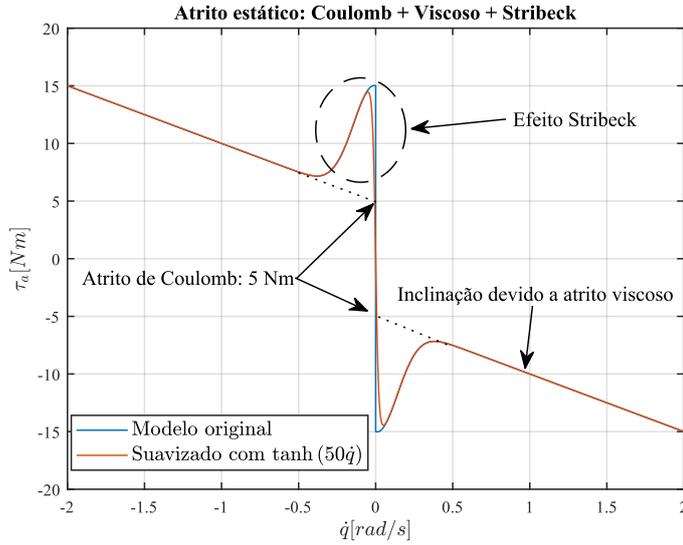


Figura 5 – Componentes do modelo estático de atrito e sua suavização. Valores apenas para ilustração.

O lado direito desta expressão contém um vetor  $\mathbf{u}$  com tensões de entrada nos cabos e três matrizes,  $\mathbf{J}_s^T$ ,  $\mathbf{W}$  e  $\mathbf{B}$  os quais representam a atuação do robô e serão descritos a seguir. O vetor  $\mathbf{u}$  contém as tensões de entrada de cada cabo na base no robô, atuadas externamente, com  $n_c$  elementos. Essas tensões são propagadas ao longo dos cabos, dissipando potência em cada disco devido ao atrito. A matriz  $\mathbf{B}$  faz essa projeção de tensões aplicadas para tensões locais dos cabos em cada módulo, seguindo o modelo de propagação de tensões em polias descrito pela Equação (14), onde  $\mu_c$  é o coeficiente de atrito estático entre cabo e disco e  $\theta$  é o ângulo de abraçamento do cabo com o disco, levando  $\mathbf{B}$  a ter a estrutura apresentada na Eq. (15).

$$T_{i+1} = T_i \exp(-\mu_c \theta) \quad (14)$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{I}_{n_c} \exp\left(-\mu_c \frac{\theta_1}{2}\right) \\ \mathbf{I}_{n_c} \exp\left(-\mu_c \left(\theta_1 + \frac{\theta_2}{2}\right)\right) \\ \vdots \\ \mathbf{I}_{n_c} \exp\left(-\mu_c \left(\theta_1 + \dots + \frac{\theta_{n_j}}{2}\right)\right) \end{bmatrix} \quad (15)$$

A matriz  $\mathbf{W} = [\mathbf{W}_1 \ \dots \ \mathbf{W}_i \ \dots \ \mathbf{W}_{n_j}]^T$  transforma as tensões  $\mathbf{T}$  existentes nos cabos atuantes em cada módulo para *wrenches*  $\mathcal{F}_i$  aplicados no centro de massa de cada elo. Em geral, essa matriz é dependente da geometria de cada elo. Para um problema bidimensional com dois cabos, ela pode ser formulada por inspeção, conforme apresentado na Figura 6, e exposta para um elo do manipulador como na Eq. (16).

$$\mathbf{W}_i = \begin{bmatrix} \sin\left(\frac{\theta_i}{2}\right) & \sin\left(\frac{\theta_i}{2}\right) & \sin\left(\frac{\theta_{i+1}}{2}\right) & \sin\left(\frac{\theta_{i+1}}{2}\right) \\ \cos\left(\frac{\theta_i}{2}\right) & \cos\left(\frac{\theta_i}{2}\right) & -\cos\left(\frac{\theta_{i+1}}{2}\right) & -\cos\left(\frac{\theta_{i+1}}{2}\right) \\ a_i \cos\left(\frac{\theta_i}{2}\right) & -a_i \cos\left(\frac{\theta_i}{2}\right) & -a_i \cos\left(\frac{\theta_{i+1}}{2}\right) & a_i \cos\left(\frac{\theta_{i+1}}{2}\right) \end{bmatrix} \quad (16)$$

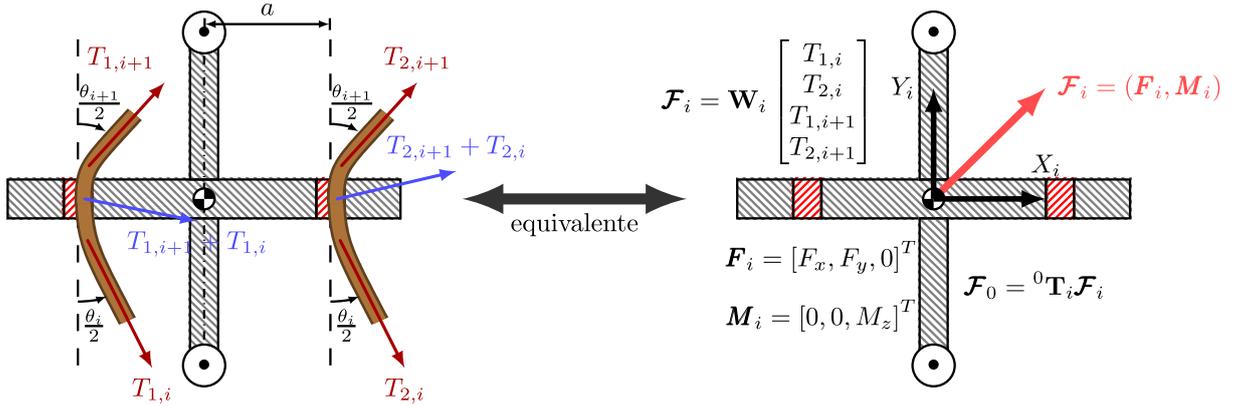


Figura 6 - Tensões devido a cabos e *wrench*  $\mathcal{F}$  equivalente atuando em um elo. A matriz  ${}^i\mathbf{W}$  é calculada para cada elo em função dos ângulos de rotação. Os *wrenches* são transformados do sistema local para o sistema da base via uma transformação adjunta  ${}^0\mathbf{T}_i$ .

Finalmente, a matriz de jacobianos seriais  $\mathbf{J}_s^T$  contém as matrizes jacobianas geométricas transpostas de cada elo com relação à base,  $\mathbf{J}_i^T$ . Estas servem para mapear  $\mathcal{F}_i$  de cada elo para os torques  $\tau_i$  correspondentes nas juntas. Nota-se que devido à estrutura serial deste robô  $\mathbf{J}_s^T$  será uma matriz de bloco diagonal superior, implicando que forças aplicadas nos elos mais próximos na base não gerarão torques de atuação equivalentes nas juntas mais adiante na cadeia cinemática. Essa matriz é explicitada na Equação (17) para um manipulador bidimensional.

$$\boldsymbol{\tau}_{ext} = \begin{bmatrix} \tau_1 \\ \vdots \\ \tau_i \\ \vdots \\ \tau_{n_j} \end{bmatrix} = \begin{bmatrix} \underbrace{(\mathbf{n}_1 \times \mathbf{r}_{1,1})^T \quad \mathbf{n}_1^T}_{\mathbf{J}_1^T} & \cdots & \underbrace{(\mathbf{n}_{n_j} \times \mathbf{r}_{n_j, n_j})^T \quad \mathbf{n}_{n_j}^T}_{\mathbf{J}_{n_j}^T} \\ \vdots & & \vdots \\ \mathbf{0}^T & \mathbf{0}^T & \cdots & (\mathbf{n}_i \times \mathbf{r}_{i, n_j})^T & \mathbf{n}_i^T \\ \vdots & & \vdots & \vdots & \vdots \\ \mathbf{0}^T & \mathbf{0}^T & & \underbrace{(\mathbf{n}_{n_j} \times \mathbf{r}_{n_j, n_j})^T \quad \mathbf{n}_{n_j}^T}_{\mathbf{J}_{n_j}^T} \end{bmatrix} \begin{bmatrix} F_{x,1} \\ F_{y,1} \\ M_{z,1} \\ \vdots \\ F_{x,n_j} \\ F_{y,n_j} \\ M_{z,n_j} \end{bmatrix} \quad (17)$$

O modelo dinâmico apresentado na Equação (12) pode ser estendido para uso com arquiteturas mais gerais de robôs *Snake Arm* multiseção e com roteamento geral de cabos entre os discos. Deve-se notar que esse modelo é limitado, uma vez que presume que os cabos estarão sempre sob tração e desconsidera efeitos como elasticidade, atrito dinâmico ou contato externo com o ambiente, que se mostram fundamentais para o controle em ambientes não-estruturados. Apesar disso considera-se que ele serve como uma primeira aproximação para o estudo da dinâmica deste manipulador no plano de flexão.

### 3. DESENVOLVIMENTO

Visando à validação dos modelos apresentados anteriormente, foram desenvolvidos programas para o cálculo automático das variáveis cinemáticas ( $\mathbf{q}$ ,  $\dot{\mathbf{z}}$ ,  $\mathbf{x}$ ) e simulação da dinâmica direta visando atuação por tensões nos cabos. Os modelos cinemáticos foram implementados em um pacote escrito em C++ e Python para o ambiente ROS2 Humble, com visualização do manipulador no software RViz. O modelo da dinâmica direta, por sua vez, foi implementado em MATLAB para um manipulador *Snake Arm* planar inextensível de uma seção com 2 cabos. Em particular, para validação do modelo dinâmico desenvolvido, foram realizados testes com um robô protótipo de uma seção deste tipo desenvolvido pelo LAMECC (Laboratório de Mecatrônica e Controle) da UFRGS. Neste capítulo detalhar-se-á a estrutura e o funcionamento desses dois simuladores, bem como o *setup* experimental para a validação do modelo dinâmico.

#### 3.1. SIMULADOR CINEMÁTICO

O robô foi descrito através do formato URDF (*Universal Robot Description Format*), com um arquivo para cada seção. Os mapas cinemáticos foram implementados como nodos divididos por seção no ROS 2, onde cada nodo assina e publica tópicos contendo valores dos respectivos espaços de configuração, conforme detalhado na Figura 2. Ao receber mensagens de posição ou velocidade os nodos então mapeiam estes mesmos para o outro espaço de acordo com os modelos cinemáticos previamente apresentados. A conexão entre os espaços das seções e os espaços globais cartesianos e de cabos é feita via nodos dedicados para estes mapas. Mais detalhes sobre o ROS 2 e este pacote estão disponibilizados no Apêndice A.

#### 3.2. SIMULADOR DINÂMICO

Tipicamente, a simulação de sistemas multicorpos é realizada utilizando algoritmos otimizados, como Newton-Euler recursivo ou o algoritmo de corpos articulados (SICILIANO; KHATIB, 2016). Apesar disso, optou-se pelo uso do sistema obtido por meio da formulação Lagrangeana apresentada na Eq. (12) para o simulador dinâmico devido à sua facilidade de implementação e a facilidade para incorporar termos ou estados adicionais. Este é um sistema de equações diferenciais de segunda ordem acopladas e não lineares nos ângulos das juntas. Para resolvê-las, primeiro elas devem ser reescritas em forma de espaço de estados, como descrito em (GHOSAL, 2006), gerando um total de  $2N$  equações para um sistema com  $N$  graus de liberdade. Para o manipulador estudado, isto leva à Equação (18).

$$\begin{aligned} \dot{\mathbf{q}}_i &= \mathbf{q}_{i+N}, i = 1, \dots, N \\ \ddot{\mathbf{q}}_i &= \mathbf{M}^{-1}(\mathbf{J}_S^T \mathbf{W} \mathbf{B} \mathbf{u} - \mathbf{C} \dot{\boldsymbol{\theta}} - \mathbf{K}(\boldsymbol{\theta} - \boldsymbol{\theta}_0) - \mathbf{G} - \mathbf{F}), i = N + 1, \dots, 2N \end{aligned} \quad (18)$$

O método de integração numérica de equações diferenciais de um sistema dinâmico é dependente não só da escala de tempo desejada para a solução, mas também da natureza do sistema estudado. Para sistemas "rígidos"<sup>1</sup>, utiliza-se tipicamente métodos implícitos, uma vez que estes são capazes de tomar grandes passos com relativo baixo erro de integração. Como um sistema multicorpo com grandes rotações é "rígido", foi utilizado o *solver* de equações diferenciais *ode15s* do MATLAB neste trabalho, que é um método *multistep* de tamanho de passo variável com ordens de 1 até 5, conforme recomendado por (SHAMPINE; REICHEL, 1996).

<sup>1</sup> De forma pragmática, problemas ditos "rígidos" são aqueles que requerem um passo de integração proibitivamente pequeno para métodos explícitos (HAIRER; WANNER, 1996).

1997) para situações como essa. Foram utilizados passos de tamanho máximo de 0,01 segundos, com tolerâncias absoluta de  $10^{-5}$  e relativa de  $10^{-4}$  para o erro na solução. Estes valores foram escolhidos por apresentarem resultados convergentes com relativa precisão e baixa demora.

### 3.3. CASO DE ESTUDO: DINÂMICA DE UM MANIPULADOR PLANAR

No caso de estudo foi considerado um manipulador *Snake Arm* protótipo atuado por cabos movimentando-se no plano. Esse manipulador consiste de 5 elos com discos espaçadores, conectados por juntas esféricas entre cada elo e com uma mola helicoidal de compressão montada entre cada um dos discos. Cada disco conta com um total de 12 furos dispostos em torno do eixo principal de sua coluna. Essa arquitetura é modular, uma vez que os cabos podem ser dispostos tal que uma seção pode ter mais ou menos elos em função da aplicação. Os atuadores em si, que aplicam tensões nos cabos, são localizados na base. Os parâmetros de um elo do manipulador estudado neste trabalho podem ser vistos na Tabela 1; como única exceção, o último elo apresenta  $m_{elo,5} = 201,33$  g e  $I_{z,elo,5} = 76822$  gmm<sup>2</sup>. A Figura 7 apresenta o manipulador protótipo como utilizado na validação experimental.

Tabela 1 - Parâmetros geométricos e inerciais de um módulo do robô, obtidos via modelo CAD, adicionados da massa e inércia dos sensores e placas de madeira.

$n_j$ [-]	$n_{c,p}$ [-]	$a$ [mm]	$b_1$ [mm]	$b_2$ [mm]	$m_{elo,i}$ [g], $i = 1 \dots 4$	$I_{z,elo}$ [gmm <sup>2</sup> ]
5	4	32,0	25,0	13,5	222,57	79398

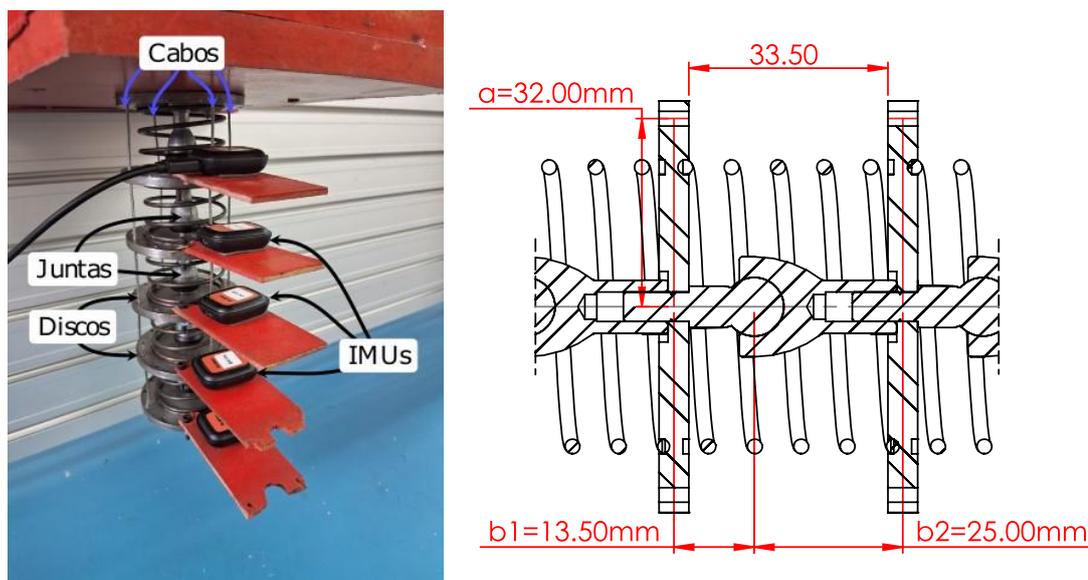


Figura 7 – À esquerda: *Setup* do robô com IMUs montados em placas de madeira. O sensor da primeira junta ficou conectada a um carregador devido à bateria defeituosa. À direita: vista em corte de um dos módulos do robô, com dimensões marcadas.

Cada um dos elos do robô foi equipado com sensores inteligentes do tipo IMU (*Inertial Measurement Units*) XSens DOT®, posicionados em placas de madeira fixadas nos discos. Esses sensores contém um acelerômetro, um giroscópio e um magnetômetro de 3 eixos cada. Idealmente, estas 9 medidas são fundidas através de uma implementação proprietária de um filtro de Kalman do sensor (XSENS, 2022); porém, como os discos do robô são feitos de uma liga magnetizável, as medidas do magnetômetro não são confiáveis e devem ser descartadas. Similarmente, não é possível utilizar um filtro complementar para combinar as estimativas de ângulo resultantes do giroscópio e do acelerômetro para medições dinâmicas devido à presença

de acelerações tangenciais e angulares com alta magnitude. Portanto, foram utilizados apenas os sinais do giroscópio com bias corrigido no eixo de rotação para o caso dinâmico, e um filtro complementar com o acelerômetro e o giroscópio para o caso estático. Os sinais foram amostrados a 120 Hz e sincronizados entre si, resultando nos ângulos e velocidades das juntas medidos,  $\theta_m$  e  $\dot{\theta}_m$ , pela diferença entre o ângulo estimado por um sensor e o seu anterior.

O modelo dinâmico proposto na Eq. (12) é um modelo caixa-cinza (AGUIRRE, 2015) que contém duas características desconhecidas por junta: a rigidez à flexão equivalente das molas e os parâmetros do atrito equivalente das juntas esféricas. Para identificar esses parâmetros foi realizado um teste em bancada, na qual um dos cabos do robô foi tracionado lentamente até que este atingisse um ângulo de dobra de 90°, e em seguida o mesmo foi solto. Esse teste foi repetido 8 vezes, com 4 repetições de cada lado.

De posse de  $\theta_m$  e  $\dot{\theta}_m$ , os parâmetros  $K, \tau_v, \tau_c, S$  e  $v_d$  foram identificados via algoritmo *particle swarm* de otimização heurística, implementado conforme descrito em (GOMES, 2016). Metade dos sinais foram utilizados como conjunto de dados para identificação dos parâmetros, enquanto a outra metade foi utilizada para validação dos parâmetros obtidos. A função objetivo utilizada para o *particle swarm* foi a média das raízes dos erros médios quadrados normalizados (*Normalized Root Mean Square Error, NRMSE*) pelo desvio padrão da velocidade medida de cada junta,  $\text{std}(\dot{\theta}_{m,i})$ , dado pela Equação (19).

$$\overline{NRMSE} = \frac{1}{n_j} \sum_i^{n_j} \frac{\sqrt{\frac{1}{n_{t,k}} \sum_k^t (\dot{\theta}_k - \dot{\theta}_{m,k})^2}}{\text{std}(\dot{\theta}_{m,i})} \quad (19)$$

## 4. RESULTADOS

Nesta seção são apresentadas as validações dos modelos desenvolvidos e a sua implementação nos simuladores. O modelo cinemático foi validado utilizando um problema de seguimento de trajetória, analisando assim a solução da cinemática diferencial inversa. O modelo dinâmico foi validado com a resposta à gravidade sem acionamento nos cabos. Finalmente, os modelos cinemático e dinâmico e o protótipo são comparados com relação à curvatura prevista para uma flexão completa de 90° com o robô montado na vertical.

### 4.1. CINEMÁTICA DIFERENCIAL INVERSA

Para validação do modelo cinemático inverso, foi gerada uma trajetória circular a ser seguida por um robô *Snake Arm* com os mesmos parâmetros cinemáticos apresentados na Tabela 1. O caminho a ser percorrido consiste de uma hélice de raio de 15 centímetros iniciando a 1 centímetro do chão, com passo de 8 centímetros e terminando em 30 centímetros de altura. O caminho foi gerado via 300 *via points* com suavização entre eles, gerado com auxílio da biblioteca OROCOS KDL. O perfil de velocidade usado foi trapezoidal, com velocidade máxima de 0,1 m/s e aceleração de 0,02 m/s<sup>2</sup>. A Figura 8 apresenta os comprimentos totais de cabos obtidos pelo simulador no ROS 2, bem como a velocidade dos cabos obtidas por jacobiano quando comparadas com a sua aproximação obtida por diferenças finitas da posição. Nota-se a irregularidade nos pontos espaçados, decorrentes da qualidade de serviço dos tópicos no ROS 2. Apesar das irregularidades e dos picos devido à leitura, os resultados de velocidade são muito similares, validando assim a cinemática inversa diferencial.

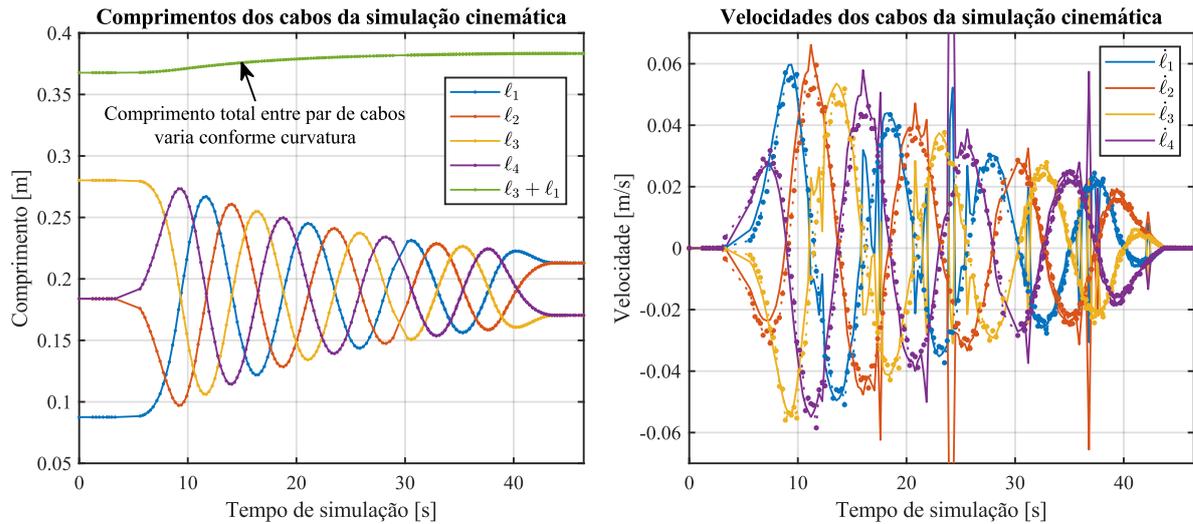


Figura 8 – À esquerda: Comprimentos totais dos cabos. A variação do comprimento total de par de cabos indica que uma atuação diferencial (via polias, por exemplo) não seria livre de variação na tensão dos cabos. À direita: Velocidades dos cabos via tópico `\cable_states` (linhas sólidas) e obtidas por diferenças finitas (pontos).

## 4.2. RESPOSTA DINÂMICA À GRAVIDADE

Os parâmetros utilizados para a otimização via algoritmo de *particle swarm* podem ser vistos na Tabela 2; os valores físicos identificados para o robô em estudo podem ser vistos na Tabela 3. A Tabela 4 apresenta os valores de *NRMSE* médios para os sinais de velocidade e de posição. Como exemplo, a Figura 9 apresenta os resultados previstos com os parâmetros identificados comparado com os valores medidos para posição das juntas na primeira repetição.

Tabela 2 - Parâmetros utilizados para o algoritmo de *particle swarm* e parâmetros identificados do manipulador.

Parâmetro	Número de partículas $n_p$	Número máximo de iterações $n_{iter}$	Tolerância da função objetivo $\epsilon_f$	Tolerância do coeficiente de variação $\epsilon_{COV}$
Valor	20	40	0,001	0,0001
Parâmetro	Componente cognitivo $\lambda_1$	Componente social $\lambda_2$	Momento para velocidade $\varpi$	Coefficiente para evitar divergência $\chi$
Valor	3	2	0,1	0,1146

Tabela 3 – Parâmetros físicos identificados para o manipulador com o algoritmo PSO.

Parâmetro	$K$ [Nm/rad]	$\tau_\mu$ [Nms/rad]	$\tau_c$ [Nm/rad]	$S$ [–]	$v_d$ [rad/s]
Valor	1,2379	0,0314	0,0413	0,7472	0,2994

Tabela 4 -  $\overline{NRMSE}$  de posição e velocidade para repetições. Os sinais 1, 2, 5 e 8 foram os sinais de identificação.

SINAL	1	2	3	4	5	6	7	8
$\overline{NRMSE}_p$	0,43335	0,73348	0,47457	0,54565	0,67894	0,61284	0,58105	0,5226
$\overline{NRMSE}_v$	0,68125	0,65181	0,62494	0,60868	0,58333	0,64235	0,63264	0,67874

A Figura 9 evidencia que, apesar de não reproduzir perfeitamente os resultados medidos com o robô protótipo, o modelo dinâmico proposto é capaz de aproximar importantes efeitos presentes no robô real, a notar a zona-morta de atrito estático para as juntas, bem como a característica oscilatória similar a um sistema de segunda-ordem. As principais discrepâncias ocorrem com a amplitude das oscilações e no início da simulação, sugerindo a possibilidade de os valores de inércia considerados serem significativamente diferentes dos valores reais do manipulador.

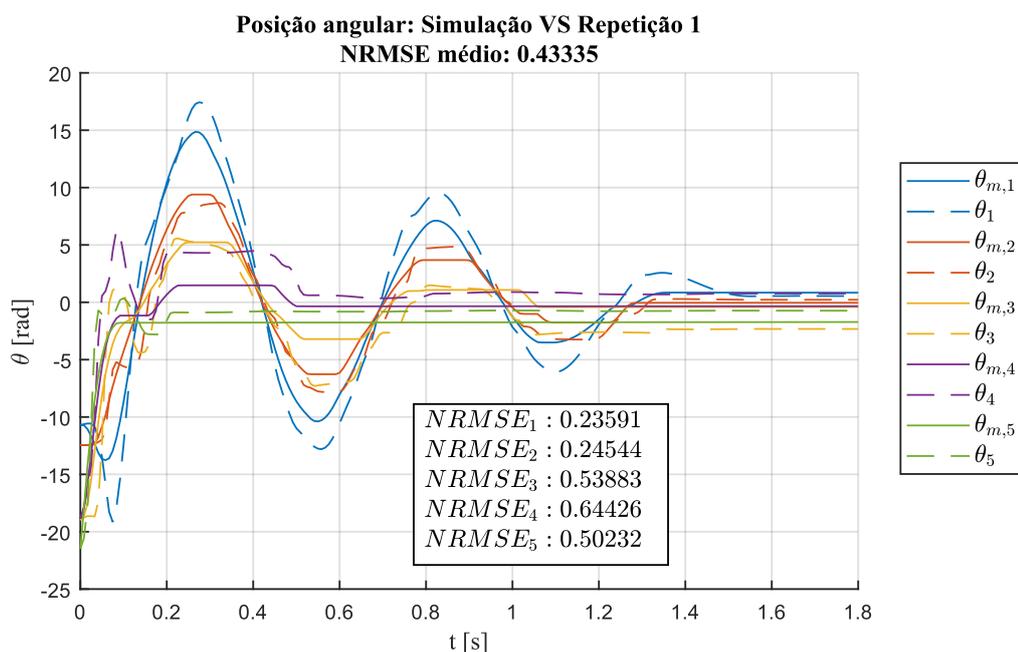


Figura 9 - Posição angular prevista (linhas sólidas) VS medida (linhas tracejadas) das juntas para a repetição 1. O modelo se encaixa razoavelmente bem com o medido.

### 4.3. CURVATURA PREVISTA PARA FLEXÃO

Como resultado comum aos modelos cinemático e dinâmico e o robô protótipo, é possível analisar os ângulos de junta no plano resultantes para uma mesma flexão total prevista. A Figura 10 apresenta esta comparação. No gráfico à esquerda, as barras de erro dos dados experimentais representam um intervalo entre quantis de 2,5% e 97,5% da função distribuição acumulada, e os pontos marcados são a mediana das medidas tomadas. No gráfico à direita, os círculos representam as posições das juntas do manipulador.

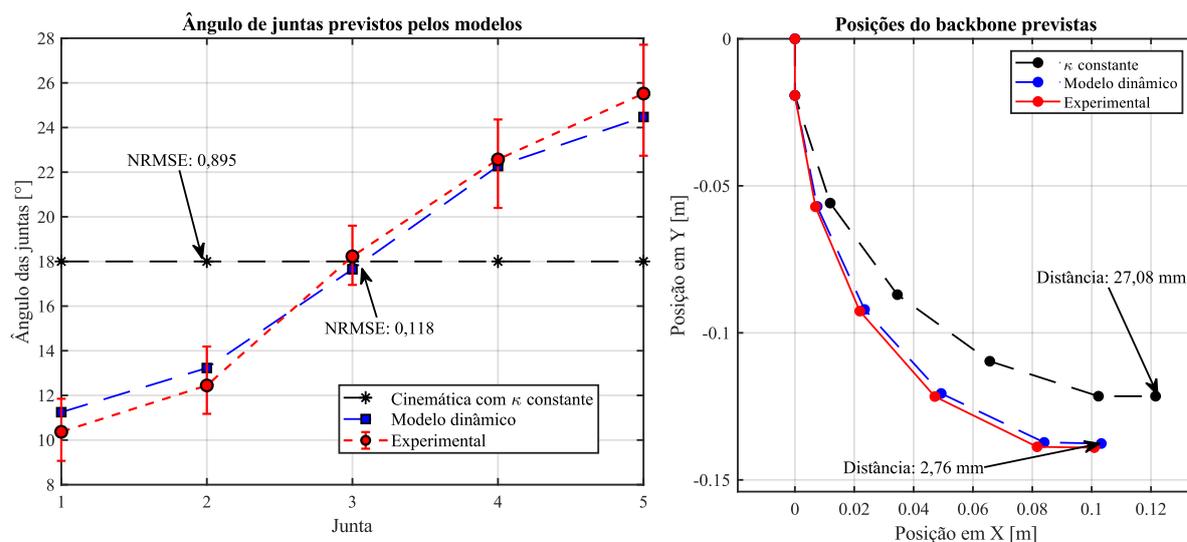


Figura 10 – À esquerda: Comparação entre ângulos de juntas previstos para uma flexão total de  $90^\circ$ . A simulação dinâmica utilizou os parâmetros estimados anteriormente e  $\mu_c = 0,2$ . À direita: Espinha dorsal do robô prevista pelos modelos versus experimental, medida via filtro complementar do acelerômetro e giroscópio. Valores de distância são entre a posição do *frame* final prevista pelos modelos e a posição estimada com valores experimentais de  $\theta_m$ . O modelo dinâmico consegue captar fielmente a curvatura medida.

É possível notar que o modelo dinâmico proposto aproxima a curvatura variável intrínseca ao manipulador mais fielmente que o modelo cinemático. Isto ocorre tanto pela perda de tensão ao longo dos cabos em função do atrito quanto pela massa do manipulador em si.

## 5. CONCLUSÕES

Este estudo avaliou os modelos de cinemática e dinâmica para um manipulador *Snake Arm* atuado por cabos, montado com juntas esféricas e sob efeito de seu peso próprio. Foram apresentadas as principais representações da cinemática direta e inversa desta classe de manipuladores, bem como deduzido um modelo dinâmico para uma seção de um destes manipuladores. Esses modelos foram implementados computacionalmente e os resultados das simulações comparados com um protótipo com relação à flexão do mecanismo no plano.

Os resultados do modelo cinemático mostram que, apesar de servir como uma consideração útil para o desenvolvimento de algoritmos de planejamento de trajetória, ele não é adequado para situações onde o atrito dos cabos ou a massa do próprio manipulador não possam ser desprezados, como no protótipo estudado no trabalho. Já os resultados do modelo dinâmico com os parâmetros ajustados foram capazes de reproduzir apenas aproximadamente o comportamento dinâmico observado. Apesar disto, o modelo dinâmico apresentou uma melhor aproximação com relação à curvatura no plano observada no protótipo.

Deve-se notar que o procedimento de estimação de parâmetros aqui apresentado foi realizado de uma maneira diferente da usualmente realizada na literatura. Para fins de controle, é mais interessante que a identificação do sistema seja realizada em conjunto com o sistema de atuação, com uma entrada com banda de frequências com excitação persistente e em diferentes pontos de operação (já que o sistema é não-linear nos parâmetros). Ademais, a hipótese de parâmetros iguais para todos os elos não é condizente do ponto de vista de identificação de sistemas, apesar de reduzir o tempo de computação.

Como sugestões de trabalhos futuros, há a extensão do modelo dinâmico para considerar todos os três graus de liberdade (i.e., incluindo a torção) das juntas esféricas, o que é interessante na medida em que permitirá levar em conta explicitamente os efeitos da torção nas juntas do manipulador. Ademais, uma extensão deste modelo dinâmico para um caso multiseção levando em conta efeitos de acoplamento entre seções também é pertinente. Finalmente, uma adaptação do modelo cinemático para planejamento de trajetórias levando em conta os esforços dinâmicos do manipulador também é considerado importante.

## REFERÊNCIAS BIBLIOGRÁFICAS

- AGUIRRE, L. A. **Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais**. 4. ed. Belo Horizonte, Brasil: Editora UFMG, 2015.
- ALLEN, T. F. *et al.* Closed-Form Non-Singular Constant-Curvature Continuum Manipulator Kinematics. *Em: 2020 3RD IEEE INTERNATIONAL CONFERENCE ON SOFT ROBOTICS (ROBOSOFT)*. 2020. p. 410–416.
- ANDERSON, V. C.; HORN, R. C. **Tensor Arm Manipulator - US Patent 3497083.pdf**. Depositante: US Department of Navy. 3497083. Depósito: 1968. Concessão: 1968.
- ARISTIDOU, A.; LASENBY, J. FABRIK: A fast, iterative solver for the Inverse Kinematics problem. **Graphical Models**, v. 73, n. 5, p. 243–260, 2011.
- EGLE, T. **Whole-body Joint Friction Modeling in a Torque-controlled Humanoid Walking Robot**. 48 f. - Technische Universität München, Garching bei München, 2021.
- FRAZELLE, C. G. **Continuum Robots: Interfacing, Modeling, and Automation**. Clemson University, 2021.
- GHOSAL, A. **ROBOTICS: FUNDAMENTAL CONCEPTS AND ANALYSIS**. OUP India, 2006.
- GOMES, H. M. Multi-objective optimization of quarter car passive suspension design in the frequency domain based on PSO. **Engineering Computations**, v. 33, n. 5, p. 1422–1434, 2016.
- HAIRER, E.; WANNER, G. **Solving Ordinary Differential Equations II**. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996.
- HANNAN, M. W.; WALKER, I. D. Kinematics and the Implementation of an Elephant's Trunk Manipulator and Other Continuum Style Robots. **Journal of Robotic Systems**, v. 20, n. 2, p. 45–63, 2003.
- JONES, B. A.; WALKER, I. D. Practical Kinematics for Real-Time Implementation of Continuum Robots. **IEEE Transactions on Robotics**, v. 22, n. 6, p. 1087–1099, 2006.
- LAGES, W. F. Geração de Trajetória. NOTAS DE AULA DE ENG10026 - ROBÓTICA A. 2022. p. 20. *E-book*. Disponível em: <http://www.ece.ufrgs.br/~fetter/eng10026/>.
- MURRAY, R. M.; LI, Z.; SASTRY, S. S. **A Mathematical Introduction to Robotic Manipulation**. 1. ed. CRC Press, 1994.
- MUSTAFA, S. K. **Design and kinematic analysis of a cable-driven anthropomorphic robotic arm**. 225 f. Doutorado - Nanyang Technical University, Singapore, 2008.
- NEPPALLI, S. *et al.* Closed-Form Inverse Kinematics for Continuum Manipulators. **Advanced Robotics**, v. 23, n. 15, p. 2077–2091, 2009.
- OKOLI, F. *et al.* Cable-Driven Parallel Robot Simulation Using Gazebo and ROS. *Em: ARAKELIAN, V.; WENGER, P. (org.). ROMANSY 22 – Robot Design, Dynamics and Control*. Cham: Springer International Publishing, 2019. (CISM International Centre for Mechanical Sciences). v. 584, p. 288–295.
- QUIGLEY, M. *et al.* ROS: an open-source Robot Operating System. **ICRA workshop on open source software**, v. 3, n. 2, p. 5, 2009.

- RAO, P. *et al.* How to Model Tendon-Driven Continuum Robots and Benchmark Modelling Performance. **Frontiers in Robotics and AI**, v. 7, p. 630245, 2021.
- RUSSO, M. *et al.* Continuum Robots: An Overview. **Advanced Intelligent Systems**, v. 5, n. 5, p. 2200367, 2023.
- SHAMPINE, L. F.; REICHEL, M. W. The MATLAB ODE Suite. **SIAM Journal on Scientific Computing**, v. 18, n. 1, p. 1–22, 1997.
- SICILIANO, B.; KHATIB, O. (org.). **Springer Handbook of Robotics**. Cham: Springer International Publishing, 2016. (Springer Handbooks).
- TEDRAKE, R. **Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying and Manipulation**. 2023. Disponível em: <https://underactuated.csail.mit.edu/>. Acesso em: 26 jun. 2023.
- WEBSTER, R. J.; JONES, B. A. Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review. **The International Journal of Robotics Research**, v. 29, n. 13, p. 1661–1683, 2010.
- XSENS. **Xsens DOT User Manual**, Manual de uso, ver. G. Publicação eletrônica, 2022.
- ZHANG, W. *et al.* FABRIKc: an Efficient Iterative Inverse Kinematics Solver for Continuum Robots. *Em*: 2018 IEEE/ASME INTERNATIONAL CONFERENCE ON ADVANCED INTELLIGENT MECHATRONICS (AIM). Auckland: IEEE, 2018. p. 346–352.

## APÊNDICE A – Implementação do simulador cinemático em ROS 2

Este apêndice descreve o meta-pacote “continuum” para ROS 2 Humble, desenvolvido como parte deste trabalho para estudo da cinemática de manipuladores multiseção, bem como a estrutura do código do simulador cinemático. O metapacote inclui 4 pacotes e segue a estrutura de árvore apresentada na Figura 11. Este meta-pacote contém um programa gerador de descrições de robô *Snake Arm* multiseção, além de um pacote gerador de trajetórias baseado na biblioteca OROCOS KDL e em código originalmente desenvolvido pelo Prof. W. F. Lages (LAGES, W. F., 2022). Estes arquivos podem ser adquiridos junto ao autor, e estão disponibilizados publicamente no URL [https://github.com/gustavocomerlato/continuum\\_tcc](https://github.com/gustavocomerlato/continuum_tcc).

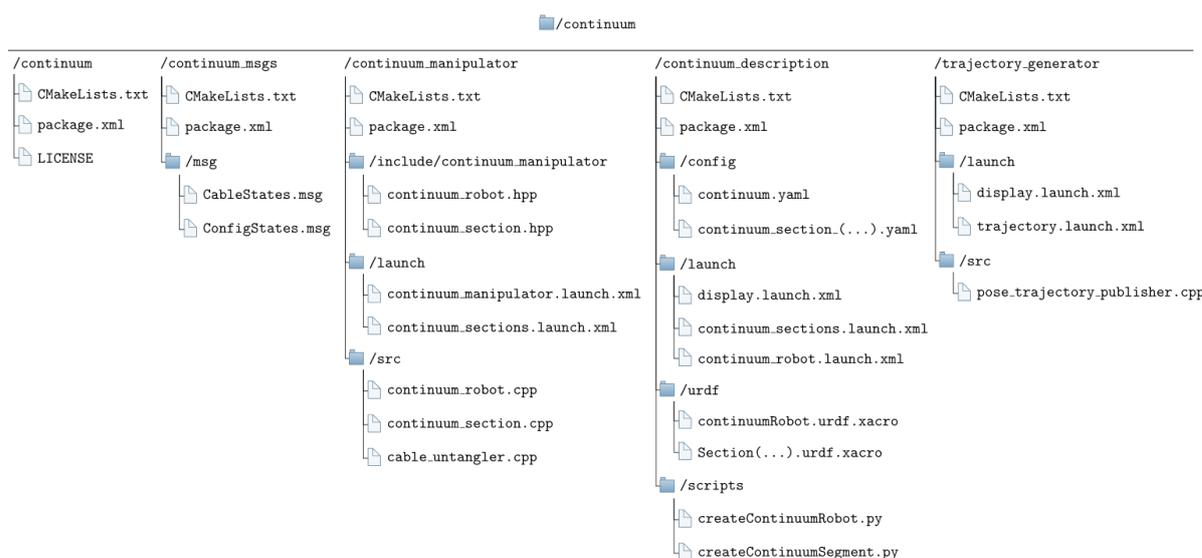


Figura 11 – Conteúdos do meta-pacote continuum desenvolvido para o presente trabalho utilizados no simulador cinemático.

O pacote “continuum” contém apenas os arquivos “CMakeLists.txt” e “package.xml”, que contém informações do meta-pacote e são necessários para descoberta pelo ROS 2.

O pacote “continuum\_description” contém quatro subpastas: a pasta de *launch*, a pasta URDF, a pasta de *config* e a pasta de *scripts*. Esse pacote contém a descrição do robô *Snake Arm* em formato urdf. Visando modularidade, essa é dividida em um arquivo principal, *continuum.urdf.xacro*, e vários arquivos individuais para descrição de cada uma das seções do manipulador. O formato urdf descreve o robô como uma série de elos conectados por juntas em uma cadeia cinemática em formato de árvore. Apesar do nome universal, esse formato na verdade é limitado a cadeias cinemáticas abertas, como é o caso de manipuladores seriais, e não tem suporte para juntas esféricas. Portanto, a descrição cinemática do robô *Snake Arm* via URDF foi limitada à estrutura serial do manipulador, excluindo os cabos. Por conveniência, foi desenvolvido os *scripts* utilitários *createContinuumRobot* e *createContinuumSegment* na linguagem Python para gerar automaticamente a descrição de um manipulador multiseção com parâmetros variáveis entre seções. A pasta de *config* contém parâmetros geométricos lidos pelos nodos das seções.

As juntas esféricas tipicamente são descritas como um conjunto de três juntas revolutas com eixos intersectantes no mesmo ponto, o que é uma parametrização de rotações 3D via 3 ângulos de Euler. A rigor, isto não é uma parametrização livre de singularidades (refletidas no famoso problema de *gimbal lock*), já que qualquer parametrização do grupo ortogonal especial de 3

dimensões  $SO(3)$  com três parâmetros é incapaz de fornecer uma carta global com esse grupo. Uma parametrização mais adequada é via quaternions unitários, que cobrem duplamente esse grupo (MURRAY; LI; SASTRY, 1994), mas isto não está disponível no formato urdf. Aqui, escolheu-se descrever uma seção  $i$  deste robô como uma junta revoluta na base proximal da seção (ângulo  $\varphi_i$ ) seguida de juntas revolutas posicionadas no centro das juntas esféricas (ângulos  $\theta_i/n_{j,i}$ ), e finalmente uma junta revoluta na extremidade distal, mimicking a junta na base em  $-\varphi$  radianos. Essa estratégia tem a vantagem de evitar uma grande quantidade de transformações cinemáticas intermediárias e se aproxima da parametrização de Jones, descrevendo uma seção apenas com os ângulos  $\varphi_i$  e  $\theta_i$ ; em contrapartida, os discos espaçadores intermediários são implementados como elos separados conectados via juntas fixas, sem cancelar a torção devido à rotação da base. Essa torção nos discos deve ser então cancelada ativamente pelo nodo `/continuum_section` da seção. A Figura 12 apresenta a estrutura cinemática e a representação dos *frames* de um robô multiseção conforme implementado.

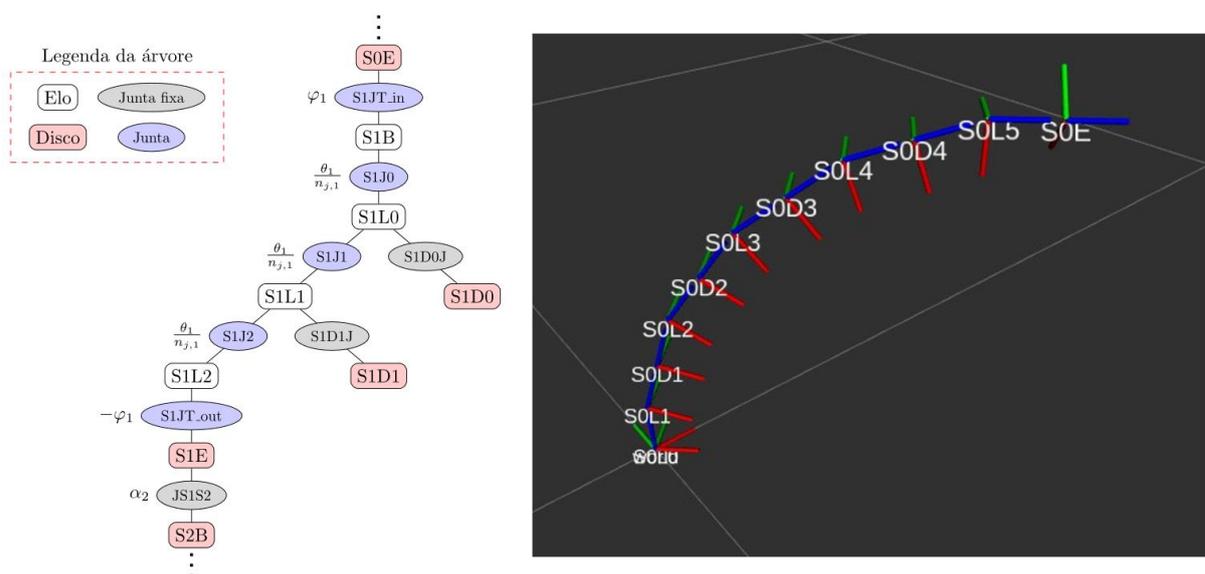


Figura 12 – À esquerda, cadeia cinemática de uma seção gerada pelo pacote `continuum_description`. À direita, sua representação no RViz como *frames*.

O pacote `continuum_manipulator` é o pacote “principal” do simulador cinemático. Ele contém a implementação de três classes-nodo em C++ para descrição de um robô *Snake Arm* multiseção atuado por 4 cabos por seção. A classe `continuum_section` contém a descrição de um nodo que realiza as transformações cinemáticas de uma seção do manipulador. Em operação cinemática inversa, ela assina o tópico `/joint_states`, que transmite mensagens do tipo `sensor_msgs/JointState` com valores de posições e velocidades angulares das juntas pertencentes à cadeia cinemática serial do robô. Essas posições e velocidades são transformadas em posições e velocidades no espaço de configurações  $\zeta_i$  da seção, que são então publicadas no tópico `/section_config_states` via mensagens do tipo `continuum_msgs/ConfigStates`. Além disso, este nodo também calcula os comprimentos e velocidades dos cabos, tanto atuantes como passantes, no espaço  $\mathbf{q}_i$ , e esses são então publicados no tópico `/section_cable_states`, que transmite mensagens do tipo customizado `continuum_msgs/CableState`, que foi implementado nesse trabalho pelo pacote `continuum_msgs`. A Figura 13 apresenta o gráfico de computação com nodos e tópicos durante a simulação de seguimento de trajetória de um robô com três seções.

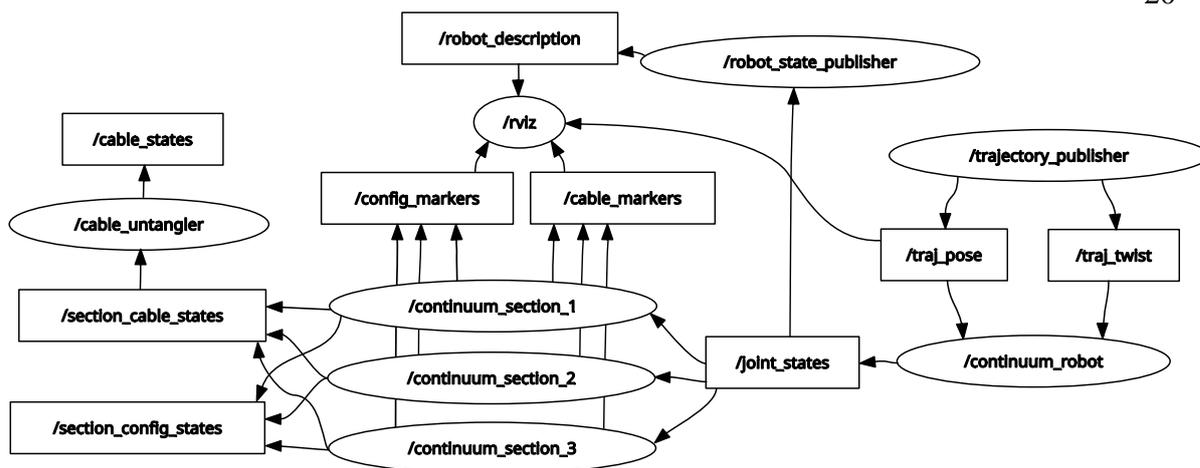


Figura 13 – Grafo de computação do simulador cinemático deste trabalho aplicado para um manipulador de 3 seções. As elipses são nós do ROS 2, e os retângulos são tópicos. Os tópicos /cable\_markers e /section\_config\_markers são para visualização no RViz.

O pacote `trajectory_generator` implementa um nó de nome `trajectory_publisher` que publica as trajetórias cartesianas nos tópicos `traj_pose` e `traj_twist` através de mensagens dos tipos `geometry_msgs/PoseStamped` e `geometry_msgs/TwistStamped`. Este tópico é assinado pelo nó `continuum_robot`, que realiza a cinemática inversa via `FABRIKc` para obter os ângulos e velocidades de juntas, que são então publicadas no tópico `/joint_states`.

Para visualização no RViz, os cabos são publicados como marcadores via tópico `/cable_markers`, enquanto o vetor  $\mathbf{w} = [u, v, 0]^T$  de parâmetros de Allen é publicado via tópico `/config_markers`, posicionado no centro de curvatura da seção. A fins de ilustração, a Figura 14 apresenta um exemplo de manipulador multiseção representado no RViz, com torção não-nula entre as seções e parâmetros geométricos variáveis. Os cabos são representados em azul, verde, vermelho e preto, incluindo cabos passantes e distais.

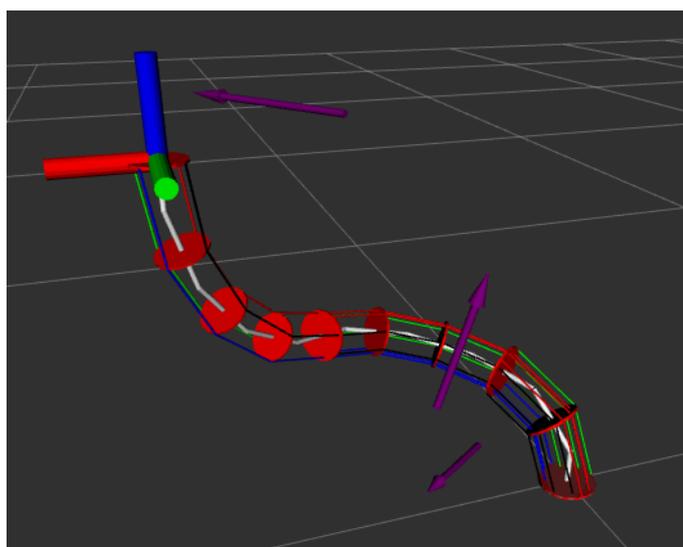


Figura 14 – Visualização de um manipulador de 3 seções no ambiente RViz seguindo uma trajetória. As seções são torcidas em 30 graus entre si, com  $a_i$  decrescente entre elas. Descrição URDF gerada pelo pacote desenvolvido neste trabalho.

## APÊNDICE B – Relações de cinemática diferencial

Neste apêndice são apresentadas as relações cinemáticas diferenciais inversas implementadas no simulador cinemático. As matrizes jacobianas foram obtidas via computação simbólica através do programa MATLAB. A Figura 15 apresenta em forma de diagrama as relações de cinemática diferenciais implementadas neste trabalho e descritas nesse apêndice. Por clareza, os índices de seção serão omitidos no desenvolvimento a seguir.

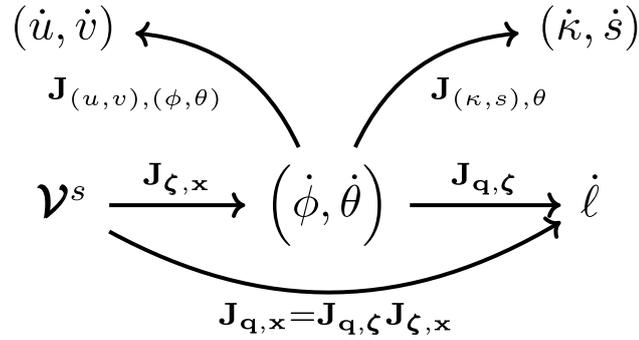


Figura 15 – Relações de cinemática diferencial via jacobianos presentes neste trabalho.

Como a rigor a configuração no espaço cartesiano pertence ao grupo especial euclidiano em 3 dimensões  $SE(3)$ , a expressão de  $\mathbf{J}_{x,\zeta}$  pode ser obtida via álgebra de *twists*, conforme exposto em Allen *et al.*, (2020) e desenvolvida em Murray, Li e Sastry, (1994). O *twist* espacial  $\mathcal{V}^s$  que contém as velocidades linear  $\mathbf{v}$  e angular  $\boldsymbol{\omega}$  de uma dada transformação homogênea  $T$  em função da  $i$ -ésima coordenada generalizada  $\zeta_i$  pode ser obtido pela Eq. (11), onde  $\mathcal{V}$  é um operador que transforma uma matriz de transformação homogênea  $4 \times 4$  para um *twist*  $6 \times 1$ . Esses *twists*, por sua vez, formam as colunas da matriz jacobiana  $\mathbf{J}_{x,\zeta}$ .

$$\mathbf{J}_{T,\zeta_i} = \left( \frac{dT}{d\zeta_i} T^{-1} \right)^{\mathcal{V}} \quad (20)$$

Aplicando a Eq. (11) para uma dada seção, temos a expressão do *twist*  $\mathcal{V}^s$  em função das coordenadas generalizadas  $(\varphi, \kappa, s)$  que descrevem a configuração de uma seção do manipulador. Para uma seção inextensível que depende apenas de  $\theta$  e  $\varphi$  como é o caso desse trabalho, o seu jacobiano está na Eq. (21) em seguida.

$$\mathbf{v}^s = \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = [\mathbf{J}_{x,\varphi} \mathbf{J}_{x,\theta}] \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\frac{bs_\varphi(c_\theta - 1)}{\tan\left(\frac{\theta}{2n_j}\right)} & \frac{-b(c_\theta - 1)}{n_j(c_{\theta/n_j} - 1)}c_\varphi \\ \frac{bc_\varphi(c_\theta - 1)}{\tan\left(\frac{\theta}{2n_j}\right)} & \frac{-b(c_\theta - 1)}{n_j(c_{\theta/n_j} - 1)}s_\varphi \\ 0 & \frac{-b}{2n_j} \cdot \frac{(s_\theta - n_j s_{\theta/n_j})}{s_{\theta/n_j}^2} \\ -s_\theta c_\varphi & -s_\varphi \\ -s_\theta s_\varphi & c_\varphi \\ 1 - \cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} \quad (21)$$

Como uma seção do manipulador descrito possui apenas dois graus de liberdade, o mapa  $IK_{gen}$  é restrito a atingir no máximo 2 das 6 coordenadas possíveis de um *frame* prescrito em  $\mathbf{x}$ . Escolhendo-se as coordenadas de posição e assumindo  $z > 0$  resulta em  $\mathbf{J}_{\zeta,\mathbf{x}}$  na Eq. (22). Para fins de geração de trajetórias no espaço de cabos, isso é equivalente a ignorar qualquer orientação prescrita.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \overbrace{\begin{bmatrix} \mathbf{J}_{\zeta,x} \\ \mathbf{J}_{\zeta,y} \\ \mathbf{J}_{\zeta,z} \end{bmatrix}}^{\mathbf{J}_{\zeta,\mathbf{x}}} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} -y/(x^2 + y^2) & x/(x^2 + y^2) & 0 \\ \frac{2xz(x^2 + y^2)^{-\frac{1}{2}}}{(x^2 + y^2 + z^2)} & \frac{2yz(x^2 + y^2)^{-\frac{1}{2}}}{(x^2 + y^2 + z^2)} & -\frac{2\sqrt{x^2 + y^2}}{x^2 + y^2 + z^2} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \quad (22)$$

Para obter as taxas das coordenadas de Allen  $u$  e  $v$  em função das variações de  $\varphi$  e  $\theta$ , basta derivar a Eq. (3), resultando na Eq. (23). Similarmente,  $\dot{\kappa}$  e  $\dot{s}$  são produto da jacobiana da Eq. (6) por  $\dot{\theta}$ , e são obtidos via Eq. (24).

$$\begin{bmatrix} \dot{u} \\ \dot{v} \end{bmatrix} = \overbrace{\begin{bmatrix} \mathbf{J}_{(u,v),\varphi} \\ \mathbf{J}_{(u,v),\theta} \end{bmatrix}}^{\mathbf{J}_{(u,v),(\varphi,\theta)}} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} -\theta c_\varphi & -s_\varphi \\ -\theta s_\varphi & c_\varphi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} \quad (23)$$

$$\begin{bmatrix} \dot{\kappa} \\ \dot{s} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\kappa}{2n_j} + \frac{1}{2bn_j} \\ \frac{1}{\kappa} - \frac{s}{n_j s_\theta / n_j} \end{bmatrix}}_{\mathbf{J}_{(\kappa,s),\theta}} \dot{\theta} \quad (24)$$

Para obter a jacobiana de  $\zeta$  para  $\mathbf{q}$ , é necessário tomar a derivada da Eq. (8). Isto pode ser simplificado com ajuda da regra da cadeia se for calculada a derivada entre o quadrado do comprimento de um cabo,  $\ell_i^2$ , e um dos parâmetros de configuração  $\zeta_i$ , Eq. (25), resultando em uma expressão menor e mais tratável computacionalmente. Para cabos posicionados conforme exposto na Figura 4, isso é expresso via Eq. (26), onde  $i$  é o índice do cabo e  $\alpha_i$  é o ângulo da linha de fixação do cabo em relação ao eixo  $^P X$ .

$$\frac{\partial \ell_i^2}{\partial \varphi} = \frac{\partial \ell_i^2}{\partial \zeta_i} \cdot \frac{\partial \zeta_i}{\partial \varphi} \Rightarrow \frac{\partial \ell_i^2}{\partial \varphi} = \frac{1}{2\ell_i} \frac{\partial \ell_i^2}{\partial \varphi} \quad (25)$$

$$\dot{\ell}_i = \left[ \begin{array}{c} \overbrace{\left( \frac{4as_{\varphi-\alpha_i} \left( bs_{\frac{\theta}{n_j}} - 2as_{\frac{\theta}{2n_j}} c_{\varphi-\alpha_i} \right)}{2\ell_i} \right)}^{\mathbf{J}_{\ell_i,\varphi}} \\ \overbrace{\left( \frac{2}{n_j} \left( a^2 s_{\frac{\theta}{n_j}} c_{\varphi-\alpha_i}^2 - b^2 s_{\frac{\theta}{n_j}} - 2abc_{\frac{\theta}{n_j}} c_{\varphi-\alpha_i} \right) \right)}^{\mathbf{J}_{\ell_i,\theta}} \\ \hline 2\ell_i \end{array} \right] \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \end{bmatrix} \quad (26)$$