**UNIVERSIDADE DE SÃO PAULO
ESCOLA DE ENGENHARIA DE SÃO CARLOS**

João Cavalcanti Santos

**Análise numérica e experimental da influência da
redundância cinemática em um manipulador paralelo
planar**

São Carlos

2017

João Cavalcanti Santos

**Análise numérica e experimental da influência da
redundância cinemática em um manipulador paralelo
planar**

Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, para obtenção do título de Mestre em Ciências - Programa de Pós-Graduação em Engenharia Mecânica.

Área de concentração: Dinâmica das Máquinas e Sistemas

Orientadora: Profa. Dra. Maíra Martins da Silva

ESTE EXEMPLAR TRATA-SE DA
VERSÃO CORRIGIDA.
A VERSÃO ORIGINAL ENCONTRA-SE
DISPONÍVEL JUNTO AO
DEPARTAMENTO DE ENGENHARIA
MECÂNICA DA EESC-USP.

São Carlos

2017

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTE TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

C237a Cavalcanti Santo, João
 Análise numérica e experimental da influência da
 redundância cinemática em um manipulador paralelo
 planar / João Cavalcanti Santo; orientadora Máira
 Martins da Silva. São Carlos, 2017.

 Dissertação (Mestrado) - Programa de Pós-Graduação
 em Engenharia Mecânica e Área de Concentração em
 Dinâmica das Máquinas e Sistemas -- Escola de
 Engenharia de São Carlos da Universidade de São Paulo,
 2017.

 1. Robótica paralela. 2. Redundância cinemática. 3.
 Resolução de Redundância. 4. Modelo. 5. Desempenho. I.
 Título.

FOLHA DE JULGAMENTO

Candidato: **JOÃO CAVALCANTI SANTOS**

Título da dissertação: "Análise numérica e experimental da influência da redundância cinemática em um manipulador paralelo planar"

Data da defesa: 03/07/2017

Comissão Julgadora:

Resultado:

Prof^a. Dr^a. **Maíra Martins da Silva (Orientadora)**
(Escola de Engenharia de São Carlos/EESC)

APROVADO

Prof. Associado **Eduardo Fontoura Costa**
(Instituto de Ciências Matemáticas e de Computação/ICMC-USP)

APROVADO

Prof. Dr. **Juan Francisco Camino**
(Universidade Estadual de Campinas/UNICAMP)

Aprovado

Coordenador do Programa de Pós-Graduação em Engenharia Mecânica:
Prof. Associado **Gherhardt Ribatski**

Presidente da Comissão de Pós-Graduação:
Prof. Associado **Luis Fernando Costa Alberto**

AGRADECIMENTOS

Me sinto afortunado pelo aprendizado e pela vivência que esse projeto me propiciou. O presente texto sintetiza em resultados um pouco disso que me foi tão precioso. Na realidade, por esse “projeto”, me vem à mente muito mais as pessoas com as quais convivi ao longo desse período. Esse aprendizado e essa vivência foram consequências da convivência com elas. Por isso, sou muito grato à cada uma dessas pessoas.

Agradeço especialmente a orientadora e amiga Maíra Martins da Silva. Além de dar sempre importantes orientações técnicas, foi certamente quem mais me motivou para os meus avanços profissionais.

Sébastien Krut me ensinou valiosos conceitos de robótica paralela com sua grande experiência. Sua sensibilidade e dedicação para que seus orientados trabalhem bem e com prazer é inigualável.

A reflexão sobre diversos temas com amigos do laboratório aumentou muito a qualidade e o escopo deste mestrado. Mais do que isso, fizemos juntos um ambiente agradável de trabalho. Dentre esse numeroso grupo, dedico agradecimento especial ao Henrique Garcia, Marcela Demarzo, Gustavo Lahr e Thales Pasqual.

Agradeço meus amigos de república e do colégio. Essas amizades criadas já há muitos anos me motivam e me dão segurança. Apesar de conhecê-los há menos tempo, o mesmo se dá para os amigos que conheci na França. Me receberam e fizeram com que eu me sinta em casa nesse país especial. Em particular, o grande amigo Guillaume Aiche.

O apoio da família foi essencial. Refletindo em momentos difíceis e compartilhando as felicidades. Igualmente importante foi o suporte da companheira Fernanda.

Finalmente, agradeço o suporte financeiro fornecido pela FAPESP através do projeto de mestrado 2014/21946-2 e a infra-estrutura disponibilizada pela EESC e LIRMM. Aqui incluo a dedicação dos técnicos do Laboratório de Dinâmica e da Université de Montpellier.

RESUMO

SANTOS, J. C. **Análise numérica e experimental da influência da redundância cinemática em um manipulador paralelo planar**. 2017. 103p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2017.

Manipuladores paralelos apresentam inércia reduzida, o que lhes permite alcançar altas acelerações e melhor eficiência energética. Porém, frequentemente seu espaço de trabalho possui desempenho pouco uniforme. De fato, a presença de singularidades paralelas é um forte limitante para essa arquitetura robótica. A redundância de atuação já é conhecida como uma alternativa para essa questão. Enquanto que a redundância cinemática ainda não possui consequências claras. Um manipulador com esse tipo de redundância apresenta um número de graus de liberdade do *end-effector* menor que o número de graus de liberdade do mecanismo como um todo. Considerando essa lacuna, o objetivo desse mestrado é analisar a influência da redundância cinemática no desempenho de manipuladores paralelos através de simulações e testes experimentais. Tal estudo não é trivial, uma vez que com o maior número de atuadores e graus de liberdade, é adicionada também inércia ao sistema. Foram definidas métricas para avaliar o quão favorável é uma dada posição e, com elas, estratégias de resolução de redundância foram analisadas. A estratégia principal proposta se compõe por duas etapas. No primeiro passo, é definida uma movimentação na qual a posição a cada instante é ótima segundo uma dada métrica de desempenho multiobjetivo. Isso resulta em um movimento de referência que em geral possui altas acelerações. Na segunda etapa, aplica-se uma otimização global, procurando manter um compromisso de proximidade com o movimento de referência e com os níveis de aceleração. Além deste, foram aplicados diversos métodos de otimização local (onde a cinemática inversa é resolvida para cada instante isoladamente) e uma estratégia global truncada. Essas opções foram comparadas numericamente e experimentalmente, trazendo uma resposta objetiva da influência da redundância cinemática no manipulador paralelo. A campanha experimental foi realizada em uma protótipo construído no Laboratório de Dinâmica da Escola de Engenharia de São Carlos. Esse protótipo consiste em um manipulador paralelo planar com 6 graus de liberdade, tendo assim, até 3 graus de redundância para a movimentação no plano. Têm-se 6 motores rotativos para atuá-los, sendo 3 deles acoplados a guias lineares com fusos para obtenção de atuação linear. O acionamento ou não destas guias define o grau de redundância do sistema, garantindo a versatilidade do protótipo.

Palavras-chave: Robótica paralela, Redundância cinemática, Resolução de redundância, Modelo, Desempenho

ABSTRACT

SANTOS, J. C. **Numerical and experimental investigation of the influences of the kinematic redundancy on the performance of a kinematically redundant parallel planar manipulator.** 2017. 103p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2017.

As a consequence of their reduced inertia, parallel manipulators present superior energetic efficiency and they are able to reach high accelerations. Nevertheless, their workspace has a poorly uniform performance. Indeed, the presence of parallel singularities is a strong limitant for this kind of robots. On the one hand, actuation redundancy is well-known as a good choice in an effort to solve this issue. On the other hand, kinematic redundancy still have unclear consequences on this matter. A kinematically redundant manipulator presents an end-effector with fewer degrees of freedom than the mechanism as a whole. Considering this gap, the objective of this research is to analyze the influence of kinematic redundancy on the performance of parallel manipulators through simulations and experimental tests. This issue turns out to be complex, once traveling actuators sum additional inertia to the system. Metrics are defined in order to evaluate how favourable is a given position, and redundancy resolution strategies are analyzed using them. The main proposed strategy is composed of two steps. In the first one, a movement is defined so that the position for each instant is optimum for a given multiobjective performance metric. This procedure delivers a reference movement which generally presents high accelerations. On the second step, a global optimization is applied, seeking for a trade-off between the proximity to the reference and the acceleration levels. In addition, several local methods (which resolve the inverse kinematics for each instant independently) and one truncated global strategy were addressed. These configurations were compared numerically and experimentally, delivering a objective analysis of the influence of kinematic redundancy on the performance of the parallel manipulator. The experimental campaign was executed with a physical prototype built at the São Carlos School of Engineering. This is a planar manipulator with 6 degrees of freedom, consequently presenting up to 3 degrees of redundancy. The mechanism is actuated by 6 rotating motors, of which 3 are coupled to leadscrews, resulting in linear actuators. These leadscrews can be locked, defining different degrees of redundancy and granting the versatility of the prototype.

Keywords: Parallel robot, Kinematic redundancy, Redundancy resolution, Model, Performance

LISTA DE FIGURAS

Figura 1 –	Plataforma de Stewart usada como simulador de vôo	20
Figura 2 –	Par4, robô de cinemática paralela com capacidade de atingir acelerações de até 15g	21
Figura 3 –	3RRR: manipulador planar sem redundância, 4RRR: manipulador planar com redundância de atuação e PRRR+2RRR: manipulador planar com redundância cinemática	23
Figura 4 –	Manipulador 3RRR em posição singular	24
Figura 5 –	Reconfiguração do manipulador planar PRRR+2RRR através da redundância cinemática	25
Figura 6 –	Manipulador planar PRRR+2RRR construído pelo Instituto de Sistemas Mecatrônicos, Universidade de Leibniz, Hanover, Alemanha	25
Figura 7 –	Versatilidade do manipulador robótico: configurações não-redundantes e redundantes	26
Figura 8 –	Relação entre limitantes dos atuadores com os limites no espaço de trabalho	34
Figura 9 –	Representação esquemática do 3(P)RRR	36
Figura 10 –	Existem infinitas posições possíveis dos atuadores para uma única posição do <i>end-effector</i>	45
Figura 11 –	Posição onde as derivadas parciais de ϕ_2 são nulas	50
Figura 12 –	Convergência de ζ para c_H crescente	56
Figura 13 –	Exemplo de movimento otimizado levando em conta suas acelerações e diferença com o movimento referência	56
Figura 14 –	Placa controladora EPOS2 50/5, motor EC 60 flat e redutor GP 52-C (componentes não estão em escala)	62
Figura 15 –	(a) Comunicação baseada em CAN e USB, e (b) Comunicação baseada em CAN e ETHERNET com interface da dSPACE 1103	62
Figura 16 –	Guia linear com fuso de esferas recirculantes	63
Figura 17 –	Acoplamento de fole BKL 003 usado para acionar as guias lineares . . .	63
Figura 18 –	Junta passiva de revolução: vista em corte e peça acabada	63
Figura 19 –	Vista superior do protótipo.	64
Figura 20 –	Painel com placas controladoras EPOS2 (enumeradas na parte superior), fontes (abaixo), disjuntor e contator (a direita).	65
Figura 21 –	Posicionamento usado para reconhecer a posição de um <i>encoder</i> de um atuador rotativo	66
Figura 22 –	Trajetórias de excitação	68
Figura 23 –	Trajetória triangular utilizada para validação	69

Figura 24 – Gráficos comparativos para trajetória de validação (A) atuadores rotativos e (B) atuadores lineares	69
Figura 25 – Posições visitadas na primeira trajetória	72
Figura 26 – Posição próxima de singularidade apresentada na trajetória 1	73
Figura 27 – Posições finais desejada e real após um experimento com o manipulador 3RRR	73
Figura 28 – Resultados numéricos para o manipulador não redundante: (A) Posições finais desejada (azul) e obtida através da dinâmica direta (vermelho), (B) Orientação desejada e simulada ao longo da tarefa	73
Figura 29 – Posições (A) cartesianas e (B) orientação simuladas com otimização global e distúrbio contínuo de 1,2Nm	74
Figura 30 – Correntes dos atuadores (A) lineares e (B) rotativos para um experimento com a primeira trajetória e a otimização global	75
Figura 31 – Correntes dos atuadores rotativos utilizando a otimização global truncada	75
Figura 32 – Posições visitadas na segunda trajetória	76
Figura 33 – Posições finais desejada (azul) e simulada (vermelho) para a segunda trajetória	77
Figura 34 – Forças demandas dos atuadores lineares para o métodos dos gradientes reduzidos	77
Figura 35 – (A) Posições cartesianas e (B) orientações desejadas, simuladas e experimentais	78

LISTA DE TABELAS

Tabela 1 – Tipos de redundância na Fig. 3	24
Tabela 2 – Especificação das dimensões do protótipo	64
Tabela 3 – Parâmetros de inércia, iniciais e ótimos	68
Tabela 4 – Parâmetros da Resolução de Redundância	78

SUMÁRIO

1	INTRODUÇÃO	19
2	REVISÃO BIBLIOGRÁFICA	29
2.1	Modelagem Cinemática e Dinâmica	29
2.2	Identificação de Parâmetros Inerciais	30
2.3	Resolução de Redundância	31
2.4	Métricas de Desempenho	33
3	MODELO	35
3.1	Modelo Cinemático	35
3.1.1	Cinemática Inversa	35
3.1.2	Cinemática Direta	37
3.1.3	Matriz Jacobiana	37
3.2	Modelo Dinâmico - Formalismo Euler-Lagrange	38
3.2.1	Matriz de massa	39
3.2.2	Equacionamento Dinâmico	40
3.2.3	Dinâmica Inversa	41
3.2.4	Dinâmica Direta	42
3.3	Modelo Dinâmico - Princípio dos Trabalhos Virtuais	44
4	RESOLUÇÃO DE REDUNDÂNCIA	45
4.1	Métricas de Desempenho	46
4.2	Métodos Locais	51
4.2.1	Método dos gradientes projetados - PG	52
4.2.2	Método dos gradientes reduzidos - RG	53
4.2.3	Gradientes reduzidos aplicados à aceleração	53
4.2.4	Condições de Contorno	54
4.3	Métodos Globais	54
4.3.1	Otimização Global por Programação Dinâmica	55
4.3.2	Otimização Global Truncada	60
5	PROTÓTIPO	61
5.1	Projeto, Dimensionamento e Montagem	61
5.2	Programação e Comunicação	64
5.3	Identificação de Parâmetros Inerciais	67
6	RESULTADOS	71

6.1	Primeira Trajetória	72
6.1.1	Não Redundante	72
6.1.2	Gradientes Reduzidos	74
6.1.3	Otimização Global por Programação Dinâmica	74
6.1.4	Otimização Global Truncada	75
6.2	Segunda Trajetória	76
6.2.1	Configuração não redundante	76
6.2.2	Gradientes Reduzidos	76
6.2.3	Otimização Global por Programação Dinâmica	77
6.2.4	Otimização Global Truncada	78
7	CONSIDERAÇÕES FINAIS	79
	REFERÊNCIAS	81
	ANEXOS	87
	ANEXO A – PLANEJAMENTO DE MOVIMENTO COM CÁLCULO VARIACIONAL	89
	ANEXO B – EQUIVALÊNCIA DE INÉRCIAS NA ATUAÇÃO LINEAR	93
	ANEXO C – CÓDIGOS DE INICIALIZAÇÃO MATLAB	95

1 INTRODUÇÃO

As máquinas de arquitetura cinemática paralela (*parallel kinematic machine* – PKM) são mecanismos com a cadeia cinemática fechada, ou seja, todos os componentes do mecanismo tem uma conexão com pelo menos dois elos. Esse tipo de manipulador tem atraído a atenção das comunidades acadêmica e industrial devido às suas vantagens sobre as de arquitetura serial. Dentre suas vantagens, pode-se citar a leveza, as elevadas velocidades e acelerações, a rigidez superior, a capacidade de carga e alta compactação (MERLET, 2012). A precisão de manipuladores similares comparando casos de arquitetura serial e paralela são comparadas objetivamente em (BRIOT; BONEV, 2007), provando resultados superiores aos manipuladores paralelos. De forma semelhante, em (MCCLOY, 1990) as duas arquiteturas foram comparadas considerando rigidez, consumo de potência e área de trabalho. Novamente, a mesma conclusão foi obtida.

O principal diferencial dos manipuladores paralelo quando comparados com os seriais é a possibilidade de fixar os atuadores à base. Isso é possível graças à presença de múltiplas cadeias cinemáticas ligando a base ao *end-effector*. Com isso, juntas passivas podem ser aplicadas. Um manipulador puramente paralelo pode ter apenas elementos passivos transladando no espaço, isto é, todos os atuadores são fixos à base. Tal característica reduz drasticamente a inércia da máquina (MERLET, 2012). Além disso, a configuração serial amplifica e acumula o erro de cada atuador. Esse problema não ocorre para manipuladores paralelos, melhorando sua precisão. Alta acuracidade obtida com mecanismos paralelos é um tema de estudos já bem estabelecido (KOSEKI et al., 1998; HESSELBACH et al., 2004).

Esse conjunto de qualidades faz com que a aplicação industrial mais promissora para esta arquitetura alternativa de sistemas robóticos sejam as operações *pega-e-põe*, necessárias nas indústrias alimentícia, farmacêutica, de componentes eletrônicos, entre outras (FINOTTI,). A redução do tempo de cada ciclo é crítica para esse procedimento. Como exemplos relevantes, podemos destacar as PKMs similares ao robô Delta, introduzido em (CLAVEL, 1988). Estes apresentam características pertinentes para tarefas de *pega-e-põe*. Outra possível aplicação é a utilização de PKMs em dispositivos *haptics* de alto desempenho que requerem baixa inércia, alta rigidez, baixo atrito, balanceamento gravitacional, múltiplos graus de liberdade e alta compactação (BONEV, 2001). Tido como o primeiro PKM, a plataforma de Stewart apresentada em (STEWART, 1965) é um robô de 6 graus de liberdade que foi utilizado como simulador de voo (vide Fig. 1). A alta rigidez e precisão também fazem com que manipuladores paralelos sejam aplicados como máquina-ferramenta. Shayya *et al.* desenvolve a síntese dimensional do manipulador *Arrow* em (SHAYYA et al., 2014). Ele foi projetado para desenvolver procedimentos de fresagem.

Robôs paralelos a cabo também apresentam grande relevância por possuírem grande espaço de trabalho e construção relativamente barata (GOUTTEFARDE; GOSSELIN, 2006). A aplicação mais conhecida é o robô suspenso por cabos usado com câmera para filmagem em estádios.

Figura 1: Plataforma de Stewart usada como simulador de vôo

Fonte: (STEWART, 1965)

Devido ao seu potencial, pesquisas têm sido realizadas com o objetivos de melhorar o desempenho e aplicabilidade desses manipuladores. O projeto, a modelagem e o controle de máquinas de configuração cinemática paralela têm sido tratados em várias referências recentes (DAVLIAKOS; PAPADOPOULOS, 2008; ABDELLATIF; HEIMANN, 2009; WANG et al., 2009b; WANG et al., 2009a; HESS-COELHO, 2007; HESS-COELHO, 2006; SILVA et al., 2010). Em (DAVLIAKOS; PAPADOPOULOS, 2008), modelos de baixa complexidade, incluindo a dinâmica de corpo rígido e a dinâmica de componentes hidráulicos, foram considerados para o projeto de um controlador. Em outro trabalho (ABDELLATIF; HEIMANN, 2009), usando o formalismo Lagrangiano, as equações de movimento para robôs paralelos foram obtidas explicitamente. Baseando-se no modelo dinâmico de corpo rígido de uma PKM, um controlador dinâmico usando torque calculado foi obtido por (WANG et al., 2009b). Para uma máquina de 5 graus de liberdade, o modelo dinâmico inverso foi obtido e uma estratégia de controle posição/força foi desenvolvida e testada experimentalmente em (WANG et al., 2009a). Hess-Coelho apresenta e averigua experimentalmente uma metodologia para a síntese topológica de sistemas robóticos de cinemática paralela em (HESS-COELHO, 2007; HESS-COELHO, 2006). A metodologia proposta em (SILVA et al., 2010) considera as flexibilidades para o projeto estrutural e de sinais de entrada do sistema, melhorando assim o procedimento convencional de projeto de PKMs de alto desempenho. Como averiguado pelos trabalhos supracitados e por (CORBEL et al., 2010), ao se projetar uma PKM de alto desempenho que será submetida a altas acelerações, modelos dinâmicos e cinemáticos devem ser avaliados.

Figura 2: Par4, robô de cinemática paralela com capacidade de atingir acelerações de até 15g

Fonte: (NABAT et al., 2005)

O projeto do manipulador de cinemática paralela Par4, ilustrado na Fig. 2, é descrito em (NABAT et al., 2005). O manipulador Par4 possui 3 graus de liberdade translacionais e um grau de liberdade rotacional sobre o eixo vertical. Isso o possibilita a fazer operações de *pega-e-põe* usuais. Sua tecnologia foi transferida para a indústria, resultando no robô industrial mais rápido para essa aplicação.

Apesar de apresentarem precisão, rigidez e propriedades dinâmicas elevadas, as PKMs sofrem com a presença de singularidades no seu espaço de trabalho (CONKUR; BUCKINGHAM, 1997). Pode-se compreender as singularidades na robótica como um fenômeno que deteriora a mobilidade do mecanismo. Singularidades seriais se caracterizam pela redução de um ou mais níveis de mobilidade, o que ocorre tipicamente no limite do espaço de trabalho da máquina ou quando juntas se posicionam colinearmente. Já as singularidades paralelas podem ocorrer no interior do espaço de trabalho do manipulador. São posições nas quais os atuadores perdem a capacidade de manter o mecanismo rígido. Como resultado, a relação entre o espaço útil de trabalho e o espaço físico ocupado pelo equipamento é baixo. Pesquisadores têm sugerido a utilização da redundância como maneira de minimizar o aparecimento de singularidades. Entre eles podem ser citados (CONKUR; BUCKINGHAM, 1997; KOTLARSKI et al., 2009; MOHAMED; GOSSELIN, 2005; KOTLARSKI; HEIMANN; ORTMAIER, 2011; ROCHA; SILVA, 2013; SANTOS; ROCHA; SILVA, 2013).

Redundância pode assumir diferentes características na robótica. Para compreendê-las, é necessário comparar três números: o número de atuadores (n_a), número de graus de liberdade do *end-effector* (n_e), o número de graus de liberdade (mobilidade) do mecanismo como um todo (n_m) e o número de sensores de posição (n_s). Os valores de n_a , n_e e n_s são evidentes. Por outro lado, a mobilidade, ou o número de graus de liberdade, do mecanismo n_m pode ser calculado pela fórmula de Grübler (AHUACTZIN; GUPTA, 1999). Para casos planares

$$n_m = 3(l - j - 1) + \sum_{i=1}^n d_i \quad (1.1)$$

onde l é o número de corpos rígidos incluindo a base, j é o número de juntas (tanto passivas como ativas) e d_i é o número de graus de liberdade da junta i .

Para um manipulador puramente paralelo sem redundância, têm-se $n_s = n_m = n_a = n_e$. Para os outros casos o conceito de redundância pode ser classificado em (MERLET, 2012; MERLET, 1996):

- **Redundância de medidas:** ($\mathbf{n}_s > \mathbf{n}_m$) Corresponde à obtenção de mais medidas do que graus de liberdade do efetuador. Essa metodologia é utilizada principalmente para a calibração e controle de sistemas robóticos.
- **Redundância de atuação:** ($\mathbf{n}_a > \mathbf{n}_m$) O número de atuadores é maior que o número de graus de liberdade do mecanismo. O manipulador se torna hiperestático. O método disponibiliza liberdade na escolha das cargas de cada atuador. Mas além do controle de tais manipuladores ser ainda um desafio (MOHAMED; GOSSELIN, 2005), a adição de uma cadeia cinemática pode reduzir a área de trabalho conforme verificado por (ROCHA; SILVA, 2013).
- **Redundância cinemática:** ($\mathbf{n}_m > \mathbf{n}_e$) O mecanismo possui um número de graus de liberdade maior que o *end-effector*. Isso disponibiliza múltiplas opções no posicionamento do mecanismo. Por consequência, o manipulador pode se reconfigurar evitando as singularidades (KOTLARSKI et al., 2009; MOHAMED; GOSSELIN, 2005; KOTLARSKI; HEIMANN; ORTMAIER, 2011; SANTOS; ROCHA; SILVA, 2013).

A Fig. 3 ilustra os conceitos de redundância de atuação e cinemática para o mecanismo planar 3RRR. Esse mecanismo apresenta 3 cadeias cinemáticas constituídas de 3 juntas de revolução (RRR). As juntas ativas são marcadas com texto sublinhado (R). O mecanismo denominado de 4RRR que ilustra o conceito de redundância de atuação apresenta 4 cadeias cinemáticas constituídas de três juntas de revolução. O mecanismo denominado de PRRR+2RRR ilustra o conceito de redundância cinemática. Uma junta

ativa prismática (P) foi adicionada a uma das cadeias cinemáticas do mecanismo planar 3RRR . O número de cadeias cinemáticas permanece constante (3), mas as possibilidades de reconfiguração fazem com que esse manipulador tenha menos posições singulares como verificado por (KOTLARSKI et al., 2009) e ilustrado pela Fig. 5.

As redundâncias de atuação e cinemática tem influência direta nas características construtivas do manipulador, sendo assim determinantes para o desempenho do manipulador. A redundância de atuação pode ser uma boa alternativa para o projeto de manipuladores de alto desempenho, devido ao aumento da capacidade dinâmica do sistema. Estudos sobre o uso de redundância para manipuladores de alto desempenho foi apresentado do LIRMM (CORBEL et al., 2010; CORBEL; GOUTTEFARDE; PIERROT, 2010), grupo de pesquisa renomado na área de robótica. O sistema estudado nesse trabalho foi um robô com estrutura semelhante ao Delta e as conclusões foram que para esse tipo de sistema robótico, a redundância de atuação poderia ser uma boa alternativa para o projeto de sistemas do tipo Delta de alto desempenho. Por outro lado, a avaliação de desempenho para manipuladores paralelos com redundância cinemática ainda não é incipiente. A fim de abordar essa avaliação, o estudo de um sistema mais simples é pertinente.

Figura 3: 3RRR : manipulador planar sem redundância, 4RRR : manipulador planar com redundância de atuação e $\text{P}\text{RRR}+2\text{RRR}$: manipulador planar com redundância cinemática

Fonte: modificado de (FONTES; SILVA, 2016)

O tipo de singularidade mais preocupante para os manipuladores estudados nesse trabalho é ilustrado na Fig. 4, o tipo paralelo, como discutido anteriormente. Considerando o robô 3RRR , pode-se ver que os elos mais afastados da base são fixos apenas por juntas de revolução passivas. Como consequência, de um ponto de vista estático, tais elos transmitem apenas forças ao longo de suas linhas de ação, representadas por linhas pontilhadas. Nessa

Tabela 1: Tipos de redundância na Fig. 3

Configuração	n_e	n_m	n_a	l	j	$\sum_{i=1}^n d_i$	Redundância
3RRR	3	3	3	8	9	9	–
4RRR	3	3	4	10	12	12	Atuação
PRRR+2RRR	3	4	4	9	10	10	Cinemática

Figura 4: Manipulador 3RRR em posição singular

Fonte: própria

posição específica, as três linhas de ação se cruzam em um único ponto, de forma que qualquer momento aplicado sobre o *end-effector* em torno do ponto destacado em vermelho, não pode ser suportado pelo mecanismo. Sendo assim, o manipulador perde completamente sua rigidez, de forma que o *end-effector* possui mobilidade infinitesimal mesmo com todos os atuadores travados. A Fig. 5 ilustra o manipulador PRRR+2RRR se reconfigurando para efetuar uma trajetória (linha preta) que atravessaria uma região singular (linha vermelha). Isso evidencia que a redundância cinemática entrega capacidade de reconfiguração ao mecanismo, o que o capacita a suprimir as singularidades.

O deslocamento da junta ativa de rotação pela junta ativa de translação faz com que o manipulador possa realizar o trajeto desejado sem problemas (KOTLARSKI et al., 2009). Segundo (KOTLARSKI et al., 2009; KOTLARSKI; HEIMANN; ORTMAIER, 2011), a reconfiguração pode acontecer durante (*ongoing positioning*) ou antes (*prepositioning*) da movimentação do *end-effector*. No caso do PRRR+2RRR, manipulador paralelo planar com 1 nível de redundância cinemática, a reconfiguração *ongoing positioning* consiste da realização do movimento linear simultaneamente com os movimentos de rotação. Já a reconfiguração *prepositioning* consiste na movimentação prévia do motor linear para uma configuração isenta de singularidades no trajeto desejado (como ilustrado na Fig. 5).

Um protótipo com essa configuração foi construído pelo Instituto de Sistemas Mecatrônicos na Universidade de Leibniz em Hanover na Alemanha (KOTLARSKI; HEIMANN; ORTMAIER, 2011). Essa máquina está ilustrado na Fig. 6 e consiste de 3

Figura 5: Reconfiguração do manipulador planar $\underline{P}RRR+2\underline{R}RR$ através da redundância cinemática

Fonte: (KOTLARSKI et al., 2009)

Figura 6: Manipulador planar $\underline{P}RRR+2\underline{R}RR$ construído pelo Instituto de Sistemas Mecatrônicos, Universidade de Leibniz, Hanover, Alemanha

Fonte: (KOTLARSKI; HEIMANN; ORTMAIER, 2011)

motores rotacionais e 1 motor linear garantindo ao sistema uma redundância cinemática ($\underline{P}RRR+2\underline{R}RR$). Dessa maneira, o mecanismo é capaz de se reconfigurar e, portanto, otimizar a sua geometria de acordo com vários critérios de desempenho e estratégias de controle de movimento (KOTLARSKI; HEIMANN; ORTMAIER, 2011).

Sendo assim, estudos realizados até então demonstraram que a redundância cinemática é uma boa alternativa para a redução do conjunto de posições singulares e aumento do espaço de trabalho. Dessa maneira, não só outros aspectos de avaliação de desempenho de manipuladores planares precisam ser investigados mas também o impacto do uso de mais que um nível de redundância cinemática. O objetivo desse trabalho é avaliar numérica e experimentalmente a melhoria trazida pela redundância cinemática para o desempenho de um manipulador planar. A resposta para essa pergunta não é trivial, pois

como a redundância cinemática consiste na adição de motores (juntas ativas), isto traz não somente mais torque disponível, mas também inércia adicional. O sistema estudado é ilustrado na Fig. 7. Grandes circunferências cinzas são motores rotativos, retas estreitas são elos, pequenas circunferências pretas são juntas rotativas passivas e o triângulo cinza é o *end-effector*.

Figura 7: Versatilidade do manipulador robótico: configurações não-redundantes e redundantes

Fonte: própria

Como indicado na Fig. 7, a habilitação ou não dos motores DC que atuam o fuso define o grau de redundância do sistema robótico, garantindo a versatilidade do protótipo. Dessa maneira, o mesmo protótipo pode ser utilizado para o estudo do sistema não-redundante (vide 3RRR na Fig. 7) com a não habilitação do movimento linear e dos possíveis sistemas redundantes (ver PRRR+2RRR, 2PRRR+RRR e 3PRRR) com a habilitação dos movimentos lineares. Modelos cinemáticos e dinâmicos do manipulador proposto devem ser desenvolvidos e explorados. Além da avaliação numérica, validação experimental deve ser executada com um protótipo físico. Este consiste de um manipulador paralelo planar atuado através de 6 motores DC. A movimentação linear é realizada por guias lineares e fuso de esferas recirculantes.

Além disso, um ponto chave dessa avaliação é a resolução de redundância. Uma vez que o manipulador possui mais graus de liberdade que o *end-effector*, a cinemática inversa é indefinida. Em outras palavras, é necessário definir uma estratégia de otimização para obter o movimento dos atuadores redundantes. Como se trata de uma otimização que recebe como argumentos as posições do manipulador para todos os instantes da tarefa,

várias abordagens podem ser levantadas. Devido à alta complexidade geométrica imposta pela arquitetura paralela, a metodologia já desenvolvida na academia pode ser deficiente para a aplicação proposta. Por isso, um novo método é introduzido com objetivo de suprir tais deficiências.

Neste cenário, os seguintes objetivos específicos deste mestrado são pontuados:

- **Modelagem:** Obtenção de modelos cinemático e dinâmico. Estes são essenciais para o controle e análise do manipulador. Uma metodologia para a dinâmica direta também é necessária;
- **Métricas de Desempenho:** Desenvolvimento de métricas capazes de medir o desempenho do manipulador para uma dada posição. Isso é indispensável para a resolução de redundância;
- **Resolução de Redundância:** Levantamento de diferentes estratégias para o planejamento do movimento dos atuadores redundantes. Métodos locais e globais devem ser discutidos. A falta de metodologia própria para um manipulador paralelo como o 3PRRR exige levantamento de uma nova estratégia. Sem ela, o desempenho deste robô será mal avaliado;
- **Protótipo:** Dimensionamento, montagem e programação de um protótipo 3PRRR. Este é indispensável para a validação experimental dos dados numéricos
- **Comparativo de Configurações:** Análise de resultados obtidos com as diversas configurações possíveis e posterior síntese da melhoria de desempenho obtida em cada caso de aplicação da redundância cinemática. Este é o principal objetivo deste trabalho.

Cada um destes pontos é abordado nos futuros capítulos. Mais especificamente, a modelagem é deduzida no capítulo 3, as métricas de desempenho junto com os métodos de resolução de redundância no capítulo 4, detalhes do protótipo experimental no capítulo 5 e a discussão de resultados no capítulo 6. A metodologia apresentada é suportada por revisão bibliográfica, sendo essa descrita no capítulo 2. Finalmente, as conclusões são dadas no capítulo 7, sintetizando as consequências dos resultados.

2 REVISÃO BIBLIOGRÁFICA

A obtenção dos resultados numéricos e experimentais finais apresentados no capítulo 6 dependeu da aplicação de diversos conceitos pertinentes à robótica. Isso só foi possível após uma detalhada análise de trabalhos já publicados sobre cada um desses temas. As seções a seguir apresentam as principais fontes estudadas e seus métodos propostos. Mais em detalhes o presente capítulo se divide da seguinte maneira:

- **2.1 Modelagem Cinemática e Dinâmica:** Métodos considerados para modelagem do sistema. Foi necessária o estudo da modelagem dinâmica por Trabalhos Virtuais e Lagrangiana. Métodos numéricos para integração de sistemas de equações algébrico-diferenciais também foram abordados;
- **2.2 Identificação de Parâmetros Inerciais:** Para correlacionar apropriadamente as simulações com os resultados experimentais, os parâmetros de inércia precisaram ser definidos. Com isso em mente, metodologias de identificação de parâmetros precisaram ser consultadas;
- **2.3 Resolução de Redundância:** A resolução da cinemática inversa foi o foco deste trabalho. Tantos métodos locais como globais foram levantados;
- **2.4 Métricas de Desempenho:** Métodos de medição de desempenho são indispensáveis para a resolução de redundância. Sendo assim, foram estudadas métricas de rigidez, manipulabilidade, desempenho dinâmico, entre outras.

2.1 Modelagem Cinemática e Dinâmica

Primeiro passo necessário para qualquer análise numérica do sistema robótico, o modelo do manipulador merece atenção especial. O modelo cinemático de robôs semelhantes já foi apresentado em diversas fontes, tal como (WU; WANG; YOU, 2011). Com isso, as restrições cinemáticas são equacionadas. Um resultado importante desta etapa é a obtenção da matriz jacobiana do manipulador. Essa matriz é definida como a matriz jacobiana da função vetorial que define as posições do *end-effector* em função das posições dos atuadores, ou seja, é a matriz jacobiana da cinemática direta (MERLET, 2012). Em outras palavras, é a matriz que mapeia linearmente as velocidades dos atuadores para as velocidades do *end-effector*.

Wu, Wang e You também deduzem em (WU; WANG; YOU, 2011) o modelo dinâmico inverso do 3RRR aplicando o método dos trabalhos virtuais. Esse modelo determina os torques necessários de cada atuador para uma dada movimentação. Contudo, é interessante obter também o modelo dinâmico direto, de difícil obtenção aplicando

o proposto em (WU; WANG; YOU, 2011). Com a dinâmica direta, é possível analisar a influência dos distúrbios, estratégias de controle e erros na definição de parâmetros geométricos e de inércia. Procurando isso, diferentes bibliografias de mecânica de corpos rígidos foram consultadas. O mais conveniente para este caso foi a modelagem pelo método Lagrangiano. Através dele, foi possível a inserção de variáveis de estado redundantes (mais que as seis coordenadas generalizadas exigidas pelo sistema de 6 *gdl*) com o uso conjunto de multiplicadores de Lagrange (GOLDSTEIN, 1965; LANCZOS, 1970). Assim, é possível simular distúrbios em cada junta e no *end-effector*, além de ter acesso direto às forças internas do sistema. Tais recursos não são disponibilizados com a aplicação do princípio dos trabalhos virtuais. Pelo método dos multiplicadores de Lagrange usando variáveis de estado adicionais, as equações diferenciais se tornam simples. Mas por outro lado, o sistema de equações deve receber também equações puramente algébricas. Apesar de frequente para modelagem dinâmica, o sistema final, denominado Sistema de Equações Algébrico-Diferenciais, não possui integração numérica simples como um sistema de equações puramente diferenciais. Métodos numéricos dedicados a esse problema foram estudados (RABIER; RHEINBOLDT, 2000; ASCHER; PETZOLD, 1998; BRENAN; CAMPBELL; PETZOLD, 1996).

Já implementada como funções *built-in* no MATLAB, a abordagem mais frequente aproxima tal sistema para um caso de equações puramente diferenciais *rígido*, com derivadas altamente variáveis (ASCHER; PETZOLD, 1998). Contudo, essa proposta não se mostrou robusta e precisa para o caso estudado. Também foi considerado método no qual se projeta as derivadas obtidas pelas equações diferenciais sobre o *manifold* correspondente às equações algébricas. Finalmente, foi implementado o *Backward Differential Formulae* (BRENAN; CAMPBELL; PETZOLD, 1996), no qual as derivadas são aproximadas interpolando as posições já integradas em instantes anteriores. Dessa forma, as equações diferenciais se tornam puramente algébricas. Consequentemente, a cada instante, um sistema de equações algébricas não lineares deve ser resolvido.

2.2 Identificação de Parâmetros Inerciais

O modelo só se torna útil quando é capaz de correlacionar resultados experimentais com numéricos. Para isso, é necessário definir parâmetros inerciais pertinentes. Devido à razoável complexidade da montagem mecânica, a medição desses parâmetros por métodos convencionais ou mesmo por predição com um software CAD não é simples (apesar de ser também factível). Sendo assim, decidiu-se aplicar uma estratégia de identificação de parâmetros inerciais.

De maneira geral, considerando um sistema qualquer com entradas e saídas, a identificação de parâmetros propõe que esse sistema seja excitado com diversas entradas e que os parâmetros incluídos no seu modelo sejam otimizados para que as saídas obtidas

numericamente e experimentalmente sejam o mais próximas possíveis.

Para o caso do manipulador estudado (Fig. 7), são desenvolvidos múltiplos testes com o protótipo, resultando em movimentos e torques lidos durante os experimentos. Sabendo que o modelo dinâmico deve correlacionar essas duas variáveis, o problema proposto é uma otimização na qual os argumentos são os parâmetros de inércia, e a função a ser minimizada é a diferença entre os torques lidos e os torques previstos pelo modelo.

Objetivamente, a discussão deste problema se foca em quais movimentações devem ser usadas, como tratar os dados lidos, e como proceder com a otimização. Swevers *et al.* em (SWEVERS; VERDONCK; SCHUTTER, 2007) propõem uma trajetória composta por uma série de Fourier finita. Essa trajetória deve ser repetida ininterruptamente diversas vezes. Seus parâmetros são definidos de forma a excitar todos os fenômenos dinâmicos relevantes e cobrir a área de trabalho da máquina. Como função objetivo, pode ser usado o critério de *d-optimality* proposto por Ljung em (LJUNG, 1999) que mede o tamanho da região de incerteza obtido pela identificação de parâmetros.

Como a movimentação é repetida, são calculados os valores médios para cada instante considerando seu período. Após isso, os dados são filtrados no domínio da frequência e finalmente convertidos para o domínio do tempo novamente. Outro problema descartado por essa proposta é o cálculo de velocidades e acelerações. Em um caso geral, isso exigiria aproximação por diferenças finitas e aplicação filtros para que isso não causasse erros inaceitáveis em razão de ruídos. Como os dados finais podem ser escritos no domínio da frequência, suas derivadas podem ser obtidas analiticamente. Por fim, os parâmetros de inércia capazes de minimizar a diferença entre dados numéricos e experimentais são obtidos através do *Linear Least Squares Parameter Estimation - LLSE*. Este método foi implementado por Swevers *et al.* em um KUKA IR 361. Os resultados estão presentes em (SWEVERS *et al.*, 1997). Com abordagem similar, Do Tahn *et al.* identificou com boa precisão em (THANH *et al.*, 2012) um manipulador semelhante ao estudado neste projeto de mestrado.

2.3 Resolução de Redundância

Manipuladores com redundância cinemática possuem maior mobilidade que a necessária para atuar seu *end-effector*. Como consequência, sua cinemática inversa é indeterminada, ou seja, existem infinitas posições possíveis dos atuadores para uma única posição do *end-effector*. Portanto, é necessário estabelecer uma metodologia de escolha de qual é o melhor posicionamento dos atuadores dentre essas infinitas possibilidades. Tal procedimento é chamado de resolução de redundância.

Sendo assim, a resolução de redundância é responsável por gerir os graus de

liberdade do mecanismo excedentes, planejando um movimento que melhore o desempenho durante a tarefa. Pode-se dividir tal campo de estudos entre dois grupos de métodos: locais e globais (KAZEROUNIAN; WANG, 1988). Os métodos locais definem velocidade ou acelerações ótimas para cada instante isoladamente, enquanto que os globais planejam o movimento considerando um desempenho geral da tarefa.

Kazerounian e Wang discutem em (KAZEROUNIAN; WANG, 1988) com uma visão geral ambas abordagens. É apresentado o método local mais bem estabelecido com nome de *Gradientes Projetados*, no qual as velocidades dos atuadores recebem a projeção do gradiente de uma métrica a ser maximizada no espaço nulo da matriz jacobiana (não produzindo movimentação do *end-effector*) somada ao vetor de norma mínima capaz de produzir a velocidade desejada do *end-effector*. Também são introduzidos métodos de otimização global fazendo uso do cálculo variacional para minimização de um funcional integrada ao longo da tarefa. O funcional proposto procura otimizar a norma das velocidades dos atuadores ao longo da movimentação como um todo. Isso se mostrou uma estratégia pertinente para um manipulador $\underline{RRR}$ serial. Porém a presença de singularidades no interior do espaço de trabalho do robô $3\underline{PRRR}$ não pode ser contornada com essa proposta, como será discutido no apêndice A.

Dentre outros métodos locais estudados, é importante ressaltar o métodos dos Gradientes Reduzidos (LUCA; ORIOLO, 1991), a Mínima Norma Ponderada Geral (XIANG; ZHONG; WEI, 2010), e a resolução no nível da aceleração. Siciliano também sintetizou alguns desses métodos junto com outras propostas semelhantes em (SICILIANO, 1990).

Já quanto aos métodos globais, detalhes quanto ao cálculo variacional (AGRAWAL; XU, 1994; GELFAND et al., 2000) foram consultados em um esforço de implementá-lo no mecanismo paralelo $3\underline{PRRR}$. Também foi revisitado um método proposto pelo mestrando anteriormente no qual a otimização global é truncada, assumindo como argumentos apenas as posições final e inicial (SANTOS et al., 2015). As posições de todos os instantes intermediários são definidas por interpolação.

Ao invés de encarar a movimento ao longo do tempo como uma função contínua, é possível aproximá-la por posições discretas no tempo. Isso possibilita a aplicação da programação dinâmica, abordando a posição a cada instante como uma decisão discreta. Graças ao Princípio de Otimalidade de Bellman (BELLMAN, 1956), uma otimização que recebe como argumento as ações de controle para todos os instantes discretizados pode ser abordada como sub-problemas à cada instante. Em acúmulo à isso, usando aproximações de Taylor a dinâmica do sistema e dos custos a serem minimizados, esses sub-problemas podem ser resolvidos com grande rapidez, dando origem à Programação Dinâmica Diferencial (MAYNE, 1966; SEN; YAKOWITZ, 1987), LQR iterativo (LI; TODOROV, 2004) e LQG iterativo (LI; TODOROV, 2004).

Além destas fontes que foram estudadas mais em detalhe, é importante ressaltar outros trabalhos que propuseram estratégias de resolução de redundância. Um método global foi implementado com sucesso para um manipulador serial em (AHUACTZIN; GUPTA, 1999). Usando otimização multi-critério e o algoritmo de busca NSGA-II (DEB et al., 2002), a resolução de redundância do manipulador RRRR foi estudado numericamente em (MARCOS; MACHADO; AZEVEDO-PERDICOÚLIS, 2012). Uma proposta local foi explorada em (KIM et al., 2012) para um braço humano. A proposta foi aplicada para controlar um exoesqueleto de 7 graus de liberdade.

2.4 Métricas de Desempenho

A definição de uma métrica de desempenho é indispensável em diversos campos da robótica. Entre eles, recebem especial atenção a síntese dimensional, planejamento de trajetória, resolução de redundância, seleção de atuadores e comparação entre diferentes manipuladores (NENCHEV, 1989; BOWLING; KHATIB, 2003; BOWLING; KHATIB, 2005; SHAYYA et al., 2014). Por essa razão, existem numerosos estudos já bem estabelecidos focados neste tema (ASADA, 1983; YOSHIKAWA, 1985; KHATIB; BURDICK, 1987; GRAETTINGER; KROGH, 1988; MA; ANGELES, 1990; KIM; DESA, 1993).

O Elipsoide de Inercia Generalizado (GIE) foi proposto em (ASADA, 1983). Estendendo o conceito de tensor de inércia de um único corpo rígido para um braço mecânico, o GIE retrata a dinâmica do manipulador sobre suas diferentes direções para uma dada posição. Analisando se a matriz de inércia é mal condicionada, o Índice de Condicionamento Dinâmico (DCI) (MA; ANGELES, 1990) mede o acoplamento entre os torques de cada atuador. A manipulabilidade dinâmica introduzida em (YOSHIKAWA, 1985) é proporcional ao volume do elipsoide de manipulabilidade dinâmica, que é o conjunto de acelerações factíveis do *end-effector* correspondentes a torques limitados por uma esfera de raio unitário. Todos estes métodos se focam na isotropia do manipulador, mas não consideram limites reais dos atuadores.

Contudo, conceitos discutidos em (KIM; DESA, 1993; GRAETTINGER; KROGH, 1988; KHATIB; BURDICK, 1987) levam em consideração restrições realistas dos atuadores (torques, acelerações e velocidades). Mais especificamente, o desempenho dos motores é mapeado para a área de trabalho através do modelo dinâmico. Com isso, acelerações realizáveis do *end-effector* podem ser calculadas.

Seguindo este conceito, trabalhos mais recentes (SHAYYA, 2015; BOWLING; KHATIB, 2008; BOWLING; KHATIB, 2005) atingiram resultados precisos com valiosos significados físicos. Bowling e Khatib (BOWLING; KHATIB, 2005) definiram as Equações de Capacidade Dinâmica (DCE), que mapeiam os limites de torques para as capacidade do *end-effector* (carregamentos toleráveis, velocidade e acelerações realizáveis). Uma aplicação simplificada é ilustrada na Fig. 8. Para isso, a pior direção relativa a cada tipo de capacidade

Figura 8: Relação entre limitantes dos atuadores com os limites no espaço de trabalho

Fonte: (BOWLING; KHATIB, 2005)

é obtida. Ou seja, a direção na qual uma restrição de torque é encontrada com menos amplitude. Consequentemente, sobre qualquer outra direção, um desempenho melhor ou igual pode ser garantida. Considerando todos os tipos de capacidade (carregamentos, velocidades e acelerações) uma *hyper-superfície* estabelece o limite de situações factíveis. Provando ser um conceito versátil, esse método foi aplicado para um manipulador com redundância cinemática em (BOWLING; KHATIB, 2008). De forma similar, Shayya (SHAYYA, 2015) propôs uma abordagem relativamente simples para o cálculo de um tipo específico de capacidade, mantendo os outros com um limite fixo. Finalmente, mapas dinâmicos sob a área de trabalho de um manipulador paralelo redundante são introduzidos em (FONTES; SILVA, 2016).

3 MODELO

O modelo do manipulador é indispensável para simulações, projeto de componentes, projeto do sistema de controle e análise de resultados. Este capítulo apresenta relações matemáticas entre variáveis (cinemáticas na seção 3.1 e dinâmicas na seção 3.2) de forma que o comportamento físico do protótipo seja previsto computacionalmente.

3.1 Modelo Cinemático

Métodos usuais propostos por (MERLET, 2012) foram aplicados para que as restrições cinemáticas fossem modeladas, de forma a ser obtido um instrumento capaz de relacionar matematicamente posições e velocidades dependentes entre si. Mais especificamente, relacionando-se dados cinemáticos do *end-effector* em função dos dados dos atuadores, é obtida a Cinemática Direta. Enquanto que o processo contrário recebe o nome de Cinemática Inversa.

As notações pertinentes ao modelo cinemático do 3PRRR são ilustradas na Fig.9. Os comprimentos dos elos AB e BC são respectivamente l_2 e l_3 . Ao longo da presente seção, o subíndice i se refere sempre ao número da cadeia cinemática, tal que $i = 1, 2, 3$. Existem atuadores rotativos em cada ponto A_i responsáveis pelo posicionamento de θ_i e atuadores lineares responsáveis pelo posicionamento de ζ_i . Os ângulos γ_i representam a orientação de cada atuador linear. Juntas B_i e C_i são rotativas e passivas. A dimensão h é a distância entre o centro do *end-effector* e um ponto C_i . A distância entre o centro do manipulador (a origem O) e os centros de cada atuador linear é dado por a . O ponto O é definido como o ponto de simetria do mecanismo. As orientações dos elos B_iC_i são dadas β_i . A posição cartesiana e orientação do *end-effector* são x, y e α .

3.1.1 Cinemática Inversa

O modelo cinemático deve correlacionar as posições dos atuadores com as posições do *end-effector*, respectivamente

$$\begin{aligned}\Theta &= [\theta_1, \zeta_1, \theta_2, \zeta_2, \theta_3, \zeta_3]^T \\ \mathbf{X} &= [x, y, \alpha]^T\end{aligned}\tag{3.1}$$

Primeiramente, será abordado o problema de definir Θ para um $\mathbf{X}$ dado (Cinemática Inversa). Como trata-se de um manipulador com redundância cinemática, o mecanismo possui maior mobilidade (6) do que o *end-effector*. Como consequência, para um dado conjunto (x, y, α) a relação matemática com as posições dos atuadores $(\zeta_1, \zeta_2, \zeta_3, \theta_1, \theta_2, \theta_3)$ é indeterminada. Por isso, seguindo a ideia de que os atuadores redundantes são os

Figura 9: Representação esquemática do 3(P)RRR

Fonte: própria

lineares, parte-se do pressuposto que $(\zeta_1, \zeta_2, \zeta_3)$ são determinados previamente. Nas seções futuras isso será discutido mais a fundo. Portanto, deseja-se determinar $(\theta_1, \theta_2, \theta_3)$ para $(\zeta_1, \zeta_2, \zeta_3, x, y, \alpha)$ conhecidos.

Para isso, deve-se eliminar as variáveis β_i usando a equação

$$\|\mathbf{r}_{OC_i} - \mathbf{r}_{OB_i}\|^2 = \left\| \begin{bmatrix} \rho_{xi} - l_2 \cos \theta_i \\ \rho_{yi} - l_2 \sin \theta_i \end{bmatrix} \right\|_2^2 = l_3^2 \quad (3.2)$$

onde $\mathbf{r}_{OC_i}$ e $\mathbf{r}_{OB_i}$ são vetores posição dos pontos C_i e B_i , respectivamente. Ainda,

$$\begin{bmatrix} \rho_{xi} \\ \rho_{yi} \end{bmatrix} = \begin{bmatrix} x \\ y \end{bmatrix} + h \begin{bmatrix} \cos(\alpha + \gamma_i + \pi/2) \\ \sin(\alpha + \gamma_i + \pi/2) \end{bmatrix} - \zeta_i \begin{bmatrix} \cos \gamma_i \\ \sin \gamma_i \end{bmatrix} - a \begin{bmatrix} \cos(\gamma_i \pm \pi/2) \\ \sin(\gamma_i \pm \pi/2) \end{bmatrix} \quad (3.3)$$

Os vetores $\mathbf{r}_{OA_i}$ são escritos como $a [\cos(\gamma_i + \pi/2) \ \sin(\gamma_i + \pi/2)]^T$, já que possuem comprimento a e direção perpendicular às guias lineares γ_i . O mesmo raciocínio é empregado para o vetor que é a segunda parcela do lado direito de (3.3).

Expandindo a norma em (3.2)

$$\underbrace{-2.l_2.\rho_{yi}}_{e_{1i}} \sin \theta_i - \underbrace{2.l_2.\rho_{xi}}_{e_{2i}} \cos \theta_i + \underbrace{\rho_{xi}^2 + \rho_{yi}^2 + l_2^2 - l_3^2}_{e_{3i}} = 0 \quad (3.4)$$

A equação transcendental (3.4) pode ser resolvida analiticamente para θ_i com a *substituição de Weierstrass* (MERLET, 2012) transformando funções trigonométricas de θ_i por funções de $\tan(\theta_i/2)$.

$$\theta_i = 2 \tan^{-1} \left(\frac{-e_{i1} \pm \sqrt{e_{i1}^2 + e_{i2}^2 - e_{i3}^2}}{e_{i3} - e_{i2}} \right) \quad (3.5)$$

Os valores de θ_i podem variar de acordo com o sinal da raiz quadrada na Eq. 3.5. Realmente, existem duas possíveis soluções para a cadeia cinemática, como discutido em (LIU; QIU; ZHANG, 2017). Uma vez obtidas as posições θ_i , é interessante conhecer os valores de β_i . Isso pode ser feito por

$$\beta_i = \tan^{-1} \left(\frac{\rho_{yi} - l_2 \sin \theta_i}{\rho_{xi} - l_2 \cos \theta_i} \right) \quad (3.6)$$

3.1.2 Cinemática Direta

As informações de posição disponíveis de um robô se limitam, em geral, às leituras dos *encoders*. Sendo assim, conhecendo-se Θ , é desejável um instrumento capaz de inferir a posição do *end-effector* $\mathbf{X}$ (Cinemática Direta). A complexidade matemática de métodos analíticos capazes disso é proibitiva. Por isso, essa relação é obtida numericamente. Uma opção clara é a solução da equação 3.2. Mais especificamente, conhecendo-se ζ_i e θ_i , a equação (3.7) é resolvida numericamente, tendo x , y e α como incógnitas.

$$\left\| -h \begin{bmatrix} \cos(\gamma_i \pm \pi/2) \\ \sin(\gamma_i \pm \pi/2) \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} + a \begin{bmatrix} \cos(\gamma_i \pm \pi/2) \\ \sin(\gamma_i \pm \pi/2) \end{bmatrix} + \dots \right. \quad (3.7)$$

$$\left. \dots + \zeta_i \begin{bmatrix} \cos \gamma_i \\ \sin \gamma_i \end{bmatrix} + l_2 \begin{bmatrix} \cos \theta_i \\ \sin \theta_i \end{bmatrix} \right\| - l_3 = 0$$

3.1.3 Matriz Jacobiana

Será relevante na análise do manipulador a sua matriz jacobiana. Esta relaciona velocidades do *end-effector* com as dos atuadores.

Pode-se determiná-la através da equação $\|\mathbf{r}_{\mathbf{BC}}\|^2 = r_{BC_x}^2 + r_{BC_y}^2 = l_3^2$. Derivando, obtêm-se $r_{BC_x} \dot{r}_{BC_x} + r_{BC_y} \dot{r}_{BC_y} = 0$. Componentes de posição r_{BC_x} e r_{BC_y} são substituídos

por $l_3 \cos \beta_i$ e $l_3 \sin \beta_i$, respectivamente. Componentes de velocidade $\dot{r}_{BC_x}$ e $\dot{r}_{BC_y}$ são obtidos derivando (3.4). Assim, cada cadeia cinemática i resulta em uma equação:

$$\begin{aligned} \dot{x} \overbrace{[l_3 \cos \beta_i]}^{a_{i1}} + \dot{y} \overbrace{[l_3 \sin \beta_i]}^{a_{i2}} + \dot{\alpha} \overbrace{[l_3 h \sin(\beta_i - \gamma_i - \alpha \pm \pi/2)]}^{a_{i3}} = \\ = \underbrace{\dot{\theta}_i [l_2 l_3 \sin(\beta_i - \theta_i)]}_{b_{i(2i-1)}} + \underbrace{\dot{\zeta}_i [l_3 \cos(\beta_i - \gamma_i)]}_{b_{i(2i)}} \end{aligned} \quad (3.8)$$

Seguindo os índices da Eq. 3.8, matrizes $\mathbf{A} = (a_{i,j})$ e $\mathbf{B} = (b_{i,j})$ podem ser construídas. Dessa forma, assumindo $\mathbf{A}$ inversível, $\mathbf{X} = [x, y, \alpha]^T$ e $\mathbf{\Theta} = [\theta_1, \zeta_1, \theta_2, \zeta_2, \theta_3, \zeta_3]^T$ têm suas velocidades correlacionadas por

$$\begin{aligned} \mathbf{A}_{3 \times 3} \dot{\mathbf{X}} &= \mathbf{B}_{3 \times 6} \dot{\mathbf{\Theta}} \\ \dot{\mathbf{X}} &= \mathbf{A}^{-1} \mathbf{B} \dot{\mathbf{\Theta}} \\ \dot{\mathbf{X}} &= \mathbf{J}_{3 \times 6} \dot{\mathbf{\Theta}} \end{aligned} \quad (3.9)$$

Assim, a matriz jacobiana $\mathbf{J}$ é calculada por $\mathbf{J} = \mathbf{A}^{-1} \mathbf{B}$. É interessante observar neste ponto, que caso $\mathbf{A}$ seja singular, uma velocidade $\dot{\mathbf{X}} \neq 0$ pode existir mesmo que $\dot{\mathbf{\Theta}} = 0$, o que caracteriza perda de rigidez do manipulador. Este é um tipo de singularidade apresentada para este manipulador.

3.2 Modelo Dinâmico - Formalismo Euler-Lagrange

Após o trato cinemático, simulações completas exigem modelagem dinâmica do sistema. Esta seção apresenta os instrumentos necessários para isso, equacionando o problema analiticamente e propondo métodos numéricos capazes de resolvê-lo. O instrumento descrito nesta seção resolve a Dinâmica Direta (prevê a movimentação a partir de torques e forças dadas) e Inversa (calcula as forças e torques necessários para uma movimentação específica).

O equacionamento analítico foi estabelecido usando o formalismo de Euler-Lagrange com multiplicador de Lagrange para contabilização das restrições cinemáticas. Este método é descrito detalhadamente em referências de Mecânica Clássica tais como (GOLDSTEIN, 1965; LANCZOS, 1970). Para a integração numérica da Dinâmica Direta é necessária a solução de Equações Algébrico-Diferenciais. Este problema é abordado em (ASCHER; PETZOLD, 1998; BRENAN; CAMPBELL; PETZOLD, 1996; HAIRER; WANNER, 1996; RABIER; RHEINBOLDT, 2000). Foi implementado um método proposto por (BRENAN; CAMPBELL; PETZOLD, 1996) utilizando *Backward Differential Formula* (BDF).

3.2.1 Matriz de massa

Para uma cadeia cinemática i ($i = 1, 2, 3$), os parâmetros de posição são $\Theta_i^* = [\zeta_i \ \theta_i \ \beta_i]^T$, de acordo com a Fig.9. O indicativo $*$ se refere ao fato do vetor receber também a posição da junta passiva β_i . Cada cadeia tem três corpos, que podem ser identificados por um índice j da seguinte forma: ($j = 1$) o corpo que pode ser considerado pontual em A_i , ($j = 2$) o elo $\overline{A_i B_i}$, e ($j = 3$) o elo $\overline{B_i C_i}$.

É definido um vetor ν_{ij} relativo ao corpo j da cadeia i que apresenta as velocidades translacionais e rotacionais

$$\nu_{ij} = \begin{bmatrix} v_{xij} \\ v_{yij} \\ \omega_{ij} \end{bmatrix} \quad (3.10)$$

onde v_{xij} e v_{yij} são as velocidades cartesianas do centro de massa em x e y respectivamente e ω_{ij} é a velocidade angular.

É necessário definir matrizes que relacionem $\dot{\Theta}_i^*$ com ν_{ij} . Essas matrizes são chamadas de matrizes de velocidades parciais denotadas por $\mathbf{K}_{ij}$.

$$\begin{aligned} \nu_{i1} &= \underbrace{\begin{pmatrix} \cos \gamma_i & 0 & 0 \\ \sin \gamma_i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}}_{\mathbf{K}_{i1}} \dot{\Theta}_i^* \\ \nu_{i2} &= \underbrace{\begin{pmatrix} \cos \gamma_i & -s_2 \sin \theta_i & 0 \\ \sin \gamma_i & s_2 \cos \theta_i & 0 \\ 0 & 1 & 0 \end{pmatrix}}_{\mathbf{K}_{i2}} \dot{\Theta}_i^* \\ \nu_{i3} &= \underbrace{\begin{pmatrix} \cos \gamma_i & -l_2 \sin \theta_i & -s_3 \sin \beta_i \\ \sin \gamma_i & l_2 \cos \theta_i & s_3 \cos \beta_i \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{K}_{i3}} \dot{\Theta}_i^* \end{aligned} \quad (3.11)$$

onde s_2 é a distância de A_i até o centro de massa do elo $\overline{A_i B_i}$ e s_3 é a distância de B_i até o centro de massa do elo $\overline{B_i C_i}$.

Denotando μ_j a matriz com os parâmetros de inércia de um corpo j

$$\mu_j = \begin{pmatrix} m_j & 0 & 0 \\ 0 & m_j & 0 \\ 0 & 0 & I_j \end{pmatrix} \quad (3.12)$$

a soma da energia cinética dos corpos da cadeia i pode ser dada por

$$\begin{aligned} T_i &= \frac{1}{2} \sum_{j=1}^3 \boldsymbol{\nu}_{ij}^T \mu_j \boldsymbol{\nu}_{ij} \\ &= \frac{1}{2} \dot{\boldsymbol{\Theta}}_i^{*T} \underbrace{\left(\sum_{j=1}^3 \mathbf{K}_{ij}^T \mu_j \mathbf{K}_{ij} \right)}_{\mathbf{M}_i} \dot{\boldsymbol{\Theta}}_i^* \end{aligned} \quad (3.13)$$

Resta incluir a contribuição do atuador-final. Sendo $\mathbf{X} = [x, y, \alpha]^T$, essa parcela é

$$T_N = \frac{1}{2} \dot{\mathbf{X}}^T \underbrace{\begin{pmatrix} m_N & 0 & 0 \\ 0 & m_N & 0 \\ 0 & 0 & I_N \end{pmatrix}}_{\mathbf{M}_N} \dot{\mathbf{X}} \quad (3.14)$$

Para assessor a energia cinética total T do sistema, define-se o vetor

$$\mathbf{q} = [\boldsymbol{\Theta}_1^{*T} \ \boldsymbol{\Theta}_2^{*T} \ \boldsymbol{\Theta}_3^{*T} \ \mathbf{X}^T]^T \quad (3.15)$$

de forma que

$$T = \frac{1}{2} \dot{\mathbf{q}}^T \mathbf{M} \dot{\mathbf{q}} \quad (3.16)$$

As submatrizes definidas em (3.13) e (3.14) dispostas na diagonal de $\mathbf{M}$ conforme abaixo formam a matriz de massa

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 & & & \\ & \mathbf{M}_2 & & \\ & & \mathbf{M}_3 & \\ & & & \mathbf{M}_N \end{pmatrix} \quad (3.17)$$

onde cada submatriz de massa possui dimensões 3x3, formando a matriz de massa final $\mathbf{M}_{12 \times 12}$.

3.2.2 Equacionamento Dinâmico

O vetor $\mathbf{q}$ não representa um conjunto de variáveis independentes (coordenadas generalizadas). Portanto, o método de Multiplicadores de Lagrange será usado. Como o mecanismo possui 6 graus de liberdade enquanto $\mathbf{q}$ apresenta 12 componentes, 6 equações de restrições e 6 multiplicadores de Lagrange serão necessários.

É necessário definir 6 equações de restrição cinemática. Cada cadeia cinemática pode impor duas equações, uma em x e outra em y , impondo que sua cadeia seja fechada. Sendo assim, para cada cadeia, apresenta-se o seguinte conjunto de equações

$$\begin{aligned} \begin{bmatrix} g_{2i-1} \\ g_{2i} \end{bmatrix} = & a \begin{bmatrix} \cos(\gamma_i \pm \pi/2) \\ \sin(\gamma_i \pm \pi/2) \end{bmatrix} + \zeta_i \begin{bmatrix} \cos \gamma_i \\ \sin \gamma_i \end{bmatrix} + l_2 \begin{bmatrix} \cos \theta_i \\ \sin \theta_i \end{bmatrix} + \dots \\ & \dots + l_3 \begin{bmatrix} \cos \beta_i \\ \sin \beta_i \end{bmatrix} - h \begin{bmatrix} \cos(\alpha + \gamma_i \pm \pi/2) \\ \sin(\alpha + \gamma_i \pm \pi/2) \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} = 0 \end{aligned} \quad i = 1, 2, 3 \quad (3.18)$$

de forma que a equação vetorial $\mathbf{g} = 0$ resulta em 6 equações algébricas de restrição.

Como neste sistema não há variação de energia potencial, o lagrangeano é $L = T$. Calculando derivadas parciais e temporais das definições em (3.16) e (3.18), a aplicação da Equação de Lagrange com Multiplicadores $\boldsymbol{\lambda} = [\lambda_1 \ \lambda_2 \ \dots \ \lambda_6]^T$ chega a

$$\mathbf{M}_{12 \times 12} \ddot{\mathbf{q}}_{12 \times 1} + \mathbf{b}_{12 \times 1}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}_{12 \times 6}^T \boldsymbol{\lambda}_{6 \times 1} = \boldsymbol{\tau}_{12 \times 1} \quad (3.19)$$

$$\mathbf{g}_{6 \times 1}(\mathbf{q}) = 0 \quad (3.20)$$

onde $\mathbf{G}$ é a matriz jacobiana da função vetorial $\mathbf{g}$ com todas as suas derivadas parciais. O vetor $\boldsymbol{\tau} = [\tau_1 \ \tau_2 \ \dots \ \tau_{12}]^T$ tem componentes τ_k representantes de torques e forças não conservativas agentes em q_k . É importante observar que alguns componentes de $\boldsymbol{\tau}$ representam efetivamente torques e forças dos atuadores do mecanismo, enquanto que os componentes restantes podem introduzir distúrbios ou atrito nas demais coordenadas generalizadas. Por fim, $\mathbf{b}$ é calculado por

$$\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) = \dot{\mathbf{M}} \dot{\mathbf{q}} - \begin{bmatrix} \frac{\partial T}{\partial q_1} \\ \frac{\partial T}{\partial q_2} \\ \vdots \\ \frac{\partial T}{\partial q_{12}} \end{bmatrix} \quad (3.21)$$

O equacionamento descrito pode ser facilmente implementado em um *software* de manipulação simbólica, resultando em todos os vetores e matrizes apresentados.

3.2.3 Dinâmica Inversa

Nesta seção, as informações relativas à movimentação são assumidas como já definidas ($\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$ conhecidos). São calculados as forças e torques necessários dos atuadores para gerar tal movimentação. Existem 6 atuadores no mecanismo: 3 lineares em ζ_i e 3 rotativos

em θ_i . Analisando (3.19), pode-se perceber que as incógnitas são $\boldsymbol{\lambda}$, τ_1 , τ_2 , τ_4 , τ_5 , τ_7 e τ_8 (os τ_k relativos a ζ_i ou θ_i). Os multiplicadores $\boldsymbol{\lambda}$ estão presentes em todas as 12 equações, enquanto que os 6 τ_k apontados estão presentes apenas nas suas respectivas equações, restando assim, 6 equações (linhas 3, 6, 9, 10, 11 e 12) em que $\boldsymbol{\lambda}$ é a única incógnita. Desta forma, este vetor pode ser facilmente determinado simplesmente isolando-o matricialmente nestas linhas. Em outras palavras, são abordadas primeiramente apenas as linhas da equação 3.19 que possuem componentes de $\boldsymbol{\tau}$ relativos aos atuadores do mecanismo. Uma vez feito isso, os componentes desconhecidos de $\boldsymbol{\tau}$ são obtidos diretamente.

De forma equacionada, da (3.19) apenas as linhas $r = 3, 6, 9, 10, 11$ e 12 são extraídas ($\tau_r = 0$ se não houverem forças não conservativas nestes componentes) e assumindo $\mathbf{G}_r$ inversível

$$\begin{aligned} \mathbf{M}_r \ddot{\mathbf{q}} + \mathbf{b}_r + \mathbf{G}_r^T \boldsymbol{\lambda} &= 0 \Rightarrow \\ \Rightarrow \boldsymbol{\lambda} &= -(\mathbf{G}_r^T)^{-1} (\mathbf{M}_r \ddot{\mathbf{q}} + \mathbf{b}_r) \end{aligned} \quad (3.22)$$

A substituição de (3.22) em (3.19) resulta diretamente nas forças e torques exigidos.

Forças não conservativas conhecidas (como atrito e distúrbios) agentes em β_i , x , y e α podem ser facilmente introduzidas em (3.22) fazendo $\tau_r \neq 0$. O caso mais simples em que as únicas forças não conservativas têm origem nos atuadores foi exposto para facilitar a compreensão.

3.2.4 Dinâmica Direta

Tendo $\boldsymbol{\tau}$, deseja-se calcular a movimentação resultante $\mathbf{q}(t)$. Isto se dá pela integração numérica do sistema (3.19) e (3.20). Este é um Sistema de Equações Algébrico-Diferenciais de ordem 2 e índice 3 (BRENAN; CAMPBELL; PETZOLD, 1996). Primeiramente, é necessário ter a ordem reduzida para 1

$$\begin{aligned} \mathbf{u} - \dot{\mathbf{q}} &= 0 \\ \mathbf{M} \dot{\mathbf{u}} + \mathbf{b}(\mathbf{q}, \mathbf{u}) + \mathbf{G}^T \boldsymbol{\lambda} - \boldsymbol{\tau} &= 0 \\ \mathbf{g}(\mathbf{q}) &= 0 \end{aligned} \quad (3.23)$$

Sendo (3.23) de índice 3, típico de sistemas mecânicos, é dificultada a aplicação de métodos numéricos usuais para sistemas algébrico-diferenciais (adaptações de métodos usados para sistemas diferenciais rígidos). Para contornar este problema, foi implementado o *Backward Differential Formula* (BDF). Este aproxima $\dot{f}_n$ a derivada em um dado instante de uma função discretizada interpolando-a com os instantes anteriores. Este dado é obtido

por

$$\dot{f}_n = \sum_{m=0}^k \frac{a_m f_{n-i}}{\Delta t} \quad (3.24)$$

Em que k é o o número de pontos anteriores interpolados e a_m são coeficientes constantes previamente conhecidos em função de k .

No caso da integração de (3.23), ao se definir um valor para as incógnitas $(\mathbf{q}_n, \mathbf{u}_n, \boldsymbol{\lambda}_n)$ em um instante n , por (3.24) são obtidas suas derivadas $(\dot{\mathbf{q}}_n, \dot{\mathbf{u}}_n, \dot{\boldsymbol{\lambda}}_n)$. Assim, é possível substituir o conjunto $(\mathbf{q}_n, \mathbf{u}_n, \boldsymbol{\lambda}_n, \dot{\mathbf{q}}_n, \dot{\mathbf{u}}_n)$ em (3.23) e aferir sua validade. Com este procedimento, (3.23) se torna um sistema de equações algébricas em $(\mathbf{q}_n, \mathbf{u}_n, \boldsymbol{\lambda}_n)$. É possível então, a cada passo, resolvê-lo numericamente com um método para sistemas de equações algébricas não lineares.

Esta abordagem implica que, para um dado instante t_n , sejam conhecidos os valores de todas as variáveis de t_{n-k} até t_{n-1} . No instante inicial t_0 , evidentemente, não se conhece valores anteriores. Mas partindo do pressuposto que todas as trajetórias estudadas partem do repouso, é consistente assumir que nos k instantes anteriores ao inicial o sistema permaneceu em condição idêntica a t_0 .

Neste sistema, o conjunto $(\mathbf{q}_n, \mathbf{u}_n, \boldsymbol{\lambda}_n)$ possui 30 componentes. Um método tradicional para solução de sistema de equações não lineares, tal como Newton-Raphson, necessita da matriz jacobiana do sistema em relação a suas incógnitas. Para sistemas de menor dimensão, é aceitável o cálculo das derivadas parciais numericamente a cada iteração. Mas para este seriam necessários 30 cálculos adicionais para cada iteração, o que indica a necessidade do cálculo analítico da jacobiana. Esta matriz pode ser obtida por derivadas parciais diretas do sistema (3.23).

Como o intervalo Δt deve ser pequeno, a solução de (3.23) como sistema algébrico a cada instante exige poucas iterações, já que as funções $(\mathbf{q}(t), \mathbf{u}(t), \boldsymbol{\lambda}(t))$ contínuas têm valores próximos para instantes vizinhos.

Anteriormente, foi desenvolvido outro modelo dinâmico para o sistema, como descrito em (SANTOS et al., 2015), baseado no Princípio dos Trabalhos Virtuais. Apesar de mais enxuto, abordando apenas os 6 graus de liberdade necessários, apresenta duas grandes limitações: dificulta a Dinâmica Direta e a simulação de qualquer força não conservativa atuante fora de ζ_i e θ_i . Desta forma, o método detalhado nesta seção é mais conveniente. Ainda assim, tal abordagem entrega dados importantes para o estabelecimento de métricas de desempenho (vide capítulo 4). Por isso ele será introduzido brevemente em seguida.

3.3 Modelo Dinâmico - Princípio dos Trabalhos Virtuais

Um modelo que correlacione diretamente os torques dos atuadores com as acelerações do *end-effector* será pertinente para a definição de medidas de desempenho dinâmico na seção 4.1. Sendo assim, a presente seção busca um modelo sem variáveis de estado redundantes.

Utilizando notação similar à apresentada até agora, as matrizes de velocidades parciais reduzidas $\mathbf{K}_{ij}^R$ são definidas tal que

$$\boldsymbol{\nu}_{ij} = \mathbf{K}_{ij}^R \dot{\boldsymbol{\Theta}} \quad (3.25)$$

onde $\boldsymbol{\Theta}$ é definido pela Eq. 3.1. O trabalho virtual sob um dado corpo é

$$\delta w = \delta \boldsymbol{\Theta}^T \left[\left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \mathbf{K}_{ij}^R \ddot{\boldsymbol{\Theta}} + \left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \dot{\mathbf{K}}_{ij}^R \dot{\boldsymbol{\Theta}} \right] \quad (3.26)$$

Observando que a matriz de velocidades parciais do *end-effector* é a matriz jacobiana $\mathbf{J}$ e somando os trabalhos virtuais efetuados sob todos os corpos rígidos dos sistema

$$\begin{aligned} \delta \boldsymbol{\Theta}^T \boldsymbol{\tau}^R = \delta \boldsymbol{\Theta}^T & \left\{ \left(\sum_{i,j=1}^3 \left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \mathbf{K}_{ij}^R + \mathbf{J}^T \boldsymbol{\mu}_N \mathbf{J} \right) \ddot{\boldsymbol{\Theta}} + \dots \right. \\ & \left. \dots + \left(\sum_{i,j=1}^3 \left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \dot{\mathbf{K}}_{ij}^R + \mathbf{J}^T \boldsymbol{\mu}_N \dot{\mathbf{J}} \right) \dot{\boldsymbol{\Theta}} \right\} \end{aligned} \quad (3.27)$$

onde $\boldsymbol{\tau}^R$ é o vetor de torques dos atuadores. A eq. 3.27 deve ser válida para qualquer deslocamento virtual $\delta \boldsymbol{\Theta}$. Por consequência, os torques podem ser dados por

$$\boldsymbol{\tau}^R = \mathbf{M}^R \ddot{\boldsymbol{\Theta}} + \mathbf{b}^R(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) \quad (3.28)$$

com a matriz de massa $\mathbf{M}^R$ e vetor de forças centrífugas e de Coriolis $\mathbf{b}^R(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})$ calculadas por

$$\begin{aligned} \mathbf{M}^R &= \sum_{i,j=1}^3 \left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \mathbf{K}_{ij}^R + \mathbf{J}^T \boldsymbol{\mu}_N \mathbf{J} \\ \mathbf{b}^R(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) &= \left(\sum_{i,j=1}^3 \left(\mathbf{K}_{ij}^R \right)^T \boldsymbol{\mu}_j \dot{\mathbf{K}}_{ij}^R + \mathbf{J}^T \boldsymbol{\mu}_N \dot{\mathbf{J}} \right) \dot{\boldsymbol{\Theta}} \end{aligned} \quad (3.29)$$

A modelagem dinâmica pela eq. 3.28 possui dificuldade a integração necessária para a modelo direto e exige mais esforço computacional para a execução do modelo inverso. Entretanto, será visto no capítulo 4 que a matriz de massa pode trazer importantes informações a respeito do desempenho dinâmico do manipulador para uma dada posição.

4 RESOLUÇÃO DE REDUNDÂNCIA

Como consequência da redundância cinemática, para uma dada posição do *end-effector*, as posições dos atuadores é indeterminada, como ilustrado na Fig. 10. Com isso em mente, e seguindo a notação introduzida na Eq. 3.1, este capítulo se foca no seguinte problema: dado um $\mathbf{X}(t)$, defina um $\Theta(t)$ conveniente. Tal problema é chamado Resolução de Redundância.

Figura 10: Existem infinitas posições possíveis dos atuadores para uma única posição do *end-effector*

Fonte: própria

Métricas de performance são essenciais neste processo de resolução de redundância. Evidentemente, a definição de um movimento $\Theta(t)$ ótimo demanda a definição de uma função objetivo avaliadora do desempenho desenvolvido. Este ponto é abordado na seção 4.1, discutindo métodos propostos para o manipulador 3PRRR.

Tendo métricas de desempenho definidas, a resolução de redundância pode ser discutida diretamente. Diferentes métodos foram considerados e estudados. Os métodos de Gradientes Projetados (*Projected Gradients* - PG) (SICILIANO, 1990) e dos Gradientes Reduzidos (*Reduced Gradients* - RG) (LUCA; ORIOLO, 1991) são estratégias que resolvem a cinemática inversa no nível de velocidade. Em outras palavras, as velocidades ótimas dos atuadores são definidas para uma dada posição. Por consequência, possuem âmbito local já propõem otimização de cada instante isoladamente. Esses conceitos são discutidos na seção 4.2. A possibilidade de resolução no nível de aceleração também é considerada.

Em geral, principalmente em aplicações industriais, a trajetória $\mathbf{X}(t)$ é conhecida por completo previamente, antes do início da tarefa. Sendo assim, a aplicação de um método local desconsidera esta valiosa informação. Uma estratégia de otimização global é

portanto desejável, levando em consideração a trajetória como um todo. Buscando isso, uma abordagem inovadora é apresentada na seção 4.3. Ela propõe a obtenção de um $\Theta_r(t)$ com posições ideais para cada instante t isoladamente, e uma posterior otimização global que obtém o movimento ótimo final buscando se manter próximo de $\Theta_r(t)$ mas reduzindo também a incidência de altas acelerações. Este é o principal resultado deste projeto de mestrado.

Também será descrito um método global truncado, no qual o problema de otimização de Θ para cada instante $t \in [t_i \ t_f]$ é simplificado para uma otimização onde apenas $\Theta(t_i)$ e $\Theta(t_f)$ são argumentos. Dessa forma, $\Theta(t)$ para $t \neq t_i, \ t_f$ é definido por interpolação dos valores extremos.

Sendo assim, as principais seções do presente capítulo se organizam como apresentado a seguir:

- **4.1 Métricas de desempenho** - Descrição e dedução de todas as métricas utilizadas para medição de desempenho para uma dada posição do manipulador;
- **4.2 Métodos Locais** - Aborda os métodos de resolução de redundância em que a cinemática inversa é definida para cada instante isoladamente. As estratégias apresentadas são as seguintes:
 - **4.2.1 Gradientes Projetados**
 - **4.2.2 Gradientes Reduzidos**
 - **4.2.3 Gradientes reduzidos nos nível das acelerações**
- **4.3 Métodos Globais** - Este métodos otimizam a movimentação considerando a tarefa como um todo. O presente capítulo aborda as seguintes estratégias:
 - **4.3.1 Otimização Global por Programação Dinâmica**
 - **4.3.2 Otimização Truncada**

4.1 Métricas de Desempenho

A avaliação de desempenho é indispensável em qualquer processo de decisão na robótica. Como exemplos se destacam o projeto mecânico, planejamento de trajetória, a comparação entre manipuladores e, finalmente, a resolução de redundância. Consequentemente, esse tema já possui várias pesquisas bem estabelecidas (ASADA, 1983; YOSHIKAWA, 1985; KHATIB; BURDICK, 1987; GRAETTINGER; KROGH, 1988; MA; ANGELES, 1990; KIM; DESA, 1993).

Entretanto, a obtenção de uma métrica pertinente é fortemente dependente da sua aplicação. Como o robô estudado possui espaço de trabalho altamente heterogêneo e

mobilidade tanto angular quanto linear, trata-se de um caso merecedor de atenção especial. Diversas métricas foram consideradas e aplicadas na resolução de redundância. A presente seção detalha as de maior relevância e que obtiveram melhor resultados. Somando-se ponderadamente múltiplos métodos, estes podem ser usados a fim de avaliar o desempenho sob diversas perspectivas simultaneamente. Com isso em mente, cada métrica recebe um sub-índice, facilitando sua indicação quando apresentada na soma ponderada necessária para a obtenção da métrica final.

Número de condição da matriz $\mathbf{A}$ normalizada

Uma das principais desvantagens apresentadas pelo 3PRRR é a presença de singularidades de natureza paralela, como ilustrado na Fig. 4. Nestes pontos, o *end-effector* possui mobilidade mesmo com os atuadores travados. Ou seja, independente do limite de torque dos atuadores, o mecanismo perde completamente a rigidez. Estas posições são inaceitáveis e estão presentes no interior da área de trabalho.

Estudando a Eq. 3.9, este fenômeno é consequência de uma matriz $\mathbf{A}$ singular, de forma que um $\dot{\mathbf{X}}$ não nulo é possível mesmo para um $\dot{\mathbf{\Theta}} = \mathbf{0}$. Isto sugere que o número de condicionamento $\kappa(\mathbf{A})$ pode ser uma boa indicação de proximidade a singularidades deste tipo. Porém, esta matriz não possui consistência de unidade, uma vez que possui relação com deslocamentos lineares (colunas 1 e 2) e angulares (coluna 3). Para normalizá-la, a terceira coluna é dividida pelo comprimento característico do manipulador $L_c = \sqrt{2}h$, como proposto em (ALBA-GOMEZ; WENGER; PAMANES, 2005). Logo:

$$\mathbf{A}' = \begin{bmatrix} a_{11} & a_{12} & a_{13}/L_c \\ a_{21} & a_{22} & a_{23}/L_c \\ a_{31} & a_{32} & a_{33}/L_c \end{bmatrix} \quad (4.1)$$

Nas seções 4.2 e 4.3 será esclarecido que as métricas serão maximizadas. Desta forma, é pertinente utilizar o negativo do número de condicionamento. Assim, a primeira métrica é definida portanto como

$$\phi_1 = -\kappa(\mathbf{A}') = -\frac{\sigma_{\max}(\mathbf{A}')}{\sigma_{\min}(\mathbf{A}')} \quad (4.2)$$

Distância entre A_i e C_i

Como descrito na modelagem cinemática presente na seção 3.1, a matriz $\mathbf{A}$ depende apenas das posições do *end-effector* e dos elos $\overline{B_iC_i}$. Evitando a incidência de $\mathbf{A}$ mal condicionada, singularidades paralelas são contornadas. Contudo, a ocorrência de elos $\overline{A_iB_i}$ e $\overline{B_iC_i}$ se configura como singularidade serial. Nestes casos, o mecanismo não-redundante (desconsiderando a movimentação dos atuadores lineares) se apresenta no limite da sua área de trabalho. Estas posições devem ser evitadas.

Uma forma clara e direta evitar a incidência de singularidades seriais é a manipulação da distância entre pontos A_i e C_i . Tal distância pode ser calculado de acordo com a eq. 4.3

$$\begin{aligned} \|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| = & \left\| -h \begin{bmatrix} \cos(\alpha + \gamma_i + \pi/2) \\ \sin(\alpha + \gamma_i + \pi/2) \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} + \right. \\ & \left. + a \begin{bmatrix} \cos(\gamma_i + \pi/2) \\ \sin(\gamma_i + \pi/2) \end{bmatrix} + \zeta_i \begin{bmatrix} \cos(\gamma_i) \\ \sin(\gamma_i) \end{bmatrix} \right\| \end{aligned} \quad (4.3)$$

Primeiro, define-se uma distância ideal l_t entre estes dois pontos. Tipicamente, essa variável pode receber o valor $l_t = \sqrt{l_2^2 + l_3^2}$ de forma que os elos $\overline{A_iB_i}$ e $\overline{B_iC_i}$ estejam perpendiculares. Com isso, a métrica relacionada deve receber valor máximo quando $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| = l_t$ para todas as cadeias cinemáticas i .

Em acúmulo a isso, é possível considerar que posições nas quais $\overline{A_iB_i}$ e $\overline{B_iC_i}$ se apresentem colineares são inaceitáveis. Nestas situações, têm-se $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| = |l_2 - l_3|$ ou $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| = l_2 + l_3$.

Levando esse conjunto de informações em consideração, a seguinte métrica é proposta:

$$\begin{aligned} \phi_2 = & - \sum_{i=1}^3 \left[(l_t - \|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\|)^2 + \dots \right. \\ & \left. \dots + c'_2 \left(\frac{1}{\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| - |l_2 - l_3|} + \frac{1}{l_2 + l_3 - \|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\|} \right) \right] \end{aligned} \quad (4.4)$$

Vê-se que ϕ_2 é maximizado para $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| = l_t$. Além disso, $\phi_2 \rightarrow -\infty$, quando $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| \rightarrow |l_2 - l_3|^+$ ou $\|\mathbf{r}_{\mathbf{A}_i\mathbf{C}_i}\| \rightarrow (l_2 + l_3)^-$, indicando que tais posições não são cabíveis. Para isso, a constante c'_2 regula a proximidade aceita para pares de elos colineares.

Desempenho dinâmico

Foi esclarecido na seção 3.3 que o modelo dinâmico pode ser equacionado como

$$\boldsymbol{\tau} = \mathbf{M}\ddot{\boldsymbol{\Theta}} + \mathbf{b}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) \quad (4.5)$$

Para uma dada posição, os fenômenos dinâmicos relativos à parcela $\mathbf{M}\ddot{\boldsymbol{\Theta}}$ são de maior relevância que os relativos à parcela $\mathbf{b}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})$ (MERLET, 2012). Com isso, uma análise simplificada é proposta eliminando as forças de Coriolis e centrífugas, ou seja $\boldsymbol{\tau} \approx \mathbf{M}\ddot{\boldsymbol{\Theta}}$. Com isso, os torques dos motores se tornam uma função linear das acelerações

$\ddot{\Theta}$. Fazendo proveito dessa simplicidade, pode-se ver que as acelerações do *end-effector* são descritas por

$$\ddot{\mathbf{X}} = \mathbf{J} \mathbf{M}^{-1} \boldsymbol{\tau} \quad (4.6)$$

Assim, conhecendo os limites de torques, os limites de aceleração podem ser facilmente definidos. Entretanto, similar ao estudo da matriz $\mathbf{A}$, o vetor $\mathbf{X}$ possui grandezas lineares e angulares. Tendo em mente este problema, é aplicado o conceito proposto em (KIM; RYU, 2003). Pertinente a este caso, as acelerações $\ddot{\mathbf{X}}$ serão mapeadas para acelerações dos pontos C_i . Para isso, define-se um vetor contendo as posições cartesianas dos três pontos C_i

$$\mathbf{X}_e = [x_{C_1} \ y_{C_1} \ x_{C_2} \ y_{C_2} \ x_{C_3} \ y_{C_3}]^T \quad (4.7)$$

Novamente desconsiderando a influência dos termos relativos a $\dot{\mathbf{X}}$, as acelerações são correlacionadas linearmente

$$\ddot{\mathbf{X}}_e = \underbrace{\begin{bmatrix} 1 & 0 & -\cos(\alpha + \lambda_1) \\ 0 & 1 & -\sin(\alpha + \lambda_1) \\ 1 & 0 & -\cos(\alpha + \lambda_2) \\ 0 & 1 & -\sin(\alpha + \lambda_2) \\ 1 & 0 & -\cos(\alpha + \lambda_3) \\ 0 & 1 & -\sin(\alpha + \lambda_3) \end{bmatrix}}_{\mathbf{K}_e} \ddot{\mathbf{X}} \quad (4.8)$$

Finalmente, acelerações cartesianas dos pontos C_i podem ser escritas como

$$\ddot{\mathbf{X}}_e = \underbrace{\mathbf{K}_e \mathbf{J}}_{\mathbf{J}_d} \mathbf{M}^{-1} \boldsymbol{\tau} \quad (4.9)$$

Sabendo que $\mathbf{J}_d$ possui posto 3, correspondente aos 3 graus de liberdade do *end-effector*, pode-se deduzir que o terceiro valor singular desta matriz corresponde ao limite de aceleração $\ddot{\mathbf{X}}_e$ para o pior caso em relação à sua direção, considerando um limite de torque. Com este conceito, a seguinte métrica é introduzida

$$\phi_3 = \sigma_3(\mathbf{J}_d) \quad (4.10)$$

Maximizando $\sigma_3(\mathbf{J}_d)$, o terceiro valor singular da matriz $\mathbf{J}_d$, maximiza-se o desempenho dinâmica geral, já que qualquer outra direção de aceleração do vetor $\ddot{\mathbf{X}}_e$ terá desempenho igual ou superior.

Limitação de amplitude de ζ

Durante qualquer tipo de resolução de redundância, é necessário levar em consideração a limitação de amplitude das juntas. No caso da máquina estudada, os atuadores lineares têm comprimento finito. Isso pode ser descrito matematicamente como $|\zeta_i| \leq \zeta_{max}$. Qualquer posição $\zeta_i \rightarrow -\zeta_{max}^+$ ou $\zeta_i \rightarrow +\zeta_{max}^-$ devem ser evitadas. Para isso, uma métrica que receba valores $-\infty$ para estas posições é necessária:

$$\phi_4 = \sum_{i=1}^3 \left(\frac{1}{\zeta_i - \zeta_{max}} - \frac{1}{\zeta_i + \zeta_{max}} \right) \quad (4.11)$$

Divisão de ζ em intervalos

Posições tais como a ilustrada na Fig. 11 podem ser problemáticas para algumas movimentações. Nestas configurações, principalmente se $\|\mathbf{r}_{AC_i}\|$ é pequeno, é provável que haja grande irregularidade na resolução de redundância. Esse fenômeno foi percebido repetidamente em simulações.

Figura 11: Posição onde as derivadas parciais de ϕ_2 são nulas

Fonte: própria

Posições dos atuadores lineares ζ tais como a representada na Fig. 11 devem ser evitadas. Para uma dada cadeia i , denotando essa posição por ζ_s , e a respectiva distância entre A_i e C_i por r_{ACs} conforme indicado na figura, a seguinte equação deve ser respeitada

$$\begin{aligned} \zeta_s \begin{bmatrix} \cos \gamma_i \\ \sin \gamma_i \end{bmatrix} + r_{ACs} \begin{bmatrix} \cos(\gamma_i + \pi/2) \\ \sin(\gamma_i + \pi/2) \end{bmatrix} = \dots \\ \dots = \underbrace{\mathbf{X} + h \begin{bmatrix} \cos(\gamma_i + \pi/2 + \alpha) \\ \sin(\gamma_i + \pi/2 + \alpha) \end{bmatrix}}_{\mathbf{p}} - a \begin{bmatrix} \cos(\gamma_i + \pi/2) \\ \sin(\gamma_i + \pi/2) \end{bmatrix} \end{aligned} \quad (4.12)$$

Que pode ser resolvido para ζ_s e r_{ACs} por (4.13), onde $\mathbf{p}$ é o vetor definido em (4.12).

$$\begin{bmatrix} \zeta_s \\ r_{ACs} \end{bmatrix} = \begin{pmatrix} \cos \gamma_i & \cos(\gamma_i + \pi/2) \\ \sin \gamma_i & \sin(\gamma_i + \pi/2) \end{pmatrix}^{-1} \mathbf{p} \quad (4.13)$$

Para que $\zeta_i(t)$ não se aproxime de ζ_s a seguinte métrica é proposta

$$\phi_5 = -c_5 \sum_{i=1}^3 \frac{1}{|\zeta_i - \zeta_s|} \quad (4.14)$$

Com a aplicação de ϕ_5 , ζ_i tende a se manter em apenas um lado de ζ_s ao longo de toda a movimentação, ou seja, ou no intervalo $-\zeta_{max} < \zeta_i < \zeta_s$ ou no $\zeta_s < \zeta_i < \zeta_{max}$.

Métrica geral

Com as métrica propostas nas seções anteriores, é possível construir uma função somando ponderadamente múltiplas ϕ_k para $i = 1, \dots, 5$. Com isso, todos os aspectos comentados podem ser abordados em uma única análise. Sendo assim:

$$\phi = \sum_{k=1}^5 c_k \phi_k \quad (4.15)$$

O ajuste dos pesos c_k pode ser feito manualmente, uma vez que a ação de cada métrica ϕ_k é clara ao longo das simulações. Pode-se ver ainda que algumas abordagens podem obter resultados semelhantes. Um exemplo evidente é que a análise dinâmica proposta com ϕ_3 já leva em consideração a proximidade com singularidades tanto seriais quanto paralelas. Com isso, a aplicação de ϕ_3 pode eventualmente eliminar ϕ_1 e ϕ_2 . Na seção de resultados esses detalhes serão discutidos tomando exemplo numérico é experimental.

4.2 Métodos Locais

Os métodos de resolução de redundância em que cada instante é abordado independentemente são denominados métodos locais. A presente seção aborda estas estratégias. Primeiramente são apresentados resoluções que definem a cinemática inversa no nível de velocidade e, após isso, uma proposta de resolução no nível de aceleração. Como discutido ao longo da revisão bibliográfica, tais métodos são já bem reconhecidos na literatura. As próximas seções revisitam os conceitos já bem estabelecidos na academia. Os procedimentos propostos não se tratam de otimizações, propriamente ditas. Se trata apenas de métodos capazes de planejar um $\Theta(t)$ pertinente para um dado $\mathbf{X}(t)$.

4.2.1 Método dos gradientes projetados - PG

Para uma dada trajetória $\mathbf{X}(t)$, têm-se diretamente suas velocidades relacionadas $\dot{\mathbf{X}}(t)$. Para o 3PRRR, possuindo redundância cinemática, da mesma forma que as posições dos atuadores $\Theta(t)$ não são definidos para um $\mathbf{X}(t)$, suas velocidades $\dot{\Theta}$ e acelerações $\ddot{\Theta}(t)$ também são indefinidas.

Entretanto, para um instante qualquer, a definição dada na eq. 4.16 parece pertinente:

$$\dot{\Theta}(t) = \mathbf{J}^\dagger \dot{\mathbf{X}}(t) \quad (4.16)$$

Isso porque, de acordo com a definição da pseudo-inversa $\mathbf{J}^\dagger$ da matriz jacobiana, a norma das velocidades das juntas ativas será a menor possível.

Contudo, uma vez que nenhuma análise de desempenho de cada posição é feita, um movimento resultante da eq. 4.16 é altamente sujeito a encontrar posições desfavoráveis, como singularidades. Pensando nisso, (SICILIANO, 1990) a adição de uma nova parcela:

$$\dot{\Theta}(t) = \mathbf{J}^\dagger \dot{\mathbf{X}}(t) + (\mathbf{I} - \mathbf{J}^\dagger \mathbf{J}) \dot{\Theta}_0 \quad (4.17)$$

Uma vez que $\mathbf{J}(\mathbf{I} - \mathbf{J}^\dagger \mathbf{J}) \dot{\Theta}_0 = \mathbf{0}$, é evidente que a segunda parcela da eq. 4.17 não induz qualquer velocidade do *end-effector*. Mesmo assim, uma movimentação do mecanismo é induzida. De fato, esta parcela projeta um vetor de velocidade arbitrário $\dot{\Theta}_0$ no espaço nulo de $\mathbf{J}$. Para fazer uso disso, tipicamente

$$\dot{\Theta}_0 = \nabla \phi(\Theta) \quad (4.18)$$

onde ϕ é uma métrica de desempenho do manipulador para uma dada posição. Desta forma, o vetor $\dot{\Theta}_0$ é direcionado para conduzir uma melhora de desempenho de acordo com ϕ . Finalmente, a parcela $(\mathbf{I} - \mathbf{J}^\dagger \mathbf{J}) \dot{\Theta}_0$ projeta este gradiente no espaço nulo da matriz jacobiana, não alterando a velocidade do *end-effector* obtida pela parcela $\mathbf{J}^\dagger \dot{\mathbf{X}}(t)$. Este conceito caracteriza o método com o nome de Gradientes Projetados (PG) (SICILIANO, 1990).

Em resumo, a primeira parcela da eq. 4.17 resulta em velocidades dos atuadores de norma mínima capazes de produzir a velocidade $\dot{\mathbf{X}}$ desejada. Enquanto isso, a segunda parcela direciona a movimentação para posições mais favoráveis segundo ϕ .

É importante comentar que a aplicação da Eq. 4.17 pode ser feita em tempo real (implementando o método dos Gradientes Projetados no controlador do manipulador) ou previamente à movimentação (integrando a equação diferencial 4.17).

4.2.2 Método dos gradientes reduzidos - RG

O método dos gradientes projetados resolve a cinemática inversa no nível da velocidade procurando simultaneamente a redução da norma de $\dot{\Theta}$ e a melhoria de desempenho. Como resultado, o movimento completo de todas as juntas ativas é obtido.

Por outro lado, se o movimento de três das seis juntas ativas for definido, as restantes podem ser deduzidas diretamente. Pensando nisso, os atuadores lineares são denominados atuadores redundantes e estes serão os únicos argumentos variáveis de todas as análises de desempenho. Em outras palavras, será usado $\phi(\zeta)$ ao invés de $\phi(\Theta)$. Isso é possível já que θ pode ser deduzida para um conjunto de ζ e $\mathbf{X}$ conhecido. Esse método é descrito em (LUCA; ORIOLO, 1991).

Matematicamente, a aplicação deste raciocínio na resolução de redundância resulta em

$$\dot{\zeta}(t) = \nabla \phi(\zeta(t)) \quad (4.19)$$

A simplicidade matemática desta opção é evidente. Da mesma forma, o esforço computacional para a integração da eq. 4.19 também é reduzido em comparação com o necessário para a integração da eq. 4.17. Evidentemente, no cálculo de $\nabla \phi(\zeta(t))$ há necessidade de atualizar a posição de todo corpo do mecanismo para cada instante, de forma que a cinemática do sistema seja respeitada.

4.2.3 Gradientes reduzidos aplicados à aceleração

Diferentemente dos métodos locais apresentados até este ponto, é possível resolver a cinemática inversa no nível das acelerações. De forma análoga à aplicada anteriormente para o RG, as acelerações das juntas redundantes serão direcionadas no sentido do gradiente de uma métrica geral:

$$\ddot{\zeta} = \nabla \phi(\zeta) - c_d \dot{\zeta} \quad (4.20)$$

O termo $-c_d \dot{\zeta}$ é necessário para garantir convergência em posições de ϕ máximo. Para compreender essa necessidade, é simples analisar um estado no qual $\mathbf{X}$ permanece estático dentro da área de trabalho e Θ está em uma posição em que ϕ não é máximo. Se $c_d = 0$, a equação 4.20 se torna a de um oscilador não linear conservativo. Como resultado, Θ oscilará indeterminadamente em torno de uma posição na qual a métrica é maximizada.

Esse fenômeno é claramente indesejável. Para evitá-lo, a parcela $-c_d \dot{\zeta}$ funciona como termo de dissipação, garantindo convergência para a posição de máximo desempenho. A constante c_d deve ser ajustada tendo em vista o compromisso entre rápida atuação das juntas redundantes e o impedimento deste tipo de oscilação.

Após simulações em diversas trajetórias, comparado com os métodos de resolução no nível de velocidade, a presente estratégia apresenta como principal vantagem a estabilidade numérica na integração de 4.20. A aplicação de PG ou RG demandam necessariamente um filtro após a integração numérica, consequência de uma sensível instabilidade numérica. Para o caso em que a resolução é feita no nível das acelerações, esse problema é suavizado.

4.2.4 Condições de Contorno

As seções anteriores apresentaram uma método de determinar velocidades $\dot{\zeta}(t)$ ou acelerações $\ddot{\zeta}(t)$ para uma dada trajetória $\mathbf{X}(t)$. Condições de contorno precisam ser definidas para especificar $\zeta(t)$. Se posições e velocidades iniciais $\zeta(t_i)$ e $\dot{\zeta}(t_i)$ são determinadas, um simples Problema de Valor Inicial é estabelecido. Trajetórias discutidas aqui apresentam $\dot{\mathbf{X}}(t_i) = \mathbf{0}$ e $\dot{\mathbf{X}}(t_f) = \mathbf{0}$, o que faz $\dot{\zeta}(t_i) = \mathbf{0}$ uma escolha consistente. No entanto, a variável $\zeta(t_i)$ restante ainda precisa ser definida.

Como descrito, ϕ deve ser uma métrica confiável do quão favorável uma dada posição é. Em acordo com isso, ela pode ser usada para determinar $\zeta(t_i)$.

Uma trajetória auxiliar $\bar{\mathbf{X}}(\bar{t})$ é definida dentro de um intervalo auxiliar $0 < \bar{t} < \bar{t}_f$ tal que $\bar{\mathbf{X}}(\bar{t}_f) = \mathbf{X}(t_i)$ (pode-se usar $\bar{\mathbf{X}}(\bar{t}_i) = \mathbf{0}$). Então, eq. 4.17, 4.19 ou 4.20 é aplicada na trajetória $\bar{\mathbf{X}}(\bar{t})$. Se as posições dos atuadores redundantes $\bar{\zeta}$ convergirem conforme $\bar{t} \rightarrow \bar{t}_f$, a posição final $\bar{\zeta}(\bar{t}_f)$ da trajetória auxiliar é atribuída à posição inicial $\zeta(t_i)$ da trajetória principal. É claro que a trajetória auxiliar representa apenas um recurso teórico, que não deve ser executado em protótipos físicos.

Independentemente da $\bar{\zeta}(\bar{t}_i)$ escolhida, a aplicação de um dos métodos de resolução de redundância pode guiar $\bar{\zeta}$ para posições consistentes com o procurado por ζ na trajetória principal, induzindo convergência em um promissor $\bar{\zeta}(\bar{t}_f)$. Na tentativa de garantir isso, o intervalo de tempo $[0, \bar{t}_f]$ deve ser longo o bastante. Uma boa escolha de $\bar{\mathbf{X}}$ é um polinômio de grau 5 ligando $\bar{\mathbf{X}}(\bar{t}_i) = \mathbf{0}$ a $\bar{\mathbf{X}}(\bar{t}_f) = \mathbf{X}(t_i)$ com velocidades e acelerações nulas em $\bar{t}_i$ e $\bar{t}_f$.

4.3 Métodos Globais

Salvo em situações em que há interferência significativa do usuário ou do meio em tempo real na atuação do robô, a trajetória $\mathbf{X}(t)$ desejada para uma da tarefa é bem definida antes do seu início. Como consequência, os métodos locais apresentados nas seções anteriores perdem proveito desse dado relevante. Além disso, no caso do robô 3PRRR estudado, as juntas redundante ζ possuem alta inércia. Com isso, o manipulador não tem capacidade de se reconfigurar com grande rapidez. Tendo isso em mente, um método de resolução de redundância global se mostra promissor. Esse tema é discutido nas futuras seções.

4.3.1 Otimização Global por Programação Dinâmica

Principal resultado obtido durante este projeto de mestrado, foi desenvolvido um método em que o movimento ótimo final é um compromisso entre acelerações reduzidas e proximidade com um movimento de referência com posições ideais em cada instante isoladamente. Como será discutido nas próximas seções, o algoritmo proposto apresenta versatilidade e simplicidade computacional.

O cálculo do movimento de referência faz uso do método de gradientes reduzidos já descrito. Uma breve discussão sobre essa aplicação se dará na próxima seção. Contudo, esta referência apresenta em geral acelerações e velocidades ineficazes. No esforço de contornar este problema, uma otimização global é desenvolvida posteriormente aplicando a Programação Dinâmica Diferencial (DDP). Por isso, é necessário discutir mais em detalhe este conceito.

Movimento de Referência

O objetivo desta seção é definir uma estratégia capaz de obter um $\zeta_r(t)$ contínuo de referência na qual a posição em cada instante seja ideal, desconsiderando os níveis de velocidades e acelerações.

Para isso, é possível usar o método dos gradientes reduzidos com ganhos de maior ordem de grandeza. Aplicando RG da com $c_H \phi$, em que c_H é uma constante suficientemente grande, a integração numérica da eq. 4.19 entrega um movimento em que cada posição é suficientemente próximo de um ótimo local de ϕ . Formalmente, a Eq. 4.19 é integrada numericamente com o a função ϕ multiplicada por uma constante c_H suficientemente grande.

Para ilustrar a coerência dessa proposta, a fig. 12 apresenta múltiplos $\zeta(t)$ obtidos para uma mesma trajetória $\mathbf{X}(t)$ através da RG com $c_H \phi$ como métrica, variando c_H . São apresentados curvas relativas a c_H de 10^{-1} até 10^3 com uma sequência logarítmica de 10 valores. Isso esclarece que, dentre estas 10 curvas, as 5 com os maiores valores de c_H são aproximadamente iguais, indicando que para tal perfil se aproxima do ótimo de ϕ para cada instante.

É interessante destacar que tal procedimento é conveniente tanto por não exigir a implementação de um novo método como por facilitar a obtenção de um movimento contínuo.

Programação Dinâmica Diferencial

A programação dinâmica diferencial (DDP) será usada para obter um movimento ótimo global dos atuadores redundantes $\zeta(t)$ que se mantenha o mais próximo possível

Figura 12: Convergência de ζ para c_H crescente

Fonte: própria

de um movimento de referência $\zeta_{ref}(t)$ mas também apresente nível de acelerações $\ddot{\zeta}(t)$ factíveis. Um exemplo de resultado simples deste procedimento é apresentado na fig. 13.

Figura 13: Exemplo de movimento otimizado levando em conta suas acelerações e diferença com o movimento referência

Fonte: própria

O uso de DDP para planejamento de trajetórias tomando em conta a dinâmica do sistema robótico e seus limitantes já é bem conhecido. Fontes como (TASSA; MANSARD; TODOROV, 2014; TODOROV; LI, 2005) aplicaram tais métodos para obter estratégias

de controle eficientes de sistemas complexos. Inspirado em resultados como estes, uma otimização simplificada é introduzida, com complexidade matemática e computacional reduzida.

A programação dinâmica é por definição um método de otimização que recebe como função objetivo um acúmulo de custos ao longo de diversas decisões subsequentes. Uma aplicação bem conhecida é a otimização de funções discretizadas no tempo. Este é o caso da resolução de redundância, sendo que no presente problema o argumento é uma sequência $\zeta_k = \zeta(k)$, correspondentes uma trajetória dos atuadores redundantes no intervalo de tempo da tarefa discretizado em $t_k = t(k) \in (t_i, t_f)$, $k = 1, \dots, n-1$ e n o número total de instantes. Além disso a DDP aplica sequencialmente nestes instantes o conhecimento das derivadas do modelo e dos custos para sua minimização. Deste pontos se inspira o nome Programação Dinâmica Diferencial.

Como comentado, a função objetivo é um acúmulo de custos ao longo da tarefa. Esse acúmulo se dá pela soma de custos instantâneos. Ao longo deste movimento discretizado, é desejado reduzir $\zeta_r(k) - \zeta(k)$ e as acelerações relacionadas $\mathbf{a}(k)$, que são obtidas por diferenças finitas de $\zeta(k)$. Dessa forma, o custo instantâneo para um momento t_k é

$$l(\zeta(k), \mathbf{a}(k)) = c_r \|\zeta_r(k) - \zeta(k)\|^2 + c_a \mathbf{a}(k)^T \mathbf{a}(k), \quad (4.21)$$

onde c_r e c_a são fatores que distribuem a relevância entre a proximidade com o movimento de referência e as acelerações, respectivamente. As acelerações $\mathbf{a}(k)$ são calculadas por aproximação de diferenças finitas de $\zeta(k)$.

É necessário garantir que o manipulador apresente velocidades e acelerações nulas ao fim da tarefa. Com esse objetivo, um função custo aplicada para t_n o instante final penaliza valores não nulos destas variáveis além da diferença com a trajetória de referência

$$l_f(\zeta(n), \mathbf{v}(n), \mathbf{a}(n)) = c_r \|\zeta_r(n) - \zeta(n)\|^2 + \dots \quad (4.22)$$

$$\dots + c_{fv} \mathbf{v}(n)^T \mathbf{v}(n) + c_{fa} \mathbf{a}(n)^T \mathbf{a}(n),$$

onde $\mathbf{v}(k)$ é a aproximação por diferença finitas da derivada de $\zeta(k)$. Constantes c_{fv} e c_{fa} devem assumir valores altos o suficiente para que $\mathbf{v}(n)$ e $\mathbf{a}(n)$ sejam negligenciáveis.

As variáveis $\zeta(k)$, $\mathbf{v}(k)$ e $\mathbf{a}(k)$ são relacionadas por aproximação de diferenças finitas. Como resultado, o argumento da otimização pode ser definido como o vetor de *jerk* $\mathbf{j}(k)$, a aproximação da terceira derivada de $\zeta(k)$. As demais variáveis são integradas numericamente baseadas no seus valores iniciais $\zeta(0)$, $\mathbf{v}(0)$ e $\mathbf{a}(0)$. Isso é conveniente ao invés de usar como argumento $\mathbf{a}(k)$ pois essa variável não deve ser um argumento para que possa ser ponderada na função custo final da eq. 4.22. Sendo assim, a aceleração é vista como variável de estado.

Seguindo neste tema, para usar a notação usual aplicada à DDP, é necessário definir

as variáveis de controle e de estado. Dessa forma, o vetor de variáveis de estado pode ser definido como $\mathbf{x}_k = [\zeta(k)^T \mathbf{v}(k)^T \mathbf{a}(k)^T]^T$, e o vetor de variáveis de controle $\mathbf{u}_k = \mathbf{j}(k)$.

Usando integração de Euler, o vetor de estado $\mathbf{x}_{k+1}$ pode ser descrito como combinação linear do vetor $\mathbf{x}_k$ e o vetor de controle $\mathbf{u}_k$:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k, \quad (4.23)$$

onde $\mathbf{f}$ é uma função vetorial, $\mathbf{A}_k$ é a matriz constante de estados e $\mathbf{B}_k$ é a matriz constante de controle. Reunindo todos os vetores de controle de cada instantena matriz $\mathbf{U} = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$, a função de custo total $J(\mathbf{U}, \mathbf{x}_0)$ pode ser definida sobre a trajetória completa pela suma do custo instatâneo para todos os t_k :

$$J(\mathbf{U}, \mathbf{x}_0) = l_f(\mathbf{x}_n) + \sum_{k=0}^{n-1} l(\mathbf{x}_k). \quad (4.24)$$

Em outra palavras, se o estado inicial $\mathbf{x}_0$ e o conjunto de ações de controle $\mathbf{U} = \{\mathbf{u}_0, \dots, \mathbf{u}_{n-1}\}$ são definidos, os estados ao longo da trajetória $\mathbf{x}$ bem como o custo total J pode ser calculado. Esse processo é chamado *Forward Pass*.

Um problema de otimização considerando como argumento $\mathbf{U}$ e $\mathbf{x}_0$ pode ser proibitivo devido à sua dimensão ($3n$). O Princípio de Otimalidade (BELLMAN, 1956) conclui que este problema pode ser dividido em sub-problemas resolvidos para cada instante. Considerando $\mathbf{U}_p = \{\mathbf{u}_p, \dots, \mathbf{u}_{n-1}\}$ como um sub-conjunto de $\mathbf{U}$ contendo os últimos $n - p$ vetores de $\mathbf{U}$, o custo parcial $J_p(\mathbf{U}_p, \mathbf{x}_p)$ é calculado como

$$J_p(\mathbf{U}_p, \mathbf{x}_p) = l_f(\mathbf{x}_n) + \sum_{k=p}^{n-1} l(\mathbf{x}_k). \quad (4.25)$$

A função V_p é definida como o mínimo de J_p para um dado $\mathbf{x}_p$. Essa função, também conhecida como função *cost-to-go*, pode ser descrita pelo seguinte problema de otimização:

$$V_p(\mathbf{x}) = \min_{\mathbf{U}_p} \left\{ l_f(\mathbf{x}_n) + \sum_{k=p}^{n-1} l(\mathbf{x}_k) \right\}. \quad (4.26)$$

Graças ao Princípio de Otimalidade introduzido por Bellman, se o valor de V_{p+1} já é conhecido, o V_p anterior pode ser obtido por:

$$V_p(\mathbf{x}) = \min_{\mathbf{U}_p} \left\{ l(\mathbf{x}) + V_{p+1}(\mathbf{f}(\mathbf{x}, \mathbf{u}_p)) \right\}. \quad (4.27)$$

Usando essa estratégia, o procedimento começa pelo cálculo de $V_n(\mathbf{x}_n) = l_f(\mathbf{x}_n)$. Então, um $\mathbf{u}_p$ que minimize V_p é procurado para cada estágio p . Esse processo recebe o nome de *Backward Pass*. Como l , l_f e $\mathbf{f}$ são funções quadráticas dos vetores de controle de estados, esses sub-problemas de otimização podem ser resolvidos com métodos de

programação quadrática bem conhecidos. Para todos os estágios p , o procedimento entrega o *jerk* ótimo (vetor de controle $\mathbf{u}_p$), capaz de minimizar o custo J_p , e suas derivadas em relação aos estados $\mathbf{x}_p$. Mais detalhes de como esse processo de otimização podem ser encontrados em (TASSA; EREZ; TODOROV, 2012).

Algoritmo Proposto

A fim de sintetizar o método de otimização global proposto nesta dissertação, sua visão geral é detalhada na presente seção. Primeiramente, o método dos gradientes reduzidos é aplicado:

$$\dot{\boldsymbol{\zeta}} = c_H \nabla (\phi(\boldsymbol{\zeta})) \quad (4.28)$$

com c_H suficientemente grande para garantir que a integração numérica da eq. 4.28 leve a um perfil $\boldsymbol{\zeta}_{ref}(t)$ seguindo posições próximas de ótimos locais da métrica $\phi(\boldsymbol{\zeta})$. Dessa forma, $\boldsymbol{\zeta}_{ref}(t)$ pode ser visto como um perfil contínuo no qual a posição de cada instante isolado é ideal.

Tipicamente, esse movimento referência apresenta níveis de velocidade e aceleração impraticáveis. Sendo assim, a programação dinâmica diferencial (DDP) é aplicada conforme discutido na seção anterior. No custo instantâneo para um momento k , são minimizadas simultaneamente a aceleração $\mathbf{a}(k)$ e a diferença entre a posição atual com a referência $\|\boldsymbol{\zeta}_{ref}(k) - \boldsymbol{\zeta}(k)\|$. Com isso, é alcançado um pertinente compromisso entre esses dois aspectos.

Uma vez determinados, os valores dos atuadores redundantes $\boldsymbol{\zeta}$ podem ser usados para o cálculo das posições dos atuadores restantes $\boldsymbol{\theta}$. Isso pode ser feito aplicando a cinemática inversa discutida na seção 3.1.1, conhecendo também as posições do *end-effector* $\mathbf{X}$.

O método proposto é considerado versátil pois dá ao usuário opção de ajustar as prioridades ao longo da otimização seja pelas constantes l e l_f . Tal compromisso deve ser ajustado através da proporção entre as constantes da eq. 4.15 como da eq. 4.21. A influência de todas essas constantes sobre o movimento resultante é evidente através de simulações, o que facilita esse procedimento de ajuste.

Inclusive, a abordagem global é feita de forma simplificada. O função $\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$ que relaciona um estágio ao posterior é uma função linear. Os custos l e l_f também apresentam clara simplicidade, minimizando somente a aceleração da função final e sua diferença com a função referência. Dessa forma, o maior fardo computacional que geralmente é originário da otimização global é drasticamente reduzido. Mais detalhes desse método estão descritos em (SANTOS; SILVA, 2017).

4.3.2 Otimização Global Truncada

Um outro método de otimização global foi proposta em (SANTOS et al., 2015) e (FONTES; SILVA, 2016). Essa se dá truncando o problema de definir ζ para cada instante da tarefa. Ao invés disso, o argumento da otimização são apenas seus valores iniciais $\zeta(t_i) = \zeta^i$ e finais $\zeta(t_f) = \zeta^f$, de forma que a curva completa é dada por interpolação destes pontos extremos. Sendo assim, a otimização recebe como argumento seis variáveis, a saber: $\zeta_{1,2,3}^i$ e $\zeta_{1,2,3}^f$.

Objetivamente, o perfil das posições $\zeta(t)$ é formado por polinômios de grau 5 com velocidades e acelerações nulas em t_i e t_f . Além disso, a função objetivo proposta é uma combinação linear do maior torque e da pior avaliação de desempenho de posições (obtida por ϕ) ao longo de toda a tarefa. Consequentemente, definindo c_τ e c_ϕ como as constantes dessa combinação linear, a otimização pode ser apresentada como

$$\begin{aligned}
 \min_{\zeta_{1,2,3}^i, \zeta_{1,2,3}^f} & \left(c_\tau \|\boldsymbol{\tau}\|_\infty + c_\phi \|\phi^{-1}\|_\infty \right) \\
 \text{s.t.} & \\
 & - \zeta_{max} \leq \zeta_{1,2,3}^i \leq \zeta_{max} \\
 & - \zeta_{max} \leq \zeta_{1,2,3}^f \leq \zeta_{max}
 \end{aligned} \tag{4.29}$$

5 PROTÓTIPO

Em paralelo ao desenvolvimento dos instrumentos teóricos presentes nos capítulos 3 e 4, procedeu-se com o projeto, montagem e programação do protótipo experimental. Para apresentar uma descrição clara da máquina, a construção do protótipo é sucintamente abordada neste capítulo.

5.1 Projeto, Dimensionamento e Montagem

Utilizando Modelo Dinâmico Inverso descrito em (SANTOS et al., 2015) (torques necessário para realizar trajetórias predefinidas), os componentes de atuação e de movimentação linear foram selecionados.

Os motores escolhidos foram EC60 flat da *Maxon* com 100 W de potência, *brushless*, com redutores planetários GP52 C de redução 3,5:1, disponibilizando na rotação nominal de 1020 rpm, o torque máximo contínuo de 0,82 Nm, utilizando uma corrente máxima contínua de 2,3A e tensão de 48 Vdc. O controlador desse motor DC é a placa EPOS2 50/5, com alimentação de até 50 Vdc e corrente de 5 A. Nesse controlador, além da possibilidade do uso dos algoritmos de controle (PID), há a possibilidade da comunicação entre as diversas unidades de controle via comunicação CAN. Isso garante ao sistema segurança e robustez. A Fig. 14 ilustra estas peças.

A Fig. 15 ilustra as possibilidades de comunicação entre as 6 unidades de controle e o computador. A opção (A) utiliza a comunicação com o computador via USB. Esta opção permite a programação direta do computador com as placas controladoras, seguindo comandos simples em linguagem C. Porém, sofre com limitações de processamento do computador e da transferência de dados da rede CAN para a comunicação USB, inibindo possibilidades de controle em tempo real. A opção (B) utiliza a comunicação com o computador através do equipamento de aquisição de dados dSPACE 1103. Entre esse equipamento e as EPOS, a comunicação é via CAN e entre o computador e a dSPACE 1103 é via ETHERNET. Esta placa de aquisição de dados possui processador de 1 GHz, interface com o *Simulink*, e estaria conectada diretamente à rede CAN. Nesse caso, o usuário poderá programar a lógica de controle via *Simulink* a ser processada na placa controladora dSPACE 1103 abrangendo as possibilidades de controle.

A movimentação linear é estabelecida pelo mesmo conjunto exposto na Fig. 14 acoplado a uma guia linear acionada por fuso de esferas recirculantes. A peça KK60-10-C-E-600-A-1-F0-S3 (vide Fig. 16) produzida pela *Hiwin* possui 600 mm de amplitude de movimento e passo de 10 mm/rotação. Para o acoplamento entre o redutor e o eixo da guia linear foram usados acoplamentos elásticos de fole de pequena dimensão BKL 003

Figura 14: Placa controladora EPOS2 50/5, motor EC 60 flat e redutor GP 52-C (componentes não estão em escala)

Fonte: própria

Figura 15: (a) Comunicação baseada em CAN e USB, e (b) Comunicação baseada em CAN e ETHERNET com interface da dSPACE 1103

Fonte: própria

(Fig. 17) fabricados pela *R+W*. Este componente é apropriado por apresentar alta rigidez torsional (994 Nm/rad).

Peças estruturais tais como elos, suportes para motores e flanges foram manufatu-

Figura 16: Guia linear com fuso de esferas recirculantes

Fonte: própria

Figura 17: Acoplamento de fole BKL 003 usado para acionar as guias lineares

Fonte: própria

rados seguindo medidas de acordo com as simulações prévias. Juntas passivas de revolução foram projetadas e executadas cada uma com uso de um par de rolamentos de esferas garantindo mínimo atrito, alta resistência e mínima folga axial. Uma vista em corte e a peça pode ser vista na Fig. 18. A fim de reduzir a inércia do sistema, a maioria das peças móveis foram manufaturadas em alumínio.

Figura 18: Junta passiva de revolução: vista em corte e peça acabada

Fonte: própria

Para alimentação elétrica de cada placa controladora EPOS2 é usada uma fonte

DC CHM 48-10 fabricada pela *MCE*, com saída de 48Vdc e corrente limite de 10A. Como garantia de segurança, foram instalados fusíveis de ação rápida para casos de sobrecarga e dois botões de emergência ligados ao contator que alimenta as fontes.

Como existem três motores móveis (posicionados sobre a guia linear), foram usadas esteiras porta cabos para sua fiação de alimentação e controle.

Ao fim da montagem, o protótipo apresentado na Fig. 19 foi obtido. O sistema está fixo a um bloco inercial de ferro fundido através de uma placa de alumínio quadrada com largura de 1 metro. O painel com fontes, placas controladoras, disjuntor e contator pode ser visto na Fig. 20. Finalmente, as dimensões da máquina de acordo com a notação introduzida no capítulo 3 estão indicadas na tabela 2.

Figura 19: Vista superior do protótipo.

Fonte: própria

Tabela 2: Especificação das dimensões do protótipo

h	a	ζ_{max}	l_2	l_3	γ_1	γ_2	γ_3
59.7 mm	259.8 mm	249.2 mm	191 mm	232 mm	-180°	60°	-60°

5.2 Programação e Comunicação

Até o momento, foi implementado apenas a comunicação via USB entre o computador e a rede CAN (conforme esquematizado em (A) da Fig. 15). Com isto, é possível comandar as placas controladoras através de diversas interfaces. Foi escolhido proceder a comunicação usando o *MATLAB*, já que os estudos vinham sendo feitos neste ambiente. Este software, tendo acesso à biblioteca original em C fornecida pela *Maxon*, compila as rotinas internas e envia os comandos às placas controladoras.

Figura 20: Pannel com placas controladoras EPOS2 (enumeradas na parte superior), fontes (abaixo), disjuntor e contator (a direita).

Fonte: própria

Existem diversos modos de operação previstos na programação de uma EPOS2. As mais usadas neste projeto são as seguintes:

- Modo de Posição: dada uma posição desejada, age com o controlador PID sobre o erro de posição;
- Modo de Velocidade: dada uma velocidade desejada, age com o controlador PID sobre o erro de velocidade;
- Modo de Corrente: dada uma corrente desejada, age com o controlador PI sobre o erro de corrente;
- Modo de Perfil de Posição: movimentação ponto-a-ponto com perfil de acelerações e velocidades predefinidas;
- Modo de Posições Interpoladas: dados diversos conjuntos [tempo, posição, velocidade], um perfil desejado contínuo é formado por polinômios de grau 3 interpolando conjuntos subsequentes no tempo.

Para iniciar o robô, é necessário que sejam reconhecidas as posições de cada *encoder*. Para isso, uma velocidade baixa é estabelecida para um dado motor com o Modo de Velocidade até que um limitante físico seja encontrado. Este limitante é reconhecido por um pico de corrente na armadura e pode ter sua posição deduzida pelo modelo cinemático.

Para os atuadores lineares, quando a posição atinge a amplitude máxima da guia o pico de corrente é reconhecido e a posição é atribuída como $\pm\zeta_{max}$ conhecido conforme a direção da velocidade. Essa ação é procedida para os três atuadores lineares simultaneamente, estando os atuadores rotativos desabilitados.

Para iniciar um atuador rotativo, todos os lineares são posicionados em ζ_{max} e uma velocidade $\dot{\theta}_i$ é estabelecida enquanto os demais atuadores rotativos estão livres. Em algum momento, θ_i atingirá um limitante para esse posicionamento. Sendo esta posição conhecida pela cinemática, ela é atribuída ao atuador θ_i . Um exemplo dessa situação é mostrada na Fig. 21.

Figura 21: Posicionamento usado para reconhecer a posição de um *encoder* de um atuador rotativo

Fonte: própria

Mais especificamente e a título de exemplo, para o caso da calibração de θ_1 como indicado na Fig. 21, o ângulo θ_{1ref} pode ser obtido da equação

$$\left\| \begin{bmatrix} 0 \\ a \end{bmatrix} - \begin{bmatrix} \zeta_{max} \\ 0 \end{bmatrix} + l_2 \begin{bmatrix} \cos \theta_{1ref} \\ \sin \theta_{1ref} \end{bmatrix} - a \begin{bmatrix} \cos(-\frac{2\pi}{3}) \\ \sin(-\frac{2\pi}{3}) \end{bmatrix} - \dots \right. \\ \left. \dots \zeta_{max} \begin{bmatrix} \cos(-\frac{\pi}{3}) \\ \sin(-\frac{\pi}{3}) \end{bmatrix} \right\| = 2l_2 + l_3 + \sqrt{3}h \quad (5.1)$$

É interessante apontar que, do ponto de vista geométrico, ao menos duas cadeias cinemáticas poderiam ser calibradas no mesmo procedimento (θ_3 também é univocamente determinado pela geometria na Fig. 21). Mas a precisão ao se tentar determinar θ_3 nesta configuração é especialmente baixa. Observe que quando $\theta_1 \rightarrow \theta_{1ref}$, a proporção $\delta\theta_3/\delta\theta_1 \rightarrow \infty$. Ou seja, para $\theta_1 \approx \theta_{1ref}$, a posição θ_3 sofre grandes deslocamentos.

5.3 Identificação de Parâmetros Inerciais

No capítulo 3 foi descrito o modelo dinâmico do manipulador. Ele é essencial para o controle e análise do sistema. Entretanto, os parâmetros inerciais não são conhecidos de início. Como cada corpo rígido do modelo corresponde de fato à uma montagem complexa do protótipo (vide por exemplo, o objeto de massa m_1 , relacionado aos objetos movimentados diretamente pela guia linear) a medição direta destes parâmetros não é simples.

Com isso em mente, foi aplicado a identificação de parâmetros inerciais. Essa se dá com a execução de múltiplos movimentos pelo protótipo. Através disso, obtêm-se torques exigidos de cada atuador. Comparando torques previstos pelo modelo dinâmico e torques obtidos experimentalmente, um conjunto de parâmetros inerciais que minimiza a diferença entre tais dados pode ser calculado.

O esquema de comunicação implementado até o momento apresenta limitação na duração de cada experimento. Dessa forma, a repetição ininterrupta de trajetórias de excitação como proposto em (SWEVERS; VERDONCK; SCHUTTER, 2007) não é possível. Por outro lado, a execução de movimentos não periódicos também traz resultados satisfatórios. Os dados apresentados nessa seção também foram publicados em (VIEIRA et al., 2017).

No total, foram assumidos 4 parâmetros a serem definidos: m_1 , m_2 , m_3 e m_n . O manipulador foi considerado simétrico, de forma que corpos rígidos idênticos no projeto tivessem parâmetros inerciais iguais no protótipo. Inclusive, os momentos de inércia dos elos foram considerados como barras delgadas. Já para o *end-effector* foi assumido $I_n = 3.2 \cdot 10^{-6} m_n$, uma aproximação para um triângulo perfeito.

Procurando desacoplar os fenômenos dinâmicos dos atuadores lineares e rotacionais, na primeira movimentação de excitação apenas um motor linear foi atuado, enquanto todos os outros atuadores permaneceram estáticos. Objetivamente, foi executada uma oscilação entre $-0.06 < \zeta_1 < 0.06$.

No segundo grupo trajetórias, foram desempenhados 5 quadrados concêntricos defasados de 10° , como mostrado na fig. 22.

Com esse conjunto de movimentos, foi proposta a otimização da diferença entre os torques experimentais $\tau_e(t)$ e os previstos pela dinâmica inversa $\tau_s(t)$:

$$\begin{aligned} \mathbf{m}_{iopt} = \arg \min & \left(\sum_{t=t_0}^{t_f} (\tau_e(t) - \tau_s(t))^2 \right) \\ \text{subject to } & \mathbf{m}_{min} \leq \mathbf{m}_{iopt} \leq \mathbf{m}_{max}. \end{aligned} \quad (5.2)$$

onde $\mathbf{m}_{iopt} = \{m_1, m_2, m_3, m_n\}$. Para solução da eq. 5.2 foi aplicada com sucesso a

Figura 22: Trajetórias de excitação

Fonte: (VIEIRA et al., 2017)

programação quadrática sequencial (SQP). Os valores obtidos são detalhados na tabela 3.

Tabela 3: Parâmetros de inércia, iniciais e ótimos

Massa	Valor Inicial (kg)	Valor Final (kg)
m_1	780	603.62
m_2	0.30	0
m_3	0.354	0.5723
m_n	0.78	0.2269

É importante lembrar que o procedimento de identificação de parâmetros inerciais não almeja a obtenção dos valores reais. O único objetivo é estabelecer boa correlação entre modelo e dados experimentais. De fato, a massa nula $m_2 = 0$ é fisicamente impossível. Mas isso indica a possibilidade de que um modelo simplificado com massas pontuais localizadas nos extremos do respectivo elo pode apresentar boa precisão. A massa de 603.62 kg de m_1 é consequência do fato que ela representa uma inércia pontual em A_i equivalente à inércia de todo o mecanismo responsável pela atuação linear. Tal mecanismo possui alta redução (observar as forças equivalentes nos gráficos da Fig. 24). Acumulando a inércia do motor, redutor planetário, fuso de esferas, patins e estrutura em translação, esse valor é consistente. Esse ponto é melhor discutido no anexo B.

Finalmente, tais dados devem ser validados. Para isso, uma trajetória triangular foi executada como apresentado na fig. 23. O triângulo equilátero executado possui altura de 0,18m, e a orientação é mantida $\alpha = -20^\circ$.

Os resultados podem ser vistos na fig. 24. A precisão obtida é satisfatória. A principal

Figura 23: Trajetória triangular utilizada para validação

Fonte: (VIEIRA et al., 2017)

Figura 24: Gráficos comparativos para trajetória de validação (A) atuadores rotativos e (B) atuadores lineares

Fonte: (VIEIRA et al., 2017)

fonte de divergência entre estes dados advém de imprecisões nos testes experimentais. O atual protótipo ainda é sujeito à erros dimensionais significantes e a estratégia de controle ainda merece melhorias. Como o movimento usado para calcular os torques numéricos é o movimento desejado, há diferenças entre este e o movimento real.

6 RESULTADOS

Com a metodologia introduzida nos capítulos 3 e 4, simulações do manipulador podem ser desenvolvidas abordando diversos pontos pertinentes. Entre eles, destacam-se diferenças de comportamento para configurações com métodos de resolução de redundância diferentes e a influência de estratégias de controle e distúrbios. Exemplos dessas análises estão na presente seção.

Similarmente, o protótipo apresentado no capítulo 5 é utilizado para a execução de testes experimentais. Com isso, fenômenos previstos durante as simulações podem ser validados e pontos merecedores de melhorias são destacados.

Procurando abordar todos os conceitos relevantes, duas trajetórias foram estudadas. Todas elas se baseiam em tarefas de *pega-e-põe*, nas quais o *end-effector* visita posições da área de trabalho com velocidades e acelerações nulas, movendo-se entre estes em linha reta. De fato, a movimentação entre dois pontos a serem visitados definem um polinômios de grau 5 no espaço da tarefa resultando em posições final e inicial desejadas e com primeira e segunda derivadas nulas nos instantes extremos.

As duas trajetórias discutidas nas próximas seções se focaram nos seguintes fenômenos:

- **6.1 Trajetória 1:** Movimento de *pega-e-põe* que se aproxima rapidamente de uma singularidade. Testa a capacidade de cada método de reconfigurar o manipulador com agilidade;
- **6.2 Trajetória 2:** Movimento que visita posições de grande porção da área de trabalho apresentando igualmente pontos de singularidade. Avalia a habilidade de movimentações amplas e de reconfiguração simultaneamente.

Para cada trajetória, as seguintes configurações foram aplicadas (por simulações e experimentalmente):

- Não redundante (3RRR)
- Gradientes Reduzidos (no nível das velocidades)
- Otimização Global por Programação Dinâmica
- Otimização Truncada

No anexo C são apresentadas as rotinas de inicialização do MATLAB. Nelas, detalhes do manipulador, seu controle e sua simulação podem ser esclarecidos. Não foram incluídas todas as rotinas devido ao grande volume de programação.

6.1 Primeira Trajetória

Na primeira trajetória, três posições são visitadas, sendo que a primeira e a última são iguais (vide fig. 25).

Figura 25: Posições visitadas na primeira trajetória

Fonte: própria

Essa trajetória é interessante de ser analisada pois, se um manipulador não redundante é utilizado, a posição intermediária é próxima de uma singularidade, como apresentado na fig. 26. Há um intervalo de 0,65s para cada trecho, totalizando 1,3s de movimentação.

6.1.1 Não Redundante

Se um manipulador 3RRR (não redundante) executa essa trajetória, essa proximidade se mostra crítica. Isso pode ser analisado por simulações. Aplicando a dinâmica direta, como descrito no capítulo 3, é possível verificar que com o desempenho deteriorado nessa posição o manipulador atravessa uma singularidade e perde a rigidez. Após isso, os atuadores seguem as posições desejadas, mas o mecanismo já apresenta posição inconsistente. Isso é detalhado na figs. 28 e 27.

Figura 26: Posição próxima de singularidade apresentada na trajetória 1

Fonte: própria

Figura 27: Posições finais desejada e real após um experimento com o manipulador 3RRR

Fonte: (SANTOS; SILVA, 2017)

Figura 28: Resultados numéricos para o manipulador não redundante: (A) Posições finais desejada (azul) e obtida através da dinâmica direta (vermelho), (B) Orientação desejada e simulada ao longo da tarefa

Fonte: própria

6.1.2 Gradientes Reduzidos

Procurando evitar a proximidade à singularidades, o método dos gradientes reduzidos pode ser utilizado. Porém, sabe-se que esse método resolve a cinemática inversa localmente. Como consequência, para que o desempenho de cada posição se mantenha aceitável é necessário que os atuadores lineares se movimentem rapidamente.

Em termos práticos, as constantes c_k na eq. 4.15 podem ser aumentadas sucessivamente para acelerar a movimentação de ζ . Isso foi executado junto com testes experimentais. Com valores baixos, o mesmo fenômeno descrito para o 3RRR se repetia. A movimentação foi acelerado até um ponto em que os atuadores lineares foram incapazes de executar a tarefa. Isso se deu com uma movimentação que exigia acelerações de até 17m/s^2 .

6.1.3 Otimização Global por Programação Dinâmica

O método de resolução de redundância global proposto é capaz de contornar o problema imposto pelo método dos gradientes reduzidos. Como a movimentação é analisada por inteiro, os atuadores lineares já iniciam a tarefa levando em consideração que uma singularidade pode ser encontrada mais a frente.

De fato, foi simulado um distúrbio de $1,2\text{Nm}$ no sentido anti-horário (tendendo a provocar a singularidade). Os resultado são apresentados na Fig. 29. Mesmo com este distúrbio de grande magnitude, considerando que o torque nominal entregue pelos motores é de 1Nm , o manipulador é capaz de manter boa precisão.

Figura 29: Posições (A) cartesianas e (B) orientação simuladas com otimização global e distúrbio contínuo de $1,2\text{Nm}$

Fonte: própria

Pode-se ver finalmente que as correntes demandas de cada motor (vide fig. 30) possuem um limite razoável, levando em consideração que a corrente nominal (máxima corrente contínua) é de 2300mA .

Figura 30: Correntes dos atuadores (A) lineares e (B) rotativos para um experimento com a primeira trajetória e a otimização global

Fonte: própria

6.1.4 Otimização Global Truncada

Assim como para a otimização por programação dinâmica, a otimização truncada analisa a movimentação como um todo. Essa estratégia trouxe os melhores resultado para a trajetória 1.

O método foi capaz de manter o manipulador distante de singularidades e também reduziu os torque necessários. Isso pode ser visto claramente comparando-se as correntes obtidas por esse métodos na fig. 31 com os da fig. 30. As correntes nesse caso não alcançam 2000mA, enquanto o anterior se aproxima de 4000mA.

Figura 31: Correntes dos atuadores rotativos utilizando a otimização global truncada

Fonte: própria

É importante comentar que o resultado da otimização foram posições aproximadamente estáticas dos atuadores lineares. As posições final e inicial ótimas são aproximadamente iguais. Por isso, as correntes demandas dos atuadores lineares foi aproximadamente nula.

6.2 Segunda Trajetória

Um diferencial do método de otimização global por programação dinâmica introduzido é sua capacidade de cobrir a área de trabalho do manipulador. Uma segunda trajetória é estudada para exemplificar essa característica.

Tal trajetória é composta por 4 posições a serem visitadas. Elas podem ser vistas junto com a respectiva sequência na fig. 32. O tempo total da trajetória é de 9s

Figura 32: Posições visitadas na segunda trajetória

Fonte: própria

6.2.1 Configuração não redundante

Novamente, a trajetória passa por singularidades na configuração não redundante. Com isso, o mesmo fenômeno descrito anteriormente ocorre. Como mostrado na fig. 33, a posição final simulada é completamente inconsistente com a desejada, mesmo sem a presença de distúrbios.

6.2.2 Gradientes Reduzidos

Como essa trajetória é efetuada com menores acelerações e velocidades, o método dos gradientes reduzidos é capaz de manter posições razoáveis ao longo de toda a tarefa mesmo se tratando de uma estratégia local.

Por outro lado, como discutido anteriormente, tal proposta resulta em uma movimentação numericamente problemática, apresentando $\ddot{\xi}$ altamente irregular. Como consequência, as forças necessárias para desempenhar essa movimentação são pouco suaves (vide fig. 34).

Figura 33: Posições finais desejada (azul) e simulada (vermelho) para a segunda trajetória

Fonte: própria

Figura 34: Forças demandas dos atuadores lineares para o métodos dos gradientes reduzidos

Fonte: própria

Em adição a isso, são necessárias altas velocidades e acelerações para manter posições aceitáveis, a saber, 150mm/s e 6,8m/s².

6.2.3 Otimização Global por Programação Dinâmica

Como esperado, a otimização global por programação dinâmica possibilita uma movimentação que mantém posições com alto desempenho e com baixos níveis de aceleração e velocidade.

Similarmente à primeira trajetória, foi executada um teste experimental com o protótipo físico. Além disso, foi simulado a movimentação com presença de distúrbio de 1,2Nm. Os resultados destes procedimentos estão na fig. 35.

É importante esclarecer que tanto nas simulações quanto no protótipo foram utilizados sinais de controle *feedforward* lineares. Isso elimina a necessidade da dinâmica inversa para a previsão do torque calculado a ser usado como sinal *feedforward*. Os resultados numéricos e experimentais trouxeram precisão razoável. Isso indica que a aplicação desta estratégia é aceitável para o presente âmbito.

Figura 35: (A) Posições cartesianas e (B) orientações desejadas, simuladas e experimentais

Fonte: própria

Comparando com os resultados obtidos pelo método dos gradientes reduzidos, se destaca a diferença entre as velocidades e acelerações demandadas (68mm/s contra 150mm/s, e 0.11m/s^2 contra 6.8m/s^2).

A título de referência, os principais dados relativos à resolução de redundância são apresentados na tabela 4.

Tabela 4: Parâmetros da Resolução de Redundância

c_1	c_2	c_3	c_4	c_5	c_H	l_t	c_r	c_a
0,02	1	0,01	10^{-3}	0,01	25	$1.2 \sqrt{l_2^2 + l_3^2}$	0,2	$3 \cdot 10^{-3}$

6.2.4 Otimização Global Truncada

Por fim, a otimização global truncada foi aplicada para a segunda trajetória. Tal método não obteve êxito. A escolha de posições final e inicial não possibilita que o manipulador se mantenha distante de singularidade e dentro da sua área de trabalho ao longo das posições intermediárias.

7 CONSIDERAÇÕES FINAIS

O presente texto apresenta uma análise objetiva da influência da redundância cinemática em um manipulador paralelo 3PRRR. Antes de qualquer estudo mais detalhado, foi proposto um modelo do robô. O modelo cinemático aplica as restrições cinemáticas do mecanismo e correlaciona posições e velocidades dos corpos rígidos do mecanismo. Dois modelos dinâmicos são introduzidos. Um modelo aplicando o formalismo de Euler-Lagrange com multiplicadores de Lagrange permite simulação inversa e direta incluindo a influência de distúrbios e estratégias de controle. Outro modelo utilizando o Princípio dos Trabalhos Virtuais obtêm a matriz de massa do manipulador para uma dada posição. Essa matriz tem relação direta com o desempenho dinâmico desenvolvido.

Como discutido, para um robô com redundância cinemática, é necessário definir um método de resolução de redundância para execução de qualquer tarefa. Esse ponto se mostra crítico para que as melhorias de desempenho sejam de fato demonstradas durante testes experimentais e simulações. Por isso, atenção especial foi dedicada ao levantamento dos possíveis métodos de resolução de redundância. Métodos locais, tal como o método dos Gradientes Reduzidos (RG), definem velocidades ou acelerações dos atuadores pertinentes para uma dada métrica, considerando cada instante isoladamente. Em contraste, métodos globais definem uma função objetivo que meça o desempenho da movimentação como um todo. Essa abordagem se mostra promissora mas também desafiadora em consequência da sua complexidade. Foi proposta uma estratégia de resolução global por programação dinâmica que possui duas principais etapas. Primeiramente, uma movimentação referência é obtida na qual cada instante apresente uma posição ótima localmente. Depois disso, é executada uma otimização global utilizando programação dinâmica que procura seguir esta movimentação de referência e reduzir suas acelerações.

Evidentemente, a definição de uma métrica de desempenho é essencial para a resolução de redundância. Sendo assim, foram estabelecidos diversas possibilidades. De fato, como cada uma aborda aspectos específicos de desempenho, a métrica final é definida como uma soma ponderada delas.

Por fim, como foi proposta validação experimental, um protótipo do 3PRRR foi construído. Foi executada sua programação e os seus controladores foram calibrados. Seus parâmetros de inércia foram identificados, possibilitando uma boa correlação entre modelo e resultados experimentais.

Com a metodologia apresentada, resultados numéricos e experimentais são obtidos. A fim de avaliar a influência da redundância cinemática, tais resultados se focam na comparação entre estratégias de resolução de redundância com a configuração não-redundante.

Duas trajetórias são analisadas apresentando diferentes desafios.

A primeira trajetória é aplicada com o intuito de analisar a capacidade de cada estratégia de suprimir singularidades. Foi visto que o método dos gradientes reduzidos, sendo um método local, não pôde manter posições aceitáveis ao longo de toda a tarefa. A resolução global por programação dinâmica se mostrou foi capaz de manter bom desempenho e níveis razoáveis de forças e torques. Já a otimização global truncada obteve o melhor resultado para essa trajetória, suprimindo as singularidades e mantendo baixas exigências de torque. Isso foi possível já que tal método aborda diretamente esses pontos na sua otimização.

A segunda trajetória exige capacidade de reconfiguração ao longo de toda a tarefa, cobrindo grande porção da área de trabalho do manipulador. Os gradientes reduzidos trazem movimentação aceitável, mas exigindo altas acelerações e velocidades. O método truncado não é capaz de reconfigurar o robô continuamente, já que recebe como argumentos apenas posições iniciais e finais. Já a otimização global por programação dinâmica entrega reconfiguração contínua sem necessidade de altas acelerações e velocidades, resultando na melhor opção para esse caso.

Fica claro que o método truncado é ideal para situações nas quais não é necessária contínua reconfiguração, já que este otimiza os torques diretamente. Por outro lado, a otimização por programação dinâmica também pode reduzi-los indiretamente (considerando que as métricas tem correlação com os torques demandados) e pode reconfigurar as posições para cada instante. Além disso, o esforço computacional é muito menor, já que a abordagem global deste método resolve uma programação dinâmica simples. Enquanto isso, para cada avaliação da função custo no caso da otimização truncada, é necessário o cálculo completo da dinâmica inversa. Dessa forma, a resolução de redundância proposta se mostra um método versátil e pertinente para aplicação no manipulador $3P\underline{R}RR$.

Finalmente, a redundância cinemática se mostra um meio de melhoria de desempenho do manipulador planar $3R\underline{R}RR$. É verificado que essa alternativa possibilita a supressão singularidades, redução de torques e aumento da área de trabalho útil.

Os resultados deste projeto serão utilizados em futuros estudos. Dentre eles, destacam-se a análise de melhorias na eficiência energética, impacto de incertezas, aplicação de diferentes estratégias de controle, uso de servovisão e de componentes flexíveis.

REFERÊNCIAS

ABDELLATIF, H.; HEIMANN, B. Computational efficient inverse dynamics of 6-DOF fully parallel manipulators by using the Lagrangian formalism. **Mechanism and Machine Theory**, Elsevier, v. 44, n. 1, p. 192–207, 2009. ISSN 0094-114X.

AGRAWAL, O. P.; XU, Y. On the global optimum path planning for redundant space manipulators. **IEEE Transactions on Systems, Man and Cybernetics**, v. 24, n. 9, p. 1306–1316, 1994. ISSN 00189472.

AHUACTZIN, J. M.; GUPTA, K. K. The kinematic roadmap: A motion planning based global approach for inverse kinematics of redundant robots. **IEEE Transactions on Robotics and Automation**, IEEE, v. 15, n. 4, p. 653–669, 1999. ISSN 1042-296X.

ALBA-GOMEZ, O.; WENGER, P.; PAMANES, A. Consistent kinetostatic indices for planar 3-DOF parallel manipulators, application to the optimal kinematic inversion. In: **ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference**. [S.l.]: American Society of Mechanical Engineers, 2005. p. 765–774.

ASADA, H. A geometrical representation of manipulator dynamics and its application to arm design. **Journal of dynamic systems, measurement, and control**, American Society of Mechanical Engineers, v. 105, n. 3, p. 131–142, 1983. ISSN 0022-0434.

ASCHER, U. M.; PETZOLD, L. R. **Computer methods for ordinary differential equations and differential-algebraic equations**. [S.l.]: Siam, 1998. v. 61. ISBN 0898714125.

BELLMAN, R. Dynamic programming and Lagrange multipliers. **Proceedings of the National Academy of Sciences**, National Acad Sciences, v. 42, n. 10, p. 767–769, 1956. ISSN 0027-8424.

BONEV, I. **Delta Parallel Robot - the Story of Success**. 2001. Disponível em: <http://www.parallemic.org/Reviews/Review002.html>.

BOWLING, A.; KHATIB, O. Dynamic loading criteria in actuator selection for desired dynamic performance. **Advanced Robotics**, Taylor & Francis, v. 17, n. 7, p. 641–656, 2003. ISSN 0169-1864.

_____. The dynamic capability equations: a new tool for analyzing robotic manipulator performance. **IEEE transactions on robotics**, IEEE, v. 21, n. 1, p. 115–123, 2005. ISSN 1552-3098.

BOWLING, A. P.; KHATIB, O. Dynamic Performance and Modular Design of Redundant Macro-/Minimanipulators. **Journal of Mechanical Design**, American Society of Mechanical Engineers, v. 130, n. 9, p. 92301, 2008. ISSN 1050-0472.

BRENAN, K. E.; CAMPBELL, S. L.; PETZOLD, L. R. **Numerical solution of initial-value problems in differential-algebraic equations**. [S.l.]: Siam, 1996. v. 14. ISBN 1611971225.

BRIOT, S.; BONEV, I. Are parallel robots more accurate than serial robots? **CSME Transactions**, v. 31, n. 4, p. 445–456, 2007.

CLAVEL, R. Delta, a fast robot with parallel geometry. In: **Proc. 18th Int. Symp. on Industrial Robots, Lausanne, 1988**. [S.l.: s.n.], 1988. p. 91–100.

CONKUR, E. S.; BUCKINGHAM, R. Clarifying the definition of redundancy as used in robotics. **Robotica**, Cambridge Univ Press, v. 15, n. 05, p. 583–586, 1997. ISSN 1469-8668.

CORBEL, D. et al. Towards 100G with PKM. Is actuation redundancy a good solution for pick-and-place? In: **Robotics and Automation (ICRA), 2010 IEEE International Conference on**. [S.l.]: IEEE, 2010. p. 4675–4682. ISBN 1424450381.

CORBEL, D.; GOUTTEFARDE, M.; PIERROT, F. Actuation redundancy as a way to improve the acceleration capabilities of 3T and 3T1R pick-and-place parallel manipulators. **Journal of Mechanisms and Robotics**, American Society of Mechanical Engineers, v. 2, n. 4, p. 41002, 2010. ISSN 1942-4302.

DAVLIAKOS, I.; PAPADOPOULOS, E. Model-based control of a 6-dof electrohydraulic Stewart–Gough platform. **Mechanism and machine theory**, Elsevier, v. 43, n. 11, p. 1385–1400, 2008. ISSN 0094-114X.

DEB, K. et al. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. **IEEE Transactions on the Evolutionary Computation**, v. 6, p. 182–197, 2002.

FINOTTI, G. **Cálculo explícito dos torques dos atuadores de um robô paralelo plano empregando o método de Kane**. [S.l.]: Universidade de São Paulo.

FONTES, J. V.; SILVA, M. M. da. On the dynamic performance of parallel kinematic manipulators with actuation and kinematic redundancies. **Mechanism and Machine Theory**, Elsevier, v. 103, p. 148–166, 2016. ISSN 0094-114X.

GELFAND, I. M. et al. **Calculus of variations**. [S.l.]: Courier Corporation, 2000. ISBN 0486414485.

GOLDSTEIN, H. **Classical mechanics**. [S.l.]: Pearson Education India, 1965. ISBN 8131758915.

GOUTTEFARDE, M.; GOSSELIN, C. M. Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms. **IEEE Transactions on Robotics**, IEEE, v. 22, n. 3, p. 434–445, 2006. ISSN 1552-3098.

GRAETTINGER, T. J.; KROGH, B. H. The acceleration radius: a global performance measure for robotic manipulators. **IEEE Journal on Robotics and Automation**, IEEE, v. 4, n. 1, p. 60–69, 1988. ISSN 0882-4967.

HAIRER, E.; WANNER, G. Solving ordinary differential equations ii: Stiff and differential-algebraic problems second revised edition with 137 figures. **Springer series in computational mathematics**, Springer-Verlag, v. 14, 1996. ISSN 0179-3632.

HESS-COELHO, T. A. Topological synthesis of a parallel wrist mechanism. **Journal of Mechanical Design**, American Society of Mechanical Engineers, v. 128, n. 1, p. 230–235, 2006. ISSN 1050-0472.

_____. A redundant parallel spherical mechanism for robotic wrist applications. **Journal of Mechanical Design**, American Society of Mechanical Engineers, v. 129, n. 8, p. 891–895, 2007. ISSN 1050-0472.

HESSELBACH, J. et al. Aspects on design of high precision parallel robots. **Assembly Automation**, Emerald Group Publishing Limited, v. 24, n. 1, p. 49–57, 2004. ISSN 0144-5154.

KAZEROUNIAN, K.; WANG, K. Global versus local optimization in redundancy resolution of robotic manipulators. **International Journal of Robotics Research**, Univ of Connecticut, United States, v. 7, n. 5, p. 3–12, 1988. ISSN 02783649 (ISSN).

KHATIB, O.; BURDICK, J. Optimization of dynamics in manipulator design: The operational space formulation. **INT. J. ROBOTICS AUTOM.**, v. 2, n. 2, p. 90–98, 1987.

KIM, H. et al. Redundancy resolution of the human arm and an upper limb exoskeleton. **IEEE Transactions on Biomedical Engineering**, v. 59, n. 6, p. 1770–1779, 2012.

KIM, S.-G.; RYU, J. New dimensionally homogeneous Jacobian matrix formulation by three end-effector points for optimal design of parallel manipulators. **IEEE Transactions on Robotics and Automation**, IEEE, v. 19, n. 4, p. 731–736, 2003. ISSN 1042-296X.

KIM, Y.; DESA, S. The definition, determination, and characterization of acceleration sets for spatial manipulators. **The international journal of robotics research**, Sage Publications Sage CA: Thousand Oaks, CA, v. 12, n. 6, p. 572–587, 1993. ISSN 0278-3649.

KOSEKI, Y. et al. Design and accuracy evaluation of high-speed and high precision parallel mechanism. In: **Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on**. [S.l.]: IEEE, 1998. v. 2, p. 1340–1345. ISBN 078034300X.

KOTLARSKI, J. et al. Enlarging the useable workspace of planar parallel robots using mechanisms of variable geometry. In: **Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFTOMM International Conference on**. [S.l.]: IEEE, 2009. p. 63–72. ISBN 8889007370.

KOTLARSKI, J.; HEIMANN, B.; ORTMAIER, T. Experimental validation of the influence of kinematic redundancy on the pose accuracy of parallel kinematic machines. In: **Robotics and Automation (ICRA), 2011 IEEE International Conference on**. [S.l.]: IEEE, 2011. p. 1923–1929. ISBN 1612843867.

LANCZOS, C. **The variational principles of mechanics**. [S.l.]: Courier Corporation, 1970. v. 4. ISBN 0486650677.

LI, W.; TODOROV, E. Iterative linear quadratic regulator design for nonlinear biological movement systems. In: **ICINCO (1)**. [S.l.: s.n.], 2004. p. 222–229.

LIU, S.; QIU, Z.-c.; ZHANG, X.-m. Singularity and path-planning with the working mode conversion of a 3-dof 3-rrr planar parallel manipulator. **Mechanism and Machine Theory**, Elsevier, v. 107, p. 166–182, 2017.

LJUNG, L. **System identification**. [S.l.]: Wiley Online Library, 1999. ISBN 047134608X.

LUCA, A. D.; ORIOLO, G. Reduced gradient method for solving redundancy in robot arms. **Robotersysteme**, v. 7, n. 2, p. 117–122, 1991. ISSN 0178-0026.

MA, O.; ANGELES, J. The concept of dynamic isotropy and its applications to inverse kinematics and trajectory planning. In: **Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on**. [S.l.]: IEEE, 1990. p. 481–486. ISBN 0818690615.

MARCOS, M. da G.; MACHADO, J. A. T.; AZEVEDO-PERDICOÚLIS, T.-P. A multi-objective approach for the motion planning of redundant manipulators. **Applied Soft Computing**, Elsevier, v. 12, n. 2, p. 589–599, 2012. ISSN 1568-4946.

MAYNE, D. A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems. **International Journal of Control**, Taylor & Francis, v. 3, n. 1, p. 85–95, 1966. ISSN 0020-7179.

MCCLOY, D. Some comparisons of serial-driven and parallel-driven manipulators. **Robotica**, Cambridge Univ Press, v. 8, n. 04, p. 355–362, 1990. ISSN 1469-8668.

MERLET, J. Redundant parallel manipulators. **Laboratory Robotics and Automation**, Wiley Online Library, v. 8, n. 1, p. 17–24, 1996. ISSN 1098-2728.

MERLET, J.-P. **Parallel robots**. [S.l.]: Springer Science & Business Media, 2012. v. 74. ISBN 9401095876.

MOHAMED, M. G.; GOSSELIN, C. M. Design and analysis of kinematically redundant parallel manipulators with configurable platforms. **Robotics, IEEE Transactions on**, IEEE, v. 21, n. 3, p. 277–287, 2005. ISSN 1552-3098.

NABAT, V. et al. Par4: very high speed parallel robot for pick-and-place. In: **Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on**. [S.l.]: IEEE, 2005. p. 553–558. ISBN 0780389123.

NENCHEV, D. N. Redundancy resolution through local optimization: A review. **Journal of robotic systems**, Wiley Online Library, v. 6, n. 6, p. 769–798, 1989. ISSN 1097-4563.

RABIER, P. J.; RHEINBOLDT, W. C. **Nonholonomic motion of rigid mechanical systems from a DAE viewpoint**. [S.l.]: Siam, 2000. v. 68. ISBN 089871446X.

ROCHA, D. M.; SILVA, M. M. da. Workspace and singularity analysis of redundantly actuated planar parallel kinematic machines. In: **Proc. of the XV International Symposium on Dynamic Problems of Mechanics, Búzios, RJ, Brazil**. [S.l.: s.n.], 2013. p. 1–10.

SANTOS, J. C. et al. Numerical Analysis and Prototyping Details of a Planar Parallel Redundant Manipulator. In: **12th Latin American Robotics Symposium, LARS 2015**. [S.l.: s.n.], 2015.

SANTOS, J. C.; ROCHA, D. M.; SILVA, M. M. da. Performance Evaluation of Kinematically Redundant Planar Parallel Manipulators. In: **22nd International Congress of Mechanical Engineering, COBEM 2013**. [S.l.: s.n.], 2013.

SANTOS, J. C.; SILVA, M. M. da. Redundancy Resolution of Kinematically Redundant Parallel Manipulators Via Differential Dynamic Programing. **Journal of Mechanisms and Robotics**, American Society of Mechanical Engineers, v. 9, n. 4, p. 41016, 2017. ISSN 1942-4302.

SEN, S.; YAKOWITZ, S. J. A quasi-Newton differential dynamic programming algorithm for discrete-time optimal control. **Automatica**, Elsevier, v. 23, n. 6, p. 749–752, 1987. ISSN 0005-1098.

SHAYYA, S. **Towards Rapid and Precise Parallel Kinematic Machines**. [S.l.]: Ph. D. thesis, University of Montpellier, Montpellier, 2015.

SHAYYA, S. et al. Dimensional synthesis of 4 DoFs (3T-1R) actuatedly redundant parallel manipulator based on dual criteria: Dynamics and precision. In: **Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on**. [S.l.]: IEEE, 2014. p. 1716–1723. ISBN 1479969346.

SICILIANO, B. Kinematic control of redundant robot manipulators: A tutorial. **Journal of intelligent and robotic systems**, Springer, v. 3, n. 3, p. 201–212, 1990. ISSN 0921-0296.

SILVA, M. M. da et al. Integrating structural and input design of a 2-DOF high-speed parallel manipulator: A flexible model-based approach. **Mechanism and Machine Theory**, Elsevier, v. 45, n. 11, p. 1509–1519, 2010. ISSN 0094-114X.

STEWART, D. A platform with six degrees of freedom. **Proceedings of the institution of mechanical engineers**, SAGE Publications, v. 180, n. 1, p. 371–386, 1965. ISSN 0020-3483.

SWEVERS, J. et al. Optimal robot excitation and identification. **Robotics and Automation, IEEE Transactions on**, IEEE, v. 13, n. 5, p. 730–740, 1997. ISSN 1042-296X.

SWEVERS, J.; VERDONCK, W.; SCHUTTER, J. D. Dynamic model identification for industrial robots. **Control Systems, IEEE**, IEEE, v. 27, n. 5, p. 58–71, 2007. ISSN 1066-033X.

TASSA, Y.; EREZ, T.; TODOROV, E. Synthesis and stabilization of complex behaviors through online trajectory optimization. In: **Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on**. [S.l.]: IEEE, 2012. p. 4906–4913. ISBN 1467317373.

TASSA, Y.; MANSARD, N.; TODOROV, E. Control-limited differential dynamic programming. In: **Robotics and Automation (ICRA), 2014 IEEE International Conference on**. [S.l.]: IEEE, 2014. p. 1168–1175.

THANH, T. et al. Dynamics identification of kinematically redundant parallel robots using the direct search method. **Mechanism and machine theory**, Elsevier, v. 52, p. 277–295, 2012. ISSN 0094-114X.

TODOROV, E.; LI, W. A generalized iterative LQG method for locally-optimal feedback control of constrained nonlinear stochastic systems. In: **American Control Conference, 2005. Proceedings of the 2005.** [S.l.]: IEEE, 2005. p. 300–306. ISBN 0780390989.

VIEIRA, H. L. et al. Dynamic Modelling and Identification of a Planar PKM with Several Levels of Kinematic Redundancy. In: **XVII International Symposium on Dynamic Problems of Mechanics - DINAME 2017.** São Sebastião: [s.n.], 2017.

WANG, J. et al. Dynamic feed-forward control of a parallel kinematic machine. **Mechatronics**, Elsevier, v. 19, n. 3, p. 313–324, 2009. ISSN 0957-4158.

WANG, L. et al. An experimental study of a redundantly actuated parallel manipulator for a 5-DOF hybrid machine tool. **Mechatronics, IEEE/ASME Transactions on**, IEEE, v. 14, n. 1, p. 72–81, 2009. ISSN 1083-4435.

WU, J.; WANG, J.; YOU, Z. A comparison study on the dynamics of planar 3-DOF 4-RRR, 3-RRR and 2-RRR parallel manipulators. **Robotics and computer-integrated manufacturing**, Elsevier, v. 27, n. 1, p. 150–156, 2011. ISSN 0736-5845.

XIANG, J.; ZHONG, C.; WEI, W. General-weighted least-norm control for redundant manipulators. **IEEE Transactions on Robotics**, v. 26, n. 4, p. 660–669, 2010.

YOSHIKAWA, T. Dynamic manipulability of robot manipulators. In: **Robotics and Automation. Proceedings. 1985 IEEE International Conference on.** [S.l.]: IEEE, 1985. v. 2, p. 1033–1038.

Anexos

ANEXO A – PLANEJAMENTO DE MOVIMENTO COM CÁLCULO VARIACIONAL

Similarmente ao detalhado no capítulo 4 foi desenvolvido método de Planejamento de Movimento com Otimização usando Cálculo Variacional. Não foram obtidos resultados significativos, como será detalhado mais adiante. Por isso, os conceitos serão descritos brevemente. Justificativas aprofundadas dos instrumentos do Cálculo de Variações podem ser encontrados em (GELFAND et al., 2000).

O procedimento do capítulo 4 indica uma atuação local para dados locais. Isto é, dadas as posições e velocidades de cada corpo num dado instante t_n , é proposta uma aceleração dos atuadores redundantes $\ddot{\zeta}(t_n)$, de forma que não são necessárias informações de instantes $t \neq t_n$. O Planejamento de Movimento se dá na repetição deste procedimento para cada instante.

Mas como consequência da abordagem ser local, não há possibilidade de otimização da movimentação como um todo (GELFAND et al., 2000). Pensando neste ponto, ao invés de introduzir uma métrica local ϕ , será estudado um parâmetro global integrando toda a movimentação, a saber, o funcional na equação (A.1). Métodos semelhantes foram propostos em (AGRAWAL; XU, 1994; KAZEROUNIAN; WANG, 1988) para manipuladores seriais.

$$I = \int_{t_i}^{t_f} (T(\mathbf{q}, \dot{\mathbf{q}}) - \phi(\mathbf{q})) dt \quad (\text{A.1})$$

Em (A.1), o termo $T(\mathbf{q}, \dot{\mathbf{q}})$ é a energia cinética do sistema e ϕ é uma função que recebe valores altos para posições favoráveis. Se para uma movimentação com um $\zeta(t)$ o funcional I for minimizado, o termo $T(\mathbf{q}, \dot{\mathbf{q}}) - \phi(\mathbf{q})$ foi minimizado ao longo do tempo, mantendo baixos valores da energia cinética e altos valores da função ϕ durante toda a tarefa. Por consequência, a movimentação poderia ser chamada de ótima perante a análise de T e ϕ .

Matematicamente, para que ζ resulte em um I mínimo, sua primeira variação deve ser nula

$$\delta I = \delta \int_{t_i}^{t_f} (T(\mathbf{q}, \dot{\mathbf{q}}) - \phi(\mathbf{q})) dt = 0$$

Esmiuçando esta variação

$$\begin{aligned} \delta I &= \int_{t_i}^{t_f} \sum_{k=1}^n \left(T_{\dot{q}_k} \delta \dot{q}_k + (T_{q_k} - \phi_{q_k}) \delta q_k \right) dt = \\ &= \left[T_{\dot{q}_k} \sum_{k=1}^n \delta q_k \right]_0^{t_f} + \int_{t_i}^{t_f} \sum_{k=1}^n \left[(T_{q_k} - \phi_{q_k} - \dot{T}_{\dot{q}_k}) \delta q_k \right] dt = 0 \end{aligned}$$

Onde os subíndices indicam derivadas parciais. Para um caso no qual q_k são mutuamente dependentes, δq_k ($k = 2 \dots n$) podem ser escritos como $\delta q_k = \rho_k \delta q_1$. Assim, condições suficientes para um funcional estacionário (com primeira variação nula) são apresentadas abaixo.

$$\sum_{k=1}^n (T_{q_k} - \phi_{q_k} - \dot{T}_{\dot{q}_k}) \rho_k = 0 \quad (\text{A.2})$$

$$\sum_{k=1}^n (T_{\dot{q}_k} \rho_k) \delta q_1 \Big|_{t_a} = 0 \quad \left\{ t_a = t_i, t_f \right\} \quad (\text{A.3})$$

Aplicando este conceito para o manipulador estudado e fazendo $\mathbf{q}_i$ serem os parâmetros de posições de uma cadeia específica $\mathbf{q}_i = [\zeta_i \ \theta_i \ \beta_i]^T$ (de forma que θ_i e β_i sejam dependentes de $\zeta_i = q_1$), as derivadas parciais podem ser calculadas analiticamente, e o sistema formado pelas equações (A.2) e (A.3) pode ser resolvido para as três cadeias $i = 1, 2, 3$ simultaneamente, resultando em uma movimentação ótima.

Pormenorizando, para cada $\mathbf{q}_i$ a equação (A.2) resulta em uma equação diferencial ordinária em ζ_i e a equação (A.3) exige condições de contorno nos instantes limites $t_a = t_i, t_f$. Tais condições de contorno podem ser respeitadas de duas formas diferentes: (i) com variação δq_1 nula em t_a , ou seja, o ponto $\zeta_i(t_a)$ é fixo e não está em otimização, e (ii) com o termo $\sum_{k=1}^n (T_{\dot{q}_k} \rho_k)$ nulo para $t = t_a$.

A maneira mais simples de proceder é escolher um ponto inicial pertinente fixo (fazendo $\delta q_1(t_i) = 0$) e integrar numericamente (A.3). Observe que, por consequência, ao se escolher a condição inicial, toda a movimentação é definida diretamente, incluindo para t_f de forma que $\delta q_1(t_f) = 0$, respeitando todas as condições de contorno. A interpretação de tal resultado é que a movimentação obtida é a ótima dados os pontos em t_a fixos. Contudo, em geral, as condições finais em t_f não são favoráveis, em contraste com as iniciais que foram escolhidas pelo usuário. A integração numérica de (A.2) dirige o sistema para baixos valores de ϕ , ou seja, configurações desfavoráveis.

Para superar este impasse a solução por (ii) em t_f seria indicada. Isso pode ser feito iterativamente, com a repetição do seguinte procedimento: para um dado ponto inicial com $\sum_{k=1}^n (T_{\dot{q}_k} \rho_k) = 0$, (A.2) é integrada e a condição (ii) é testada em t_f , aferindo se $\sum_{k=1}^n (T_{\dot{q}_k} \rho_k) = 0$. A movimentação resultante é ótima perante a análise de I (os pontos em t_a não foram fixos e respeitam as condições de contorno). Entretanto, o procedimento

iterativo descrito acima é problemático, apresentando dificuldade de convergência. Só foram obtidos resultados coerentes para trajetórias extremamente curtas.

ANEXO B – EQUIVALÊNCIA DE INÉRCIAS NA ATUAÇÃO LINEAR

Existem diversas peças móveis responsáveis por transladar o ponto A_i , por exemplo: rotor do motor, engrenagens do redutor, fuso de esferas, patins... O modelo procura uma massa equivalente m_{eq} para todo esse mecanismo como se toda essa inércia estivesse concentrada apenas em A_i . Há grande amplitude de rotação do motor, redutor e fuso para uma pequena translação do ponto A_i . Por isso o valor de m_{eq} recebe valor alto. Isso é justificado a seguir.

Chamando q a posição angular do motor que aciona o fuso e ζ a posição linear do ponto A_i , temos

$$\delta\zeta = k \delta q \quad (\text{B.1})$$

Considerando a redução do fuso e do redutor planetário, $k = 4.55 \cdot 10^{-4}$ m/rad.

Na realidade, há duas opções: (i) considerar toda a inércia do mecanismo como uma massa pontual m_{eq} em A_i transladando com ζ ou (ii) considerar toda a inércia do mecanismo como um único momento de inércia I_{eq} rotacionando conjuntamente com q . Pelo princípio dos trabalhos virtuais, respectivamente aos casos (i) e (ii), uma força F responsável por transladar ζ pode ser relacionado a um torque T responsável por rotacionar q de acordo com

$$T = k F \quad (\text{B.2})$$

com k a mesma constante da Eq. B.1. Também é claro que

$$\begin{aligned} T &= I_{eq} \ddot{q} \\ F &= m_{eq} \ddot{\zeta} \end{aligned} \quad (\text{B.3})$$

Ainda, como consequência da Eq. B.1

$$\ddot{\zeta} = k \ddot{q} \quad (\text{B.4})$$

Usando todas as equações acima:

$$\begin{aligned}
 \underbrace{I_{eq}}_T \ddot{q} &= k \underbrace{m_{eq}}_F \ddot{\zeta} \\
 I_{eq} \ddot{q} &= k m_{eq} \underbrace{k \ddot{q}}_{\ddot{\zeta}} \\
 I_{eq} &= k^2 m_{eq}
 \end{aligned} \tag{B.5}$$

Aplicando o valor atual de $m_{eq} = 634 \text{ kg}$, chegamos ao valor de $I_{eq} = 1.31 \cdot 10^{-4} \text{ kg.m}^2$. Esse valor é razoável. Analisando todas as peças e seus respectivos datasheets, a peça com maior inércia é o próprio rotor do motor, com $1.21 \cdot 10^{-4} \text{ kg.m}^2$.

ANEXO C – CÓDIGOS DE INICIALIZAÇÃO MATLAB

```

function output = Simulate(simopt)
%
% Simulate(input) --      input = 1 -> (wo FD)
%                        input = 2 -> (w  FD)
%                        input = 3 -> (wo FD - wo Animations - wo Charts)

clc; close all; addpath(genpath(cd));
robot      = RobotConfig;
model      = ModelConfig;
[mpath,ipath] = PathConfig(model);
control    = ContMotConfig(robot,model,mpath,ipath);
mdes       = MotionPlanning(robot,model,mpath,ipath,control);
ddes       = InverseDynamics(robot,mdes,model,control);

output.robot      = robot;
output.model      = model;
output.mdes       = mdes;
output.ddes       = ddes;
output.cont       = control;

if simopt == 2 && mdes.tend == length(mdes.T)
    [msim, dsim] = ForwardDynamics(robot,ddes,mdes,control,model);
    output.dsim  = dsim;
    output.msim  = msim;
    AnimateSync(output,true,15)
    newfig('Forward Dynamics Error'); plot(output.msim.T,output.dsim.fnorm)
    ShowTau(output)
elseif simopt == 1
    SAnimate(output,1,true,2,10,'')
    ShowTau(output)
end

[maxtorq, limmot] = max(max(abs(ddes.tauconv)));
fprintf('\n%f,%d\n\n',maxtorq,limmot)

```

```
function robot = RobotConfig() %#ok<STOUT>

LoadRobot          = true;

if LoadRobot
    load('robot_ref.mat')
else
    MassMultiplier    = 1.0e-0; %#ok<UNRCH>
    robot.h           = 59.73502692e-3;
    robot.a           = 259.8076211e-3;
    robot.zmax        = 210e-3; % real: 249.2e-3;
    robot.l           = [0 .191 .232];
    robot.m           = [603.6 1e-5 0.572]*MassMultiplier;
    robot.bsl         = 1e-2; % Ball Screw Lead
    robot.TorqConstant = 114e-6; % Nm/mA
    robot.MotorReduct  = 3.5;
    robot.mn          = 0.2268*MassMultiplier;
    robot.s           = robot.l/2;
    robot.I           = robot.m.*4.*(robot.s.^2)/3;
    robot.In          = 3.2e-6*robot.mn;
    robot.gamma       = [0; -2*pi/3; 2*pi/3]+pi;
    robot.lambda      = -pi/2 + robot.gamma;
    robot.actdof       = [1,2,4,5,7,8]; % [z1 t1 b1 z2 t2 b2
    % z3 t3 b3 x y z] Active joints

    robot.linact       = [1,4,7];
    robot.rotact       = [2,5,8];
    robot.pssdof       = [3,6,9,10,11,12];

    m                 = robot.m;
    I                 = robot.I;
    mn                = robot.mn;
    In                = robot.In;
    robot.mu          = {diag([m(1),m(1),I(1)]),diag([m(2),m(2),I(2)]),diag([m(3)
    robot.mun          = diag([mn,mn,In]);
end

robot.TauConv        = robot.bsl/(2*pi); %#ok<NODEF>
```

```

function model = ModelConfig()
%% Variables Input
so          = 6;                % BDF order
bc          = defcoef(so);      % BDF coefficients
DCheck      = 'off';           % Derivative Check
Jac         = 'on';            % Analytic Jacobian
                                % (Forward Dynamics)
fsDisp      = 'off';           % Display (fsolve)
TolFun      = 1e-13;           % fsolve tolerance
TolX        = 1e-13;
MaxIter     = 1e2;             % fsolve Max Iterations
dt          = 3e-4;            % Delta t
testMF      = true;            % Test Maximum norm(fout)
                                % - Forward Dynamics
MaxF        = 1e-08;           % Maximum norm(fout) permitted
                                % - Forward Dynamics
DistOn      = 1;               % Apply disturbance
DistOpt     = 3;               % Which function is used in Disturban
DistAmp     = [[10    11    12 ];... % Components and amplitude
                                % of disturbances
                DistOn*[0.0    0.0    -1.00]];
omega       = 10;              % Frequency of disturbance
Newton      = false;           % Inverse Dynamics using
                                % Newton - Check Model
DampFac     = 1200*0;

%% Motion Planning Variables -- Needed to define it earlier
dT0         = 20;              % Time interval
                                % alternative path
ws          = 50;              % Window size (time steps)
                                % moving average filter

%% Define Model Structure
model.bc     = bc;
model.so     = so;
model.dt     = dt;
model.a0     = bc(end)/model.dt;
model.disturb = DisturbanceConfig(DistAmp,DistOpt,omega);
model.maxf   = MaxF;
model.testMF = testMF;
model.dT0    = dT0;
model.ws     = ws;

```

```
model.Newton      = Newton;
model.DampFac     = DampFac;
if strcmp(version,'7.10.0.499 (R2010a)')
    model.fsoptions = optimset('display',fsDisp,'jacobian',Jac,'derivativecheck',1,
                              'TolFun',TolFun,'TolX',TolX,'MaxIter',MaxIter);
else
    model.fsoptions = optimoptions('fsolve','display',fsDisp,'jacobian',Jac,'derivativecheck',1,
                                   'TolFun',TolFun,'TolX',TolX,'MaxIter',MaxIter)
end
```

```

function [mainpath, initpath] = PathConfig(model)

%% Load Variables
ws          = model.ws;
dt          = model.dt;

%% Path options
UserProvMotion = 'void.mat';           % if MP = 4, w
pathopt       = 'triang';              % Given a path
exttime       = dt*ws*8.0;              %
and "SavePathChange(p{1,1},p)" to save

                                           % 'pnp' - 'squ

R            = 100e-3;                  % Radius
(for circular trajectory)
CircRef      = [0 2*pi];                % Initial and
//
CircTime     = 2;                       % Trajectory 1
//

tr0          = [0    0.5 ];

%% Define and attribute paths

if strcmp(pathopt, 'circ')
    dT          = CircTime + exttime;
    Tmov        = 0:dt:CircTime;
    T           = 0:dt:dT;
    [Xf,c,tbound,totinter,c0] = GenerateCircularPath(R,CircRef,Tmov,T,tr0);
    Xr          = [];
    mainpath.Tmov = Tmov;
else
    [Xr,tr,anull] = LoadPath(pathopt,ws,dt,exttime);
    Xr0           = [zeros(3,1), Xr(:,1)];
    dT            = tr(length(tr));
    T             = 0:dt:dT;
    [Xf,c,tbound,totinter,c0] = GeneratePath(Xr0,tr0,T,anull,dt,Xr,tr);
end

mainpath.T      = T';
mainpath.R      = R;

```

```
mainpath.c                = c;
mainpath.tbound            = tbound;
mainpath.totinter         = totinter;
mainpath.Xf               = Xf;
mainpath.PathType         = pathopt;
mainpath.X                = Xfun(T,mainpath);
mainpath.Xr               = Xr;
mainpath.UPM              = UserProvMotion;
mainpath.trunc            = mainpath;
mainpath.trunc.tbound(end,2) = T(end) - exttime;

initpath.c{1}             = c0;
initpath.tbound           = tr0;
initpath.totinter         = 1;
initpath.PathType         = 'regular';
```

```

function control = ContMotConfig(robot,model,mpath,ipath)

%% Load Variables
l    = robot.l;
Nlt = sqrt(l(2)^2 + l(3)^2);           % AC distance with

%% Motion Planning
MotionPlanningOn      = 5;             % 0 Non-Redundant
| 1 Grad 1st order (RG)                % 2 Grad 2nd order
|                                     % 4 User Provided
|                                     % 5 DDP unique
|                                     % 7 GPM
| 6 DDP separated (not implemented)

%
%          +----- Distance
%          |          +----- Single
%          |          |          +----- Divide
%          |          |          |          +----- Limitat
%          |          |          |          |          +----- Single
%          |          |          |          |          |          +-- Conditio
%          |          |          |          |          |          |
PartiaLeadingFunctions = [1      0      0      1      1      0
]; % Choose Partial Leading Functions
C      = [1.0e-0  3.0e-2  1e-2   1e-3   1e-2   1.0e-2
]; % Gains of each PLF  MP1
C(2,:) = [1.0e3   1.5     0.1    0      0e-3   0.0e-3
]; %      MP2
C(3,:) = [0.0e3   0.0     0.0    0      0e-3   5.0e-3
]; %      MP3 (Max Torque = 1)
C(5,:) = [6.0e-1  0.0e-2  0e-2   1e-3   1e-2   1.0e-2
]*2.5e1; %      MP5
C(6,:) = [0.0e0   3.0e-2  1e-2   1e-3   0e-3   0.0e-3
]; %      MP6
C(7,:) = [0.0e1   0.0e-3  0e-1   1e-3   3e-4   0.0e-3
]; %      MP7
%=====
DDP_weigths(1,:) = [false   false   false   false   1.1e-3   2.0e+1
]; % Weigths for DDP - MP5
DDP_weigths(2,:) = [3.0e0   3.0e-2  1e-2   1e-3   1e-0   false
]; %      - MP6

```

```

%                                     |      |
%                                     |      +---- Unique ref
%                                     +----- Accelerati

DDP_dt_multiplier      = 1e+0*(mpath.T(end)-mpath.T(1)); % Reduce time step dur
DDP_FinalVelocityWeight = 3e3*DDP_weigths(1,end-1); % Extra penalty for fin
DDP_acc_penalty        = 1e16; % Penalty constant - f
DDP_accept_iteration   = .1; % Proportion of expect
DDP_multiplier_alpha   = .5; % Multiply alpha if
DDP_regularization_mu0 = [0e0 1e0]; % Initial regularization
DDP_regularization_rmu = [1e2 1e1 3e1 7e0]; % Regularization multipl
DDP_Final_Tolerance    = 1e-10; % Final tolerance for l
DDP_Optimize_Initial_Z0 = true; % Optimiza initial zet
Cd                     = 130; % Dissipative Factor
Cl                     = [13 3 ]; % Limiting constant (zr
ACtarget               = Nlt*1.2; % Distance between Ai (
AClimiterConst         = 1e-3; % Constant AC limiting
ZDivLim                = 0.90; % Maximum use of ZDiv
ws                     = model.ws; % Defined on model
fdem                   = (1/(ws-2))*ones(1,ws); % Moving average filter
ODEsolver              = @ode23t; % Solver used in Motion
dT0                    = model.dT0; % Defined on model
ChainConfig            = +1; % Which side theta (+1
zeta0                  = -ChainConfig*.05*ones(1,3);
Dzeta0                 = zeros(1,3);
PrintMotPlan           = {false, false}; % Animate Motion Plann
IntervalDotDisp        = 35; % Number of Dots displ
mainodeoptions          = odeset('outputfcn',@MainMotPlanDisp);
initodeoptions          = odeset('outputfcn',@InitMotPlanDisp);
FKinOptionsTolF        = 1e-8; % Tolerances for Fkin
FKinOptionsTolX        = 1e-8; %
%          1      2      3
LinearActuators         = [ 2 , 2 , 2 ]; % Linear Actuators: '0
% JUST FOR

"MP == 2"

if strcmp(version,'7.10.0.499 (R2010a)') % Truncated Optimizati
    OptOptionsMP3 = optimset('OutputFcn', @outfun,'algorithm','sqp',...
        'tolfun',1e-8,'tolx',1e-8,'display','off');
else
    OptOptionsMP3 = optimoptions(@fmincon,'OutputFcn', @outfun,...
        'algorithm','sqp','display','off');

```

```
end

%% Control
MaxForce      = 1000.0;           % Maximum Force
[ N ]
MaxTorq       = 3.00;            % Maximum Torque
ControlOn     = 1;
ControlFdFrwd = 'Linear';        % Feed Forward
%

'CalculatedTorques'
P  = 10.50*[5.00e3  1.08e1]*ControlOn; % Feed back gain
I  = 20.50*[6.50e3  2.28e1]*ControlOn;
D  = 2.00*[8.80e3  1.50e0]*ControlOn;
VFF = 0.00*[8.80e3  2.16e0];          % Feed Forward
AFF = 1.00*[8.00e2  1.00e-1];
```