



**UNIFOR**

**UNIVERSIDADE DE FORTALEZA  
CENTRO DE CIÊNCIAS TECNOLÓGICAS  
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO**

**GUSTAVO MITSUO FERNANDES VALENTE TAKEDA  
PAULO RUAN OLIVEIRA BARBOSA**

**SOFTWARE DE SIMULAÇÃO 3D PARA BRAÇOS ROBÓTICOS EM AMBIENTES  
COMPUTACIONAIS**

**FORTALEZA – CEARÁ**

**2017**

GUSTAVO MITSUO FERNANDES VALENTE TAKEDA  
PAULO RUAN OLIVEIRA BARBOSA

SOFTWARE DE SIMULAÇÃO 3D PARA BRAÇOS ROBÓTICOS EM AMBIENTES  
COMPUTACIONAIS

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Ciência da Computação do Centro de Ciências Tecnológicas da Universidade de Fortaleza, como requisito parcial à obtenção do grau de bacharel em Ciência da Computação.

Orientador: Juliano de Oliveira Pacheco

FORTALEZA – CEARÁ

2017

A ficha catalográfica deve ser gerada no site da biblioteca da Unifor através do link <https://goo.gl/XYUWSC> (link encurtado).

Preencha o formulário com as informações solicitadas e ao final será gerado um arquivo PDF da ficha catalográfica a ser anexada na versão final do TCC.

O arquivo PDF deve ser renomeado para "ficha-catalografica.pdf" (sem aspas) e colocado no diretório "elementos-pre-textuais" (sem aspas) do modelo de TCC da Unifor.

Ficha catalográfica da obra elaborada pelo autor através do programa de geração automática da Biblioteca Central da Universidade de Fortaleza

---

Batista, Bruno .

TEORIA DA RELATIVIDADE: SUBTÍTULO / Bruno Batista, Sandra  
Lima. - 2017  
40 f.

Trabalho de Conclusão de Curso (Graduação) - Universidade  
de Fortaleza. Curso de Ciência da Computação, Fortaleza, 2017.  
Orientação: Liadina Camargo.

1. FÍSICA. 2. RELATIVIDADE. 3. TEMPO. 4. ESPAÇO. I. Lima,  
Sandra. II. Camargo, Liadina. III. Título.

---

GUSTAVO MITSUO FERNANDES VALENTE TAKEDA  
PAULO RUAN OLIVEIRA BARBOSA

SOFTWARE DE SIMULAÇÃO 3D PARA BRAÇOS ROBÓTICOS EM AMBIENTES  
COMPUTACIONAIS

Trabalho de Conclusão de Curso apresentado  
ao Curso de Graduação em Ciência da Com-  
putação do Centro de Ciências Tecnológicas  
da Universidade de Fortaleza, como requisito  
parcial à obtenção do grau de bacharel em  
Ciência da Computação.

Aprovada em: 01 de Janeiro de 2017

BANCA EXAMINADORA

---

Juliano de Oliveira Pacheco (Orientador)  
Centro de Ciências Tecnológicas - CCT  
Universidade de Fortaleza - UNIFOR

---

Membro da Banca Dois  
Centro de Ciências e Tecnologia - CCT  
Universidade do Membro da Banca Dois - SIGLA

---

Membro da Banca Três  
Centro de Ciências e Tecnologia - CCT  
Universidade do Membro da Banca Três - SIGLA

---

Membro da Banca Quatro  
Centro de Ciências e Tecnologia - CCT  
Universidade do Membro da Banca Quatro - SIGLA

Este trabalho é dedicado às crianças adultas que,  
quando pequenas, sonharam em se tornar cientistas.

## **AGRADECIMENTOS**

Primeiramente a Deus que permitiu que tudo isso acontecesse, ao longo de minha vida, e não somente nestes anos como universitária, mas que em todos os momentos é o maior mestre que alguém pode conhecer.

Aos meus pais, pelo amor, incentivo e apoio incondicional.

Obrigada aos meus irmãos e sobrinhos, que nos momentos de minha ausência dedicados ao estudo superior, sempre fizeram entender que o futuro é feito a partir da constante dedicação no presente!

A esta universidade, seu corpo docente, direção e administração que oportunizaram a janela que hoje vislumbro um horizonte superior, eivado pela acendrada confiança no mérito e ética aqui presentes.

Agradeço a todos os professores por me proporcionar o conhecimento não apenas racional, mas a manifestação do caráter e afetividade da educação no processo de formação profissional, por tanto que se dedicaram a mim, não somente por terem me ensinado, mas por terem me feito aprender. a palavra mestre, nunca fará justiça aos professores dedicados aos quais sem nominar terão os meus eternos agradecimentos.

“É melhor lançar-se à luta em busca do triunfo mesmo expondo-se ao insucesso, que formar fila com os pobres de espírito, que nem gozam muito nem sofrem muito; E vivem nessa penumbra cinzenta sem conhecer nem vitória nem derrota.”

(Franklin Roosevelt)

## RESUMO

Este trabalho apresenta o desenvolvimento de uma Interface Homem-Máquina (IHM) para o controle de um braço robótico, propondo uma solução de hardware e software com joystick e interface gráfica para simulação de movimentos em tempo real em um ambiente computacional. O objetivo central é viabilizar o reparo e aprimoramento de equipamentos industriais legados, por meio de uma abordagem de baixo custo e código aberto. Além de fornecer controle manual (via joystick), a interface implementa a possibilidade de ajustes automáticos, permitindo o retrofit de maneira acessível e flexível. Ao longo do texto, aborda-se a justificativa do projeto, a metodologia utilizada, os resultados práticos e as perspectivas de aplicação em larga escala.

**Palavras-chave:** Automação Industrial. Controle de Robôs. Interface Gráfica. Simulação 3D. Retrofit.



## ABSTRACT

This work presents the development of a Human-Machine Interface (HMI) for controlling a robotic arm, proposing a hardware and software solution with a joystick and graphical interface for simulating movements in real time in a computational environment. The central objective is to enable the repair and improvement of legacy industrial equipment, through a low-cost and open source approach. In addition to providing manual control (via joystick), the interface implements the possibility of automatic adjustments, allowing retrofitting in an accessible and flexible way. Throughout the text, the justification for the project, the methodology used, the practical results and the prospects for large-scale application are discussed.

**Keywords:** Industrial Automation. Robot Control. Graphical Interface. 3D simulation. Retrofit.

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>11</b>
1.1	MOTIVAÇÃO . . . . .	11
1.2	OBJETIVOS . . . . .	12
<b>1.2.1</b>	<b>Objetivo Geral . . . . .</b>	<b>12</b>
<b>1.2.2</b>	<b>Objetivos Específicos . . . . .</b>	<b>12</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA . . . . .</b>	<b>14</b>
2.1	CONCEITOS DE ROBÓTICA INDUSTRIAL . . . . .	14
2.2	TECNOLOGIAS DE CONTROLE . . . . .	14
2.3	SIMULAÇÃO 3D EM TEMPO REAL . . . . .	14
2.4	DESIGN DE SOFTWARE . . . . .	15
<b>3</b>	<b>TRABALHOS RELACIONADOS . . . . .</b>	<b>17</b>
3.1	ROBODK . . . . .	17
3.2	WEBOTS . . . . .	17
3.3	ROS-INDUSTRIAL . . . . .	18
<b>4</b>	<b>METODOLOGIA . . . . .</b>	<b>19</b>
4.1	DEFINIÇÃO DOS REQUISITOS . . . . .	19
4.2	SELEÇÃO DAS TECNOLOGIAS . . . . .	19
4.3	MODELAGEM DO ROBÔ . . . . .	20
4.4	IMPLEMENTAÇÃO DO SOFTWARE . . . . .	20
4.5	USANDO FÓRMULAS MATEMÁTICAS . . . . .	23
4.6	USANDO ALGORITMOS . . . . .	23
4.7	USANDO CÓDIGO-FONTE . . . . .	23
4.8	USANDO TEOREMAS, PROPOSIÇÕES, ETC . . . . .	24
4.9	USANDO QUESTÕES . . . . .	25
4.10	CITAÇÕES . . . . .	25
<b>4.10.1</b>	<b>Documentos com três autores . . . . .</b>	<b>25</b>
<b>4.10.2</b>	<b>Documentos com mais de três autores . . . . .</b>	<b>25</b>
<b>4.10.3</b>	<b>Documentos de vários autores . . . . .</b>	<b>26</b>
4.11	NOTAS DE RODAPÉ . . . . .	26
<b>5</b>	<b>RESULTADOS . . . . .</b>	<b>27</b>
5.1	RESULTADOS DO EXPERIMENTO A . . . . .	27

5.2	RESULTADOS DO EXPERIMENTO B . . . . .	27
<b>6</b>	<b>CONCLUSÕES E TRABALHOS FUTUROS . . . . .</b>	<b>28</b>
6.1	CONTRIBUIÇÕES DO TRABALHO . . . . .	28
6.2	LIMITAÇÕES . . . . .	28
6.3	TRABALHOS FUTUROS . . . . .	28

## 1 INTRODUÇÃO

A automação industrial tem desempenhado um papel fundamental na otimização de processos de produção e na busca por maior eficiência operacional. Nesse contexto, braços robóticos, como os da série Mitsubishi RV-2SDB/RV-2SQB, emergem como ferramentas essenciais para a execução de tarefas complexas com precisão e repetitividade. Contudo, muitas empresas enfrentam desafios para manter esses equipamentos atualizados e funcionais, especialmente quando a manutenção depende de fornecedores externos, o que eleva custos e tempo de inatividade.

Neste trabalho, propõe-se a criação de uma Interface Homem-Máquina (IHM) para controlar um braço robótico, atendendo a uma demanda específica de reparo e retrofit do modelo Mitsubishi RV-2SDB/RV-2SQB. A solução, desenvolvida em um ambiente de baixo custo e baseada em bibliotecas open source, visa não apenas suprir a necessidade de controle manual e automático do robô, mas também oferecer uma base reutilizável para reparos e aprimoramentos em outros contextos industriais. Um dos destaques é a incorporação de uma simulação 3D em tempo real, permitindo ao operador visualizar e acompanhar os movimentos do braço robótico em um ambiente virtual sincronizado com a execução física.

A relevância do projeto reside na:

- a) Viabilizar uma alternativa econômica para manutenção de robôs em cenários com recursos limitados.
- b) Criar um software de controle que integra joystick, interface gráfica e simulação 3D, ampliando a usabilidade e o entendimento das operações
- c) Possibilitar compartilhamento e aprimoramento contínuo, já que o código segue uma filosofia open source, beneficiando o meio acadêmico e industrial.

Este documento descreve a fundamentação teórica que embasa o desenvolvimento, a abordagem metodológica adotada, os resultados obtidos e as implicações práticas no setor industrial e na comunidade acadêmica. Espera-se demonstrar que a combinação de controle manual, ajustes automáticos e visualização 3D online pode proporcionar maior confiabilidade, usabilidade e escalabilidade às soluções de automação.

### 1.1 MOTIVAÇÃO

A motivação central deste trabalho surge da necessidade de reparo e aprimoramento de um braço robótico *Mitsubishi RV-2SDB/RV-2SQB*, cujo suporte especializado tornou-se in-

viável devido a restrições orçamentárias. Assim, a abordagem escolhida foca em soluções de baixo custo e ferramentas open source, unificando controle manual, monitoramento automático e simulação 3D.

Em cenários industriais, a inatividade de um robô devido a falhas em módulos de controle gera prejuízos financeiros e produtivos. Uma IHM que facilite ajustes de velocidade, posicionamento e correção de falhas, além de oferecer simulação 3D realista, pode reduzir tempos de parada e retrabalho. A motivação, portanto, combina fatores econômicos, técnicos e operacionais, buscando uma solução modular e flexível.

## 1.2 OBJETIVOS

Os objetivos geral e específicos deste projeto visam oferecer uma solução completa para controle e monitoramento do braço robótico Mitsubishi RV-2SDB/RV-2SQB em ambiente industrial. A aplicação integra *firmware*, *software* e interface gráfica, com simulação 3D em tempo real para melhorar visualização e diagnóstico.

### 1.2.1 Objetivo Geral

Desenvolver uma **Interface Homem-Máquina** robusta, econômica e *open source* para controle de um braço robótico, permitindo tanto operação manual (por *joystick*) quanto ajustes automáticos, além de oferecer uma simulação 3D em tempo real que reflita os movimentos e estados do robô.

### 1.2.2 Objetivos Específicos

- a) Projetar e implementar o *firmware* de controle, definindo um protocolo de comunicação de 32 bits para *hardware* como *ESP32* e *FPGA*, incluindo recursos de freio, *reverse* e ajuste de velocidade;
- b) Integrar um dispositivo *joystick* à interface gráfica, de modo a fornecer um controle manual intuitivo para o operador;
- c) Desenvolver e incorporar uma simulação 3D em tempo real, sincronizada com as ações do braço robótico, possibilitando a identificação de falhas e otimização dos parâmetros de controle;
- d) Avaliar o desempenho do sistema em cenários de testes, verificando latência, confiabilidade e consumo de recursos do *host* e do *microcontrolador*;

- e) Documentar o projeto em formato *open source*, compartilhando bibliotecas e instruções que facilitem o *retrofit* de outros braços robóticos ou equipamentos industriais legados.

## 2 FUNDAMENTAÇÃO TEÓRICA

A modernização de equipamentos industriais por meio de técnicas de *retrofit* é cada vez mais relevante no contexto da Indústria 4.0. Avanços em microeletrônica e sistemas embarcados permitem atualizar módulos de controle, como os do Mitsubishi RV-2SDB/RV-2SQB, sem a necessidade de substituir máquinas inteiras, reduzindo custos e prolongando sua vida útil. Este capítulo apresenta os conceitos que sustentam o desenvolvimento da IHM proposta, com ênfase nas tecnologias de controle e simulação 3D.

### 2.1 CONCEITOS DE ROBÓTICA INDUSTRIAL

Braços robóticos industriais, como o Mitsubishi RV-2SDB/RV-2SQB, operam com base em cinemática direta e inversa para calcular posições e velocidades das juntas em tarefas precisas. A cinemática direta determina a posição e orientação do efetuador final a partir dos ângulos das juntas, enquanto a cinemática inversa faz o oposto, encontrando os ângulos das juntas para uma dada posição do efetuador. O robô Mitsubishi RV-2SDB/RV-2SQB, com 6 eixos, possui especificações como alcance de 2400 mm para a junta J1, velocidade máxima de 4.490 mm/s, repetibilidade de 0.02 mm, e capacidade de carga de 2 kg, conforme o manual técnico [1]. Esses parâmetros são essenciais para a modelagem cinemática no software de simulação.

### 2.2 TECNOLOGIAS DE CONTROLE

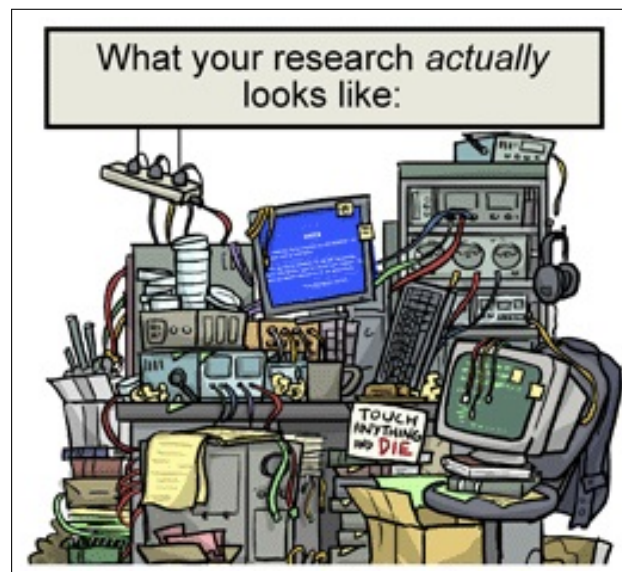
A integração de hardware para controle preciso do robô é fundamental. O firmware gerencia sinais para controlar os motores, ajustando velocidade e direção. Bibliotecas open source, como SDL (Simple DirectMedia Layer), facilitam a conexão com dispositivos de entrada, como joysticks, permitindo controle manual intuitivo. A interface gráfica, desenvolvida com ImGui, exibe estados do robô e permite ajustes interativos, melhorando a usabilidade.

A Figura 5 exibe um diagrama mostrando a interação entre o firmware, o software de controle, e a simulação 3D, com dados de entrada do joystick e saída visual na interface gráfica.

### 2.3 SIMULAÇÃO 3D EM TEMPO REAL

A simulação 3D em tempo real permite ao operador visualizar os movimentos do robô em um ambiente virtual, facilitando o monitoramento e a detecção de falhas. Utilizando

**Figura 1 – Fluxo de dados entre o sistema de controle e a simulação 3D, ilustrando a sincronia entre movimentos reais e virtuais.**



Fonte: Elaborado pelo autor

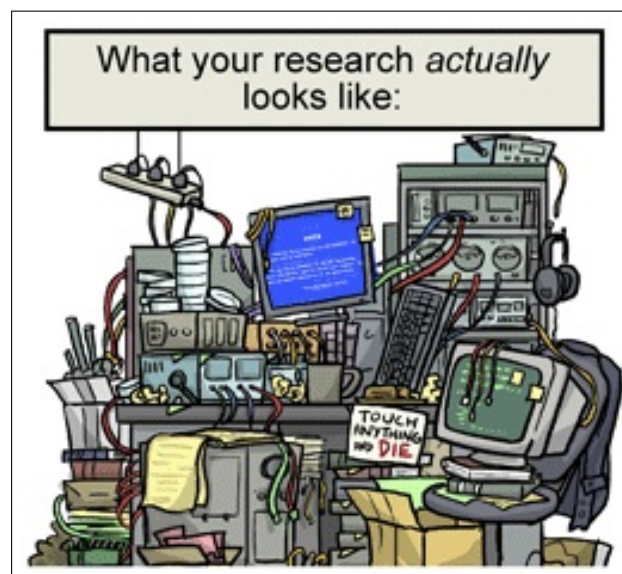
OpenGL para renderização, GLM para cálculos de álgebra linear, e Assimp para carregar modelos 3D (e.g., *rv2sdb.obj*), o software representa fielmente o robô e seus movimentos. A simulação reflete as especificações do robô, como o alcance de  $\pm 240^\circ$  da junta J1, garantindo precisão visual.

## 2.4 DESIGN DE SOFTWARE

O software foi desenvolvido seguindo princípios de modularidade, com separação de preocupações entre renderização, entrada de dados, cinemática, e interface gráfica. Isso permite manutenção e escalabilidade, facilitando a adaptação para outros robôs ou aplicações industriais. A arquitetura modular é suportada por bibliotecas como *fmt* para logging e *xmake* para gerenciamento de compilação.



**Figura 2 – Representação esquemática da camada digital, destacando a geração de sinais de controle.**



Fonte: Elaborado pelo autor

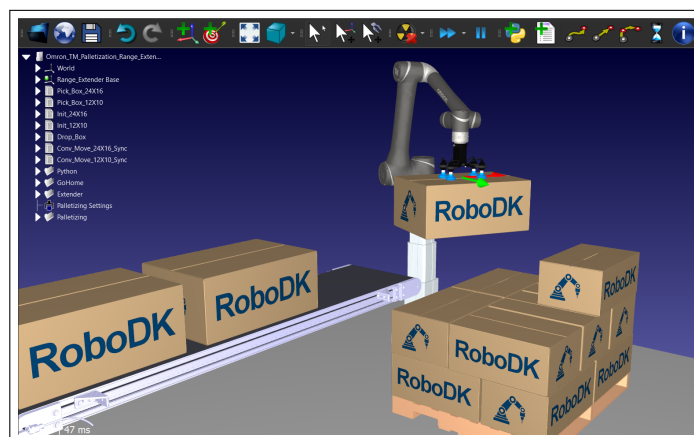
### 3 TRABALHOS RELACIONADOS

A simulação 3D e o controle de robôs industriais, como o Mitsubishi RV-2SDB/RV-2SQB, têm sido amplamente explorados em projetos industriais e acadêmicos, com ênfase em soluções open source e retrofit. Abaixo, são apresentados trabalhos e ferramentas que compartilham objetivos semelhantes aos deste projeto, destacando sua relevância para o desenvolvimento da IHM proposta.

#### 3.1 ROBODK

O RoboDK é uma plataforma de simulação e programação offline para braços robóticos, amplamente utilizada na indústria. Ele suporta uma vasta biblioteca de modelos de robôs, incluindo alguns da Mitsubishi, e permite simular aplicações como usinagem, soldagem e pick-and-place. Sua interface intuitiva não exige habilidades avançadas de programação, facilitando a criação de simulações e programas robóticos [2]. Este projeto se assemelha ao RoboDK por buscar uma simulação 3D acessível, embora com foco em controle manual e retrofit de um robô específico

**Figura 3 – Interface do Robodk simulando um pick-and-place utilizando um modelo de robô generérico**



Fonte: Elaborado pelo autor

#### 3.2 WEBOTS

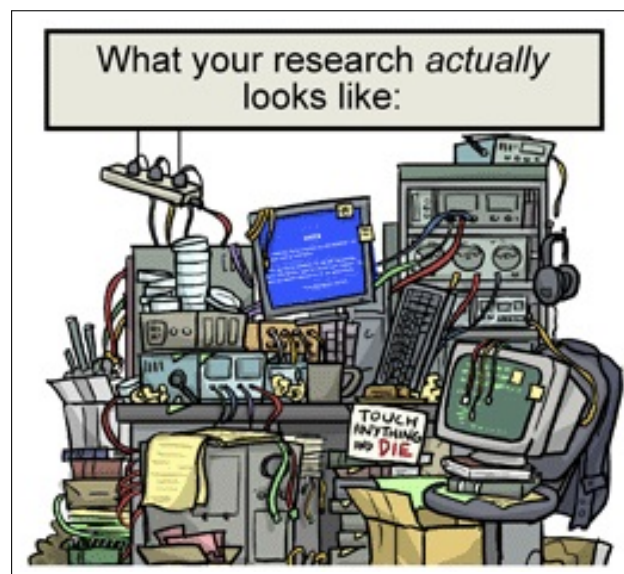
O Webots é um software open source que oferece um ambiente completo para modelagem, programação e simulação de robôs, incluindo braços industriais. Desenvolvido pela

Cyberbotics, é utilizado em educação, pesquisa e indústria, suportando a importação de modelos CAD [3]. A abordagem open source do Webots alinha-se com os objetivos deste trabalho, que também prioriza bibliotecas livres para garantir escalabilidade e acessibilidade.

### 3.3 ROS-INDUSTRIAL

O ROS-Industrial é uma extensão do Robot Operating System (ROS), um projeto open source que adapta o ROS para aplicações industriais. Liderado por um consórcio que inclui a Fraunhofer IPA, ele facilita a integração de robôs legados com tecnologias modernas, sendo particularmente relevante para retrofit [4]. Este trabalho se inspira no ROS-Industrial ao buscar soluções econômicas para modernizar o Mitsubishi RV-2SDB/RV-2SQB, com ênfase na simulação 3D e controle manual.

**Figura 4 –** *Maecenas luctus augue odio, sed tincidunt nunc posuere nec*



Fonte: Elaborado pelo autor

## 4 METODOLOGIA

O desenvolvimento da Interface Homem-Máquina (IHM) para o controle e simulação do braço robótico Mitsubishi RV-2SDB/RV-2SQB seguiu uma abordagem estruturada e iterativa, garantindo a entrega de um sistema funcional e robusto. As etapas da metodologia são detalhadas a seguir.

**Tabela 0 – Um Exemplo de tabela alinhada que pode ser longa ou curta, conforme padrão IBGE**

Nome	Nascimento	Documento
Maria da Silva	11/11/1111	111.111.111-11
Maria da Silva	11/11/1111	111.111.111-11
Maria da Silva	11/11/1111	111.111.111-11

Fonte: Produzido pelos autores

Nota: Esta é uma nota, que diz que os dados são baseados na regressão linear.

Anotações: Uma anotação adicional, seguida de várias outras.

### 4.1 DEFINIÇÃO DOS REQUISITOS

Inicialmente, foram identificados os requisitos funcionais e não funcionais do sistema, baseados nas necessidades de controle e simulação do robô:

1. Controle Manual: Operação via joystick (e.g., controle Xbox) e teclado para ajustar ângulos das juntas.
2. Simulação 3D: Visualização em tempo real dos movimentos do robô.
3. Ajustes Automáticos: Configurações de velocidade e posicionamento com limites específicos (e.g., \$240\text{r}\$ para J1).
4. Interface Gráfica: Exibição de estados e ajustes interativos.
5. Open Source: Uso de bibliotecas livres.
6. Desempenho: Baixa latência e estabilidade.

### 4.2 SELEÇÃO DAS TECNOLOGIAS

Foram selecionadas tecnologias open source adequadas aos requisitos:

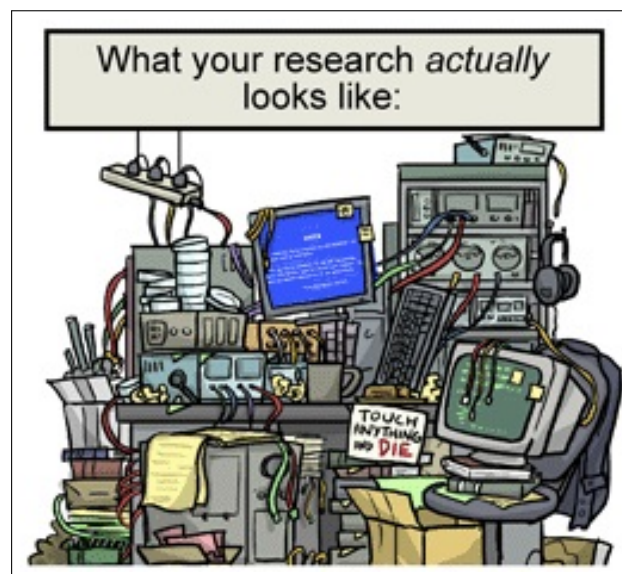
1. OpenGL e SDL: Renderização 3D e gerenciamento de janelas.

2. GLM: Álgebra linear em espaço tridimensional.
3. ImGui: Interface gráfica interativa.
4. fmt: Formatação de strings e logging.
5. xmake: Gerenciamento de compilação.

#### 4.3 MODELAGEM DO ROBÔ

Desenvolveu-se um modelo cinemático com 6 juntas, utilizando dados do manual do robô (e.g.,  $\dot{\theta}_{240}$  para J1, velocidade máxima de 4.490 mm/s). Comprimentos aproximados dos elos foram definidos (0.2m, 0.3m, 0.25m, 0.15m, 0.1m, 0.05m), a serem refinados com dados reais. A cinemática inversa foi implementada para calcular posições com base nos ângulos das juntas.

**Figura 5 – Modelo 3D renderizado do Robô**



Fonte: Elaborado pelo autor

#### 4.4 IMPLEMENTAÇÃO DO SOFTWARE

O software foi implementado em C++, com módulos separados:

1. Renderização: Carregamento e renderização do modelo 3D usando OpenGL e Assimp.
2. Input: Gerenciamento de entradas via SDL (teclado, joystick).
3. Simulação: Cálculo de transformações cinemáticas com GLM e resolução de

colisão com físicas *rigid body* em tempo real.

4. Interface Gráfica: Exibição e ajuste de parâmetros via ImGui.

Implementando estes algoritmos para cinemática inversa e detecção e resolução de colisão em tempo real:

1. FABRIK(cinemática inversa de alcance para frente e para trás)
2. GGD(Descida de coordenação cíclica)
3. GJK(Gilbert Johnson Keerthi): Cálculo e resolução de colisão com físicas *rigid body* em tempo real.
4. EPA(Algoritmo de expansão de politopo):
5. Adição de Minkowski: Função suporte para o algoritmo de GJK.
6. Sistemas de arbitragem

---

**Algoritmo 1:** Como escrever algoritmos no  $\text{\LaTeX}2\text{e}$

---

**Entrada:** o próprio texto

**Saída:** como escrever algoritmos com  $\text{\LaTeX}2\text{e}$

**início**

inicialização;

**repita**

leia o atual;

**se entendeu então**

vá para o próximo;

próximo se torna o atual;

**fim**

**senão**

volte ao início da seção;

**fim**

**até fim do texto;**

**fim**

---

Exemplo de alíneas com números:

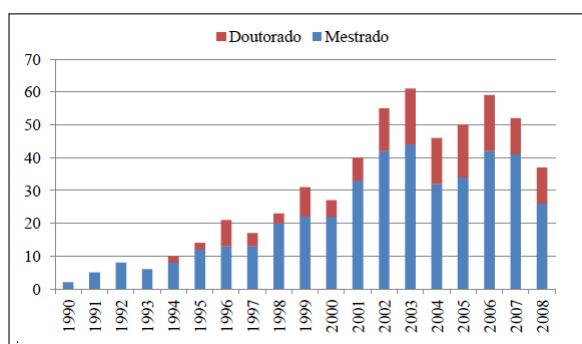
Figuras podem ser criadas diretamente em LaTeX, como o exemplo da 6.

Ou então figuras podem ser incorporadas de arquivos externos, como é o caso da Figura 6. Se a figura que ser incluída se tratar de um diagrama, um gráfico ou uma ilustração que você mesmo produza, priorize o uso de imagens vetoriais no formato PDF. Com isso, o tamanho do arquivo final do trabalho será menor, e as imagens terão uma apresentação melhor,

**Tabela 1 – Internal exon scores**

Ranking	Exon Coverage	Splice Site Support
E1	Complete coverage by a single transcript	Both splice sites
E2	Complete coverage by more than a single transcript	Both splice sites
E3	Partial coverage	Both splice sites
E4	Partial coverage	One splice site
E5	Complete or partial coverage	No splice sites
E6	No coverage	No splice sites

Fonte: os autores

**Figura 6 – Produção anual das dissertações de mestrado e teses de doutorado entre os anos de 1990 e 2008**

Fonte: os autores

principalmente quando impressas, uma vez que imagens vetoriais são perfeitamente escaláveis para qualquer dimensão. Nesse caso, se for utilizar o Microsoft Excel para produzir gráficos, ou o Microsoft Word para produzir ilustrações, exporte-os como PDF e os incorpore ao documento conforme o exemplo abaixo. No entanto, para manter a coerência no uso de software livre (já que você está usando LaTeX e abnTeX), teste a ferramenta InkScape. ao CorelDraw ou ao Adobe Illustrator. De todo modo, caso não seja possível utilizar arquivos de imagens como PDF, utilize qualquer outro formato, como JPEG, GIF, BMP, etc. Nesse caso, você pode tentar aprimorar as imagens incorporadas com o software livre Gimp. Ele é uma alternativa livre ao Adobe Photoshop.

## 4.5 USANDO FÓRMULAS MATEMÁTICAS

$$x = a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \frac{1}{a_4}}}} \quad (4.1)$$

$$k_{n+1} = n^2 + k_n^2 - k_{n-1} \quad (4.2)$$

$$\cos(2\theta) = \cos^2 \theta - \sin^2 \theta \quad (4.3)$$

$$A_{m,n} = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m,1} & a_{m,2} & \cdots & a_{m,n} \end{pmatrix} \quad (4.4)$$

$$f(n) = \begin{cases} n/2 & \text{if } n \text{ is even} \\ -(n+1)/2 & \text{if } n \text{ is odd} \end{cases} \quad (4.5)$$

## 4.6 USANDO ALGORITMOS

## 4.7 USANDO CÓDIGO-FONTE

### Código-fonte 1 – Hello World em C++

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main()
```



---

**Algoritmo 2:** Algoritmo de Otimização por Colônia de Formiga
 

---

**Entrada:** Entrada do Algoritmo

**Saída:** Saida do Algoritmo

**início**

Atribua os valores dos parâmetros;

Inicialize as trilhas de feromônios;

**enquanto** *não atingir o critério de parada* **faça**

**para** *cada formiga* **faça**

        Construa as Soluções;

**fim**

    Aplique Busca Local (Opcional);

    Atualize o Feromônio;

**fim**

**fim**

---

```

6 {
7     cout<<"Hello World!"<<endl;
8     system("pause");
9     return 0;
10 }
```

### Código-fonte 2 – Hello World em Java

```

1 public class HelloWorld {
2     public static void main(String[] args) {
3         System.out.println("Hello World!");
4     }
5 }
```

## 4.8 USANDO TEOREMAS, PROPOSIÇÕES, ETC

**Teorema 4.8.1 (Pitágoras)** *Em todo triângulo retângulo o quadrado do comprimento da hipotenusa é igual a soma dos quadrados dos comprimentos dos catetos.*

**Teorema 4.8.2 (Fermat)** *Não existem inteiros  $n > 2$ , e  $x, y, z$  tais que  $x^n + y^n = z$*

**Proposição 4.8.3** *Para demonstrar o Teorema de Pitágoras...*

**Exemplo 1** *Este é um exemplo do uso do ambiente `exe` definido acima.*

**Definição 4.8.1** *Definimos o produto de ...*

## 4.9 USANDO QUESTÕES

**Questão 1.** Esta é a primeira questão com alguns itens:

- (a) Este é o primeiro item
- (b) Segundo item

**Questão 2.** Esta é a segunda questão:

- (a) Este é o primeiro item
- (b) Segundo item

## 4.10 CITAÇÕES

### 4.10.1 Documentos com três autores

Quando houver três autores na citação, apresentam-se os três, separados por ponto e vírgula, caso estes estejam após o texto. Se os autores estiverem incluídos no texto, devem ser separados por vírgula e pela conjunção "e".

?? (??)

(??)

### 4.10.2 Documentos com mais de três autores

Havendo mais de três autores, indica-se o primeiro seguido da expressão *et al.* (do latim *et alli*, que significa e outros), do ano e da página.

?? (??)

(??)

### 4.10.3 Documentos de vários autores

Havendo citações indiretas de diversos documentos de vários autores, mencionados simultaneamente e que expressam a mesma ideia, separam-se os autores por ponto e vírgula, em ordem alfabética.

(???)

### 4.11 NOTAS DE RODAPÉ

Deve-se utilizar o sistema autor-data para as citações no texto e o numérico para notas explicativas<sup>1</sup>. As notas de rodapé podem e devem ser alinhadas, a partir da segunda linha da mesma nota, abaixo da primeira letra da primeira palavra, de forma a destacar o expoente <sup>2</sup> e sem espaço entre elas e com fonte menor (tamanho 10).

---

<sup>1</sup> Veja - se como exemplo desse tipo de abordagem o estudo de Netzer (1976)

<sup>2</sup> Encontramos esse tipo de perspectiva na 2ª parte do verbete referido na nota anterior, em grande parte do estudo de Rahner (1962).

## **5 RESULTADOS**

### **5.1 RESULTADOS DO EXPERIMENTO A**

### **5.2 RESULTADOS DO EXPERIMENTO B**

## **6 CONCLUSÕES E TRABALHOS FUTUROS**

### **6.1 CONTRIBUIÇÕES DO TRABALHO**

### **6.2 LIMITAÇÕES**

### **6.3 TRABALHOS FUTUROS**