



UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

UM NOVO MÉTODO PROBABILÍSTICO DE CINEMÁTICA INVERSA
BASEADO NA RRT EM UM ESPAÇO DE DIMENSÃO REDUZIDA

MATHEUS CARDOSO SANTOS

São Cristóvão-SE, Brasil

Janeiro de 2021



UM NOVO MÉTODO PROBABILÍSTICO DE CINEMÁTICA INVERSA BASEADO NA RRT EM UM ESPAÇO DE DIMENSÃO REDUZIDA

MATHEUS CARDOSO SANTOS

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica – PROEE, da Universidade Federal de Sergipe, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica

Orientador: Lucas Molina

Co-orientador: Elyson Adan Nunes Carvalho

São Cristóvão-SE, Brasil

Janeiro de 2021



**UNIVERSIDADE FEDERAL DE SERGIPE
PRÓ-REITORIA DE PÓS-GRADUAÇÃO E PESQUISA
COORDENAÇÃO DE PÓS-GRADUAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA-PROEE**

TERMO DE APROVAÇÃO

"UM NOVO MÉTODO PROBABILÍSTICO DE CINEMÁTICA INVERSA BASEADO NA RRT EM UM ESPAÇO DE DIMENSÃO REDUZIDA"

Discente:

Matheus Cardoso Santos

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Sergipe, como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica.

Aprovada pela banca examinadora composta por:

Prof. Dr. Jugurta Rosa Montalvão Filho (PROEE/UFS)
Presidente

Prof. Dr. Sidney Nascimento Givigi Junior (PROEE)
Examinador Interno

Prof. Dr. Francisco Guido Rossomando (INAUT / UNSJ)
Examinador Externo

Matheus Cardoso Santos
Candidato

Cidade Universitária "Prof. José Aloísio de Campos", 26 de janeiro de 2021.

**FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL
UNIVERSIDADE FEDERAL DE SERGIPE**

Santos, Matheus Cardoso
S237n Um novo método probabilístico de cinemática inversa baseado
na rrt em um espaço de dimensão reduzida / Matheus Cardoso
Santos ; orientador Lucas Molina. - São Cristóvão, 2021.
69 f. : il.

Dissertação (mestrado em Engenharia Elétrica) – Universidade
Federal de Sergipe, 2021.

1. Engenharia Elétrica. 2. Cinemática. 3. Robótica. I. Molina,
Lucas, orient. II. Título.

CDU 621.3

Agradecimentos

O ano de 2020 foi marcado pela dor global em virtude das inúmeras mortes e marcas deixadas pela COVID-19, minhas condolências à todos afetados por esse terrível momento. O Criador nos adverte das aflições que teremos nesse mundo, mas nos ensina a termos bom ânimo e sermos gratos a Ele em todos os momentos. Por isso sou grato à Deus pela graça e pelas pessoas maravilhosas que Ele colocou em minha vida.

Agradeço ao meu pai Ronaldo e a minha mãe Siziana por me ensinarem o caminho à trilhar e estarem sempre presente. De fato, tudo que tenho e que sou vem dos preciosos ensinamentos que vocês me deram. Ainda na mesma casa, agradeço aos meus avós Zé de sá e Degenalva e aos meus tios Junior Sá e Joana, sem esquecer de Yane.

Ao meu irmão Phillipe por ser referência pra mim em tudo na vida, desde as roupas que eu pego "emprestado" pra usar até a vida de pesquisador. Também à Stephanie Kamarry por ser referência desde que eu dei meus primeiros passos no ramo da pesquisa.

Agradecimento especial à minha namorada, Amanda Oliveira. Durante toda a trajetória estive comigo sendo o suporte e ponto de equilíbrio em todos os momentos. Obrigado por aceitar trilhar comigo esse caminho chamado vida. Te amo!

Ao corpo docente do GPRUFS, pois, apesar de ser orientado por Lucas e da grande contribuição do mesmo para esse trabalho, os valores e princípios que venho aprendendo desde 2015 com todos vocês são o que eu mais vou carregar pra vida. Vocês conseguiram despertar o meu interesse pela robótica ainda no 1º ano do ensino médio e me fizeram seguir o caminho da pesquisa com toda paixão e dedicação que mostram ao orientar seus alunos. Aos professores Lucas, Elyson, Eduardo e Gilmar, Muito Obrigado.

Ainda no GPRUFS, agradeço a todos aqueles que passaram pelo grupo durante esse período. Vale mencionar a galera que teve sempre comigo nas noites viradas no laboratório, Raphael, Lázaro, Matheus, Cone e Thiago (9x1 cheque).

À todos os amigos que se alegram comigo e fazem parte da minha história, Obrigado! Amo vocês!

Por fim, agradeço ao apoio financeiro da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES.

Resumo da Dissertação apresentada ao PROEE/UFS como parte dos requisitos necessários para a obtenção do grau de Mestre (Me.)

UM NOVO MÉTODO PROBABILÍSTICO DE CINEMÁTICA INVERSA BASEADO NA RRT EM UM ESPAÇO DE DIMENSÃO REDUZIDA

MATHEUS CARDOSO SANTOS

Janeiro/2021

Orientador: Lucas Molina

Co-orientador: Elyson Adan Nunes Carvalho

A evolução dos robôs manipuladores tem aumentado a complexidade de seus modelos e aplicações, exigindo que os métodos de cinemática inversa que integram seus sistemas de controle possuam características como rápida convergência, completude, baixo custo computacional e capacidade de evitar mínimos locais e singularidades. Possuindo muitas dessas vantagens, o método de busca probabilística conhecido como RRT (do inglês: *Rapidly Random Tree*), amplamente utilizado na área de planejamento de movimento, também pode ser considerado no contexto da cinemática inversa. Na maioria dos trabalhos em RRT, a árvore é aplicada no espaço de configurações, o que resulta em um espaço de dimensão elevada para grande parte dos manipuladores. Neste trabalho de mestrado é proposta uma técnica de cinemática inversa baseada na RRT aplicada diretamente no espaço de trabalho, o que implica um espaço de busca de dimensão reduzida, ao representar os nós da árvore como juntas e as arestas como elos do manipulador. A mudança de espaço realizada possibilita uma interpretação intuitiva dos parâmetros da árvore, o que facilita a alteração das diferentes variáveis que compõem o algoritmo clássico da RRT. Através dessa intuitividade, foi desenvolvida uma polarização do modelo de probabilidade em direção as regiões que contém todas as possíveis soluções para o problema de cinemática inversa a fim de obter um menor tempo de convergência sem perder sua completude probabilística. Tais características são avaliadas através de experimentos simulados.

Palavras-chaves: RRT, cinemática inversa, busca probabilística, robótica de manipuladores.

Abstract of Dissertation presented to PROEE/UFS as a partial fulfillment of the requirements for the degree of Master

A NEW PROBABILISTIC METHOD OF INVERSE KINEMATICS BASED ON RRT IN A SPACE OF REDUCED DIMENSION

MATHEUS CARDOSO SANTOS

January/2021

Supervisor: Lucas Molina

Co-supervisor: Elyson Adan Nunes Carvalho

The evolution of manipulator robots has increased the complexity of their models and applications, requiring that inverse kinematics methods integrated into their control systems have characteristics such as fast convergence, completeness, low computational cost, and the ability to avoid minimal locations and singularities. The probabilistic search method known as Rapidly Random Tree (RRT) is widely used in the area of motion planning and can also be considered in the context of inverse kinematics. In most RRT literature, a tree is applied in the configuration space, which results in a large space for most manipulators. In this thesis, an inverse kinematics technique based on RRT applied directly to the workspace is proposed, which yields a reduced-dimension search space, by representing the tree nodes as joints and the edges as links of the manipulator. This space change allows an intuitive interpretation of the tree parameters, which facilitates the alteration of the different variables that make up the classic RRT algorithm. Through this intuitiveness, a polarization direction in the probability model was developed towards the regions which contain all possible solutions to the problem of inverse kinematics in order to get a lower convergence time without losing its probabilistic completeness. Such characteristics are evaluated through simulated experiments.

Key-words: RRT, inverse kinematics, probabilistic search, manipulator robotics.

Lista de ilustrações

Figura 1 – Estrutura básica de um robô manipulador. Imagem adaptada de (PAZOS, 2002).	5
Figura 2 – Configuração do manipulador descrita pelo vetor $\mathbf{q} = (\theta_1, \theta_2, d_3, \theta_4, \theta_5)$. Imagem adaptada de (KUMAR, 2020).	6
Figura 3 – Sistemas de coordenadas de cada elo em um manipulador planar de 2-DOF.	7
Figura 4 – Interpretação visual das matrizes da equação (2.6).	10
Figura 5 – Sistemas de coordenadas posicionados de modo a satisfazer DH1 e DH2 (SPONG; HUTCHINSON; VIDYASAGAR, 2020).	11
Figura 6 – Planejamento para robô manipulador de 2-DOF se mover da configuração inicial (q_{init}) até a configuração desejada (q_{goal}). (a) Tarefa no espaço de trabalho, (b) Tarefa no $\mathcal{C}_{space}$. Imagem adaptada de (LATOMBE, 2012).	12
Figura 7 – Configuração inicial do robô (q_{init}) e configuração desejada (q_{goal}) no espaço de configurações.	14
Figura 8 – Conexão entre a amostra aleatória (q_{rand}) com o nó mais próximo (q_{near}).	15
Figura 9 – Adição do novo nó (q_{new}) conforme a limitação do crescimento da árvore por um tamanho de passo constante (d).	15
Figura 10 – Resultado da RRT clássica após conectar as configurações q_{init} e q_{goal} . O caminho gerado entre as duas configurações é destacado em verde e a árvore gerada é representada pelos pontos azuis (nós) e linhas pretas (arestas).	16
Figura 11 – Resultado do algoritmo clássico da RRT (em vermelho) no $c-space$. (a) árvore no $c-space$, (b) configuração gerada para alcançar o marcador amarelo.	28
Figura 12 – Sequência da metodologia proposta. (a) Sorteio de um novo ponto x_{rand} . (b) Encontrar o ponto mais próximo e limitar o crescimento pelo tamanho do passo que equivale ao elo correspondente do manipulador. (c) O nó x_{novo} é adicionado com índice 1. (d) O nó de índice 2 é adicionado conectado ao nó imediatamente anterior (índice 1). (e) Após o último elo sorteado é feita a checagem de solução, caso não seja possível chegar ao destino o índice retorna para 0 e o algoritmo continua. (f) Árvore após várias iterações com os índices 1,2,3 e 4 sendo representados pelas cores vermelho, verde, azul e preto, respectivamente.	30

Figura 13	–Crescimento da árvore após 3 iterações para um manipulador planar de 3-DOF. Os nós de índice 1, 2 e 3 são caracterizados pelas cores vermelho, verde e azul, respectivamente.	31
Figura 14	–Amostragem segundo a abordagem clássica. O ponto vermelho representa o destino (x_{dest}), em verde estão representadas as amostras sorteadas (x_{rand}), a linha pontilhada representa a conexão com o nó de índice imediatamente inferior mais próximo (x_{near}) e os pontos em azul representam os novos nós adicionados à árvore (x_{novo}).	33
Figura 15	–Histograma gerado a partir da amostragem no espaço de trabalho. . . .	34
Figura 16	–Regiões de interesse em torno do destino (em amarelo). As cores dos círculos correspondem a cor da junta no manipulador. Deste modo, o raio do círculo equivale ao tamanho restante do manipulador a partir da junta correspondente.	35
Figura 17	–Amostragem em direção as regiões de interesse. (a) Os pontos em verde estão representadas as amostras sorteadas (x_{rand}) e os pontos em azul representam a projeção dos pontos no círculo da junta em questão, (b) os pontos em roxo representam os novos nós adicionados à árvore (x_{novo}).	35
Figura 18	–Histograma gerado a partir da amostragem no espaço de trabalho seguida de uma projeção na região de interesse.	36
Figura 19	–Amostragem na superfície da região de interesse. O ponto vermelho representa o destino (x_{dest}), em verde estão representadas as amostras sorteadas na superfície (x_{rand}) e os pontos em azul representam os novos nós adicionados à árvore (x_{novo}).	37
Figura 20	–Histograma gerado a partir da amostragem na superfície da região de solução.	37
Figura 21	–Iteração completa da RRT no espaço de trabalho com a regra de amostragem na superfície da região de solução. (a) Ponto de destino (em amarelo), base como nó de índice 0 (em preto) e região de solução (círculo azul). (b) sorteio de índice 1 na região de solução e adição do novo nó x_{novo} a árvore com tamanho de passo limitado ao tamanho do elo d_1 , (c) sorteio de índice 2 na região de solução e adição do novo nó x_{novo} , (d) sorteio de índice 3 na região de solução e adição do novo nó x_{novo} , (e) sorteio de índice 4 como o próprio destino pois ele é a única região de solução e (f) fim da iteração.	39
Figura 22	–Regra de conexão para RRT clássica.	40
Figura 23	–A regra de conexão proposta nesse trabalho delimita um círculo de raio d_i e busca o nó mais próximo do círculo x_{perto} . Caso a regra de conexão da RRT clássica fosse utilizada, o nó x'_{perto} seria o escolhido.	40

Figura 24 – Ambiente de experimento para o robô manipulador de 3-DOF. Os pontos em preto representam os pontos de destino distribuídos uniformemente no espaço alcançável.	43
Figura 25 – Ambiente de experimento para o robô manipulador de 10-DOF. Os pontos em preto representam os pontos de destino distribuídos uniformemente no espaço alcançável.	44

Lista de tabelas

Tabela 1	– Parâmetros de Denavit-Hartenberg do manipulador planar (4-DOF).	38
Tabela 2	– Descrição das abordagens da RRT analisadas nesse trabalho.	41
Tabela 3	– Descrição das alterações da RRT analisadas nesse trabalho.	41
Tabela 4	– Parâmetros de Denavit-Hartenberg que descreve o modelo do robô manipulador de 3-DOF utilizado nos experimentos.	42
Tabela 5	– Parâmetros de Denavit-Hartenberg que descreve o modelo do robô manipulador de 10-DOF utilizado nos experimentos.	43
Tabela 6	– Desempenho em média das variantes da RRT para um manipulador de 3-DOF.	45
Tabela 7	– Desempenho em média das variantes da RRT para um manipulador de 10-DOF.	46
Tabela 8	– Desempenho em média das variantes da RRT para um manipulador de 3-DOF levando em consideração todas as alterações propostas nesse capítulo.	47
Tabela 9	– Desempenho em média das variantes da RRT para um manipulador de 10-DOF levando em consideração todas as alterações propostas nesse capítulo.	48
Tabela 10	– Desempenho em média da MB-RRT para um manipulador planar de 3-DOF com diferentes valores de tolerância.	48
Tabela 11	– Desempenho em média da MB-RRT para um manipulador planar de 10-DOF com diferentes valores de tolerância.	49

Sumário

Lista de ilustrações	VII
Lista de tabelas	X
1 Introdução	1
1.1 Objetivos	4
1.2 Estrutura do trabalho	4
2 Fundamentação Teórica	5
2.1 Cinemática Direta	5
2.2 Notação de Denavit-Hartenberg	9
2.3 Espaço de Configurações (<i>C-Space</i>)	11
2.4 RRT	13
3 Revisão Bibliográfica	17
3.1 <i>Rapidly-exploring Random Tree</i> (RRT)	17
3.2 Cinemática inversa	21
4 Desenvolvimento do Trabalho e Abordagem Proposta	27
4.1 RRT no <i>C-Space</i>	27
4.2 RRT no espaço de trabalho	28
4.3 Amostragem	32
4.4 Regra de Conexão	39
5 Resultados e Discussão	41
5.1 Configurações dos Experimentos	42
5.2 Amostragem	44
5.2.1 Experimento 3-DOF	44
5.2.2 Experimento 10-DOF	45
5.3 Regra de Conexão e Amostra no destino	46
5.3.1 Experimento 3-DOF	46
5.3.2 Experimento 10-DOF	47
5.4 <i>Manipulator-Based Rapidly-Exploring Random Tree</i> (MB-RRT)	48
6 Conclusões	50
Referências	53

1 Introdução

A evolução da automação industrial acarretou o avanço em pesquisas na área da robótica, devido à facilidade dos robôs se encaixarem em processos de produção por meio da execução de tarefas com eficiência e precisão, de forma a aumentar a capacidade de produção e reduzir os custos. Esses avanços foram responsáveis por tornar rotineira a imagem de robôs manipuladores sendo usados para soldar e pintar carrocerias de carros em linhas de montagem, inspecionar e reparar estruturas em ambientes nucleares e até na realização de cirurgias (SIEGWART; NOURBAKHS; SCARAMUZZA, 2011).

A aplicação crescente de robôs manipuladores nos diversos campos da indústria, na medicina e mesmo no cotidiano, faz com que os requisitos de eficiência, segurança, autonomia e capacidade de resposta rápida a mudanças no ambiente sejam cada vez maiores, o que impulsiona o desenvolvimento de pesquisas na área para lidar com esses novos desafios. Essas atividades exigem dos pesquisadores um entendimento da cinemática desses robôs, isto é, a relação entre a posição do seu atuador final e suas articulações.

O movimento dos robôs manipuladores é realizado através de suas articulações, também chamadas de juntas, que podem realizar movimentos de rotação ou translação. Cada junta tem associada a si um valor que representa a sua movimentação, sendo que ele é descrito em termos de ângulos para os movimentos rotacionais e deslocamentos para movimentos de translação. Com base nesses valores é possível determinar a posição de cada uma das junta e do atuador final em seu espaço alcançável, ou espaço de trabalho, no processo conhecido como cinemática direta. Como essa transformação possui apenas uma solução possível, o problema de cinemática direta é frequentemente resolvido com o uso da convenção de Denavit-Hartenberg (SPONG et al., 2006).

Por outro lado, quando a situação consiste em determinar uma configuração que leve o atuador final do corpo, também chamado de efetuador, para um determinado ponto no espaço de trabalho, encontrar uma solução não é um processo simples devido ao fato de que pode existir mais de uma solução possível. Tal problema é conhecido como cinemática inversa (IK, do inglês: *inverse kinematics*) e ainda é um desafio para pesquisadores dependendo da quantidade de juntas e das restrições do manipulador (JUNIOR et al., 2018).

Devido à sua capacidade de encontrar configurações que levem o efetuador para determinados pontos no espaço cartesiano, a cinemática inversa é essencial em tarefas como o planejamento de movimento, geração e controle de trajetórias (ROKBANI; CASALS; ALIM, 2015). Vale ressaltar que apesar da atuação nas tarefas de geração e controle de

trajetória, a solução do problema de cinemática inversa não consiste em um caminho entre duas configurações, mas somente uma configuração final na qual o efetuador é posicionado em um ponto de destino no espaço.

Desde a década de 60, quando foram publicados os primeiros trabalhos na área (WHITNEY, 1969) até o presente, vários estudos e métodos para realização de cinemática inversa foram publicados. De modo geral, as abordagens desenvolvidas para o problema de IK tem foco em problemas específicos, como desempenho computacional, prevenção de situações de conflito ou problemas de singularidade, produção de transições suaves sem oscilações, suporte de restrições antropomórficas e de contato, além da capacidade de atender a objetivos secundários, como desvio de obstáculos, controle de orientação, controle de movimento, dentre outros (ARISTIDOU et al., 2018).

Apesar de não haver taxonomia única que seja amplamente aceita para o métodos de cinemática inversa, é possível classificá-los em quatro grandes categorias: analíticos, numéricos, baseado em dados e híbridos (ARISTIDOU et al., 2018). Entretanto, os trabalhos que utilizam essa classificação não mencionam um conjunto de abordagens classificadas como meta-heurísticas (MH) (SAVSANI; JHALA; SAVSANI, 2014). Essa categoria tem como principais destaques o enxame de colônias de formigas (ABC, do inglês: *Ant Bee Colony*), algoritmo vagalume (FA, do inglês: *Firefly Algorithm*) e o enxame de partículas (PSO, do inglês: *Particle Swarm Optimization*).

As principais características das abordagens MH é que tentam resolver o problema de cinemática inversa com operações que imitam comportamentos da natureza e possuem uma componente aleatória, uma vez que se baseiam no uso de buscas probabilísticas. Essa busca é feita no espaço de configurações e cada amostra possui um valor associado, calculado através de uma função de custo, que indica quão próxima do objetivo a posição do efetuador estaria no espaço de trabalho. Os métodos de MH são recomendados para resolução de problemas complexos e apresentam a vantagem de não caírem em singularidades, pelo fato de não inverterm a matriz jacobiana (AYYILDIZ; ÇETINKAYA, 2016).

Entretanto, como essa classe de métodos é baseada em algum tipo de heurística, muitos desses métodos podem cair no problema de mínimo local (JUNIOR et al., 2018). Outra desvantagem dessa família é que a busca probabilística é feita no espaço de configurações, que não permite a existência de uma configuração de destino, pois, encontrar uma configuração que satisfaça o objetivo traçado no espaço de trabalho é a função do método IK. Além disso, esse espaço geralmente possui dimensão elevada, o que faz com que as abordagens MH não converjam rapidamente para uma solução quando comparadas com outros métodos de IK que atuam diretamente no espaço de trabalho (WICHAPONG; BUREERAT; PHOLDEE, 2018).

A dificuldade em lidar com espaços de dimensões elevadas é recorrente nas mais variadas aplicações. Por isso, diversos autores propuseram técnicas de redução de dimensionalidade com o objetivo de reduzir a complexidade do problema, tornar a solução do problema mais eficiente e melhorar o entendimento dos dados (CARREIRA-PERPINÁN, 1997; JUNIOR et al., 2018). Desta forma, no contexto em que as técnicas probabilísticas operam em um espaço de dimensão maior que outras abordagens, a redução da dimensionalidade pode fazer com que o tempo de convergência se equipare a outros métodos de cinemática inversa.

Na área da robótica, uma das técnicas probabilísticas mais conhecida é a RRT (do inglês: *Rapidly-exploring Random Tree*), criada para resolução do problema de planejamento de movimento (LAVALLE, 1998). A RRT ganhou destaque devido à sua capacidade de encontrar soluções rápidas em grandes espaços de buscas, além de ser probabilisticamente completa, ou seja, quando uma solução existe, a RRT sempre a encontra se o número de amostras tende ao infinito (SANTANA et al., 2018). Alguns trabalhos na área de manipuladores utilizam a RRT no espaço de configurações para solução do problema de planejamento (RYBUS; SEWERYN, 2015; WEI; REN, 2018).

Entretanto, para que a busca probabilística possa ter um impacto significativo no tempo de convergência da resolução do problema de cinemática inversa, a RRT pode ser aplicada em um espaço de dimensão reduzida, como o espaço de trabalho. Isso se deve ao fato que a quantidade de dimensões no espaço de configurações é equivalente ao número de juntas de manipulador. Logo, a aplicação da RRT no espaço de trabalho garante que o espaço de busca pode ter um máximo de 3 dimensões, para uma tarefa de posição do efetuador, resultando num menor espaço de busca.

Outra forma de se obter um impacto significativo no tempo de convergência da busca é aplicar técnicas de polarização para enviesar o crescimento da árvore. Entretanto, a aplicação dessas técnicas, com o uso do espaço de configurações, exige um mapeamento não-linear entre os sistemas que causa distorções e dificulta o uso de métricas comuns de distância, como a euclidiana. Além disso, técnicas de polarização podem levar a mínimos locais e comprometer a completude probabilística das abordagens.

Nesse trabalho de mestrado, foi proposto um método de cinemática inversa para manipuladores baseado na RRT focado em aspectos como rápida convergência, prevenção de mínimos locais e singularidades, simplicidade e completude probabilística. Apesar dessas vantagens, a principal contribuição da abordagem proposta é a aplicação da busca probabilística diretamente no espaço de trabalho, o que implica o uso de um espaço de dimensão reduzida e uma interpretação intuitiva de todas as etapas da abordagem. Para possibilitar essa mudança, é considerado que o crescimento da árvore equivale à construção do manipulador no espaço de trabalho, de forma que os nós representam as juntas e as arestas os elos do robô. Desta forma, ela facilita a visualização da relação entre a árvore

e as características físicas do manipulador, o que favorece a proposição de alterações na abordagem a fim de otimizar os resultados. Baseado na intuitividade obtida, é proposta uma alteração no modelo de probabilidade a fim de polarizar o crescimento da árvore sem comprometer sua completude probabilística. Além disso, o método proposto permite ainda incorporar aspectos específicos das inúmeras extensões da RRT para obter um melhor desempenho ou resolver objetivos secundários.

1.1 Objetivos

O objetivo desse trabalho é desenvolver um método probabilístico de cinemática inversa para manipuladores aplicado em um espaço de dimensão reduzida, tendo como principais características simplicidade, rápida convergência e completude probabilística.

São objetivos específicos deste trabalho:

- Realizar uma ampla revisão bibliográfica acerca dos métodos de cinemática inversa e da RRT;
- Analisar as alterações que podem ser feitas na RRT clássica a partir da nova interpretação no espaço de trabalho;
- Investigar alterações no modelo de probabilidade e viabilidade de polarização;
- Desenvolver melhorias na abordagem proposta baseadas nas investigações realizadas e implementá-las utilizando simulação;
- Comparar o impacto de cada alteração proposta através de experimentos com o algoritmo clássico da RRT.

1.2 Estrutura do trabalho

Esta dissertação está estruturada da seguinte maneira: no Capítulo 3, é apresentada uma revisão acerca das técnicas de cinemática inversa, bem como da RRT; no Capítulo 2, é apresentada a fundamentação teórica referente à modelagem matemática dos robôs manipuladores e a aplicação da RRT no espaço de configurações; no Capítulo 4 é apresentada a abordagem proposta e os aspectos investigados; no Capítulo 5, os resultados são apresentados e discutidos; e, por fim, a conclusão é apresentada no Capítulo 6.

2 Fundamentação Teórica

Neste capítulo, são detalhados os conceitos da modelagem matemática usada para descrever o movimento de um manipulador considerando apenas o ângulo das juntas. Esse tipo de modelo é chamado de modelo cinemático. Junto com a modelagem, também é introduzido o conceito de espaço de configurações e é detalhada a aplicação clássica da RRT.

2.1 Cinemática Direta

Os robôs manipuladores são caracterizados pela existência de uma base fixa e uma sequência de corpos ligados por intermédio de articulações, de modo que, na área da robótica os corpos são chamados de elos e as articulações de juntas. Esses robôs possuem em sua extremidade um efetuator que é responsável por interagir com o ambiente, como uma ferramenta de solda ou uma garra (SPONG et al., 2006). É apresentada na Figura 1 a estrutura de um robô manipulador destacando seus elos, juntas e efetuator.

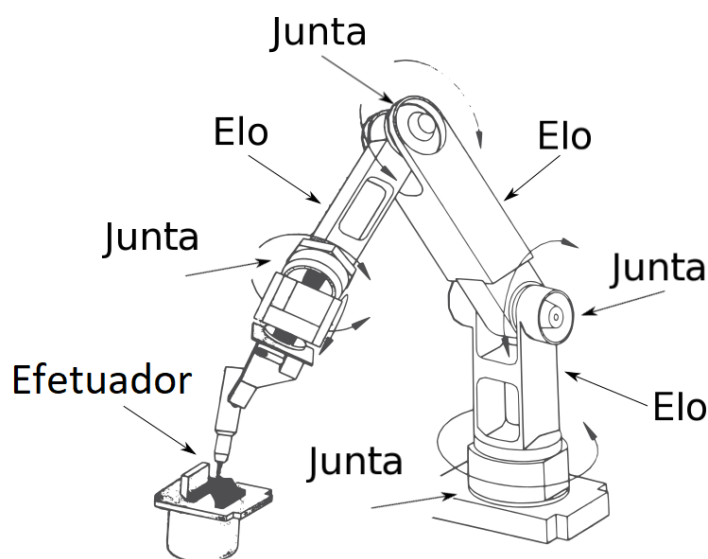


Figura 1 – Estrutura básica de um robô manipulador. Imagem adaptada de (PAZOS, 2002).

Para fins de análise, um robô manipulador pode ser considerado como uma cadeia cinemática, uma vez que ele é composto por um conjunto de elos ligados entre si por articulações (SPONG et al., 2006). Essas articulações podem ser classificadas como juntas de revolução, quando a articulação faz um movimento de rotação, ou prismática, quando

o movimento é de translação. Com a premissa de que cada uma das juntas tem um único grau de liberdade (DOF, do inglês: *degrees of freedom*), a ação de cada articulação pode ser descrita por um único número real

$$q_i = \begin{cases} \theta_i, \text{revolução} \\ d_i, \text{prismática} \end{cases}. \quad (2.1)$$

Após obter as variáveis de todas as juntas, é possível determinar a localização de cada ponto do manipulador. Sendo assim, o conjunto de valores de todas as juntas consegue expressar a posição do manipulador no espaço onde o robô pode executar seus movimentos, que é chamado de espaço de trabalho.

Dado um robô manipulador com n juntas, sua posição completa pode ser representada por um vetor

$$\mathbf{q} = (q_1, q_2, \dots, q_n), \quad (2.2)$$

que é chamado de configuração (SPONG et al., 2006). Na Figura 2 é apresentada a descrição de uma configuração de um manipulador de 5 juntas, ou 5-DOF, que possui 4 juntas de revolução ($\theta_1, \theta_2, \theta_4, \theta_5$) e uma junta prismática (d_3).

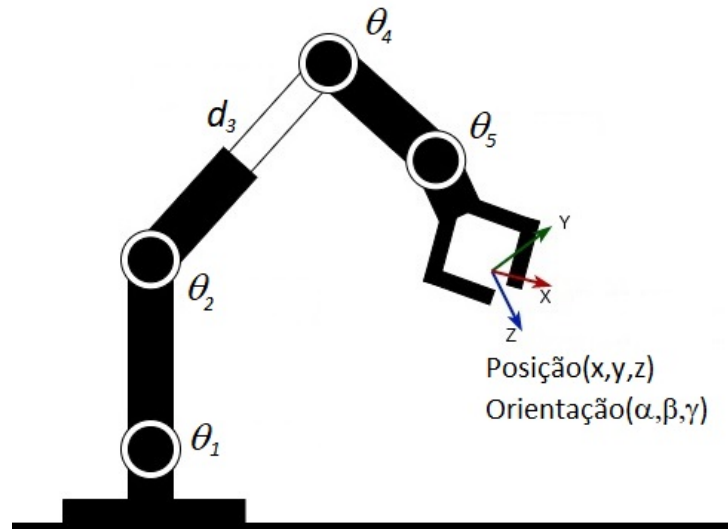


Figura 2 – Configuração do manipulador descrita pelo vetor $\mathbf{q} = (\theta_1, \theta_2, d_3, \theta_4, \theta_5)$. Imagem adaptada de (KUMAR, 2020).

Uma vez que a posição de cada ponto do robô no espaço de trabalho é representada por uma configuração, o conjunto de todas as configurações possíveis para um manipulador forma um novo espaço, chamado de espaço de configurações. Desta forma, a partir de

um ponto no espaço de configurações $\mathbf{q}$ é possível descrever a posição e orientação do efetuador no espaço de trabalho e essa transformação de espaços é a base para o estudo cinemático do robô.

A modelagem cinemática de um manipulador é o estudo do movimento do mesmo, sendo que ela pode ser realizada de forma direta ou inversa. Na cinemática direta, o modelo gerado permite determinar a posição cartesiana de cada junta e do efetuador no espaço de trabalho com base no conhecimento do ângulo das juntas. Por outro lado, a cinemática inversa tem como objetivo encontrar uma configuração de valores das juntas correspondente a uma posição e orientação desejada para o manipulador.

Para realizar a análise cinemática direta de um robô manipulador, um sistema de coordenadas é associado a cada elo a partir da base, que pode ser considerada como o elo 0. Na Figura 3 é mostrado um exemplo da atribuição dos sistemas de coordenadas de cada elo para um manipulador planar de dois graus de liberdade (2-DOF). Através do exemplo citado, é fácil perceber que para expressar a posição e orientação do sistema de coordenadas do elo 2 é necessário aplicar operações de rotação e translação no sistema de coordenadas 1 sempre que a junta 1 for movimentada.

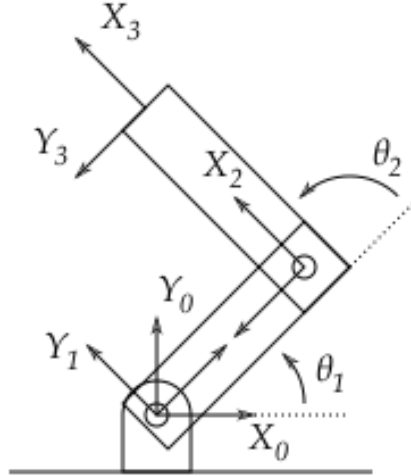


Figura 3 – Sistemas de coordenadas de cada elo em um manipulador planar de 2-DOF.

Dessa forma, é possível obter uma matriz homogênea de transformação, $\mathbf{A}_i$, que expressa a posição e orientação do sistema de coordenadas do elo i com relação ao elo anterior e que varia de acordo com a mudança da configuração do manipulador. Essa

matriz de transformação homogênea $\mathbf{A}_i$, é obtida através da composição de um vetor que represente a translação $\mathbf{d}_i^{i-1}$ e uma matriz que represente a rotação $\mathbf{R}_i^{i-1}$ entre esses sistemas, resultando na matriz

$$\mathbf{A}_i = \begin{bmatrix} \mathbf{R}_i^{i-1} & \mathbf{d}_i^{i-1} \\ 0 & 1 \end{bmatrix}. \quad (2.3)$$

Considerando o manipulador planar da Figura 3, tem-se a necessidade de expressar uma matriz homogênea relacionando elos que não sejam subsequentes. Logo, se o sistema de coordenadas 3 varia no momento em que a junta 1 realiza algum movimento, a matriz de transformação homogênea entre o sistema 0 e o sistema 3 irá mudar sem nenhuma variável que torne possível o cálculo da nova matriz. Desta forma, é necessário multiplicar as matrizes homogêneas $\mathbf{A}_1$, $\mathbf{A}_2$ e $\mathbf{A}_3$ para obter uma matriz de transformação, denotada por $\mathbf{T}_3^0$, que relaciona diretamente o sistema 0 com o sistema 3.

Com a matriz de transformação, tem-se a possibilidade de relacionar o sistema de coordenadas de um elo do manipulador a qualquer outro, ainda que não sejam subsequentes. Desta forma, é possível determinar a posição do efetuador em relação ao sistema da base. Sendo assim, considerando 0 como o sistema de coordenadas da base e n o sistema do efetuador, a cinemática direta é obtida a partir da matriz de transformação homogênea da base com relação ao efetuador

$$\mathbf{T}_n^0 = \begin{bmatrix} \mathbf{R}_n^0 & \mathbf{d}_n^0 \\ 0 & 1 \end{bmatrix}, \quad (2.4)$$

que é gerada ao percorrer a cadeia cinemática, de modo que a matriz $\mathbf{T}_n^0$ é calculada a partir do produto das matrizes homogêneas de cada elo

$$\mathbf{T}_n^0 = \mathbf{A}_1 \dots \mathbf{A}_n. \quad (2.5)$$

Portanto, a cinemática direta consiste em encontrar a matriz da equação (2.5), a fim de obter as informações do efetuador com base nas posições das juntas. Entretanto, esses cálculos são trabalhosos e intensificam-se à medida que o manipulador apresenta mais juntas, pois torna-se necessário encontrar uma matriz de transformação homogênea para cada elo e multiplicá-las.

Diante disso, existem algumas convenções que tornam a obtenção das matrizes de transformação homogênea um processo sistemático, sendo a notação de Denavit-Hartenberg a mais utilizada.

2.2 Notação de Denavit-Hartenberg

Utilizando os métodos matemáticos convencionais para modelar o manipulador, geralmente, é preciso criar uma matriz de transformação em que 6 variáveis são consideradas, sendo elas os três eixos de translação e de rotação. A notação de DH propõe a simplificação dessa matriz, onde o problema que continha 6 variáveis passa a apresentar apenas 4 variáveis.

Na convenção DH, a matriz homogênea da junta i pode ser representada como um produto de quatro transformações básicas (SPONG et al., 2006), como mostrado na equação (2.6),

$$\mathbf{A}_i = \mathbf{Rot}_{z_i, \theta_i} \mathbf{Trans}_{z_i, d_i} \mathbf{Trans}_{x_i, a_i} \mathbf{Rot}_{x_i, \alpha_i}, \quad (2.6)$$

em que:

- $\mathbf{Rot}_{z_i, \theta_i}$ representa uma rotação θ_i em torno do eixo z_i ;
- $\mathbf{Trans}_{z_i, d_i}$ representa um deslocamento d_i ao longo do eixo z_i ;
- $\mathbf{Trans}_{x_i, a_i}$ representa um deslocamento a_i ao longo do eixo x_i ;
- $\mathbf{Rot}_{x_i, \alpha_i}$ representa uma rotação α_i em torno do eixo x_i .

Na Figura 4 é apresentada uma interpretação visual das 4 matrizes da equação (2.6). A partir dessa interpretação pode-se obter os parâmetros θ_i , d_i , a_i e α_i que são chamados de parâmetros membros ou parâmetros DH, os quais recebem as seguintes denominações:

- θ_i é o ângulo da junta i , este ângulo é medido entre x_{i-1} e x_i em um plano normal ao z_{i-1} ;
- d_i é o elo de deslocamento, encontrado a partir da distância perpendicular a partir da origem da junta anterior o_{i-1} à intersecção do eixo x_i com z_{i-1} , medido ao longo do eixo z_{i-1} ;
- a_i é o elo de comprimento, encontrado a partir da distância entre os eixos z_{i-1} e z_i , e é medida ao longo do eixo x_i ;
- α_i é o elo de torção, encontrado a partir do ângulo entre os eixos z_{i-1} e z_i , medido em um plano normal a x_i . O sentido positivo para α é determinado a partir de z_{i-1} a z_i pela regra da mão direita.

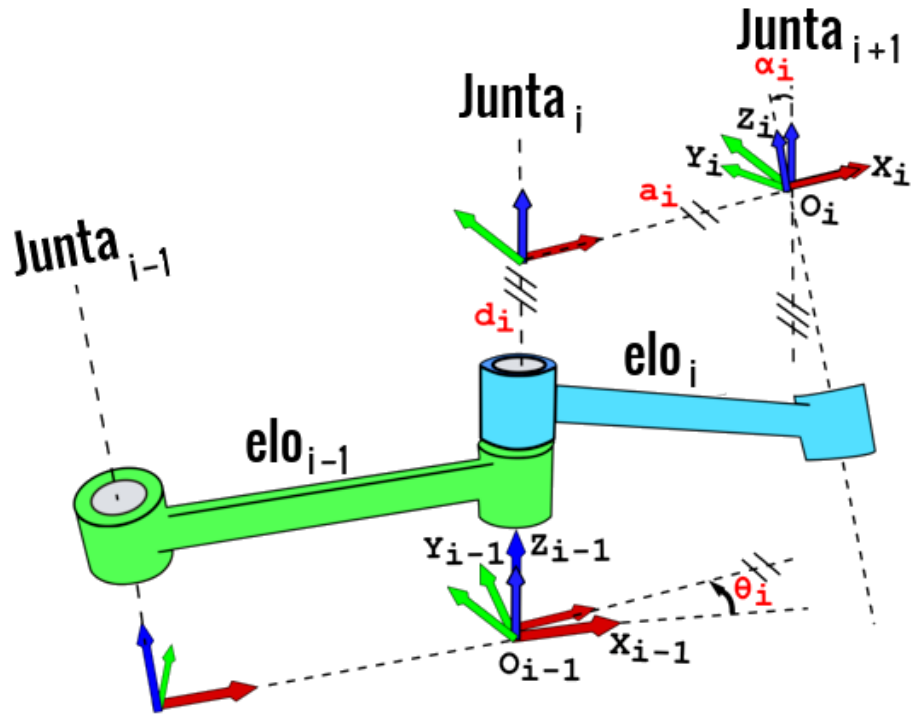


Figura 4 – Interpretação visual das matrizes da equação (2.6).

Desenvolvendo (2.6), obtém-se (os termos seno e cosseno foram simplificados para c e s , a fim de tornar a fórmula mais compacta e facilitar a visualização)

$$\mathbf{A}_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (2.7)$$

ou,

$$\mathbf{A}_i = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\alpha_i s\theta_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

que representa a versão simplificada da matriz de transformação homogênea. Entretanto, para que esta simplificação seja válida, é necessário respeitar duas regras no momento de posicionar os sistemas de coordenadas em cada junta, sendo elas:

- (DH1) O eixo x_i é perpendicular ao eixo z_{i-1} ;
- (DH2) O eixo x_i intercepta o eixo z_{i-1} ;

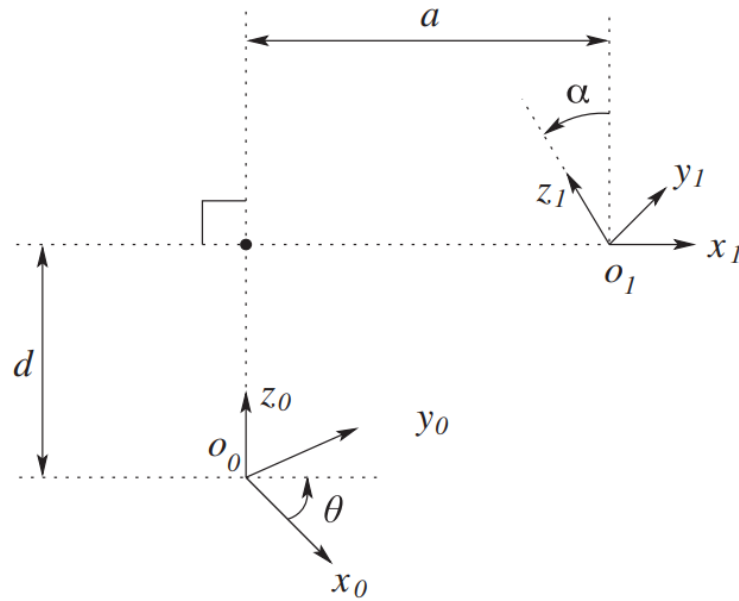


Figura 5 – Sistemas de coordenadas posicionados de modo a satisfazer **DH1** e **DH2** (SPONG; HUTCHINSON; VIDYASAGAR, 2020).

A fim de obter uma melhor visualização da aplicação das regras **DH1** e **DH2**, na Figura 5 é apresentado um exemplo de posicionamento de sistemas de coordenadas segundo a convenção de Denavit-Hartenberg.

Portanto, obedecendo a essas duas regras e com base nos valores das variáveis das juntas, é possível criar uma matriz de transformação que permite determinar a posição e orientação do efetuador para qualquer configuração. Ao obter a posição no espaço de trabalho do manipulador a partir do conjunto dos valores das variáveis de juntas, é possível abstrair as características físicas do manipulador e representá-lo como um ponto no espaço de configurações.

2.3 Espaço de Configurações (*C-Space*)

O conceito de espaço de configurações ($\mathcal{C}_{space}$, do inglês: *Configuration Space*) foi desenvolvido por Lozano-Pérez com o intuito de simplificar o problema de planejamento (LOZANO-PÉREZ; WESLEY, 1979; LOZANO-PÉREZ, 1990). A proposta é criar um espaço abstrato de n dimensões contendo todas as configurações associadas a poses possíveis para o manipulador no espaço de trabalho, sendo que a configuração $\mathbf{q} = (q_1, \dots, q_n)$ de um robô é uma representação matemática que permite tratar o robô como um ponto no espaço de configurações.

Para criar o $\mathcal{C}_{space}$, é necessário mapear todos os possíveis valores de juntas, de modo que cada dimensão do $\mathcal{C}_{space}$ represente os valores de uma única junta do manipulador. A criação do espaço de configurações permite determinar em quais configurações o manipula-

dor tem um espaço livre navegável ($\mathcal{C}_{free}$) e em quais ele colide com os obstáculos ($\mathcal{C}_{obst}$). Desta forma, o problema de planejamento consiste apenas em determinar um caminho, τ , que ligue a configuração inicial e a configuração de destino nesse espaço abstrato, tornando a solução mais simples.

Um exemplo visual de como funciona o $\mathcal{C}_{space}$ é demonstrado na Figura 6. Em 6 (a) é apresentado um robô manipulador planar de 2-DOF com as regiões em preto simbolizando os obstáculos e o destino (*goal*) é representado pelo quadrado amarelo. O mapeamento do espaço de trabalho para o espaço de configurações é feito em termos das variáveis que representam os valores das juntas q_1 e q_2 . Como existem apenas duas variáveis, ele é representado por um sistema cartesiano de duas dimensões, sendo o eixo horizontal equivalente aos possíveis valores da junta q_1 e o eixo vertical os possíveis valores da junta q_2 (Figura 6 (b)). Ainda na Figura 6 é possível perceber que as características físicas do manipulador são totalmente abstraídas no espaço de configurações.

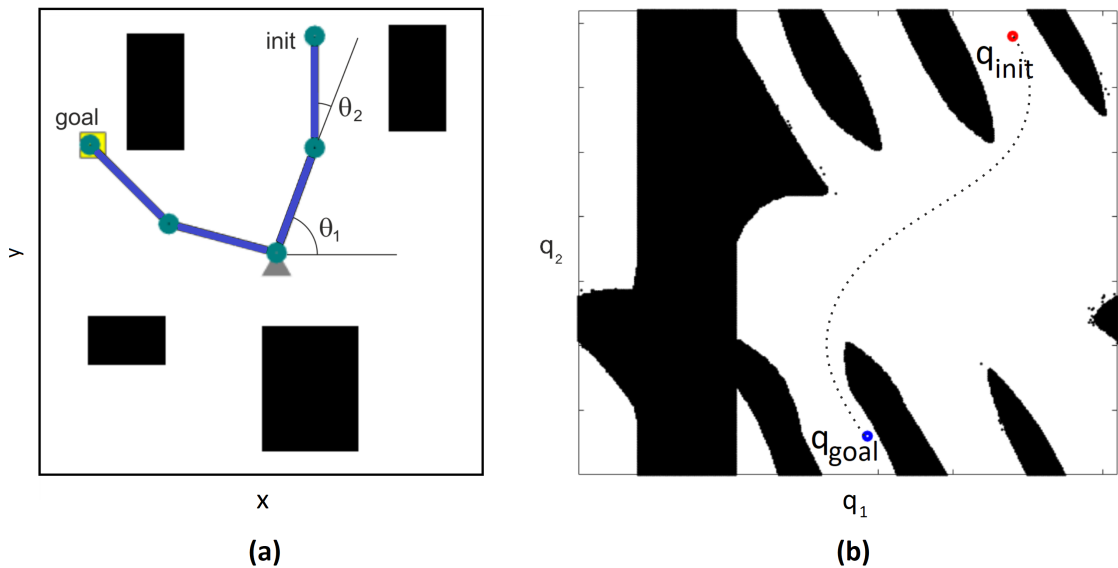


Figura 6 – Planejamento para robô manipulador de 2-DOF se mover da configuração inicial (q_{init}) até a configuração desejada (q_{goal}). (a) Tarefa no espaço de trabalho, (b) Tarefa no $\mathcal{C}_{space}$. Imagem adaptada de (LATOMBE, 2012).

Ao analisar a Figura 6, é possível perceber que a representação do $\mathcal{C}_{space}$ não tem uma relação intuitiva com o espaço de trabalho. Além disso, à medida que o manipulador possui mais graus de liberdade, esse espaço cresce em dimensionalidade e já não é mais possível uma representação visual do mesmo. O custo de criar todo o $\mathcal{C}_{space}$ para dimensões elevadas também é muito alto, por isso, o problema de planejamento no espaço de configurações costuma ser realizado através de técnicas que criem uma representação compacta do ambiente como a *Rapidly-exploring Random Tree* (RRT).

2.4 RRT

A RRT é um algoritmo clássico de planejamento que realiza uma busca aleatória, com formato de uma árvore, a fim de encontrar um caminho que conecte um estado inicial (x_{init}) com um estado desejado (x_{goal}) dentro de um determinado espaço de estados $\mathcal{X}$ (LAVALLE, 1998). O termo espaço de estados é utilizado para representar uma generalização dos possíveis espaços onde uma busca pode ser realizada.

O espaço $\mathcal{X}$ pode ser dividido em duas componentes, a região de estados admissíveis $\mathcal{X}_{free}$ e a região de rejeição $\mathcal{X}_{obst}$. A árvore gerada pelo algoritmo da RRT deve conectar o estado x_{init} ao estado x_{goal} através de um caminho contido em $\mathcal{X}_{free}$. Desta forma, a árvore tem início em x_{init} e são realizadas K tentativas de expansão na árvore pelo ambiente através da amostragem aleatória no $\mathcal{X}_{free}$.

Cada uma das K tentativas de expansão cresce a árvore na direção da amostra sorteada (x_{rand}) a partir de um nó da árvore que esteja mais próximo (x_{near}). Essa expansão é limitada por um tamanho de passo (d) que restringe a ligação entre x_{rand} e x_{near} , gerando um x_{new} . A busca deve ser interrompida caso o novo nó adicionado x_{new} esteja próximo o suficiente do x_{goal} , o critério de proximidade é calculado através de uma métrica de distância junto com um valor de tolerância tol que indica cumprimento da tarefa. A descrição da abordagem clássica da RRT é feita no Algoritmo 1.

Algoritmo 1: RRT Clássica

```

1:  $\mathcal{G}$ .inicializar( $x_{init}$ )
2: for  $i = 1$  até  $K$  do
3:    $x_{rand} = gerarAmostraAleatoria()$ 
4:    $x_{near} = verificaMaisProximo(x_{rand}, \mathcal{G})$ 
5:    $\vec{v} = x_{rand} - x_{near}$ 
6:    $x_{new} = x_{near} + limita(\vec{v}, d)$ 
7:   if  $x_{new} \subset \mathcal{X}_{free}$  then
8:      $\mathcal{G}$ .add_nó( $x_{new}$ )
9:      $dif = distancia(x_{novo}, x_{dest},)$ 
10:    if  $dif < tol$  then
11:      break
12:    end if
13:  end if
14: end for

```

Para uma melhor visualização do funcionamento do algoritmo da RRT, será feita uma aplicação da abordagem no contexto dos robôs manipuladores. No que diz respeito ao planejamento de caminho aplicado à robótica, o espaço de estados $\mathcal{X}$ pode ser utilizado como o espaço de configurações $\mathcal{C}$. Desta forma, os nós da árvore representam uma pose do robô no espaço de trabalho e as conexões, ou arestas, descrevem uma possibilidade de movimentação entre as configurações.

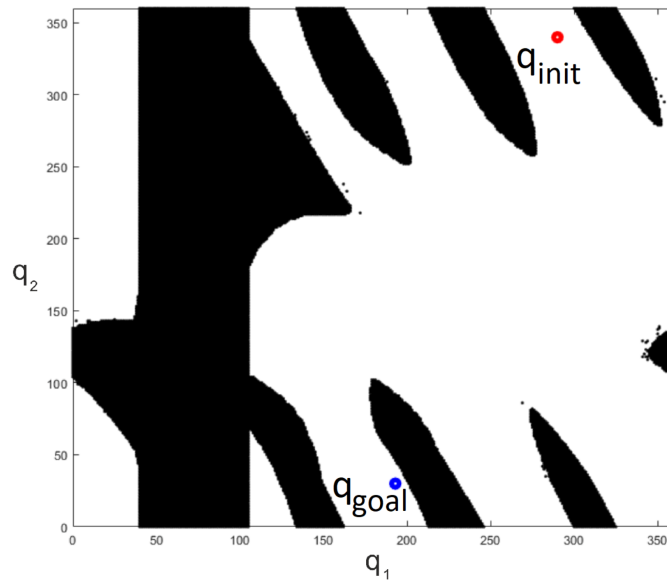


Figura 7 – Configuração inicial do robô (q_{init}) e configuração desejada (q_{goal}) no espaço de configurações.

Dado o robô manipulador de 2-DOF mostrado na Figura 6 (a) e as configurações q_{init} e q_{goal} na Figura 6 (b), a tarefa da RRT é encontrar um caminho que conecte as duas configurações. O algoritmo inicia gerando sua árvore $\mathcal{G}$ com o nó q_{init} como sua raiz (Figura 7). Após a inserção da raiz, o algoritmo segue um processo iterativo que tem como condição de parada a conexão entre as configurações desejadas ou um número máximo de iterações K . Cada iteração inicia com a amostragem aleatória de uma nova configuração q_{rand} no $\mathcal{C}_{free}$. Essa configuração, q_{rand} , é escolhida para guiar a expansão da árvore em sua direção. Na sequência, é identificado o nó de $\mathcal{G}$ que está mais próximo da configuração sorteada (q_{near}), de acordo com uma métrica de distância previamente estabelecida pelo projetista (Figura 8).

Em seguida, é iniciado o processo de expansão da árvore em direção a q_{rand} . Um vetor $\vec{v}$ é traçado do q_{near} em direção ao q_{rand} e então o algoritmo limita o crescimento da árvore na direção de q_{rand} em um tamanho máximo de passo d . Esse procedimento gera um novo nó q_{new} que será incorporado à árvore caso esse segmento d pertença ao $\mathcal{C}_{free}$ (Figura 9). Caso a aresta não esteja inteiramente no $\mathcal{C}_{free}$ o nó q_{new} é descartado e uma nova configuração é sorteada.

O procedimento descrito anteriormente deve ser repetido até que a configuração q_{goal} seja conectada à árvore ou até que um número máximo K de iterações seja alcançado, como mostrado no Algoritmo 1. Após algumas iterações, o algoritmo consegue adicionar a configuração desejada na árvore e então é possível ter um caminho entre as configurações inicial e final (Figura 10).

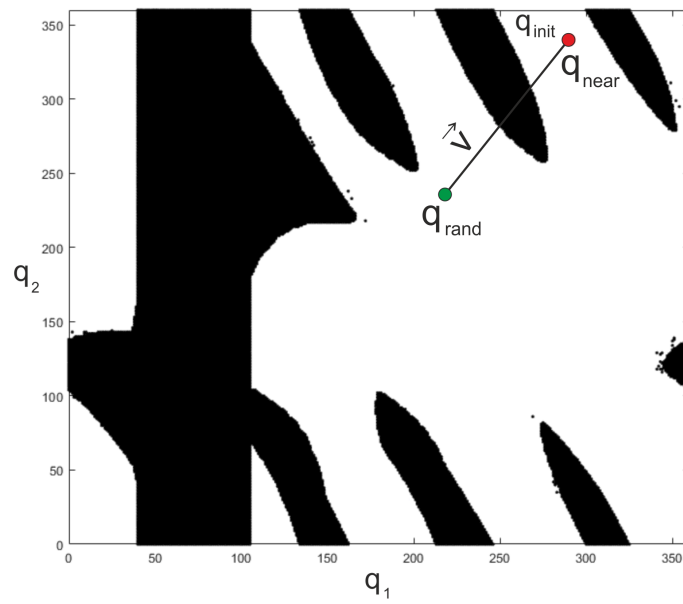


Figura 8 – Conexão entre a amostra aleatória (q_{rand}) com o nó mais próximo (q_{near}).

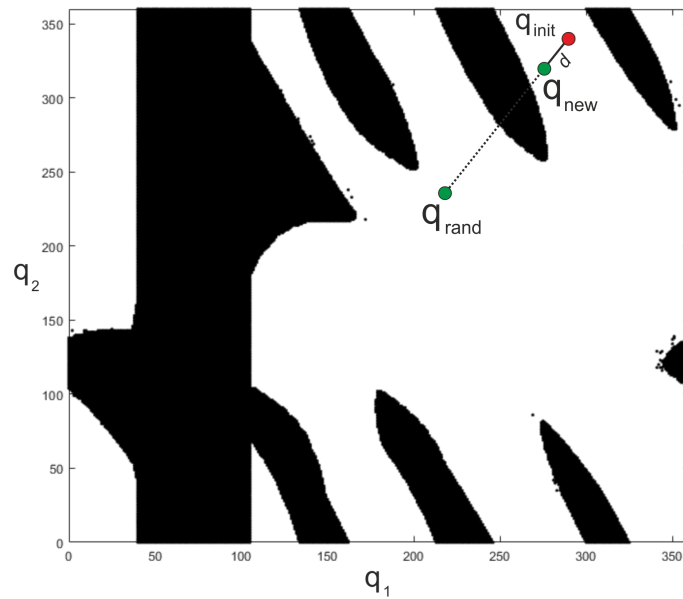


Figura 9 – Adição do novo nó (q_{new}) conforme a limitação do crescimento da árvore por um tamanho de passo constante (d).

As soluções apresentadas pela RRT não são ótimas, como pode ser visto na Figura 10. Entretanto, sua busca por uma solução sub-ótima acelera sua convergência e facilita a aplicação em grandes espaços de buscas. A RRT se destaca na área de planejamento de movimento devido a algumas características, como (LAVALLE, 1998):

- Simplicidade. Uma iteração da RRT se resume em sortear um ponto, encontrar o nó mais próximo e crescer a árvore na direção do sorteio;

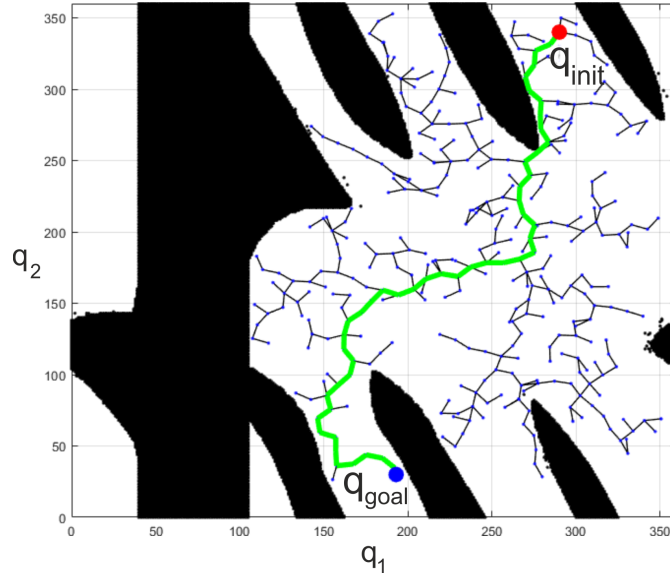


Figura 10 – Resultado da RRT clássica após conectar as configurações q_{init} e q_{goal} . O caminho gerado entre as duas configurações é destacado em verde e a árvore gerada é representada pelos pontos azuis (nós) e linhas pretas (arestas).

- Baixo custo computacional. Devido a simplicidade da abordagem o custo por iteração da abordagem é muito baixo;
- Rápida Convergência. A expansão aleatória da árvore cria uma representação compacta do ambiente, o que acelera o processo de busca. Além disso, a RRT não precisa encontrar todos os caminhos válidos, pois uma solução é apresentada ao adicionar x_{goal} à árvore;
- Aplicável em altas dimensões. A única interação da RRT com o ambiente é a verificação de pertencimento ao $\mathcal{X}_{free}$, dessa forma não se faz necessário um pré-processamento do ambiente;
- Completude probabilística. A RRT garante que, se possível, uma solução vai ser encontrada para um K tendendo ao infinito;
- Adaptável. A RRT possui uma grande quantidade de aplicações, além do planejamento de caminho (VÉRAS; MEDEIROS; GUIMARÃES, 2019).

3 Revisão Bibliográfica

Este capítulo é dedicado à apresentação da revisão bibliográfica sobre os principais conceitos e trabalhos envolvendo a RRT. Também é apresentada uma revisão da área de cinemática inversa, mostrando o estado da arte e as principais abordagens desenvolvidas para solução do problema. Para tanto, este capítulo é dividido em duas seções, a primeira referente a RRT e a segunda com foco na cinemática inversa.

3.1 *Rapidly-exploring Random Tree* (RRT)

O algoritmo da RRT foi inicialmente proposto por La Valle em (LAVALLE, 1998) para resolver o problema de planejamento de caminho através de uma busca aleatória em um espaço utilizando uma estrutura de dados que se assemelha a uma árvore. Essa abordagem permite o planejamento de caminhos através da exploração eficiente de espaços de busca multidimensionais com ou sem obstáculos (VÉRAS; MEDEIROS; GUIMARÃES, 2019).

Muitas abordagens baseadas no algoritmo da RRT foram propostas para resolver problemas de diferentes áreas. No campo da realidade virtual, Aleotti apresenta uma abordagem derivada da RRT para realizar um planejamento da desmontagem de uma pilha de objetos em um ambiente virtual que considera as características físicas. (ALEOTTI; CASELLI, 2011). No desenvolvimento de jogos, a RRT é utilizada na criação rápida de caminhos aleatórios para personagens não-jogáveis, ou seja, para personagens que se comportam de forma autônoma (FUNGE, 2004). Ela também pode ser utilizada na exploração das fases de um jogo, de modo a criar uma representação compacta da mobilidade do personagem, com o intuito de avaliar a jogabilidade (BAUER; POPOVIĆ, 2012).

No trabalho desenvolvido por Ahmadyan, a RRT é utilizada no espaço de estados para gerar casos de teste para comportamentos indesejados de circuitos analógicos (AHMADYAN; KUMAR; VASUDEVAN, 2012). Já no trabalho proposto por Yang e colaboradores, uma variação da RRT é usada para guiar a exploração de um ambiente desconhecido (YANG; GAN; SUKKARIEH, 2013). Zabolotny apresenta um planejador de trajetória baseado na RRT, visando a navegação autônoma de um veículo em rodovias sob condições ideais (ZABOLOTNY, 2019). Por fim, a RRT pode ser utilizada na robótica como um planejador de caminho para sistemas envolvendo robôs móveis, manipuladores, robôs espaciais, robôs subaquáticos, helicópteros e humanóides (VÉRAS; MEDEIROS; GUIMARÃES, 2019).

O sucesso da performance dessa abordagem em várias aplicações resultou em inúmeras extensões do algoritmo clássico (SANTANA et al., 2018). Em (LAVALLE; KUFFNER, 2001) é utilizado o conceito de árvore bidirecional, onde é gerada uma árvore a partir da configuração inicial e outra da configuração de destino com o intuito de acelerar a convergência do algoritmo. Essas árvores são expandidas conforme o algoritmo clássico da RRT até que elas possuam uma configuração em comum.

Kuffner e Lavalley apresentam a abordagem *RRT-Connect* com o intuito de diminuir consideravelmente o tempo de convergência do método ao utilizar uma árvore bidirecional e tentar estabelecer uma conexão entre elas ao fim de cada iteração. Sempre que um novo nó é adicionado, ele é estendido em direção ao nó mais próximo da árvore oposta até que as árvores estejam conectadas ou até que ocorra uma colisão (KUFFNER; LAVALLE, 2000). Apesar de apresentar uma solução rapidamente, essa abordagem só é aplicável em tarefas onde a configuração de destino já é conhecida.

Já em (PLAKU et al., 2005) é proposta a SRT, que gera uma árvore multidirecional ao sortear várias configurações no espaço de busca e crescer uma RRT em cada uma delas. Essa abordagem visa capturar a conectividade do $\mathcal{C}_{free}$ e então resolver problemas de múltiplas consultas, onde primeiro a árvore cresce no ambiente e depois, com essa mesma árvore, apresenta soluções para diferentes configurações de destino.

Em ambientes que possuem corredores estreitos, a abordagem clássica da RRT apresenta dificuldades para encontrar uma solução. A fim de resolver esse problema, Rodriguez apresenta a RRT baseada em obstáculos (OB RRT, do inglês: *Obstacle-based RRT*). Ela utiliza o formato dos obstáculos para enviesar a expansão da árvore, de modo que ela consegue resolver o problema com corredores estreitos de forma mais eficiente que outros formatos de amostragem (RODRIGUEZ et al., 2006).

Por sua vez, para problemas que visam melhorar as trajetórias geradas pela RRT em um ambiente com obstáculos, Wang propõe uma variação da RRT que também utiliza os obstáculos para enviesar o crescimento da árvore. Nessa abordagem, uma linha reta é definida entre a configuração inicial e a configuração desejada a fim de identificar os obstáculos de interesse. Após identificar os obstáculos, são gerados pontos de interesse em suas bordas e a RRT busca se conectar a essas regiões (WANG; LI; MENG, 2016).

Em (KANG et al., 2016), os autores propõem a RRT orientada ao objetivo (GO RRT, do inglês: *Goal Oriented RRT*), na qual o processo de amostragem é definido de modo a enviesar o crescimento da árvore em direção ao destino, permitindo que a RRT convirja mais rápido para uma solução. Nessa abordagem o espaço de busca é definido como um círculo de raio igual a distância entre o destino e o nó mais distante da árvore.

Apesar dos métodos baseados na RRT resolverem o problema de planejamento de forma muito rápida, eles produzem soluções não-ótimas. Para lidar com esse problema, Karaman e Frazzoli propõem a abordagem RRT estrela (RRT*) que adiciona uma operação de reconexão dos nós que geram soluções quase ótimas. Mesmo após achar uma solução, a RRT* continua seu processo de amostragem a fim de produzir melhores resultados. Por isso, essa técnica é classificada como assintoticamente ótima, pois as soluções se aproximam da melhor solução à medida que a quantidade de amostras aumenta (KARAMAN; FRAZZOLI, 2010).

Trabalhos como o de (KARAMAN; FRAZZOLI, 2010) impulsionaram um aumento do interesse acadêmico, devido à sua propriedade ótima de custo assintótico (NOREEN; KHAN; HABIB, 2016). A fim de acelerar o processo de convergência na busca de soluções assintoticamente ótimas, Kleen apresenta a *RRT*-Connect* que funciona de forma análoga ao RRT* com o crescimento de uma árvore bidirecional e a etapa de conexão após a iteração (KLEMM et al., 2015).

Apesar dos ótimos resultados obtidos com a RRT* e a *RRT*-Connect*, essas abordagens mantêm uma amostragem uniforme por todo o espaço de configurações mesmo após encontrar uma possível solução, comportamento tal que pode ser definido como ineficiente dado as características de solução única do problema. Sendo assim, Gammel propõe a abordagem *Informed RRT**, que limita o espaço de busca após encontrar uma possível solução. Essa limitação é aplicada através de uma região no formato de elipse que é recalculada sempre que um melhor caminho é encontrado (GAMMELL; SRINIVASA; BARFOOT, 2014).

A fim de melhorar os resultados obtidos na *Informed RRT**, Mashayekhi e colaboradores desenvolveram a *Informed RRT*-Connect* que funciona de forma análoga às outras variações *Connect* e mantém as mesmas características da *Informed RRT** com um tempo de convergência menor (MASHAYEKHI et al., 2020).

Em algumas aplicações, o uso da métrica euclidiana para expansão da RRT pode ser uma escolha ruim, pois duas configurações que estão fisicamente próximas podem exigir movimentações complexas (ELBANHAWI; SIMIC, 2014). No trabalho desenvolvido por Han, é apresentado o problema de planejamento de caminho para realizar o estacionamento autônomo de um veículo. Nesse caso, o carro é incapaz de realizar movimentos laterais muito próximos o que implicou na alteração na métrica de expansão da árvore ao desprezar os nós de distância euclidiana menor que um raio de giro mínimo (HAN; DO; MITA, 2011).

Com o intuito de evitar o desperdício de tempo de planejamento em nós que estão propícios a falha, Cheng e Lavalley propõem uma abordagem que é frequentemente referida como RRT de resolução completa (RC-RRT, do inglês: *Resolution Complete RRT*). A RC-RRT utiliza como métrica de expansão a taxa de falha de expansões dos nós anteriores (CHENG; LAVALLEY, 2002).

A RRT também pode ser usada em um contexto onde o objetivo não é encontrar o caminho ótimo entre duas configurações. O trabalho desenvolvido por Kamarry apresenta a DRRT (do inglês, *Dispersion RRT*), onde o objetivo é obter uma representação compacta do ambiente, com um menor custo computacional. Este método limita a quantidade de configurações numa região com base no tamanho da aresta, o que resulta numa menor quantidade de nós para representar uma dada região do ambiente, quando comparada com a RRT clássica (KAMARRY et al., 2015).

A partir da representação compacta do ambiente gerada pela DRRT, é possível partir de qualquer configuração do ambiente e chegar até o destino. Com base nisso, Kamarry e colaboradores apresentam uma abordagem híbrida para navegação em ambientes dinâmicos e/ou incertos. Essa abordagem utiliza o resultado da DRRT para gerar um caminho em direção ao destino e conta com um controlador reativo para desvio de obstáculos dinâmicos na etapa de rastreamento da trajetória gerada. Caso o controlador reativo seja ativado, a representação criada pela DRRT consegue rapidamente replanejar a trajetória (KAMARRY et al., 2016).

No que diz respeito ao planejamento de caminho para robôs manipuladores, a RRT tem sido amplamente utilizada nos últimos anos devido a seu bom desempenho em ambientes de alta dimensionalidade. Wei apresenta a S-RRT para lidar com o planejamento de movimento para manipuladores em ambientes dinâmicos (WEI; REN, 2018), enquanto que Rybus utiliza a RRT para geração de trajetória, ambos com as árvores sendo construída no espaço de configurações (RYBUS; SEWERYN, 2015).

Já no trabalho desenvolvido por Wégue e colaboradores, a RRT é aplicada diretamente no espaço de trabalho com o intuito de criar uma trajetória entre dois pontos sem se preocupar em resolver o problema de IK. Especificamente, o método do jacobiano transposto é utilizado como solucionador da IK para que o robô possa seguir a trajetória gerada (WEGHE; FERGUSON; SRINIVASA, 2007).

De modo geral, os trabalhos que utilizam a RRT no contexto de manipuladores aplicam essa técnica para solução do problema de planejamento de caminho, para tanto assumem que a tarefa de encontrar a configuração final é resolvida por um outro método de cinemática inversa. Entretanto, Bertram apresenta um método baseado na RRT que soluciona o problema de planejamento em conjunto com o problema de cinemática inversa.

Nessa abordagem o destino é conhecido apenas no espaço de trabalho e a medida que uma nova configuração é adicionada na árvore é calculada a cinemática direta a fim de obter a posição do efetuador no espaço de trabalho (BERTRAM et al., 2006).

Ainda nesse trabalho, Bertran propõe o uso de uma heurística, medida no espaço de trabalho, para enviesar o crescimento da árvore. Essa heurística atribui a cada configuração um valor que representa a distância do efetuador em relação ao destino e do efetuador em relação ao obstáculo mais próximo no espaço de trabalho, sendo que a distância em relação ao obstáculo contribui de maneira negativa para esse valor. Desta forma, a etapa de amostragem escolhe a configuração que possui a melhor avaliação e expande ele em uma direção aleatória. Caso a nova configuração possua uma melhor avaliação que o nó anterior ele é adicionado, caso contrário ele é descartado. Sempre que um nó é adicionado, a árvore segue a expansão do nó na mesma direção de forma semelhante a *RRT-Connect*.

Essa heurística adicionada pode levar o algoritmo a cair em mínimos locais. Por isso, após uma quantidade k de tentativas falhas de expansão de um mesmo nó, ele é impossibilitado de realizar a expansão de modo a sair do mínimo local. Apesar de se basear nos conceitos da *RRT-Connect* para desenvolver essa heurística, uma expansão linear da árvore no espaço de configurações não resulta em uma aproximação no espaço de trabalho.

Apesar do uso da RRT em (BERTRAM et al., 2006) para resolver o problema de cinemática inversa, ela não é amplamente utilizada para solução de IK. Como o problema de cinemática inversa visa encontrar uma configuração de destino para um determinado ponto no espaço de trabalho, essa tarefa não permite conhecer uma configuração de destino no espaço de configurações. Essa limitação dificulta a proposição de heurísticas eficientes que tornem a aplicação da RRT desejável para problemas de cinemática inversa.

3.2 Cinemática inversa

Os primeiros trabalhos em IK datam da década de 60 (WHITNEY, 1969). Desde então, a cinemática inversa continua sendo objeto de pesquisa devido à complexidade dos corpos a serem controlados. Diante da grande variedade de métodos para apresentar uma solução, nessa seção são abordados os trabalhos que mais sintetizam os conceitos de cada categoria. Como não há uma taxonomia bem definida e não é objetivo desse trabalho propor uma nova, serão utilizadas as classificações apresentadas em (ARISTIDOU et al., 2018) e (SAVSANI; JHALA; SAVSANI, 2014).

Os métodos clássicos para resolução da IK são baseados na matriz jacobiana, ou em variações dela como o jacobiano transposto, pseudoinversa e DLS (do inglês: *Damped-Least Square*) (VITO; NATALE; ANTONELLI, 2017). As soluções apresentadas através da matriz jacobiana produzem posicionamentos suaves, mas em geral sofrem de alto custo computacional, cálculos de matrizes complexas e problemas de singularidade (ZHANG et

al., 2018). A fim de lidar com os problemas de singularidade e redundâncias cinemáticas em manipuladores, a técnica DLS faz uso da matriz pseudoinversa da jacobiana com amortecimento.

Uma segunda família de métodos apresenta soluções numéricas baseadas em uma expansão da Série de Taylor de segunda ordem que utiliza uma aproximação da matriz Hessiana baseada no valor da função gradiente. Dentro dessa família, os principais métodos são o método de Broyden, o método de Powell e o método Broyden, Fletcher, Goldfarb e Shanno (BFGS) (CHIN; KONSKY; MARRIOTT, 1997; FLETCHER, 2013). Os métodos Newtonianos são apresentados como um problema de minimização, sendo assim, retornam um movimento suave sem descontinuidades erráticas e não sofrem com problemas de singularidade. No entanto, eles são complexos, difíceis de implementar e têm alto custo computacional por iteração (STARKE; HENDRICH; ZHANG, 2018).

Alguns trabalhos resolvem o problema de IK de um ponto de vista estatístico. Courty e Arnaud utilizam o método sequencial de Monte Carlo, explorando o princípio de amostragem para otimizar o processo iterativo computacional realizado. Esta abordagem não requer inversão numérica explícita e restrições de juntas podem ser aplicadas ao sistema de maneira intuitiva, sem a necessidade de algoritmos de otimização complexos (COURTY; ARNAUD, 2008). Hecker e colaboradores utilizaram um método IK iterativo com vários parâmetros para ajustar o comportamento de um corpo virtual de forma estática e dinâmica (HECKER et al., 2008). Apesar do fato que nenhum desses métodos sofre com problemas de singularidade matricial, eles apresentam a desvantagem de um alto custo computacional (ARISTIDOU; LASENBY, 2009).

Os algoritmos baseados em heurísticas resolvem o problema de cinemática inversa de forma simples, sem a necessidade de equações e cálculos complexos. Eles costumam ter um baixo custo computacional, convergindo rapidamente para uma solução, e são altamente recomendados para problemas simples (ARISTIDOU et al., 2018). As principais limitações destes métodos estão ligadas ao fato que eles produzem movimentos e gestos não naturais ou biomecanicamente inviáveis. Dentro dessa família de algoritmos pode-se destacar o triangulação, CCD (do inglês: *Cyclic Coordinate Descent*), e FABRIK (do inglês: *Forward and Backward Reaching Inverse Kinematics*).

O método da triangulação, apresentado em (MULLER-CAJAR; MUKUNDAN, 2007), corresponde a uma técnica não recursiva para solucionar o problema através do uso da lei dos cossenos. Contudo, ainda existem muitas restrições que não são abordadas, como limite de juntas e controle de angulação do efetuador.

O CCD, apresentado em (WANG; CHEN, 1991), é um dos métodos iterativos mais populares para resolução da IK. Ele tem sido muito utilizado no campo da animação gráfica e da robótica devido a sua rápida convergência e baixo custo computacional. Este método tenta minimizar erros de posição e orientação, transformando uma variável de

junta isoladamente. O algoritmo parte do efetuador em direção à base do manipulador, transformando cada articulação de modo a mover o efetuador o mais próximo possível do alvo. Este procedimento é repetido até se obter uma solução satisfatória. Entretanto, o CCD possui o viés de, muitas vezes, descrever movimentos não realísticos para robôs manipuladores pois tende a enfatizar excessivamente os movimentos das articulações mais próximas do efetuador e esse tipo de solução acaba levando a grandes rotações angulares, que geralmente produzem movimentos com descontinuidades e oscilações erráticas.

Em (KENWRIGHT, 2012) é apresentada uma revisão do CCD discutindo detalhes de implementação, extensões e limitações do algoritmo. Lander (LANDER; CONTENT, 1998) incluiu limites de juntas de forma que em cada passo os limites superior e inferior são respeitados. Já em (MERRICK; DWYER, 2004) é descrita uma extensão que lida com múltiplos efetuadores. Várias outras extensões do CCD foram desenvolvidas para melhorar sua performance no que diz respeito a produção de soluções realísticas (ARISTIDOU et al., 2018).

O FABRIK é outro método baseado em heurística que tem se destacado nos últimos anos. Essa abordagem visa minimizar o erro do sistema ajustando sequencialmente cada ângulo de junta, primeiro na direção efetuador-base, e depois na direção base-efetuador. O algoritmo inicia posicionando seu efetuador no ponto de destino e encontra as novas posições de junta projetando um ponto na reta que conecta a junta atual com a junta anterior até chegar na base. Após calcular a nova posição da base, ela retorna para seu ponto original e o mesmo processo é feito em direção ao efetuador (ARISTIDOU; LASENBY, 2011).

O método do FABRIK atrai a atenção dos pesquisadores, pois possui várias vantagens dos algoritmos heurísticos iterativos existentes. O custo computacional por iteração é baixo, o que significa que converge para uma solução muito rapidamente. O método ainda é de fácil implementação, uma vez que resolve o problema utilizando somente pontos e projeções e sempre retorna uma solução quando o alvo é alcançável. Outra vantagem dessa abordagem é que ela não requer cálculos complexos ou manipulações de matrizes, não sofre de problemas de singularidade e retorna um movimento suave. Além disso, sua ênfase nos movimentos das juntas mais próximas à base assegura uma simulação mais consistente com os movimentos naturais do que a observada em outros métodos como o CCD. Entretanto, ao aplicar as restrições de movimentação, o FABRIK perde a garantia de encontrar uma solução. Isso acontece porque o algoritmo trata cada junta independentemente e não considera as restrições das juntas anteriores e posteriores. (ARISTIDOU; LASENBY, 2011).

Nos últimos anos, diversas extensões do FABRIK foram propostas para lidar com aspectos específicos. Em (FARAJI; IJSPEERT, 2017), foi utilizada uma variação do FABRIK para resolver o problema de IK sob a perspectiva de transferência de energia.

Neste método, foi aplicado um modelo massa-mola para ajustar as posições das juntas, minimizando a energia que é conservada nas molas. Por sua vez, em (MOYA; COLLOUD, 2013), os autores demonstram que o FABRIK pode lidar com diferentes prioridades para seus alvos. Tao e Yang estenderam o FABRIK para lidar com tarefas onde é necessário evitar colisões (TAO; YANG, 2017). Por fim, no trabalho desenvolvido em (ARISTIDOU; CHRYSANTHOU; LASENBY, 2016) são apresentadas extensões para atender aos mais variados tipos de juntas.

Com o avanço da tecnologia relacionada à captura de movimentos e à grande disponibilidade de dados relacionados em bibliotecas *online*, os métodos de cinemática inversa baseados em dados passaram a ganhar visibilidade. Esses métodos usam posições conhecidas numa fase de aprendizado para encontrar uma pose que leve o efetuador a uma posição desejada no espaço.

Grochow e colaboradores apresentaram um método de dados que se baseia em um modelo aprendido a partir de comportamentos humanos (GROCHOW et al., 2004). Sendo assim, o modelo é treinado com diferentes dados de entrada, que leva a diferentes poses que podem ser representadas. Já no trabalho desenvolvido por Sumner, ao invés de usar estilos humanos como dados de treinamento, a abordagem cria um conjunto de poses significativas a partir de exemplos e usam o conjunto criado para produzir as poses desejadas (SUMNER et al., 2005).

A desvantagem dos métodos que pertencem a essa família é que requerem um procedimento de treinamento *off-line* e os resultados são altamente dependentes dos dados de treinamento e limitados aos modelos e movimentos no qual o sistema foi treinado (ARISTIDOU; LASENBY, 2009).

A crescente popularidade do *deep learning* e o seu sucesso na reconstrução de poses humanas tem instigado pesquisadores a investigar seu uso na obtenção de controladores de movimento. Nessas abordagens, são utilizadas redes neurais para aprender a movimentação das juntas através do processamento de movimentos reais. Recentemente, Holden e colaboradores apresentaram um controlador de movimento baseado totalmente no aprendizado de máquina. Especificamente, uma rede neural profunda é utilizada para possibilitar a alteração do movimento gerado, estendendo o espaço de exemplos com novas configurações (HOLDEN; SAITO; KOMURA, 2016). Entretanto, esses métodos geralmente possuem um alto custo computacional para treinamento, pois necessitam processar um volume de dados muito alto e a estrutura das redes é bastante complexa.

A última família dos solucionadores de IK é a meta-heurística (MH), representada pelas técnicas que resolvem o problema de cinemática inversa do ponto de vista de otimização e tentam minimizar uma função de custo, também chamada de função *fitness*. Os principais métodos dessa família são baseados no (FA, do inglês: *Firefly Algorithm*) e no enxame de partículas (PSO, do inglês: *Particle Swarm Optimization*).

O algoritmo *firefly* foi criado com inspiração no comportamento dos vagalumes. Esses animais usam suas luzes para atrair outros, de modo que quanto mais intensa é a luz, maior é sua capacidade de atração (ROKBANI; CASALS; ALIM, 2015). Nessa abordagem, os vagalumes são representados pelos pontos no espaço de configurações e sua intensidade é calculada através da distância entre o efetuator e o ponto final desejado no espaço de trabalho.

O PSO define inicialmente um número de indivíduos do grupo, chamados partículas, o espaço de busca e um critério de otimização. As partículas são distribuídas aleatoriamente no espaço de busca e as melhores partículas, global e localmente, são definidas de acordo com uma função de custo. Cada partícula é então movida para uma nova posição, levando em consideração sua posição atual e as melhores partículas locais e globais. As partículas continuam sendo deslocadas no espaço de busca enquanto o objetivo não for alcançado e o número máximo de iterações não é atingido. A condição de parada garante a convergência do algoritmo, mas não garante a otimização da solução (ROKBANI; ALIM, 2013).

A busca probabilística realizada pelas abordagens meta-heurísticas possui a desvantagem de ser aplicada no espaço de configurações o que implica em um espaço de dimensão elevada para grande parte dos manipuladores. Além disso, dependendo da escolha dos parâmetros, os métodos MH podem sofrer com o problema de mínimo local, pois essa classe de método se baseia em algum tipo de heurística (JUNIOR et al., 2018).

No trabalho desenvolvido por Junior e colaboradores, é proposto o PSO totalmente reamostrado (FRPSO, do inglês: *Fully Resampled PSO*) que visa melhorar a robustez da solução em relação ao problema de mínimo local através da reamostragem de todos os nós após o fim da iteração. Além disso, esse método reduz a dimensionalidade do problema pela metade ao desconsiderar as componentes de velocidade da partícula. Essa redução de dimensionalidade tem um impacto muito positivo na taxa de convergência e na quantidade de iterações, em comparação com outras variações do PSO, pois diminui a complexidade da tarefa e proporciona um menor espaço de busca (JUNIOR et al., 2018).

Por fim, vale mencionar o trabalho apresentado em (VAHRENKAMP et al., 2009), onde um método denominado IK-RRT é apresentado. Apesar da nomenclatura, esse método é usado para planejamento de movimento no espaço de configurações e funciona com o auxílio de um método de cinemática inversa para encontrar possíveis configurações que levem o efetuator para uma posição desejada.

Desta forma, durante a revisão bibliográfica realizada nesse trabalho de dissertação, apenas o método apresentado por (BERTRAM et al., 2006) utiliza a RRT para resolver o problema de cinemática inversa. Entretanto, ele atua como um planejador de movimento no espaço de configurações que explora o espaço e verifica se as configurações geradas resultam em uma solução para o problema de cinemática inversa através da transformação do espaço de configurações para o espaço de trabalho.

A revisão ainda evidencia que métodos simples, como o FABRIK e o CCD, têm tido grande destaque no contexto da cinemática inversa (ARISTIDOU et al., 2018). Essas técnicas são caracterizadas pela velocidade de convergência e baixo custo computacional ao aplicar operações simples diretamente no espaço de trabalho.

4 Desenvolvimento do Trabalho e Abordagem Proposta

Neste capítulo, é apresentado o desenvolvimento do trabalho que se inicia com um estudo do uso da RRT clássica para o problema de cinemática inversa no espaço de configurações. Desta forma, foi possível perceber que o crescimento da árvore no *Cspace* implica em uma busca dentro de um espaço de dimensão elevada para grande parte dos manipuladores e dificulta a proposição de alterações na técnica, pois os nós e arestas não possuem uma relação intuitiva com a posição espacial do manipulador no espaço de trabalho. Sendo assim, neste trabalho é proposta a aplicação da RRT diretamente no espaço de trabalho com o intuito de realizar a busca em um espaço de dimensão reduzidas e investigar as alterações que essa mudança de espaço permite.

4.1 RRT no *C-Space*

A primeira etapa deste trabalho foi focada na implementação do algoritmo clássico da RRT diretamente no espaço de configurações para resolução do problema de cinemática inversa. Ao realizar esta etapa, é possível analisar os resultados e identificar os aspectos da RRT que podem ser potencializados por uma aplicação diretamente no espaço de trabalho.

Inicialmente a RRT foi aplicada em um robô planar de 2-DOF com um comprimento de 20cm , como pode ser visto na Figura 11 (b). Uma vez que o problema de cinemática inversa consiste em encontrar uma configuração que leve o efetuador do manipulador para uma pose desejada, o resultado da tarefa é simbolizado por uma estrela vermelha na Figura 11(a). O tamanho de passo escolhido foi de 3 unidades no espaço de configurações e uma tolerância de $0,2\text{ cm}$ pro destino.

Ao analisar a implementação do algoritmo clássico para o problema de IK, alguns pontos foram levantados no que diz respeito à sua aplicação no espaço de configurações. Como a configuração final não é conhecida no *c-space*, a busca é feita sem um destino nesse espaço, dado que essa informação está presente no espaço de trabalho. Essa questão dificulta a definição de parâmetros do algoritmo como tamanho de passo e tolerância final do destino, uma vez que eles impactam de forma diferente nos diferentes espaços. Por fim, essa diferença de espaços ainda dificulta a alteração de etapas da RRT, pois a influência dessas alterações no resultado final são difíceis de serem medidas e a aplicação de técnicas de polarização também é não-trivial devido a necessidade de mapeamento entre espaços.

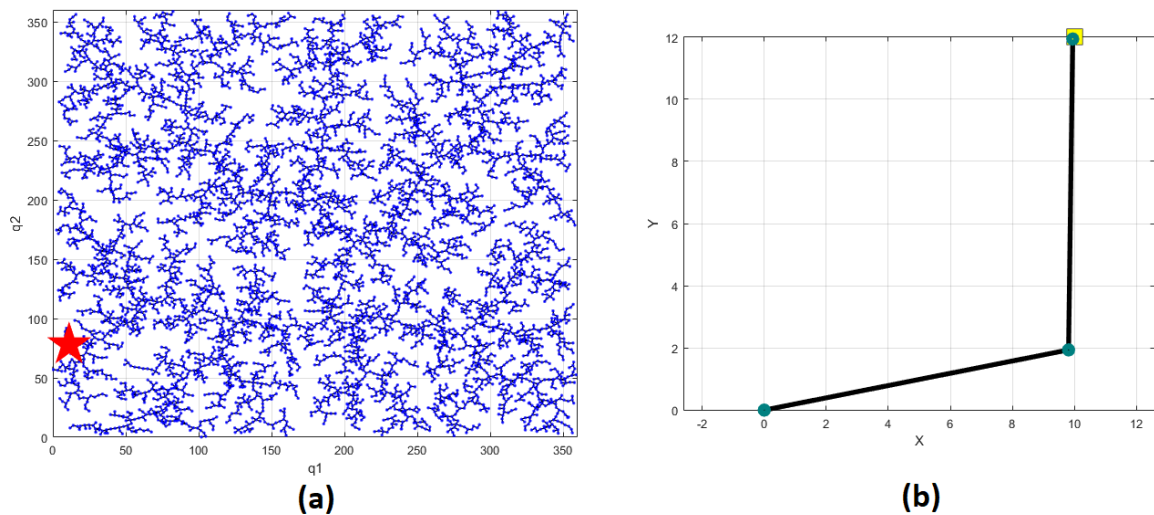


Figura 11 – Resultado do algoritmo clássico da RRT (em vermelho) no c -space. (a) árvore no c -space, (b) configuração gerada para alcançar o marcador amarelo.

Outro ponto que vale mencionar é que o espaço de busca aumenta à medida que o manipulador possui mais graus de liberdade. Sendo assim, quanto maior o número de juntas do manipulador, mais complexa é a busca nesses espaços de dimensões elevadas e maior é a dificuldade de acelerar o processo pelas complexidades impostas na transformação dos espaços.

Pode-se exemplificar essa desvantagem com um manipulador planar de 4-DOF, onde o espaço de configurações precisaria necessariamente realizar a busca em um espaço de 4 dimensões. No entanto, se essa abordagem for aplicada diretamente no espaço de trabalho, é possível realizar a busca em um espaço de apenas duas dimensões, no caso planar, e visualizar aspectos físicos dos manipuladores que podem acelerar a busca pelo objetivo.

Desta forma, no decorrer deste trabalho, foram feitas algumas alterações na RRT clássica de modo a possibilitar a sua aplicação diretamente no espaço de trabalho e verificar a influência dessas alterações no resultado final. As mudanças realizadas envolveram tamanho fixo da aresta, adição de informação aos nós, formato da amostragem e novo critério para seleção de nó mais próximo. Esses tópicos serão abordados com detalhes nas próximas seções.

4.2 RRT no espaço de trabalho

O formato de um robô manipulador no espaço de trabalho pode ser bastante complexo, o que impossibilita a interpretação natural da RRT nesse espaço. Deste modo, para o contexto dos manipuladores no espaço de trabalho, é necessária uma releitura da

abordagem clássica. Essa releitura se baseia na ideia de que a árvore criada no espaço de trabalho representa a construção do manipulador nesse espaço de forma que os nós representam as juntas e as arestas os elos do robô.

Para que a árvore possa representar o manipulador no espaço de trabalho, foi adicionado um índice a cada nó para associá-lo com as juntas do manipulador. Deste modo, um novo nó só pode ser adicionado à árvore se a conexão for feita a partir de um nó de índice imediatamente inferior ao atual, garantindo o crescimento da árvore no formato do manipulador.

Diante disso, a base do manipulador é considerada como índice 0 e o algoritmo é inicializado com índice 1, sendo incrementado a cada iteração após a adição de um novo nó até atingir o número total de juntas (nós) do manipulador, quando o índice retorna para 1 e o processo é reiniciado. Outra modificação em relação à abordagem clássica é o tamanho do passo. Como as arestas devem representar os elos do manipulador no espaço, o tamanho do passo equivale ao tamanho do elo em questão.

Na Figura 12 é apresentado um passo-a-passo dessa interpretação para um manipulador planar de 4-DOF, onde os índices podem ser identificados pelas cores das arestas. O primeiro sorteio é mostrado na Figura 12 (a) seguido das etapas de conexão e limitação pelo tamanho do elo d_1 na Figura 12 (b) e na Figura 12 (c), respectivamente. Ainda na Figura 12 (c) é possível notar que o primeiro nó adicionado à árvore possui o índice 1 e sua aresta é identificada pela cor vermelha.

O próximo passo do algoritmo consiste em conectar o nó de índice 2 com um nó de índice imediatamente inferior, como mostrado na Figura 12 (d) com a identificação da aresta na cor verde. Na Figura 12 (e), a primeira iteração do método é concluído com os nós de índice 3 e 4 adicionados à árvore e suas arestas representadas pelas cores azul e preto, respectivamente. Após várias iterações, a árvore cresce como sendo as possíveis movimentações do manipulador no espaço, como pode ser visto na Figura 12 (f), garantindo que uma solução será encontrada e que a mesma pode ser executada pelo manipulador, desde que ela exista e seja dado tempo suficiente para o algoritmo (LAVALLE, 1998).

Nessa abordagem, quando o índice atual atinge valor equivalente à quantidade de juntas, é feita uma verificação de sucesso. Essa verificação de sucesso é feita utilizando uma distância euclidiana entre o destino e o efetuador com base em um valor de tolerância. Nesse ponto é importante ressaltar que a distância é utilizada somente para verificação de cumprimento da tarefa, sem interferência no crescimento da árvore. Para essa função a distância euclidiana se encaixa perfeitamente pois os pontos de interesse estarão sempre no $\mathbb{R}^2$ ou $\mathbb{R}^3$.

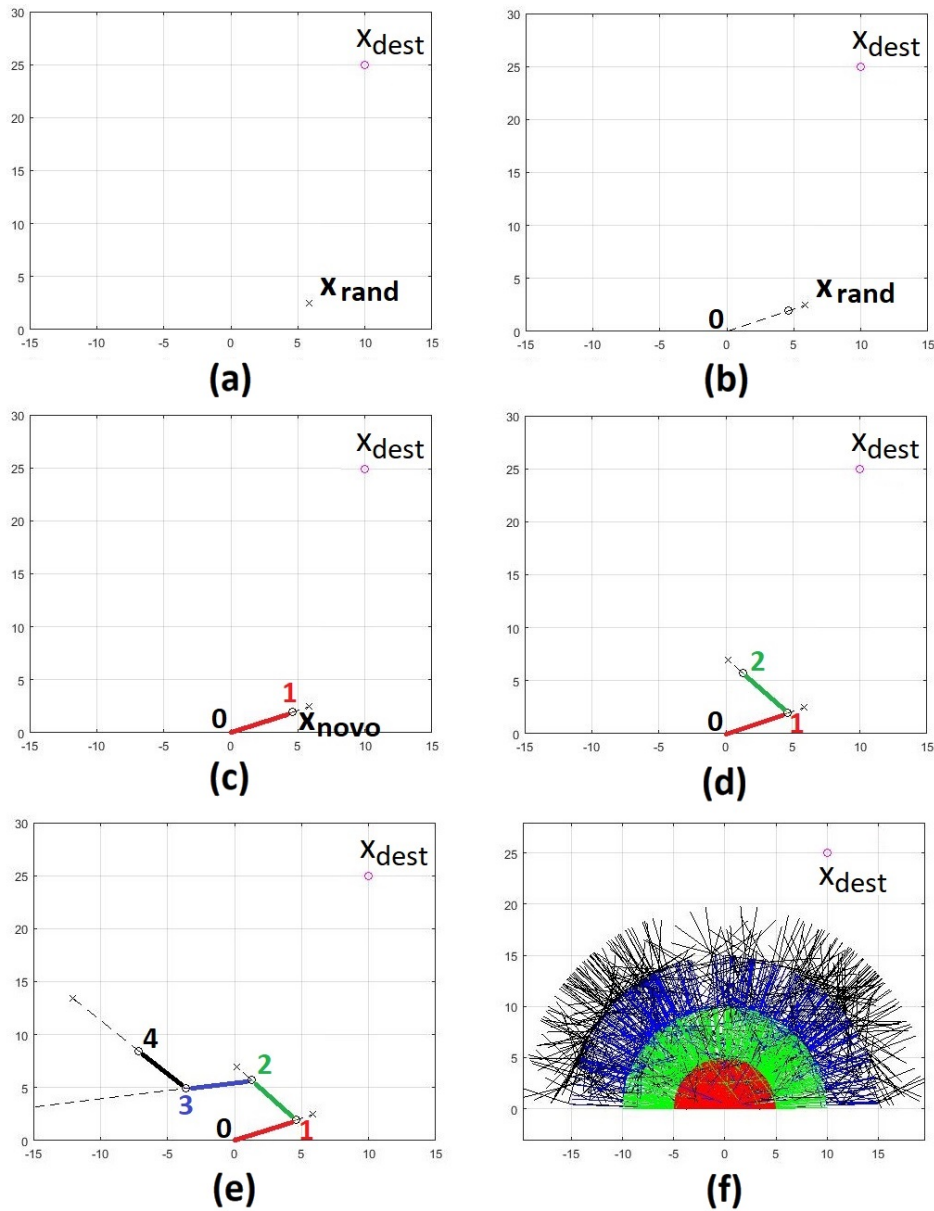


Figura 12 – Sequência da metodologia proposta. (a) Sorteio de um novo ponto x_{rand} . (b) Encontrar o ponto mais próximo e limitar o crescimento pelo tamanho do passo que equivale ao elo correspondente do manipulador. (c) O nó x_{novo} é adicionado com índice 1. (d) O nó de índice 2 é adicionado conectado ao nó imediatamente anterior (índice 1). (e) Após o último elo sorteado é feita a checagem de solução, caso não seja possível chegar ao destino o índice retorna para 0 e o algoritmo continua. (f) Árvore após várias iterações com os índices 1,2,3 e 4 sendo representados pelas cores vermelho, verde, azul e preto, respectivamente.

Caso o sucesso não ocorra, o algoritmo reinicia com o índice retornando para o valor 1. A descrição da abordagem proposta nesse trabalho é apresentada no Algoritmo 2, no qual G é a árvore, x representa um ponto no espaço de trabalho, n equivale à quantidade de juntas do manipulador e d representa os elos do manipulador.

Algoritmo 2: RRT no Espaço de Trabalho

```

1: G.inicializar()
2: while  $dif < tol$  do
3:   for  $i = 1, 2, \dots, n$  do
4:      $x_{rand} = amostra()$ 
5:      $x_{perto} = perto(x_{rand}, G, i)$ 
6:      $\vec{v} = x_{rand} - x_{perto}$ 
7:      $passo = d_i$ 
8:      $x_{novo} = x_{perto} + limita(\vec{v}, passo)$ 
9:     G.add_nó( $x_{novo}, i$ )
10:    if  $i == n$  then
11:       $dif = distancia\_euclidiana(x_{novo}, x_{dest})$ 
12:    end if
13:  end for
14: end while

```

Dada essa nova interpretação da RRT, há uma facilidade de visualização evidente da relação entre a árvore gerada e o robô manipulador, mesmo com um modelo de 4-DOF. Em termos de comparação, na Figura 11 o resultado do crescimento da árvore para um manipulador de apenas 2-DOF não possibilita ter a noção espacial do manipulador em seu espaço de trabalho. Vale ressaltar que a nova interpretação também permite a existência de bifurcações e ramos que não chegam ao último nível, o que não é possível no espaço de configurações, pois nesse espaço toda a cadeia é sorteada de uma única vez. É possível obter uma visualização dessas características através do exemplo mostrado na Figura 13. Nessa figura é utilizado um manipulador 3-DOF planar e é possível visualizar bifurcações e ramos que não alcançam o último nível após 3 iterações completas.

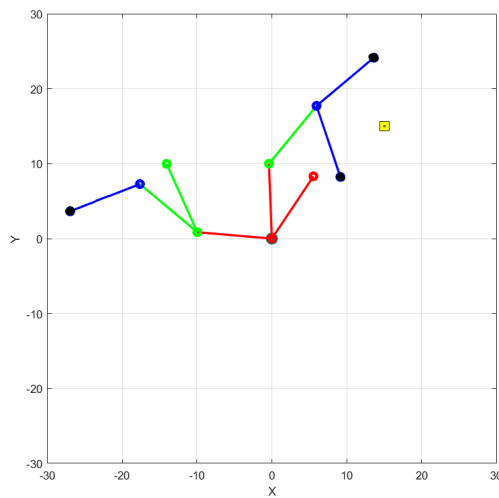


Figura 13 – Crescimento da árvore após 3 iterações para um manipulador planar de 3-DOF. Os nós de índice 1, 2 e 3 são caracterizados pelas cores vermelho, verde e azul, respectivamente.

A partir dessas observações, é possível modificar o crescimento da árvore com base na movimentação de cada junta individualmente. Além disso, como ela não sorteia uma configuração completa do manipulador, é possível utilizar os nós adicionados em iterações anteriores para enviesar o crescimento da árvore com critérios triviais como a distância euclidiana para o ponto de destino. De certo, essas discussões são possíveis devido a interpretação proposta, pois no contexto do espaço de configurações não é possível tratar uma junta de forma independente.

Sendo assim, a relação proposta nesse trabalho entre a árvore e o robô manipulador garante uma visualização mais intuitiva dos movimentos do robô, o que permite a alteração de algumas etapas da abordagem com base nas características espaciais do manipulador e no problema de IK. Como a solução da cinemática inversa consiste em uma configuração na qual o efetuador alcança o x_{dest} e não envolve necessariamente uma trajetória entre duas configurações, é possível delimitar regiões onde não é possível alcançar o destino e evitar o crescimento da árvore nessas regiões, reduzindo o espaço de busca a fim de melhorar o desempenho da técnica.

Uma vez que a mudança de representação possibilita uma alteração no desempenho da RRT no espaço de trabalho, pode-se entender que a diferença ocorre devido a uma mudança no modelo de probabilidade. Uma análise desse modelo pode ser feita através do estudo da transformação de uma variável aleatória. Entretanto, a mudança de representação realizada nesse trabalho não permite conhecer qual transformação é aplicada na variável em questão. Diante disso, o estudo do modelo de probabilidade neste trabalho é realizado através de experimentos na etapa de amostragem.

4.3 Amostragem

O processo de amostragem no algoritmo clássico da RRT ocorre de forma uniforme dentro da área de busca. Uma vez que um ponto no espaço de trabalho de 2 dimensões é descrito por um par ordenado (x,y) , a amostra sorteada deve conter um valor de x e um valor de y que descreva um ponto aleatório nesse espaço. Sendo assim, o primeiro formato de amostragem utilizado foi um sorteio uniforme em cada um dos eixos, resultando em uma região de busca retangular. Essa região foi definida como um quadrado de lado igual a duas vezes o comprimento do manipulador para garantir que todo o espaço alcançável do manipulador esteja contido na região de busca.

Na Figura 14 é possível ver um exemplo da área de busca e das amostras sorteadas (em verde) para um manipulador de 4-DOF com comprimento de 40 *cm*. Vale ressaltar que a maioria das figuras desta seção serão apresentadas com o sorteio da junta de índice 1 para facilitar a visualização e obter uma melhor ilustração da técnica.

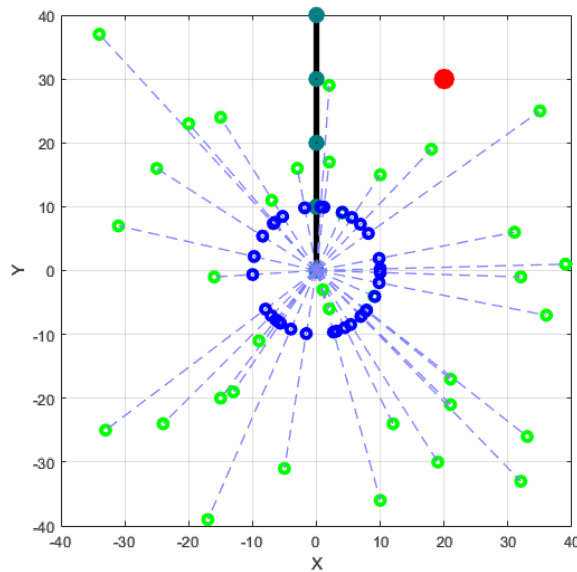


Figura 14 – Amostragem segundo a abordagem clássica. O ponto vermelho representa o destino (x_{dest}), em verde estão representadas as amostras sorteadas (x_{rand}), a linha pontilhada representa a conexão com o nó de índice imediatamente inferior mais próximo (x_{near}) e os pontos em azul representam os novos nós adicionados à árvore (x_{novo}).

Apesar da distribuição uniforme dos sorteios, a função densidade de probabilidade (FDP) dos novos nós na árvore pode não ser uniforme devido à diferença entre o formato retangular da área de busca e o formato circular dos novos nós, obtido após a limitação do tamanho da aresta na dimensão do elo do manipulador. Uma vez que o comprimento do elo (aresta) é fixo, uma estimativa da FDP pode ser obtida através de um conjunto de amostras do ângulo que descreve a posição do novo nó. Esse ângulo é calculado entre o eixo x positivo e o vetor que parte da origem $(0,0)$ até o nó x_{novo} .

Para visualizar a estimativa da FDP, foi feito um histograma baseado em 100000 amostras sob as condições descritas anteriormente. Uma vez que o interesse maior é no formato da distribuição, os histogramas apresentados nesse trabalho omitem os valores de probabilidade no eixo vertical e exibem os possíveis valores de ângulo no eixo horizontal.

No histograma apresentado na Figura 15 é possível identificar quatro picos de probabilidade que ocorrem com frequência de $(k\frac{\pi}{2} + \frac{\pi}{4})$ e se originam das quatro extremidades do quadrado. Com isso uma forma de obter uma FDP uniforme é alterar a forma de sorteio e fazer a amostragem no círculo em torno da junta em questão. Essa alteração permite que a ocorrência dos nós seja uniforme.

Apesar da distribuição uniforme ser desejável pela característica probabilística da técnica, para solução do problema de IK no espaço de trabalho dos robôs manipuladores existem áreas nesse espaço que não permitem solução pois a distância entre o nó e o destino pode ser maior que o restante da cadeia. Uma vez que um robô manipulador tem

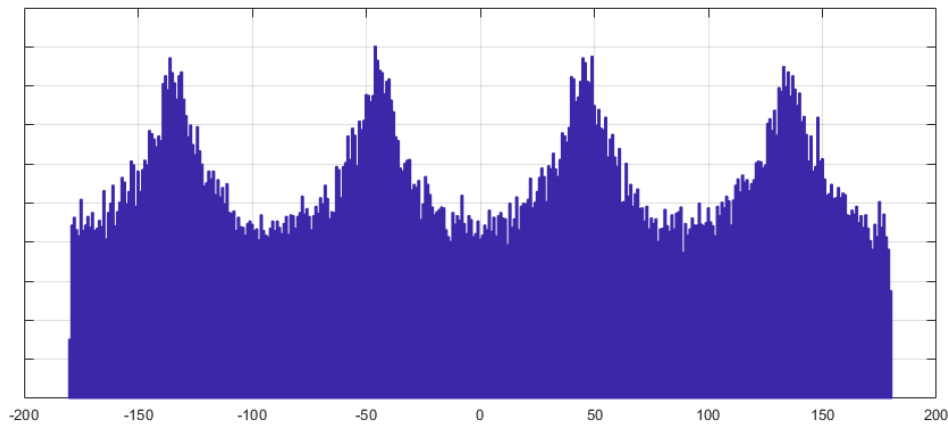


Figura 15 – Histograma gerado a partir da amostragem no espaço de trabalho.

um limite de alcance descrito pela extensão total dos seus elos, pode-se concluir que todo o espaço alcançável de um manipulador planar está contido em uma região circular onde o raio equivale a extensão máxima do manipulador.

Desta forma, para um dado ponto desejado no espaço, existe uma região circular em torno desse ponto na qual estão contidas todas as possíveis soluções para o problema de cinemática inversa. É possível perceber também que essa região é diferente para cada uma das juntas do manipulador, pois o movimento de extensão possível para cada junta depende do tamanho restante da cadeia. Uma interpretação visual dessas regiões pode ser obtida através de um exemplo utilizando um manipulador planar de 4-DOF com elos de mesmo comprimento.

Essa interpretação visual é possível devido à releitura da RRT para aplicação diretamente no espaço de trabalho que possibilita a identificação das áreas de interesse e permite a mudança no formato da amostragem de modo a alterar a distribuição de probabilidade e tornar a busca mais eficiente. Esta alteração tem como objetivo guiar a expansão da árvore em direção a regiões de solução e evitar os nós que não permitem solução. Na Figura 16 é possível visualizar as regiões de interesse para o manipulador planar de 4-DOF, delimitadas por círculos ao redor do objetivo, para um ponto de destino localizado em (20,30).

Ao analisar a Figura 16, é fácil notar que os sorteios fora das regiões circulares são sorteios que não contribuirão para solução. Isso ocorre pois o problema de cinemática inversa busca encontrar somente uma configuração que leve o efetuador ao destino e não uma trajetória entre duas configurações. Ainda destaca-se que há uma região circular diferente para cada junta, de modo que dado uma cor de junta há uma região circular correspondente de raio equivalente ao tamanho restante da cadeia cinemática até o efetuador.

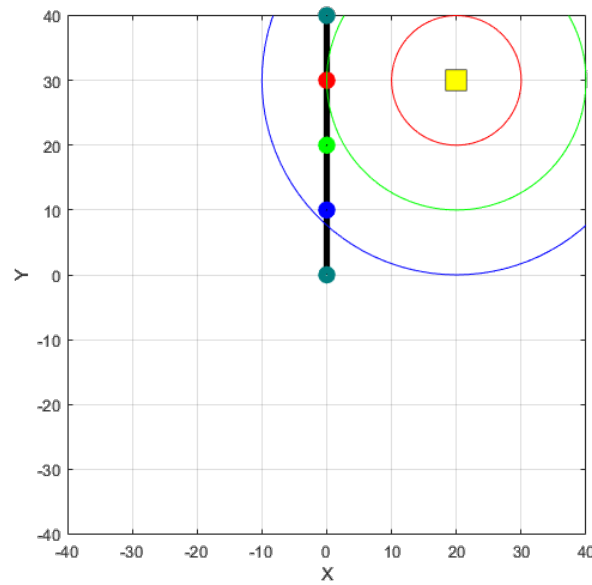


Figura 16 – Regiões de interesse em torno do destino (em amarelo). As cores dos círculos correspondem a cor da junta no manipulador. Deste modo, o raio do círculo equivale ao tamanho restante do manipulador a partir da junta correspondente.

Diante disso, a primeira tentativa de guiar o crescimento da árvore na direção das regiões de solução foi sortear aleatoriamente no espaço de trabalho e projetar o sorteio no círculo da junta em questão, Figura 17 (a). Depois de projetar no círculo, este procedimento segue os passos normais da RRT, isto é, encontrar o nó mais próximo e limitar o crescimento conforme o tamanho do passo, Figura 17 (b).

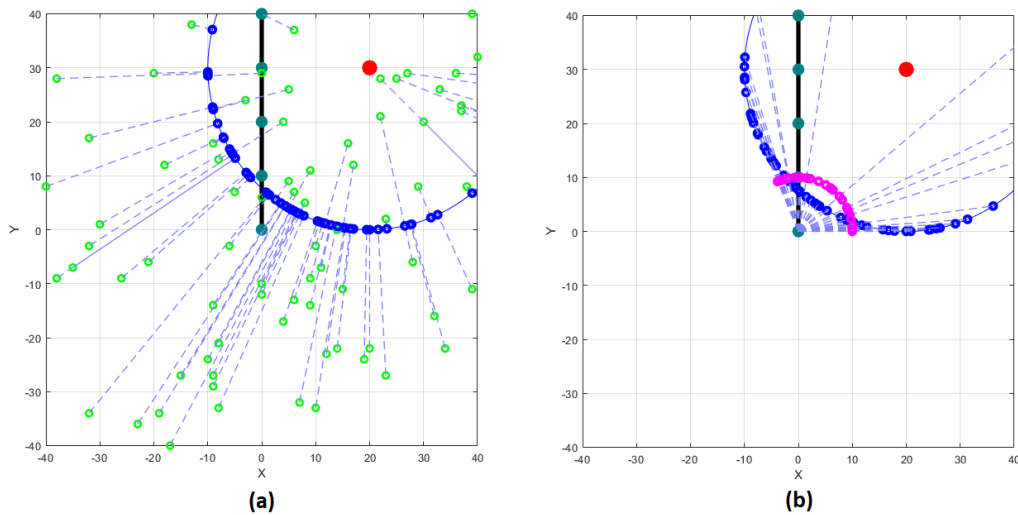


Figura 17 – Amostragem em direção as regiões de interesse. (a) Os pontos em verde estão representadas as amostras sorteadas (x_{rand}) e os pontos em azul representam a projeção dos pontos no círculo da junta em questão, (b) os pontos em roxo representam os novos nós adicionados à árvore (x_{novo}).

Ao analisar a Figura 17 (b) é notório que essa mudança no processo de amostragem gera um viés de crescimento da árvore em direção à região de solução. A fim de medir o impacto dessa alteração na distribuição de probabilidade, foram geradas 100000 amostras utilizando essa nova forma de amostragem para gerar uma estimativa de sua FDP, mostrada na Figura 18.

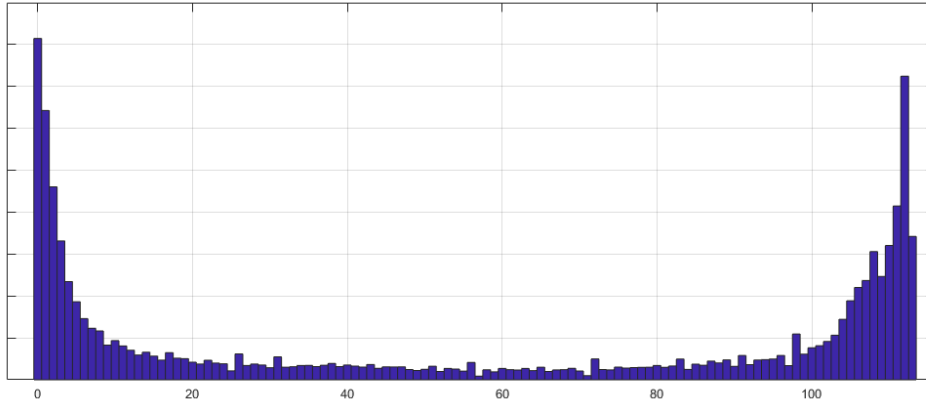


Figura 18 – Histograma gerado a partir da amostragem no espaço de trabalho seguida de uma projeção na região de interesse.

O histograma da Figura 18 mostra que para essa nova configuração de sorteio, há dois picos elevados de probabilidade nas extremidades e uma frequência de ocorrência muito baixa na região central. Sendo assim, na tentativa de obter uma FDP mais uniforme, uma nova configuração de sorteio foi testada, onde a amostragem ocorre de maneira uniforme diretamente na superfície da região de interesse, como pode ser visto na Figura 19. Esse sorteio é realizado conforme

$$x_{rand} = (x_{dest1} + r\cos\theta, x_{dest2} + r\sin\theta), \quad (4.1)$$

em que r é o comprimento restante do robô e θ é o parâmetro a ser sorteado.

Ao analisar a Figura 19, percebe-se que o viés em direção à região de solução se mantém. Além disso, essa abordagem apresenta outras vantagens, pois evita o custo de projeção no círculo e o sorteio possui uma variável a menos pois o tamanho do raio é fixo. No que diz respeito à distribuição de probabilidade, foram geradas 100000 amostras na superfície da região de solução para gerar a estimativa da FDP dessa abordagem, mostrada na Figura 20.

O histograma da Figura 20 mantém os dois picos de probabilidade nas extremidades, no entanto, eles tem importância relativamente menor do que o observado na estimativa anterior (Figura 18), o que implica em uma distribuição mais uniforme nas regiões centrais. É possível perceber que todas as configurações excluídas com essa FDP não poderiam apresentar solução de forma alguma. Isso fica evidente ao analisar a Figura 19 na qual o

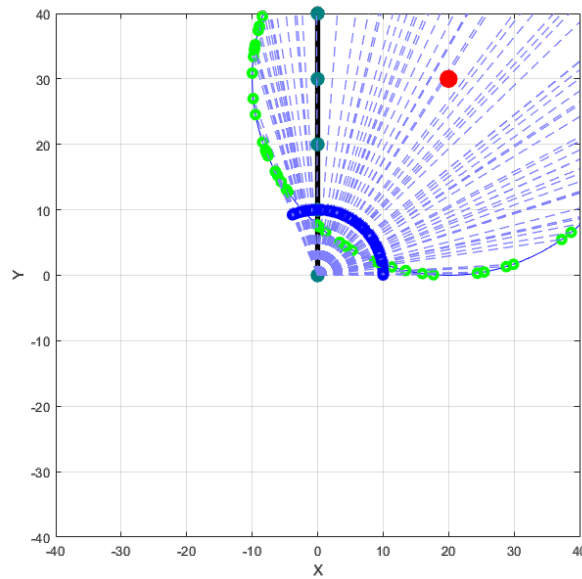


Figura 19 – Amostragem na superfície da região de interesse. O ponto vermelho representa o destino (x_{dest}), em verde estão representadas as amostras sorteadas na superfície (x_{rand}) e os pontos em azul representam os novos nós adicionados à árvore (x_{novo}).

raio do círculo azul equivale ao restante do manipulador estendido e toda configuração fora desse círculo implicaria em uma distância maior que a extensão do restante da cadeia, impossibilitando assim o cumprimento da tarefa.

Apesar disso, a região gerada não visa excluir todas as configurações que não solucionam o problema, mas sim garantir que todas as possíveis soluções estejam contidas nessa região. Desta forma, como todas as possíveis soluções estão dentro do espaço amostral e com probabilidade de ocorrência não-nula, a prova de completude probabilística da RRT continua válida para esse novo modelo de probabilidade. Vale ressaltar que apesar da amostragem na região da superfície produzir uma polarização no crescimento da árvore

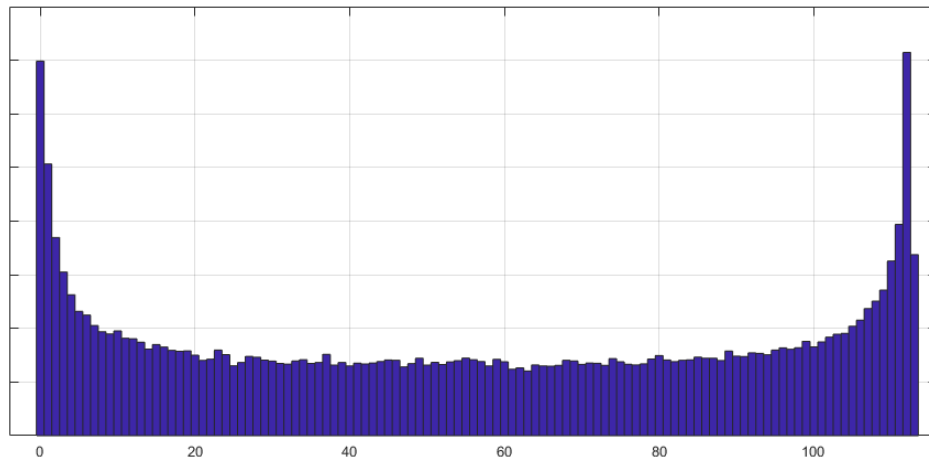


Figura 20 – Histograma gerado a partir da amostragem na superfície da região de solução.

ela não resulta em mínimos locais, uma vez que o método é probabilisticamente completo e não há um interesse em projetar trajetórias nesse espaço de busca, mas sim encontrar uma amostra nesse espaço.

Outras abordagens propõem uma polarização no crescimento da árvore no C_{space} para solução do problema de cinemática inversa (BERTRAM et al., 2006; WEGHE; FERGUSON; SRINIVASA, 2007). Entretanto, como a configuração de destino não é conhecida nesse espaço, essas abordagens dependem de uma mudança de espaço, que está ligada a necessidade de métricas complicadas de distância, tornando a tarefa não-trivial. A falta de uma configuração de destino nesse espaço também dificulta a delimitação de regiões de interesse e não é plausível a garantia de se obter uma região de busca que contém todas as soluções possíveis.

A partir da mudança de representação da RRT, do espaço de configurações para o espaço de trabalho, é possível gerar um equivalente ao processo de polarização baseado nas características físicas do manipulador que permite a construção de modelos de probabilidade para o crescimento da árvore que são mais eficientes sem um alto custo computacional. A amostragem na superfície da região de interesse se destaca devido a limitação do espaço de busca contendo todas as soluções possíveis, que só é possível através da mudança de representação proposta nesse trabalho. Ainda vale ressaltar que a proposta de polarização aqui apresentada não é fruto de um processo de reamostragem como normalmente é feito em outras abordagens probabilísticas (ROKBANI; ALIM, 2013).

Na Figura 21 é descrita uma iteração completa da RRT no espaço de trabalho com a regra de amostragem na superfície da região de solução. Esse ambiente tem como ponto de destino (20,30) e um manipulador planar (4-DOF) com parâmetros de Denavit-Hartenberg descritos na Tabela 1.

Tabela 1 – Parâmetros de Denavit-Hartenberg do manipulador planar (4-DOF).

i	$a_i(cm)$	$\alpha_i(^{\circ})$	$d_i(cm)$	$\theta_i(^{\circ})(Range)$
1	10	0	0	$-180 < \theta_1 < 180$
2	10	0	0	$-180 < \theta_2 < 180$
3	10	0	0	$-180 < \theta_3 < 180$
4	10	0	0	$-180 < \theta_4 < 180$

O primeiro passo da abordagem é definir a região de solução para o primeiro sorteio. Dado que após a junta 1 restam 30cm de comprimento para o manipulador, a primeira região é representada por um círculo de raio 30 e centro em x_{dest} , Figura 21(a). Utilizando a equação (4.1), um ponto é sorteado na superfície da região de solução (x_{rand}) e então o crescimento da árvore é feito em direção ao sorteio com tamanho de passo igual ao elo d_1 , Figura 21(b). O mesmo procedimento é realizado pro sorteio dos nós de índices 2 e 3 com raio de 20cm e 10cm, respectivamente, Figura 21(c) e (d).

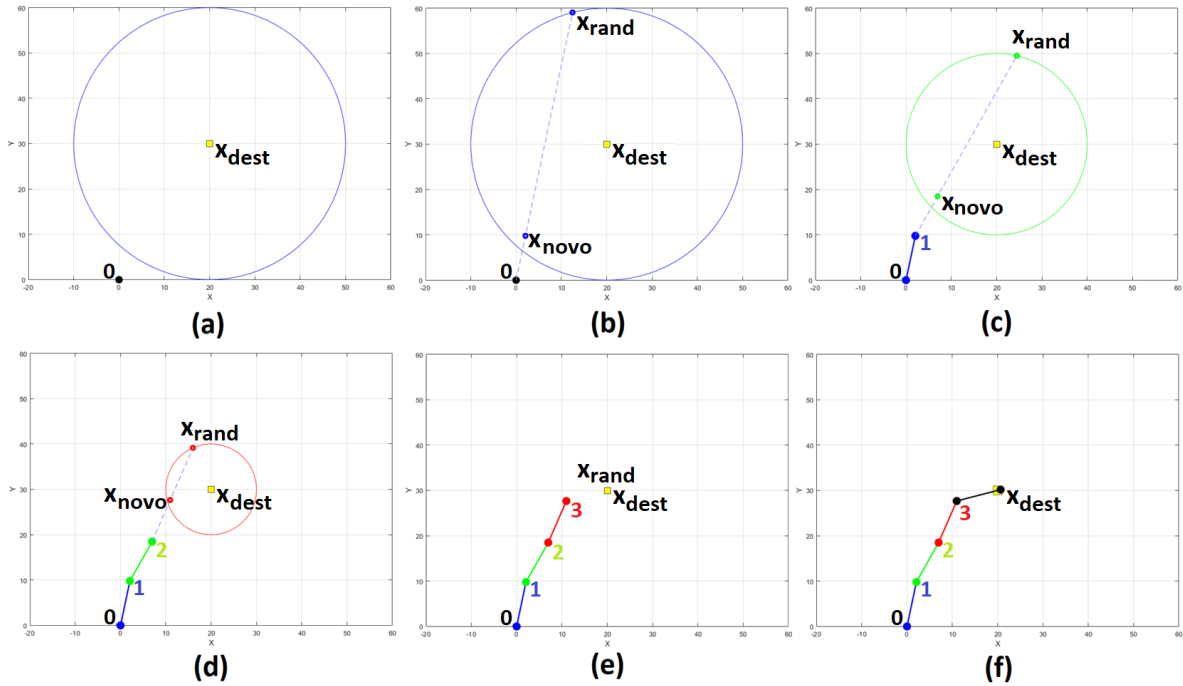


Figura 21 – Iteração completa da RRT no espaço de trabalho com a regra de amostragem na superfície da região de solução. (a) Ponto de destino (em amarelo), base como nó de índice 0 (em preto) e região de solução (círculo azul). (b) sorteio de índice 1 na região de solução e adição do novo nó x_{novo} a árvore com tamanho de passo limitado ao tamanho do elo d_1 , (c) sorteio de índice 2 na região de solução e adição do novo nó x_{novo} , (d) sorteio de índice 3 na região de solução e adição do novo nó x_{novo} , (e) sorteio de índice 4 como o próprio destino pois ele é a única região de solução e (f) fim da iteração.

No sorteio do último nó ocorre um evento interessante. Como ele representa o efetuator, não há comprimento do manipulador após ele, o que faz com que o raio do círculo seja 0 e o sorteio ocorre sempre no ponto de destino. Essa é uma característica importante para o sorteio do último nó, pois nenhuma outra direção pode gerar solução. Após limitar o crescimento no tamanho do passo, a primeira iteração é finalizada e então é feita a etapa de checagem, Figura 21(e) e (f). Caso não haja solução o processo é repetido até que uma solução seja encontrada.

4.4 Regra de Conexão

A aplicação da RRT diretamente no espaço de trabalho possibilitou alterações que não seriam aplicáveis no espaço de configurações e facilitou a visualização de possíveis formas de otimização da técnica baseada em seus aspectos físicos. Analisando o crescimento da árvore foi possível propor uma alteração na regra de conexão da RRT clássica de modo a otimizar o resultado final.

A RRT clássica tem como regra de conexão o nó da árvore de menor distância euclidiana para o nó sorteado. Uma ilustração de como ocorre essa conexão é mostrada na Figura 22.

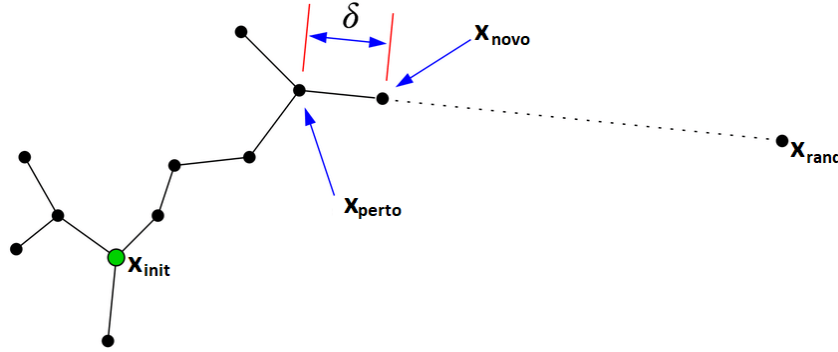


Figura 22 – Regra de conexão para RRT clássica.

Entretanto, na aplicação da RRT como uma representação espacial dos manipuladores, uma mudança nesse critério de seleção pode melhorar o desempenho do método. A proposta investigada nesse trabalho foi buscar o nó que tenha a distância mais próxima do tamanho do elo de índice i , conforme

$$\min(\|distancia_euclidiana(\acute{A}rvore, x_{rand}) - d_i\|). \quad (4.2)$$

Com essa mudança pretende-se aumentar a probabilidade do nó sorteado ser inserido na árvore. Essa mudança foi pensada após a percepção de que o sorteio do último nó sempre resulta no ponto de destino, o que aumenta a probabilidade de se adicionar o destino à árvore e resolver o problema de cinemática inversa. Na Figura 23 é ilustrado o funcionamento da nova regra de seleção proposta nesse trabalho.

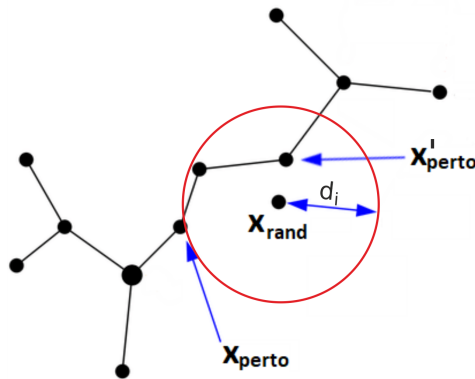


Figura 23 – A regra de conexão proposta nesse trabalho delimita um círculo de raio d_i e busca o nó mais próximo do círculo x_{perto} . Caso a regra de conexão da RRT clássica fosse utilizada, o nó x'_{perto} seria o escolhido.

5 Resultados e Discussão

Neste capítulo serão apresentados os resultados dos experimentos realizados com o objetivo de demonstrar o comportamento das técnicas desenvolvidas e ilustrar a importância dos parâmetros alterados do algoritmo clássico. Os experimentos realizados foram obtidos através de simulação utilizando a abordagem clássica da RRT no espaço de configurações e a RRT aplicada diretamente no espaço de trabalho (proposta nessa dissertação). Uma descrição dos termos utilizados nesse trabalho para representar as variantes da RRT é apresentada na Tabela 2.

Tabela 2 – Descrição das abordagens da RRT analisadas nesse trabalho.

Abordagem	Descrição
RRT C_{Space}	Abordagem clássica da RRT aplicada no espaço de configurações.
RRT Uniforme	Aplicação da RRT no espaço de trabalho com amostragem uniforme.
RRT Superfície	Aplicação da RRT no espaço de trabalho com amostragem nas regiões de solução.

Utilizando a RRT no espaço de trabalho, foram testadas a amostragem uniforme, a amostragem na região de solução (proposta nessa dissertação), a regra de conexão de nó mais próximo, a regra de conexão que busca o nó em que a distância para a amostra mais se aproxima do tamanho do elo (proposta nessa dissertação) e a influência de sortear o último nó como o próprio destino. Uma descrição dos termos utilizados para representar os aspectos investigados da RRT é apresentada na Tabela 3.

Tabela 3 – Descrição das alterações da RRT analisadas nesse trabalho.

Aspecto	Descrição
Regra de Conexão A	A árvore expande a partir do nó mais próximo da amostra sorteada
Regra de Conexão B	A árvore expande a partir do nó em que a distância para a amostra sorteada mais se aproxima do tamanho do elo i .
Amostra no Destino	A amostra correspondente ao nó de índice n é sempre o próprio objetivo. Esse aspecto é representado nas tabelas com o símbolo (*)

Todos os experimentos apresentados nesse capítulo serão realizados em 2D, como na maioria dos trabalhos em RRT, para facilitar a visualização do resultado do método. A expansão pra um espaço de 3 dimensões é análoga e problemas secundários, como restrições de movimentação e limites de junta, não serão tratados nesse trabalho.

5.1 Configurações dos Experimentos

Dado que as abordagens investigadas nesse trabalho têm características probabilísticas, a execução do algoritmo sob as mesmas condições pode produzir diferentes resultados. Com isso, para produzir resultados representativos, é necessário executar várias vezes o mesmo experimento, de modo que estes representem de forma aproximada o efeito da mudança nos parâmetros.

Adicionalmente, em cada experimento foi gerada uma nuvem de pontos distribuída uniformemente no espaço alcançável do manipulador. Esses pontos foram utilizados como pontos de destino nas várias execuções dos experimentos. É importante ressaltar que pra cada ponto de destino avaliado, o ambiente é totalmente livre de obstáculos.

Os testes foram aplicados em dois modelos diferentes de robôs manipuladores. O primeiro manipulador possui 3-DOF e tem um modelo descrito pelos parâmetros de Denavit-Hartenberg na Tabela 4. Com base neste modelo, foram gerados 35 pontos de destino distribuídos uniformemente no espaço alcançável do manipulador (Figura 24). Para cada ponto de destino foram determinadas duas condições de parada, 10000 iterações ou uma distância para o destino que equivale a 1% do comprimento do manipulador (0,3 cm).

Tabela 4 – Parâmetros de Denavit-Hartenberg que descreve o modelo do robô manipulador de 3-DOF utilizado nos experimentos.

i	$a_i(cm)$	$\alpha_i(^{\circ})$	$d_i(cm)$	$\theta_i(^{\circ})(Range)$
1	10	0	0	$-180 < \theta_1 < 180$
2	10	0	0	$-180 < \theta_2 < 180$
3	10	0	0	$-180 < \theta_3 < 180$

O segundo manipulador testado possui 10-DOF e tem um modelo descrito pelos parâmetros de Denavit-Hartenberg na Tabela 5. Com base neste modelo, foram gerados 100 pontos de destino distribuídos uniformemente no espaço alcançável do manipulador (Figura 25). Para cada ponto de destino foram determinadas duas condições de parada, 10000 iterações ou uma distância para o destino que equivale a 1% do comprimento do manipulador (1 cm).

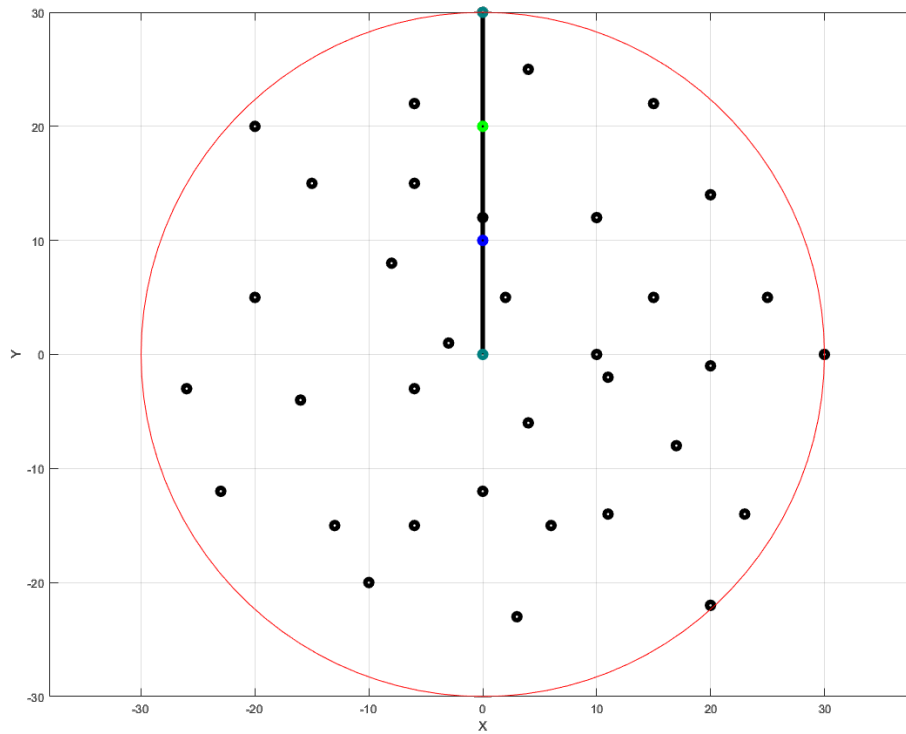


Figura 24 – Ambiente de experimento para o robô manipulador de 3-DOF. Os pontos em preto representam os pontos de destino distribuídos uniformemente no espaço alcançável.

O desempenho das abordagens foi medido em termos de taxa de convergência (T_C), média de iterações (μ) e a quantidade de nós adicionados fora da região de solução (ρ). O parâmetro ρ foi adicionado pois buscas realizadas fora da região de solução são desnecessárias e ele evidencia o ganho proporcionado pela mudança do modelo de probabilidade proposta nesse trabalho. Para obter uma maior confiabilidade nesse índice, todas as abordagens foram testadas uma segunda vez apenas com a condição de parada de 10000 iterações (ρ_{10000}).

Tabela 5 – Parâmetros de Denavit-Hartenberg que descreve o modelo do robô manipulador de 10-DOF utilizado nos experimentos.

i	$a_i(cm)$	$\alpha_i(^{\circ})$	$d_i(cm)$	$\theta_i(^{\circ})(Range)$
1	10	0	0	$-180 < \theta_1 < 180$
2	10	0	0	$-180 < \theta_2 < 180$
3	10	0	0	$-180 < \theta_3 < 180$
4	10	0	0	$-180 < \theta_4 < 180$
5	10	0	0	$-180 < \theta_5 < 180$
6	10	0	0	$-180 < \theta_6 < 180$
7	10	0	0	$-180 < \theta_7 < 180$
8	10	0	0	$-180 < \theta_8 < 180$
9	10	0	0	$-180 < \theta_9 < 180$
10	10	0	0	$-180 < \theta_{10} < 180$

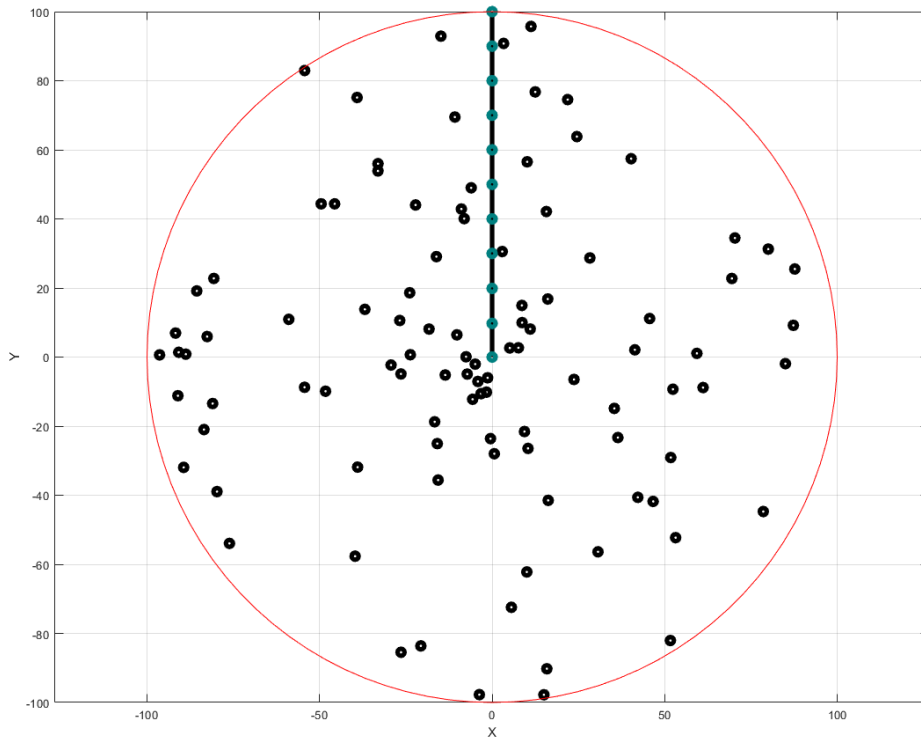


Figura 25 – Ambiente de experimento para o robô manipulador de 10-DOF. Os pontos em preto representam os pontos de destino distribuídos uniformemente no espaço alcançável.

5.2 Amostragem

O primeiro parâmetro a ser analisado nos experimentos é o impacto da nova regra de amostragem. Vale ressaltar que a mudança dos parâmetros não impactam na RRT aplicada diretamente no C_{space} pois as alterações foram propostas baseadas nas características físicas do manipulador e essa interpretação não faz sentido no espaço de configurações. Desta forma, todos resultados apresentados da RRT no C_{space} estão sob as condições da RRT clássica.

Sendo assim, foram avaliadas três tipos de abordagem nesse primeiro quesito, a RRT no espaço de configurações (RRT C_{space}), RRT no espaço de trabalho com amostragem uniforme (RRT Uniforme) e a RRT no espaço de trabalho com amostragem na superfície da região de solução (RRT Superfície).

5.2.1 Experimento 3-DOF

Os resultados obtidos nesse primeiro experimento são apresentados na Tabela 6. Analisando os resultados, é possível notar que, sob essas condições, nenhum dos algoritmos foi capaz de convergir para todos os pontos de destino. Isso se deve ao fato de que a busca probabilística garante uma solução para um número infinito de sorteios. Como a

quantidade de iterações foi limitada a 10000, nos casos onde os algoritmos não conseguiram convergir para uma solução dentro dessa quantidade de iterações, a tarefa foi finalizada com um resultado de não convergência.

Tabela 6 – Desempenho em média das variantes da RRT para um manipulador de 3-DOF.

	T_C	μ	ρ	ρ_{10000}
RRT C_{space}	28,5714%	5074,6154	0,64941	0,66067
RRT Uniforme	14,2857%	2681,2731	0,66690	0,66791
RRT Superfície	34,2857%	5,4286	0,22309	0,21782

Com a menor taxa de convergência (T_C) das abordagens analisadas, a RRT Uniforme mostra que uma simples reinterpretação da RRT para o espaço de trabalho não garante um melhor desempenho do que a RRT no espaço de configurações. Por outro lado, a amostragem na superfície de solução apresenta uma T_C superior à RRT C_{space} e um número de iterações para convergir consideravelmente menor, esperado por conta das polarizações explícitas e implícitas nos sorteios. Além disso, o valor de ρ indica que aproximadamente 80% dos nós foram adicionados na região de solução. De fato, a maior convergência e o baixo número de iterações estão relacionados a um valor inferior de ρ pois ele indica que a busca está sendo realizada somente nas regiões de interesse o que diminui bastante o espaço de busca.

5.2.2 Experimento 10-DOF

Os resultados obtidos com o manipulador de 10-DOF são apresentados na Tabela 7. Analisando os resultados, é possível notar que o desempenho das abordagens aplicadas no espaço de trabalho foram superiores ao apresentado na Tabela 6, enquanto que a RRT C_{space} obteve um desempenho inferior.

O ganho de desempenho das abordagens aplicadas no espaço de trabalho se deve ao fato de que nos experimentos com o robô 10-DOF o valor de tolerância foi alterado para 1cm. O mesmo não pode ser observado na RRT C_{space} pois, apesar do maior valor de tolerância, seu espaço de busca foi alterado de 3 para 10 dimensões. Desta forma a alta dimensionalidade associada ao uso do espaço de configurações contribui de maneira negativa para o desempenho da técnica. Esse aspecto evidencia a vantagem de realizar a redução de dimensionalidade pois o aumento de graus de liberdade do manipulador não apresentam impacto significativo na aplicação da RRT no espaço de trabalho.

De modo semelhante ao que aconteceu no experimento para o robô 3-DOF, a RRT superfície apresentou um desempenho superior com a mudança no modelo de probabilidade evidenciado pelo baixo número de iterações (μ) e uma T_C de 57%. A menor ocorrência de nós fora da região de solução, evidenciada pelo baixo valor de ρ , contribui significativamente para essa otimização.

Tabela 7 – Desempenho em média das variantes da RRT para um manipulador de 10-DOF.

	T_C	μ	ρ	ρ_{10000}
RRT C_{space}	21%	3309,0476	0,54964	0,55211
RRT Uniforme	53%	2813,2075	0,58094	0,57329
RRT Superfície	57%	7,1053	0,22309	0,21780

Apesar do bom desempenho, a taxa de convergência obtida está longe da desejada, pois em quase metade dos pontos não foi possível convergir para uma solução. Como a RRT Superfície precisa de poucas iterações para chegar ao destino, os casos de não-convergência aparentam sofrer com o problema de mínimo local. Isso se deve à amostra do último nó como o próprio destino e a regra de conexão com o nó mais próximo, pois caso um nó de índice $n - 1$ esteja a uma distância inferior ao tamanho do elo d_n , não será possível encontrar uma solução.

5.3 Regra de Conexão e Amostra no destino

Nos experimentos realizados para a regra de conexão, dois modos de conexão foram avaliados. O primeiro modo foi a abordagem clássica da RRT, onde a conexão é feita com o nó mais próximo (Conexão A), que vinha sendo utilizada em todos os experimentos até então. Já o segundo modo realiza a conexão com o nó em que a distância para a amostra mais se aproxima do tamanho do elo (Conexão B).

No que diz respeito à RRT Uniforme, a abordagem foi avaliada com e sem a amostra do último nó no objetivo. A nomenclatura utilizada nas tabelas para indicar a utilização do último nó como o próprio objetivo foi uma estrela (Conexão A*). Vale mencionar que, assim como nos outros parâmetros, a aplicação dessa alteração no espaço de configurações não possui fundamento pois o tamanho do elo não tem sentido físico naquele espaço. Além disso, para o caso específico da RRT Superfície não foi utilizada a abordagem de conexão do último nó diretamente no destino de forma explícita, por ser esta uma característica já presente no método, como explicado na seção 4.3.

5.3.1 Experimento 3-DOF

Os resultados apresentados na Tabela 8 relacionam o desempenho de todas as alterações investigadas nesse trabalho. A inserção da nova regra de seleção (Conexão B) produz uma melhor taxa de convergência T_C em todos os cenários possíveis e elimina o problema de mínimo local. Quanto ao seu desempenho na RRT Uniforme, apesar de se obter uma maior T_C o número médio de iterações μ aumenta para a nova regra. Já na RRT Superfície a nova regra de seleção possui um desempenho superior ou equivalente em todos os quesitos.

Tabela 8 – Desempenho em média das variantes da RRT para um manipulador de 3-DOF levando em consideração todas as alterações propostas nesse capítulo.

	T_C	μ	ρ	ρ_{10000}
RRT C_{space}	28,5714%	5074,6154	0,64941	0,66067
RRT Uniforme				
Conexão A	14,2857%	2681,2577	0,66690	0,66791
Conexão B	34,2857%	3742,0833	0,66007	0,65968
Conexão A*	5,7142%	74,5346	0,66692	0,66791
Conexão B*	100%	245,6286	0,64211	0,66027
RRT Superfície				
Conexão A	34,2857%	5,4286	0,195240	0,203200
Conexão B	100%	6,7844	0,139750	0,078041

Outro ganho evidente dessa nova regra de seleção em conjunto com a RRT Superfície é um número menor de nós fora da região de solução ρ . Ao analisar o valor de ρ para 10000 amostras (ρ_{10000}) ele se mantém em torno de 8% das amostras o que reforça o ganho da FDP gerada com a amostragem na região de solução. Por fim, é possível notar um bom desempenho da RRT Uniforme com a Conexão B* o que indica que as alterações propostas nesse trabalho já apresentam bons resultados mesmo sem uma mudança no modelo de probabilidade.

5.3.2 Experimento 10-DOF

Ao aplicar a técnica em manipuladores com mais graus de liberdade fica ainda mais evidente a vantagem de se aplicar a mudança no modelo de probabilidade proposta nesse trabalho. Na Tabela 9 são apresentados os resultados dos experimentos realizados para o robô de 10-DOF. Nesses testes, em nenhum cenário a RRT Uniforme foi capaz de encontrar todos os objetivos, mas, a variação Conexão B* continua apresentando os melhores resultados. No que diz respeito a RRT superfície, os dados apontam para um ótimo desempenho técnica ao utilizar a regra de conexão B.

Diante de todos os experimentos realizados, constata-se que a variante Conexão B da RRT Superfície apresenta os melhores resultados para resolução do problema de cinemática inversa. Ela é uma técnica simples e de fácil implementação, possui uma rápida convergência e garante que uma solução vai ser encontrada para um número infinito de sorteios. Com isso, neste trabalho nós propomos essa variante como uma nova técnica probabilística de cinemática inversa baseada na RRT, a *Manipulator-Based Rapidly-Exploring Random Tree* (MB-RRT).

Tabela 9 – Desempenho em média das variantes da RRT para um manipulador de 10-DOF levando em consideração todas as alterações propostas nesse capítulo.

	T_C	μ	ρ	ρ_{10000}
RRT C_{space}	21%	3309,0476	0,54964	0,55211
RRT Uniforme				
Conexão A	53%	2813,2075	0,58094	0,57329
Conexão B	45%	2546,4119	0,57982	0,56532
Conexão A*	38%	1671,5263	0,57409	0,57756
Conexão B*	71%	1333,0282	0,56263	0,56213
RRT Superfície				
Conexão A	57%	7,1053	0,22309	0,217820
Conexão B	100%	6,6812	0,24629	0,083249

5.4 Manipulator-Based Rapidly-Exploring Random Tree (MB-RRT)

Após a definição da melhor composição da MB-RRT foram realizados novos experimentos com diferentes valores de tolerância (0,3 *cm*, 0,01 *cm* e 0,001 *cm*.) para medir a capacidade da técnica de apresentar uma solução diante de tarefas que exijam mais precisão.

Tabela 10 – Desempenho em média da MB-RRT para um manipulador planar de 3-DOF com diferentes valores de tolerância.

MB-RRT	T_C	μ	ρ	ρ_{10000}
0,3 <i>cm</i>	100%	6,7844	0,139750	0,078041
0,01 <i>cm</i>	100%	31,0286	0,104440	0,078041
0,001 <i>cm</i>	100%	95,7429	0,083567	0,078041

Ao analisar os resultados obtidos para o robô 3DOF (Tabela 10) é possível notar que a abordagem continua encontrando uma solução em todos os casos com uma quantidade média de iterações baixas. Vale mencionar também que quanto maior o tempo de busca, menor a porcentagem de nós existentes fora da região de solução o que indica um viés de crescimento em direção à região de solução.

Apesar dos bons resultados, esse modelo é bastante simples pois se trata de um robô de apenas 3-DOF e espera-se que a dificuldade da tarefa aumente consideravelmente com o aumento do número de juntas. Diante disso, foram realizados também os experimentos para o modelo do manipulador de 10-DOF com valores de tolerância em 1 *cm*, 0,01 *cm* e 0,001 *cm*.

Tabela 11 – Desempenho em média da MB-RRT para um manipulador planar de 10-DOF com diferentes valores de tolerância.

MB-RRT	T_C	μ	ρ	ρ_{10000}
1 <i>cm</i>	100%	6,68	0,24629	0,083249
0,01 <i>cm</i>	100%	34,33	0,18359	0,083249
0,001 <i>cm</i>	100%	95,22	0,14158	0,083249

Os resultados obtidos na Tabela 11 são bastantes similares aos encontrados com o robô 3-DOF. Isso se deve ao fato de que mesmo com um aumento no número de juntas, o espaço de busca continua sendo de apenas duas dimensões pois a técnica se aplica no espaço de trabalho. Essa característica traz a vantagem de que a área de busca terá no máximo 3 dimensões para qualquer tipo de manipulador, enquanto a RRT clássica tem espaço de busca igual à quantidade de graus de liberdade do manipulador. Desta forma, a MB-RRT se destaca por manter suas principais características mesmo com o aumento do número de juntas e com valores de tolerância menores.

6 Conclusões

Nesse trabalho foi apresentado o desenvolvimento de um novo método probabilístico de cinemática inversa baseado na RRT, a *Manipulator-Based Rapidly-exploring Random Tree* (MB-RRT). A MB-RRT produz uma configuração que alcança o objetivo traçado no espaço de trabalho, mas, não gera uma trajetória para conectar uma configuração inicial à uma configuração de destino. Apesar disso, abordagens de cinemática inversa que possuem tal característica podem ser utilizadas como etapa *a priori* para métodos de planejamento de caminho ou geração de trajetórias que vão ser executados no espaço de configurações.

A abordagem proposta realiza uma busca diretamente no espaço de trabalho através do crescimento da árvore no formato espacial do manipulador, de forma que os nós representam as juntas e as arestas os elos do robô. Essa interpretação da árvore como o próprio manipulador é particularmente interessante pois possibilita uma fácil interpretação do significado das componentes da árvore, propiciando a elaboração de alterações baseadas nas características físicas do problema, como as que foram investigadas nesse trabalho. É válido mencionar que todas as otimizações propostas com base nas características físicas só fazem sentido no espaço de trabalho pois elas são abstraídas na criação do espaço de configurações.

Ainda sobre as vantagens de se aplicar a técnica diretamente no espaço de trabalho, há um ganho evidente quanto à dimensionalidade do espaço de busca. Uma vez que o espaço de configurações possui espaços de altas dimensões para grande parte dos robôs manipuladores, a execução da técnica no espaço de trabalho garante que o espaço de busca pode ter no máximo 3 dimensões, para uma tarefa de posição do efetuador, o que contribui significativamente para uma busca mais efetiva.

Outro ganho significativo da interpretação da RRT no espaço de trabalho, se evidencia na polarização do modelo de probabilidade em direção as regiões de solução. De fato, delimitar essas regiões onde estão presentes todas as possíveis soluções no C_{space} é uma tarefa improvável, mas ela se torna possível e intuitiva com a mudança de representação apresentada nesse trabalho. Com isso, foi proposta uma alteração no processo de amostragem, de modo a polarizar o crescimento da árvore em direção às regiões de solução, tornando a busca ainda mais eficiente e mantendo sua completude probabilística.

A partir da nova forma de amostragem proposta, também foi possível perceber que o sorteio do último nó como o próprio destino resulta em um ganho significativo no tempo de convergência da abordagem pois exclui outras direções desnecessárias que não teriam capacidade de gerar solução. Vale ressaltar que nenhuma das alterações propostas

compromete a completude probabilística da técnica pois elas evitam o crescimento da árvore em direções que não permitem soluções, mas garantem que todos os nós de solução existentes possam ocorrer.

Por fim, a MB-RRT ainda potencializa sua busca ao associar a nova regra de amostragem com uma nova regra de conexão. Como a amostragem proposta realiza os sorteios na superfície da região de interesse, a nova regra de conexão deixa de buscar o nó mais perto da amostra sorteada para buscar o nó que vai gerar um novo ramo que mais se aproxime da amostra sorteada. O impacto dessa nova regra é facilmente percebido ao imaginar que ao sortear o ponto de destino a nova regra de conexão busca o nó que posso gerar um ramo que mais se aproxime do destino, acelerando o processo de convergência.

Os experimentos realizados comprovam a eficácia da MB-RRT e evidenciam características como simplicidade, baixo custo computacional, baixo número de iterações e completude probabilística. Outros aspectos desejados em técnicas de cinemática inversa se mostram promissores com essa abordagem, dado que algumas variações da RRT já lidam com problemas similares. Além disso, todos os experimentos foram conduzidos sem obstáculos, pois, o principal objetivo desse trabalho foi dar o primeiro passo na direção da técnica proposta e mostrar que essa representação permite visualizar de forma intuitiva uma polarização do modelo de probabilidade. Diante disso, investigações sobre a adaptação de variações da RRT e o desempenho da abordagem considerando um ambiente com obstáculos serão considerados em trabalhos futuros.

Com base nos resultados obtidos nesse trabalho, a técnica MB-RRT possui potencial para apresentar um bom desempenho em comparação com as principais técnicas de IK utilizada atualmente, como o FABRIK, CCD e PSO. Sendo assim, em trabalhos futuros pretende-se realizar uma comparação da abordagem proposta com as principais técnicas de solução do problema de IK, incorporar objetivos secundários comuns no contexto da cinemática inversa de manipuladores e avaliar a inserção de restrições de movimentação no espaço de 3 dimensões.

Apesar do uso da MB-RRT no espaço 3D não ser testado nesse trabalho, não significa que ela não seja aplicável em um espaço de 3 dimensões. A extrapolação da técnica do espaço 2D para o 3D pode ser realizada através da alteração da região de solução da superfície de um círculo para a superfície de uma esfera. No entanto, essa aplicação desconsidera a existência de restrições de movimentação das juntas o que implica que maiores estudos são necessários para avaliar o impacto de restrições impostas por juntas de 1-DOF em robôs no espaço 3D.

Outro ponto a ser investigado é o modelo de probabilidade proposto nesse trabalho, pois ele pode apresentar dois picos de probabilidade nas extremidades. Uma alternativa pensada para eliminar esses picos e garantir que os sorteios sempre estão contidos na região de interesse seria verificar os ângulos que representam a intersecção entre a superfície da

região de interesse e a superfície do círculo alcançável da junta em questão. Entretanto, esse procedimento aumentaria consideravelmente o custo computacional do algoritmo que dificultaria o processo de aplicação da técnica em um espaço 3D.

Sendo assim, em trabalhos futuros pretende-se investigar a alteração do modelo de probabilidade gerado pela amostragem na superfície da região de solução pois ele prioriza o crescimento da árvore em direção a regiões limites o que aumenta a probabilidade de se ter um sorteio fora da região de interesse. Com isso, espera-se obter uma busca mais concentrada nas regiões que amplificam a quantidade de soluções possíveis.

Referências

- AHMADYAN, S. N.; KUMAR, J. A.; VASUDEVAN, S. Goal-oriented stimulus generation for analog circuits. In: *Proceedings of the 49th Annual Design Automation Conference*. [S.l.: s.n.], 2012. p. 1018–1023. Citado na página 17.
- ALEOTTI, J.; CASELLI, S. Physics-based virtual reality for task learning and intelligent disassembly planning. *Virtual reality*, Springer, v. 15, n. 1, p. 41–54, 2011. Citado na página 17.
- ARISTIDOU, A.; CHRYSANTHOU, Y.; LASENBY, J. Extending fabrik with model constraints. *Computer Animation and Virtual Worlds*, Wiley Online Library, v. 27, n. 1, p. 35–57, 2016. Citado na página 24.
- ARISTIDOU, A.; LASENBY, J. Inverse kinematics: a review of existing techniques and introduction of a new fast iterative solver. University of Cambridge, Department of Engineering, 2009. Citado 2 vezes nas páginas 22 e 24.
- ARISTIDOU, A.; LASENBY, J. Fabrik: A fast, iterative solver for the inverse kinematics problem. *Graphical Models*, Elsevier, v. 73, n. 5, p. 243–260, 2011. Citado na página 23.
- ARISTIDOU, A. et al. Inverse kinematics techniques in computer graphics: A survey. In: WILEY ONLINE LIBRARY. *Computer Graphics Forum*. [S.l.], 2018. v. 37, n. 6, p. 35–58. Citado 5 vezes nas páginas 2, 21, 22, 23 e 26.
- AYYILDIZ, M.; ÇETINKAYA, K. Comparison of four different heuristic optimization algorithms for the inverse kinematics solution of a real 4-dof serial robot manipulator. *Neural Computing and Applications*, Springer, v. 27, n. 4, p. 825–836, 2016. Citado na página 2.
- BAUER, A. W.; POPOVIĆ, Z. Rrt-based game level analysis, visualization, and visual refinement. In: *Eighth artificial intelligence and interactive digital entertainment conference*. [S.l.: s.n.], 2012. Citado na página 17.
- BERTRAM, D. et al. An integrated approach to inverse kinematics and path planning for redundant manipulators. In: IEEE. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. [S.l.], 2006. p. 1874–1879. Citado 3 vezes nas páginas 21, 25 e 38.
- CARREIRA-PERPINÁN, M. A. A review of dimension reduction techniques. *Department of Computer Science. University of Sheffield. Tech. Rep. CS-96-09*, v. 9, p. 1–69, 1997. Citado na página 3.
- CHENG, P.; LAVALLE, S. M. Resolution complete rapidly-exploring random trees. In: IEEE. *Proceedings 2002 IEEE international conference on robotics and automation (cat. no. 02CH37292)*. [S.l.], 2002. v. 1, p. 267–272. Citado na página 20.

- CHIN, K. W.; KONSKY, B. V.; MARRIOTT, A. Closed-form and generalized inverse kinematics solutions for the analysis of human motion. In: IEEE. *Proceedings of the 19th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. 'Magnificent Milestones and Emerging Opportunities in Medical Engineering'* (Cat. No. 97CH36136). [S.l.], 1997. v. 5, p. 1911–1914. Citado na página 22.
- COURTY, N.; ARNAUD, E. Inverse kinematics using sequential monte carlo methods. In: SPRINGER. *International Conference on Articulated Motion and Deformable Objects*. [S.l.], 2008. p. 1–10. Citado na página 22.
- ELBANHAWI, M.; SIMIC, M. Sampling-based robot motion planning: A review. *Ieee access*, IEEE, v. 2, p. 56–77, 2014. Citado na página 19.
- FARAJI, S.; IJSPEERT, A. J. Singularity-tolerant inverse kinematics for bipedal robots: An efficient use of computational power to reduce energy consumption. *IEEE Robotics and Automation Letters*, IEEE, v. 2, n. 2, p. 1132–1139, 2017. Citado na página 23.
- FLETCHER, R. *Practical methods of optimization*. [S.l.]: John Wiley & Sons, 2013. Citado na página 22.
- FUNGE, J. D. *Artificial intelligence for computer games: an introduction*. [S.l.]: CRC Press, 2004. Citado na página 17.
- GAMMELL, J. D.; SRINIVASA, S. S.; BARFOOT, T. D. Informed rrt*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In: IEEE. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.], 2014. p. 2997–3004. Citado na página 19.
- GROCHOW, K. et al. Style-based inverse kinematics. In: *ACM SIGGRAPH 2004 Papers*. [S.l.: s.n.], 2004. p. 522–531. Citado na página 24.
- HAN, L.; DO, Q. H.; MITA, S. Unified path planner for parking an autonomous vehicle based on rrt. In: IEEE. *2011 IEEE International Conference on Robotics and Automation*. [S.l.], 2011. p. 5622–5627. Citado na página 19.
- HECKER, C. et al. Real-time motion retargeting to highly varied user-created morphologies. *ACM Transactions on Graphics (TOG)*, ACM New York, NY, USA, v. 27, n. 3, p. 1–11, 2008. Citado na página 22.
- HOLDEN, D.; SAITO, J.; KOMURA, T. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, ACM New York, NY, USA, v. 35, n. 4, p. 1–11, 2016. Citado na página 24.
- JUNIOR, J. d. L. S. et al. Frpso: Inverse kinematics using fully resampled particle swarm optimization. In: IEEE. *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*. [S.l.], 2018. p. 402–407. Citado 4 vezes nas páginas 1, 2, 3 e 25.
- KAMARRY, S. et al. Uma abordagem híbrida para navegação em ambientes dinâmicos baseadas na drrt. *2015 12th Simpósio Brasileiro de Automação Inteligente (SBAI)*, 2016. Citado na página 20.

- KAMARRY, S. et al. Drrt: Uma nova abordagem para aumentar a dispersao dos nós na rrt aplicadaa representacao do ambiente e planejamento de caminho. 2015. Citado na página 20.
- KANG, G. et al. Sampling-based path planning with goal oriented sampling. In: IEEE. *2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. [S.l.], 2016. p. 1285–1290. Citado na página 18.
- KARAMAN, S.; FRAZZOLI, E. Incremental sampling-based algorithms for optimal motion planning. *Robotics Science and Systems VI*, v. 104, n. 2, 2010. Citado na página 19.
- KENWRIGHT, B. Inverse kinematics–cyclic coordinate descent (ccd). *Journal of Graphics Tools*, Taylor & Francis, v. 16, n. 4, p. 177–217, 2012. Citado na página 23.
- KLEMM, S. et al. Rrt-connect: Faster, asymptotically optimal motion planning. In: IEEE. *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. [S.l.], 2015. p. 1670–1677. Citado na página 19.
- KUFFNER, J. J.; LAVALLE, S. M. Rrt-connect: An efficient approach to single-query path planning. In: IEEE. *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*. [S.l.], 2000. v. 2, p. 995–1001. Citado na página 18.
- KUMAR, A. *Advanced Control Techniques for Collaborative Manipulators*. 2020. Url <https://control.com/technical-articles/advanced-control-techniques-for-collaborative-manipulators/>. Citado 2 vezes nas páginas VII e 6.
- LANDER, J.; CONTENT, G. Making kine more flexible. *Game Developer Magazine*, v. 1, n. 15-22, p. 2, 1998. Citado na página 23.
- LATOMBE, J.-C. *Robot motion planning*. [S.l.]: Springer Science & Business Media, 2012. Citado 2 vezes nas páginas VII e 12.
- LAVALLE, S. M. Rapidly-exploring random trees: A new tool for path planning. Citeseer, 1998. Citado 5 vezes nas páginas 3, 13, 15, 17 e 29.
- LAVALLE, S. M.; KUFFNER, J. J. Rapidly-exploring random trees: Progress and prospects. *Algorithmic and computational robotics: new directions*, n. 5, p. 293–308, 2001. Citado na página 18.
- LOZANO-PEREZ, T. Spatial planning: A configuration space approach. In: *Autonomous robot vehicles*. [S.l.]: Springer, 1990. p. 259–271. Citado na página 11.
- LOZANO-PÉREZ, T.; WESLEY, M. A. An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, ACM New York, NY, USA, v. 22, n. 10, p. 560–570, 1979. Citado na página 11.
- MASHAYEKHI, R. et al. Informed rrt*-connect: An asymptotically optimal single-query path planning method. *IEEE Access*, IEEE, v. 8, p. 19842–19852, 2020. Citado na página 19.

- MERRICK, D.; DWYER, T. Skeletal animation for the exploration of graphs. In: *Proceedings of the 2004 Australasian symposium on Information Visualisation-Volume 35*. [S.l.: s.n.], 2004. p. 61–70. Citado na página 23.
- MOYA, S.; COLLOUD, F. A fast geometrically-driven prioritized inverse kinematics solver. In: *Proceedings of the XXIV Congress of the International Society of Biomechanics (ISB 2013)*. [S.l.: s.n.], 2013. p. 1–3. Citado na página 24.
- MULLER-CAJAR, R.; MUKUNDAN, R. Triangulation-a new algorithm for inverse kinematics. University of Canterbury. Computer Science and Software Engineering., 2007. Citado na página 22.
- NOREEN, I.; KHAN, A.; HABIB, Z. Optimal path planning using rrt* based approaches: a survey and future directions. *Int. J. Adv. Comput. Sci. Appl*, v. 7, n. 11, p. 97–107, 2016. Citado na página 19.
- PAZOS, F. *Automação de sistema 6 robótica*. [S.l.]: Axel Books, 2002. Citado 2 vezes nas páginas VII e 5.
- PLAKU, E. et al. Sampling-based roadmap of trees for parallel motion planning. *IEEE Transactions on Robotics*, IEEE, v. 21, n. 4, p. 597–608, 2005. Citado na página 18.
- RODRIGUEZ, S. et al. An obstacle-based rapidly-exploring random tree. In: IEEE. *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. [S.l.], 2006. p. 895–900. Citado na página 18.
- ROKBANI, N.; ALIM, A. M. Inverse kinematics using particle swarm optimization, a statistical analysis. *Procedia Engineering*, Elsevier, v. 64, p. 1602–1611, 2013. Citado 2 vezes nas páginas 25 e 38.
- ROKBANI, N.; CASALS, A.; ALIM, A. M. Ik-fa, a new heuristic inverse kinematics solver using firefly algorithm. In: *Computational Intelligence Applications in Modeling and Control*. [S.l.]: Springer, 2015. p. 369–395. Citado 2 vezes nas páginas 1 e 25.
- RYBUS, T.; SEWERYN, K. Application of rapidly-exploring random trees (rrt) algorithm for trajectory planning of free-floating space manipulator. In: IEEE. *2015 10th International Workshop on Robot Motion and Control (RoMoCo)*. [S.l.], 2015. p. 91–96. Citado 2 vezes nas páginas 3 e 20.
- SANTANA, A. et al. The polarized rrt-edge approach. In: IEEE. *2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)*. [S.l.], 2018. p. 271–276. Citado 2 vezes nas páginas 3 e 18.
- SAVSANI, P.; JHALA, R.; SAVSANI, V. J. Comparative study of different metaheuristics for the trajectory planning of a robotic arm. *IEEE Systems Journal*, IEEE, v. 10, n. 2, p. 697–708, 2014. Citado 2 vezes nas páginas 2 e 21.
- SIEGWART, R.; NOURBAKHSH, I. R.; SCARAMUZZA, D. *Introduction to autonomous mobile robots*. [S.l.]: MIT press, 2011. Citado na página 1.
- SPONG, M. W.; HUTCHINSON, S.; VIDYASAGAR, M. *Robot modeling and control*. [S.l.]: John Wiley & Sons, 2020. Citado 2 vezes nas páginas VII e 11.

- SPONG, M. W. et al. *Robot modeling and control*. [S.l.]: wiley New York, 2006. Citado 4 vezes nas páginas 1, 5, 6 e 9.
- STARKE, S.; HENDRICH, N.; ZHANG, J. Memetic evolution for generic full-body inverse kinematics in robotics and animation. *IEEE Transactions on Evolutionary Computation*, IEEE, v. 23, n. 3, p. 406–420, 2018. Citado na página 22.
- SUMNER, R. W. et al. Mesh-based inverse kinematics. In: ACM. *ACM transactions on graphics (TOG)*. [S.l.], 2005. v. 24, n. 3, p. 488–495. Citado na página 24.
- TAO, S.; YANG, Y. Collision-free motion planning of a virtual arm based on the fabrik algorithm. *Robotica*, Cambridge University Press, v. 35, n. 6, p. 1431–1450, 2017. Citado na página 24.
- VAHRENKAMP, N. et al. Humanoid motion planning for dual-arm manipulation and re-grasping tasks. In: IEEE. *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*. [S.l.], 2009. p. 2464–2470. Citado na página 25.
- VÉRAS, L. G. D.; MEDEIROS, F. L.; GUIMARÃES, L. N. Systematic literature review of sampling process in rapidly-exploring random trees. *IEEE Access*, IEEE, v. 7, p. 50933–50953, 2019. Citado 2 vezes nas páginas 16 e 17.
- VITO, D. D.; NATALE, C.; ANTONELLI, G. A comparison of damped least squares algorithms for inverse kinematics of robot manipulators. *IFAC-PapersOnLine*, Elsevier, v. 50, n. 1, p. 6869–6874, 2017. Citado na página 21.
- WANG, J.; LI, X.; MENG, M. Q.-H. An improved rrt algorithm incorporating obstacle boundary information. In: IEEE. *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. [S.l.], 2016. p. 625–630. Citado na página 18.
- WANG, L.-C.; CHEN, C.-C. A combined optimization method for solving the inverse kinematics problems of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, IEEE, v. 7, n. 4, p. 489–499, 1991. Citado na página 22.
- WEGHE, M. V.; FERGUSON, D.; SRINIVASA, S. S. Randomized path planning for redundant manipulators without inverse kinematics. In: IEEE. *2007 7th IEEE-RAS International Conference on Humanoid Robots*. [S.l.], 2007. p. 477–482. Citado 2 vezes nas páginas 20 e 38.
- WEI, K.; REN, B. A method on dynamic path planning for robotic manipulator autonomous obstacle avoidance based on an improved rrt algorithm. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 18, n. 2, p. 571, 2018. Citado 2 vezes nas páginas 3 e 20.
- WHITNEY, D. E. Resolved motion rate control of manipulators and human prostheses. *IEEE Transactions on man-machine systems*, IEEE, v. 10, n. 2, p. 47–53, 1969. Citado 2 vezes nas páginas 2 e 21.
- WICHAPONG, K.; BUREERAT, S.; PHOLDEE, N. Solving inverse kinematics of robot manipulators by means of meta-heuristic optimisation. In: IOP PUBLISHING. *IOP Conference Series: Materials Science and Engineering*. [S.l.], 2018. v. 370, n. 1, p. 012056. Citado na página 2.

YANG, K.; GAN, S. K.; SUKKARIEH, S. A gaussian process-based rrt planner for the exploration of an unknown and cluttered environment with a uav. *Advanced Robotics*, Taylor & Francis, v. 27, n. 6, p. 431–443, 2013. Citado na página 17.

ZABOLOTNY, A. Planning of driverless vehicle motion trajectory taking into account its dimensional characteristics and under conditions of dynamically changing environment. In: IOP PUBLISHING. *IOP Conference Series: Materials Science and Engineering*. [S.l.], 2019. v. 534, n. 1, p. 012033. Citado na página 17.

ZHANG, W. et al. Fabrikc: an efficient iterative inverse kinematics solver for continuum robots. In: IEEE. *2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. [S.l.], 2018. p. 346–352. Citado na página 22.