

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**ALISSON HENRIQUE INACIO**

**DESENVOLVIMENTO DE UM ROBÔ AUTÔNOMO PARA CONTROLE  
DE NÍVEL DE OXIGÊNIO EM TANQUES DE PISCICULTURA**

**TOLEDO**

**2022**

**ALISSON HENRIQUE INACIO**

**DESENVOLVIMENTO DE UM ROBÔ AUTÔNOMO PARA CONTROLE  
DE NÍVEL DE OXIGÊNIO EM TANQUES DE PISCICULTURA**

**Development of an autonomous robot to control the oxygen level in fish  
farming ponds**

Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Bacharel em Engenharia Eletrônica do Curso de Bacharelado em Engenharia Eletrônica da Universidade Tecnológica Federal do Paraná.

Orientador: Dr. Fábio Rizental Coutinho

**TOLEDO**

**2022**



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**ALISSON HENRIQUE INACIO**

**DESENVOLVIMENTO DE UM ROBÔ AUTÔNOMO PARA CONTROLE  
DE NÍVEL DE OXIGÊNIO EM TANQUES DE PISCICULTURA**

Trabalho de Conclusão de Curso de Graduação  
apresentado como requisito para obtenção do  
título de Bacharel em Engenharia Eletrônica  
do Curso de Bacharelado em Engenharia  
Eletrônica da Universidade Tecnológica  
Federal do Paraná.

Data de aprovação: 15/dezembro/2022

---

Prof. Fábio Rizental Coutinho  
Doutorado  
Universidade Tecnológica Federal do Paraná - Campus Toledo

---

Prof. Edson Tavares de Camargo  
Doutorado  
Universidade Tecnológica Federal do Paraná - Campus Toledo

---

Prof. José Dolores Vergara Dietrich  
Doutorado  
Universidade Tecnológica Federal do Paraná - Campus Toledo

**TOLEDO**  
**2022**

## RESUMO

O controle do teor de oxigênio de açudes é o elemento chave para a rentabilidade de produção intensiva na piscicultura. Visto isso, o objetivo desse trabalho é desenvolver um robô paralelo acionado por cabos, *Cable Driven Parallel Robot* (CDPR), a fim de direcionar uma sonda para a medição de oxigênio dissolvido na água de vários açudes utilizando um único sensor e, assim, diminuir os custos do processo de automatizar e controlar o nível de oxigênio dissolvido para os produtores. A construção desse tipo de robô possibilita versatilidade, baixo custo, durabilidade, fácil manutenção e instalação devido ao sistema ser composto somente por cabos e postes nos quais os atuadores de tensão são instalados. Dessa forma, foi desenvolvido um sistema CDPR de 4 cabos, baseado em 1 atuador mecânico mestre e outro três escravos. O robô foi desenvolvido a partir do tensionador mecânico mestre como referência, no qual é composto por um microcontrolador Arduino Nano, que gere e calcula como será feito o movimento dos cabos e também se comunica via rádio através do módulo NRF24L01, para os demais tensionadores escravos localizados a longas distâncias. O robô abrange motores monofásicos para tracionar e desatar cabos e sensores rotativos para monitorar estes motores. Para tal feito, o robô foi testado em uma área de trabalho controlado e em escala menor que a planejada. Com esse trabalho foi possível fornecer movimento em três graus de liberdades para uma sonda, na qual atingiu 6 pontos de medições distribuídos em um espaço de aproximadamente 100 metros cúbicos, com erro de posição máximo de 12,57 cm entre os pontos estipulados e os alcançados na prática.

**Palavras-chave:** robô de cabos; sensor suspenso; qualidade de água; cinemática; automação agropecuária.

## ABSTRACT

The control of the oxygen content of reservoirs is the key element for the profitability of intensive production in fish farming. The objective of this work is to develop a cable-driven parallel robot, or CDPR, in order to direct a probe for the measurement of dissolved oxygen in water from several reservoirs using a single sensor and, thus, reduce the costs of the process of automate and control dissolved oxygen level for producers. The construction of this type of robot allows versatility, low cost, durability, easy maintenance and installation due to the system being composed only of cables and posts on which the tension actuators are installed. A 4-cable CDPR system was developed, based on 1 master mechanical actuator and another 3 slaves. The robot was developed from the master mechanical tensioner as a reference, which is composed of an Arduino Nano microcontroller, which manages and calculates how the cables will move and also communicates by radio through the NRF24L01 module, to the other slave tensioners located at long distances. The robot includes single-phase motors to pull and untie cables and rotary sensors to monitor these motors. For this purpose, the robot was tested in a controlled work area and on a smaller scale than planned. With this work it was possible to provide movement in three degrees of freedom for a probe, in which it reached 6 measurement points distributed in a space of approximately 100 cubic meters, with a maximum position error of 12.57 cm between the stipulated and reached points in practice.

**Keywords:** cable robot; suspended sensor; wather quality; cinamatics; automation farming.

## LISTA DE FIGURAS

<b>Figura 1</b>	<b>– Visão superior de um sistema CDPR de quatro cabos</b>	<b>13</b>
<b>Figura 2</b>	<b>– Parâmetros de deslocamento e trajetória do EE no plano</b>	<b>14</b>
<b>Figura 3</b>	<b>– Distância de cabos entre pontos e atuador no plano</b>	<b>15</b>
<b>Figura 4</b>	<b>– Visão lateral do sistema CDPR e parâmetros de deslocamento do EE no espaço</b>	<b>16</b>
<b>Figura 5</b>	<b>– Digrama de composição dos quatro atuadores mecânicos</b>	<b>19</b>
<b>Figura 6</b>	<b>– Componentes do atuador mecânico</b>	<b>20</b>
<b>Figura 7</b>	<b>– Diagrama de blocos do circuito eletrônico de um atuador mecânico</b>	<b>20</b>
<b>Figura 8</b>	<b>– Motor monofásico VariVelox, VXRSA</b>	<b>22</b>
<b>Figura 9</b>	<b>– Esquema de ligação para cada sentido de rotação do motor monofásico</b>	<b>23</b>
<b>Figura 10</b>	<b>– Encoder rotativo LPD3806-600BM-G5-24C</b>	<b>24</b>
<b>Figura 11</b>	<b>– Transceptor NRF24I01</b>	<b>24</b>
<b>Figura 12</b>	<b>– Esquema resumido do ambiente de testes</b>	<b>27</b>
<b>Figura 13</b>	<b>– Peças impressas em 3D: (a) Rolete encoder, (b) Extruder peça 1, (c) Extruder peça 2</b>	<b>28</b>
<b>Figura 14</b>	<b>– Encoder rotativo montado</b>	<b>28</b>
<b>Figura 15</b>	<b>– Placa de testes com componentes</b>	<b>30</b>
<b>Figura 16</b>	<b>– Tensionadores Mecânicos: (a) Tensionador escravo 1, (b) Tensionador mestre</b>	<b>31</b>
<b>Figura 17</b>	<b>– Diagrama de blocos controlador de cabos mestre</b>	<b>32</b>
<b>Figura 18</b>	<b>– Diagrama de blocos controlador de cabos escravo</b>	<b>33</b>
<b>Figura 19</b>	<b>– Movimento por iteração, com suas respectivas distâncias de cabos entre dois atuadores</b>	<b>34</b>
<b>Figura 20</b>	<b>– Distâncias de cabos para deslocar o EE entre quatro pontos (a) Sem iterações, (b) 4 iterações</b>	<b>39</b>
<b>Figura 21</b>	<b>– Distâncias de cabos para deslocar o EE entre quatro pontos: (a) 8 iterações, (b) 16 iterações</b>	<b>40</b>
<b>Figura 22</b>	<b>– Pontos de visitas (rotulados) e pontos medidos (não rotulados) com 4 iterações de resolução entre pontos de visita</b>	<b>42</b>

**Figura 23 – Pontos de visitas (rotulados) e pontos medidos (não rotulados) com 16 iterações de resolução entre pontos de visita . . . . . 44**

## LISTA DE TABELAS

<b>Tabela 1 – Orçamento dos materiais . . . . .</b>	<b>25</b>
<b>Tabela 2 – Posição dos atuadores mecânicos no ambiente de teste . . . . .</b>	<b>26</b>
<b>Tabela 3 – Leituras das distâncias sem intervalos . . . . .</b>	<b>29</b>
<b>Tabela 4 – Leituras das distâncias com intervalos a cada 20 cm . . . . .</b>	<b>29</b>
<b>Tabela 5 – 6 pontos de visitas com 4 iterações de resolução entre pontos de visita .</b>	<b>42</b>
<b>Tabela 6 – 6 pontos de visitas com 16 iterações de resolução entre pontos de visita</b>	<b>43</b>



## LISTAGEM DE CÓDIGOS FONTE

<b>Listagem 1</b>	<b>– Rotinas de calculos de distâncias no plano . . . . .</b>	<b>33</b>
<b>Listagem 2</b>	<b>– Variáveis locais da função <code>sequenceOfIterations</code> . . . . .</b>	<b>35</b>
<b>Listagem 3</b>	<b>– <i>Loop</i> para ponto inicial mais próximo que o final da função <code>sequenceOfIterations</code> . . . . .</b>	<b>36</b>
<b>Listagem 4</b>	<b>– <i>Loop</i> para ponto inicial mais distante que o final da função <code>sequenceOfIterations</code> . . . . .</b>	<b>37</b>
<b>Listagem 5</b>	<b>– <i>Loop</i> referênte ao cálculo das variações de distâncias e movimento do motor 4 da função <code>sequenceOfIterations</code> . . . . .</b>	<b>38</b>

## LISTA DE ABREVIATURAS E SIGLAS

### Siglas

CDPR	<i>Cable Driven Parallel Robot</i>
CNC	<i>Computer numerical control</i>
EE	<i>End Efeator</i>
EMBRAPA	Empresa Brasileira de Pesquisa Agropecuária
FAST	<i>Five-hundred-meter Aperture Spherical Telescope</i>
IoT	<i>Internet of Things</i>
ISM	<i>The industrial, scientific, and medical radio band</i>
LoRaWan	<i>Long Range Wide Area Network</i>
OD	Oxigênio dissolvido
PCP	Ponto de Contato com o Plano
Ph	<i>Potential of Hydrogen</i>
RF	<i>Radio frequency</i>
SPI	<i>Serial Peripheral Interface</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>10</b>
<b>1.1</b>	<b>Considerações iniciais</b>	<b>10</b>
<b>1.2</b>	<b>Objetivos</b>	<b>11</b>
1.2.1	Objetivo geral	11
1.2.2	Objetivos específicos	11
<b>1.3</b>	<b>Justificativa</b>	<b>12</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>13</b>
<b>2.1</b>	<b>Determinação da posição do EE no plano</b>	<b>13</b>
<b>2.2</b>	<b>Equação do trabalho do atuador no plano</b>	<b>14</b>
<b>2.3</b>	<b>Movimento do EE no espaço</b>	<b>15</b>
<b>3</b>	<b>MATERIAIS E MÉTODOS</b>	<b>18</b>
<b>3.1</b>	<b>Metodologia</b>	<b>18</b>
3.1.1	Atuadores mecânicos	19
3.1.2	Circuito controlador de cabos	19
3.1.3	Tensionador	21
<b>3.2</b>	<b>Materiais</b>	<b>21</b>
3.2.1	Microcontrolador Arduino Nano	21
3.2.2	Motor monofásico	22
3.2.3	Rotary encoder ótico digital	23
3.2.4	Transceptor de rádio NRF24L01	23
3.2.5	Custos de implementação	24
<b>4</b>	<b>RESULTADOS</b>	<b>26</b>
<b>4.1</b>	<b>Ambiente de testes</b>	<b>26</b>
<b>4.2</b>	<b>Placa de circuito impresso</b>	<b>30</b>
<b>4.3</b>	<b>Software</b>	<b>31</b>
<b>4.4</b>	<b>Resultados na prática</b>	<b>41</b>
<b>5</b>	<b>CONCLUSÃO</b>	<b>45</b>
	<b>REFERÊNCIAS</b>	<b>46</b>

## 1 INTRODUÇÃO

A piscicultura está atrelada à história brasileira desde o século XVIII, na qual a prática teve suas raízes estabelecidas no processo de colonização dos holandeses na região Nordeste do Brasil. Porém, essa prática data mais de 3 mil anos atrás. Ela surgiu durante a dinastia Zhou na China, que de tamanha influência e grandeza, 500 anos depois, durante a dinastia Tang, a carpa se tornou símbolo de prosperidade e de boa sorte, e ainda o imperador regente da época tomou o nome carpa como seu – ‘Fan Li’. Após a descoberta, nunca mais houve um império no mundo onde não se cultivasse peixes.

Apesar dessa história milenar da piscicultura, as técnicas empregadas nessa prática só tiveram grandes avanços depois do século XIX. Antes, os estudos voltados para aperfeiçoar as práticas envolvidas se limitavam à ciência da época, como as observações do comportamento de grupos de peixes e da sazonalidade da sua reprodução. Somente depois do século XIX, com a genética e posteriormente com tecnologias mais modernas, foi possível estabelecer vários parâmetros de boas práticas para maximizar o rendimento no cultivo de peixes, como a utilização de indivíduos geneticamente modificados, qualidade da alimentação e, mais importante, controlar o ambiente dos viveiros, monitorando parâmetros ambientais como a temperatura, a turbidez, o cloro, o *Potential of Hydrogen* (Ph) e o oxigênio dissolvido da água.

### 1.1 Considerações iniciais

O Oxigênio dissolvido (OD), nos viveiros de piscicultura ao atingir níveis baixos por longos períodos, faz com que os peixes não se alimentem, cresçam mais lentamente e fiquem mais suscetíveis a contrair doenças. Para evitar esse cenário são empregados aeradores mecânicos, os quais são acionados imediatamente quando o nível de concentração atinge um nível crítico (QUEIROZ, 2016). Dentro do mercado existem vários tipos de aeradores, porém poucos medidores de oxigênio dissolvido, gerando escassez no mercado e também preços a cima de R\$50.000, inviabilizando os produtores de terem um sensor para cada açude, devido à quantidade de açudes por produtor estar na casa das dezenas de unidades. Além disso, a fragilidade dos sensores de OD e a necessidade de se monitorarem os açudes 24 horas por dia inviabiliza o uso do equipamento de forma manual, portanto um sistema automatizado acaba sendo a forma ideal de se tratar com esse tipo de medida.

Dentro da indústria, a automação de processos com sensores é primordial, onde a linha de produção é projetada conforme o tipo de trabalho dos sensores empregados, com isso, existe um esforço para empregar e atualizar essas plantas industriais, para aportarem novas tecnologias de medição para intensificar e otimizar a produção, porém, algumas soluções foram necessárias para que esses novos sensores conseguissem ser instalados e utilizados, sem que, dentro da indústria, para se adaptar à essas novas tecnologias, precisa-se redefinir toda a cadeia de produção, pelas quais, suas estruturas e métodos de produção já foram bem estabelecidas. Uma

dessas soluções foi o CDPR, sendo um tipo de manipuladores paralelos, em que o *End Effector* (EE) é suportado por dois ou mais cabos, controlados por atuadores de tensionamento. O EE pode ser qualquer tipo de carga, desde sensores, atuadores, motores ou simplesmente deslocar algum material de um lado ao outro da fábrica (MERSI SINA VALI, 2018).

A pesquisa sobre CDPR levou a uma série de desafios, questões científicas no campo da cinemática, estática, dinâmica, controle e design (BRUCKMANN, 2013). Um exemplo de CDPR, é o *Five-hundred-meter Aperture Spherical Telescope* (FAST) localizado na China, que é o maior radiotelescópio do mundo. Ao ser desenvolvido pela Academia de Ciência Chinesa, os engenheiros responsáveis se depararam com o problema de mover a cabine receptora do telescópio sobre a cúpula de mais de 500 metros de diâmetros. A solução encontrada foi empregar um CDPR de seis cabos, projetado para mover a cabine de foco ou EE, sem precisar de uma estrutura rígida, dessa forma diminuiu o preço e o nível de complexidade do projeto, além de ter proporcionado maior versatilidade do movimento da cabine.

## 1.2 Objetivos

A pretensão desse trabalho é desenvolver um robô autônomo paralelo acionado por cabos CDPR genérico, com o intuito de seu EE ser uma sonda de medição de OD em açudes de piscicultura. O sistema precisa ser genérico a ponto de ser simples de dimensionar para campo de trabalhos maiores, e que, a cada caso empregado, necessite de poucos parâmetros para se adaptar e poder trabalhar no local desejado.

### 1.2.1 Objetivo geral

Desenvolver um protótipo de robô paralelo acionado por cabos para deslocar uma sonda de medição de OD, a fim de diminuir custos de aplicação das técnicas de controle de OD em viveiros de piscicultura, voltado para pequenos e médios produtores.

### 1.2.2 Objetivos específicos

O desenvolvimento do robô é dividido em quatro objetivos específicos:

- Pesquisar de forma ampla motores disponíveis no mercado, bem como outros sensores que serão utilizados para fazer o sistema de controle e de comunicação, selecionando de acordo com sua robustez e custo.
- Aplicar o modelo matemático do sistema de movimentação em três dimensões da sonda.

- Programar e simular o modelo a partir de um microcontrolador em um sistema embarcado.
- Projetar e construir um ambiente de testes, com a estrutura de motores e sistema embarcado.

### **1.3 Justificativa**

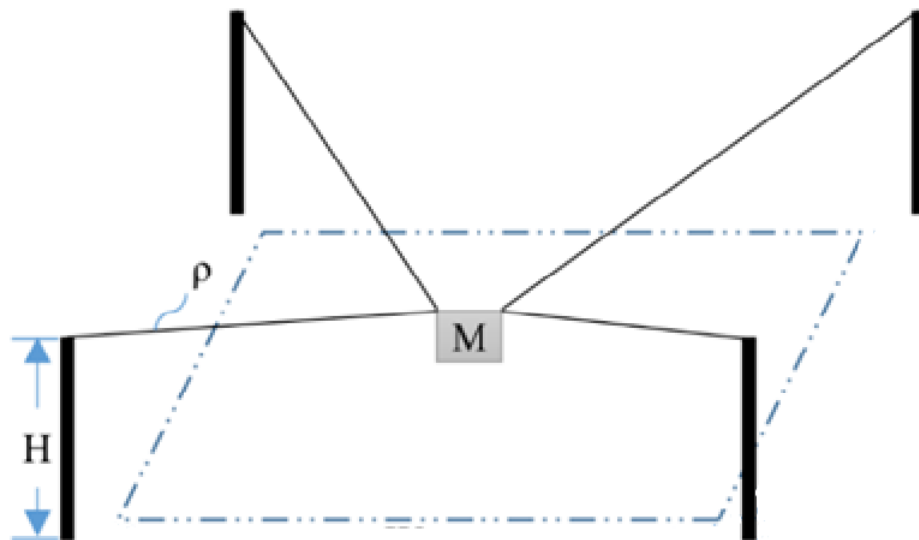
A Empresa Brasileira de Pesquisa Agropecuária (EMBRAPA) em um comunicado técnico, apresentou um conjunto de boas práticas de manejo para manter uma concentração adequada de OD em viveiros (QUEIROZ, 2016). Levando em consideração os impactos ambientais que a troca de água dos viveiros causa, é fortemente recomendados e empregados aeradores mecânicos que geram turbidez na água, assim elevando o nível de OD quando é detectado um nível crítico de OD. Mas em geral, grande parte dos produtores não tem meios de fazer uma medida confiável do nível de OD, e em alguns casos se quer fazem algum tipo de medida e optam por buscarem experiências empíricas de outros produtores, causando dois extremos que prejudicam o rendimento da produção: a ativação dos aeradores periodicamente durante o dia, o que não consegue satisfazer os níveis ideais de OD, devido o teor variar de acordo com a quantidade de peixes que habitam o açude; e a perda de OD elevada durante a noite. Para contornar esse problema é aumentado a periodicidade dos aeradores, o que gasta mais energia e reflete no aumento dos gastos de produção.

Portando o objetivo do projeto é criar um CDPR no qual o EE é uma sonda que produz medições do OD desses viveiros e remotamente sincroniza o nível medido com o acionamento dos aeradores, melhorando a produção e reduzindo gastos com energia, além de diminuir drasticamente os custos de implantação, por só precisar de um único sensor de oxigênio dissolvido.

## 2 REFERENCIAL TEÓRICO

Este capítulo aborda os parâmetros físicos, mecânicos e matemáticos que determinam o movimento do EE no plano e no espaço pelos atuadores de tensionamento. Robôs paralelos são definidos como robôs que fecham cadeias cinemáticas, ou seja, vários atuadores se encontram em um EE ou articulação (PIAO XUEJUN JIN, 2017). Essa composição de componentes e a forma que são dispostos, é observado na Figura 1.

**Figura 1 – Visão superior de um sistema CDPR de quatro cabos**



**Fonte: Adaptado de Newman (2013).**

O robô é composto de poucos elementos, sendo eles: altura das hastes verticais de apoio dos cabos e atuadores mecânicos “H”; densidade de Cabo “ $\rho$ ”; região em pontilhado representa a área de atuação do EE no espaço e no plano; “M” massa do EE (NEWMAN., 2017).

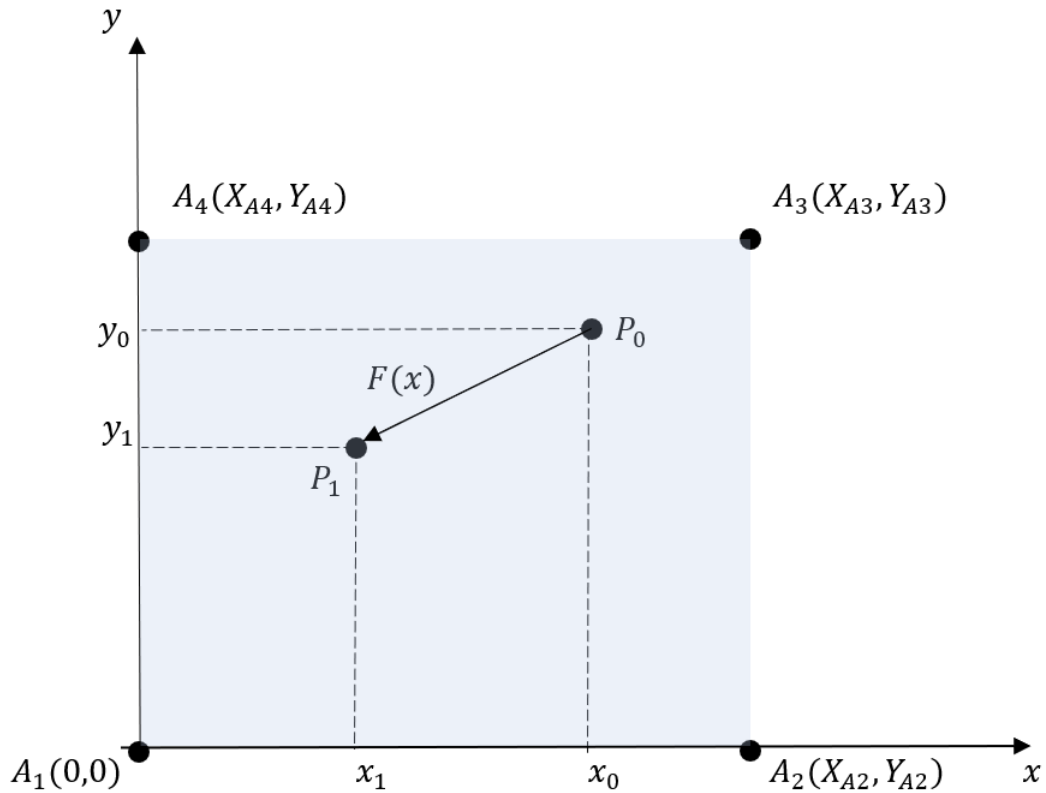
### 2.1 Determinação da posição do EE no plano

Para determinar a posição inicial e a posição final do EE, a fim de simplificar o movimento, primeiro trabalha-se no deslocamento simples no plano  $x,y$ , definindo como sendo o plano de deslocamento, e situando o atuador principal estando posicionado na origem, e os demais, como outros pontos distribuídos da forma escolhida. Dessa forma, a área formada entre os pontos  $A_1, A_2, A_3$  e  $A_4$  determina a região de atuação do EE em duas dimensões, o qual consegue se deslocar do ponto  $P_0$  ao ponto  $P_1$ , que se localiza dentro da área de atuação Figura 2.

A equação reduzida da reta:

$$F(x) = mx + n \quad (1)$$

**Figura 2 – Parâmetros de deslocamento e trajetória do EE no plano**



Fonte: Autoria própria (2022).

formada pelos pontos  $P_0$  e  $P_1$  representa o trajeto que o EE irá traçar no plano. Ela será importante para o programa que fará os cálculos algébricos, porque ela estabelece uma trajetória, que se analisada evita que o EE colida com obstáculos encontrado em algumas áreas de trabalho.

## 2.2 Equação do trabalho do atuador no plano

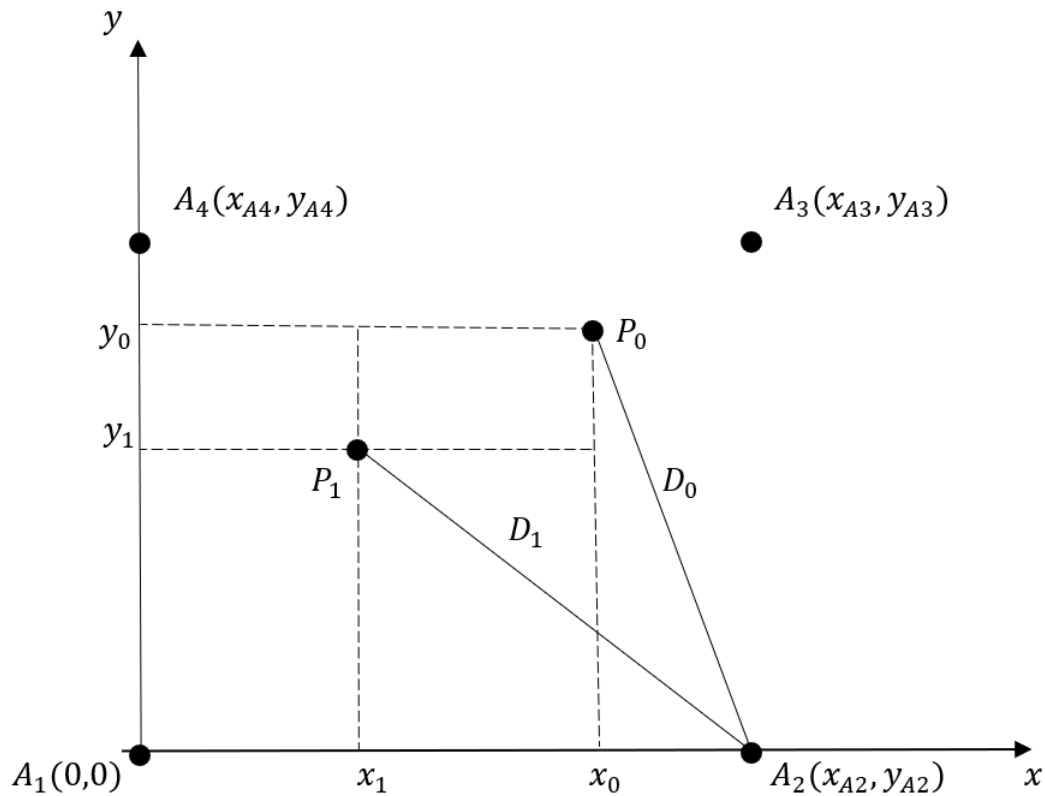
A fim de estabelecer a equação principal que determinará a forma que os atuadores irão trabalhar, precisa-se estabelecer uma relação entre o movimento do motor com a quantidade necessária de cabo para movimentar o EE (MARLET, 2006). Pode-se observar que a distância dos pontos  $P_0$  e  $P_1$  até o atuador  $A_2$  corresponde ao comprimento do cabo  $D_0, D_1$  conectado ao EE (Figura 3).

O EE se deslocando no plano, será determinado somente pelas componentes de distâncias entre pontos de medições até os atuadores mecânicos. Portanto a variação entre as distâncias  $D_1$  e  $D_2$  dos pontos  $P_0$  e  $P_1$  até o atuador, corresponde com a variação da distância de cabo que o atuador irá tracionar ou soltar (BRUCKMANN, 2013), que pode ser formulado como:

$$\delta D_{A2} = \sqrt{(x_0 - x_{A2})^2 + (y_0 - y_{A2})^2} - \sqrt{(x_1 - x_{A2})^2 + (y_1 - y_{A2})^2}, \quad (2)$$



Figura 3 – Distância de cabos entre pontos e atuador no plano



Fonte: Autoria própria (2022).

onde o resultado da operação, caso seja positivo corresponde ao atuador trabalhar distendendo o cabo, e caso negativo corresponde ao atuador tracionando o cabo. Aplicando a Equação (2) para cada atuador, sendo que cada um corresponde a um ponto no plano,  $X_{AN}, Y_{AN}$ , somando as diferenças de todas as distâncias de cabo, a fim de que todos estejam totalmente tracionadas, a variação total pode ser aproximada como igual a zero:

$$\sum_{N=1}^4 \Delta D_{AN} \approx 0, \quad (3)$$

caso essa diferença seja maior que zero significa que o peso da sonda está tracionando o sistema para baixo, funcionando assim como um quinto atuador móvel que sempre está puxando os cabos na direção perpendicular ao chão.

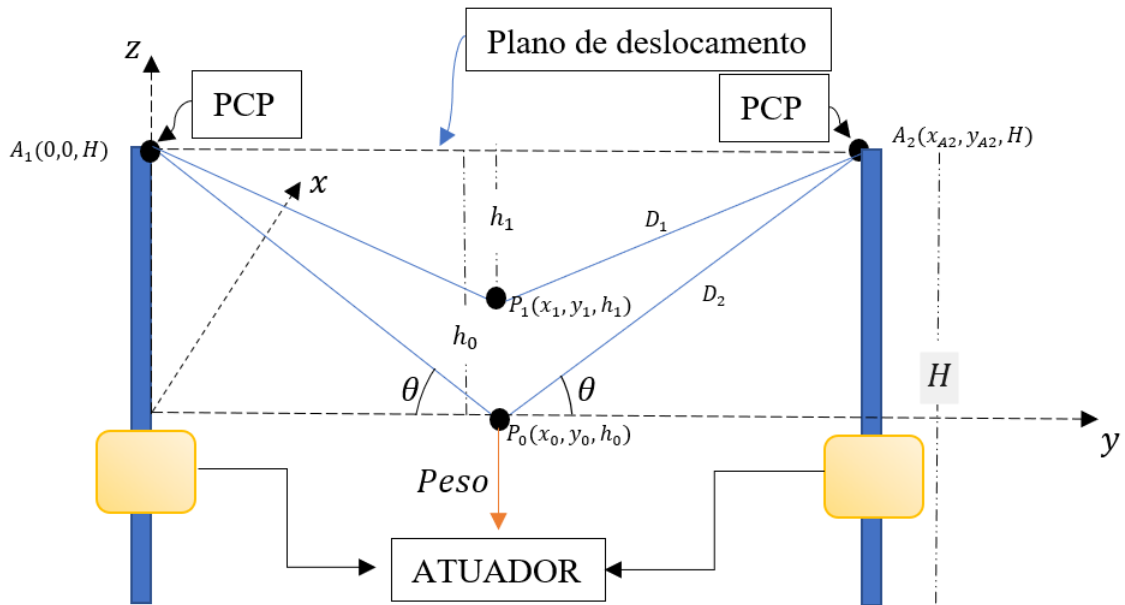
### 2.3 Movimento do EE no espaço

Supondo que o EE tenha se deslocado no plano e que esteja acima da região de medição, então, o peso total dos cabos e da sonda, irão influenciar como se fosse um quinto cabo no sistema. Então, para fazer o mergulho do EE, três parâmetros são necessários para fazer um

modelo: a densidade do material,  $\rho$ , que compõe os cabos; a massa total  $M$  do EE; a altura  $H$  em relação aos postes que os atuadores estão instalados até o plano de deslocamento localizado em cima do sistema (Figura 4).

Nos cálculos considera-se os postes estejam na mesma altura, fazendo com que o plano de deslocamento não esteja inclinado. A distância de cabos será formulada como a distância entre o ponto de medição  $P_{xyz}$ , até o Ponto de Contato com o Plano (PCP) no espaço. A distância  $h$  varia de acordo com a altura do ponto de medição até o plano de deslocamento.

**Figura 4 – Visão lateral do sistema CDPR e parâmetros de deslocamento do EE no espaço**



**Fonte: Autoria própria (2022).**

Portanto a variação entre as distâncias  $D_1$  e  $D_2$  dos pontos  $P_o$  e  $P_1$  até o atuador  $A_1$  no espaço corresponde com a variação da distância de cabo que o atuador irá tracionar ou distender, que pode ser formulado como:

$$\delta D_{A2} = \sqrt{(x_0 - x_{A2})^2 + (y_0 - y_{A2})^2 + (h_0 - H)^2} - \sqrt{(x_1 - x_{A2})^2 + (y_1 - y_{A2})^2 + (h_1 - H)^2}, \quad (4)$$

se diferenciando do deslocamento no plano, cujo deslocamento do EE através dele, apresenta a somatória de todos os cabos tracionados ou distendidos, como aproximadamente zero Equação (3). No caso do deslocamento no espaço, a soma dessas distâncias, a serem trabalhadas, para que o EE varia a sua altura, precisa ser diferente de zero. Sendo assim, para atingir esse movimento, o somatório das quantidades de cabos de todos os quatro atuadores, caso seja maior que zero, significa que a sonda diminuiu sua altura, e caso seja negativa, a sonda aumentou sua altura.

O vetor peso representa o quinto cabo tracionando perpendicular ao solo, sendo que a força será distribuída de forma não uniforme entre os quatro cabos dos atuadores, devido ao fato, que, conforme o EE se desloque entre os pontos de interesse, as distâncias de cabos variam conforme o movimento vai acontecendo, e somente quando a sonda está posicionada diretamente no centro do sistema, significa que os quatro cabos estão sofrendo o mesmo esforço do peso. Portanto, considerando que os esforços referente a força do peso no EE, para acontecer o movimento dos cabos, os motores dos atuadores precisam vencer essa força peso, porém, como esse esforço varia conforme a posição do EE, os mesmos motores são comumente superdimensionados para o peso do EE quando algum sistema CDPR é projetado. Porém, pela perspectiva da tração nos cabos, o peso corresponde à força do quinto cabo, no qual será o único fator que deixará os mesmos tracionados, e também sendo o motivo pelo qual o sistema consegue se mover no espaço (WANG, 2021), dessa forma, existe um balanço entre, ter um EE que não seja pesado o suficiente para impedir que os motores trabalhem, mas que seja pesado o suficiente para que os cabos se mantenham tracionados.

O momento em que o esforço de tração sera menor em um cabo de um tensionador mecânico, é quando o EE se posicionar o mais longe possível do mesmo, portanto, uma forma de melhorar essa tração, sem precisar aumentar o peso do EE, é utilizando cabos com maior densidade  $\rho$ , fazendo que, nessa condições em que o peso do EE esteja exercendo menos esforço no cabo, o próprio peso do cabo gera uma tração.

### 3 MATERIAIS E MÉTODOS

Este capítulo descreve os materiais e métodos que foram empregados para o desenvolvimento do CDPR. É apresentado também um protótipo funcional, empregando a fundamentação teórica na prática, sendo uma primeira etapa antes de escalar suas dimensões para campos maiores de atuação.

#### 3.1 Metodologia

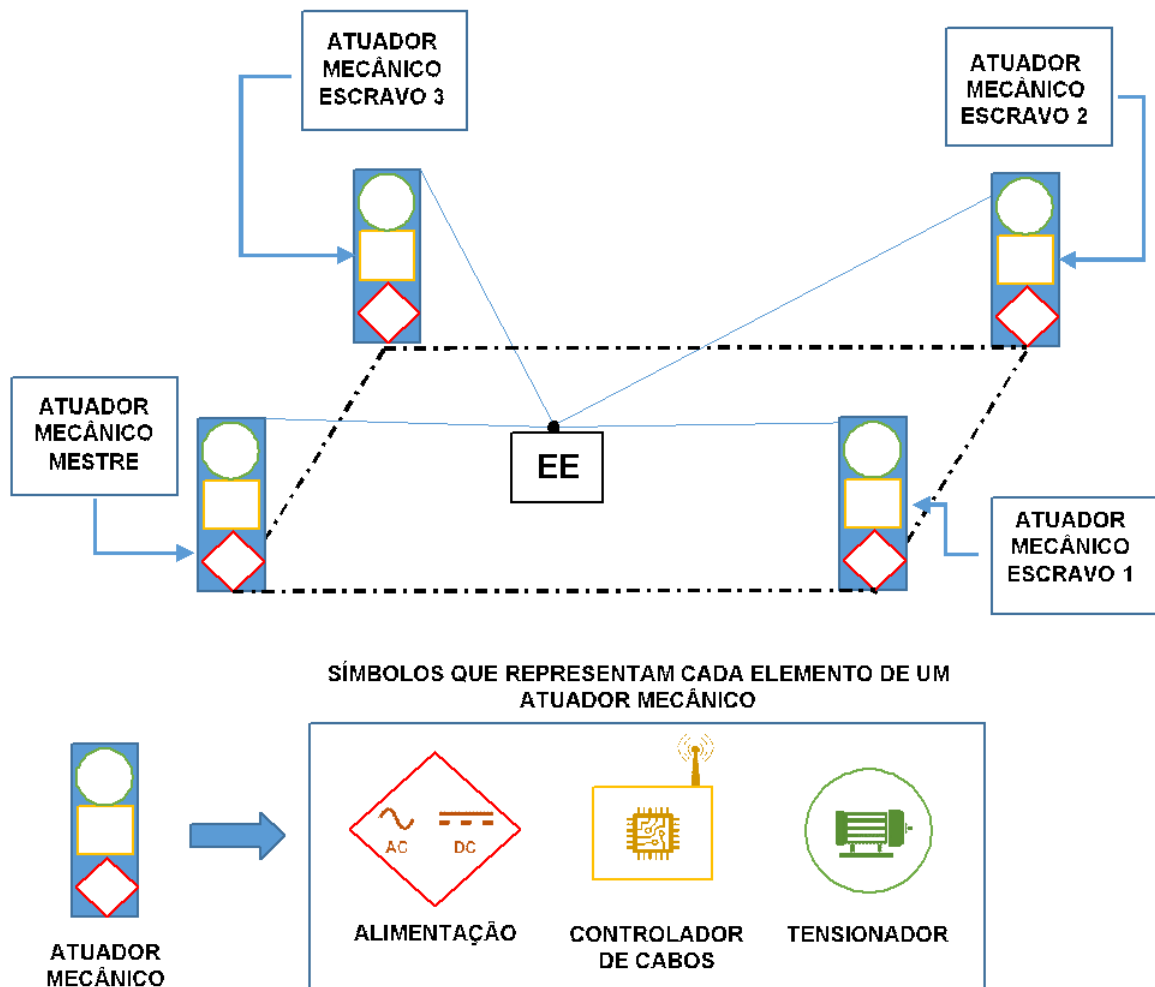
O dispositivo é dividido em 3 principais partes as quais compõem os atuadores mecânicos. Tensionador: abrange as coordenadas, altura, peso e motor; controlador de cabos: se trata do circuito eletrônico composto por microcontrolador, transceptor de rádio e encoder rotativo; alimentação: fornece energia para o tensionador e o controlador de cabos.

O sistema trabalha com poucos parâmetros fixos, onde o usuário entra com as coordenadas iniciais do EE sendo um ponto no plano, e os pontos de medição sendo coordenadas no espaço. Depois disso o circuito controlador de cabos do atuador mecânico mestre calcula as distâncias de cabos a partir dessa coordenada inicial, e comanda cada atuador mecânico escravo via rádio para soltar e tracionar os cabos suficientes para atingir os pontos de medições definidas pelo usuário. Cada atuador monitora por meio do encoder rotativo se as distâncias  $D$  foram atingidas de acordo com o que foi determinado pelo atuador mestre. Todos os atuadores são compostos pelos mesmos componentes, portanto todos os 4 podem ser definidos como o atuador mestre.

Com o objetivo de simplificar o movimento do EE, o deslocamento do mesmo acontece somente em duas dimensões, de tal forma que, calcula-se as distâncias de cabos a partir de pontos distribuídos no plano de deslocamento, de tal forma que, os pontos de interesse que o EE visite terão coordenadas situadas em  $x$  e  $y$ . E uma vez que esses pontos foram atingidos o sistema faz o procedimento de cálculo de mergulho do EE.

Devido as longas distâncias entre os atuadores, e para evitar o máximo intervenções nas propriedades em que o sistema possa ser implementado, é necessário que a comunicação entre o atuador mecânico mestre e os escravos, seja feita via wireless, assim evitando o transtorno de instalar cabos de comunicação. Levando em consideração não só as longas distâncias, mas também a durabilidade, preço e disponibilidade no mercado, os transceptores de *Radio frequency* (RF) de 2,4GHz é a forma mais viável e simples de se implementar esse recurso. Em comparação com o concorrente mais próximo o protocolo de comunicação *Long Range Wide Area Network* (LoRaWan) que apesar de conseguir se comunicar por maiores distâncias, perde nos demais requisitos de custo e disponibilidade no mercado.

Figura 5 – Digrama de composição dos quatro atuadores mecânicos



Fonte: Autoria própria (2022).

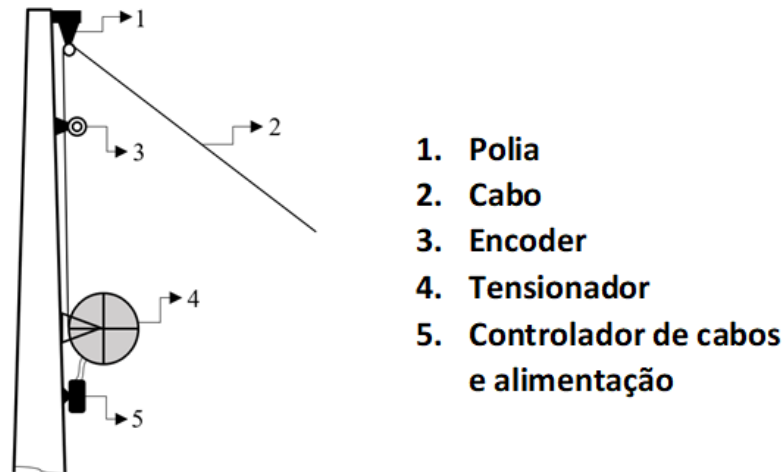
### 3.1.1 Atuadores mecânicos

O atuador mecânico consiste no equipamento responsável por fechar a cadeia cinemática, o qual vai criar a ponte entre o movimento dos cabos com o EE (PIAO XUEJUN JIN, 2017). Os principais componentes podem ser vistos na Figura 6.

### 3.1.2 Circuito controlador de cabos

A configuração básica do controlador de cabos é composta por: circuito de alimentação; microcontrolador; circuitos de condicionamento de direção do motor; sensor de rotação encoder.

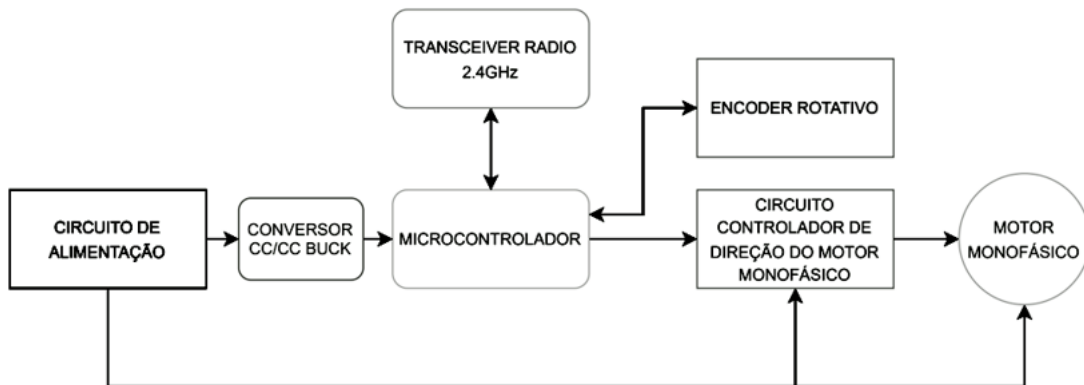
**Figura 6 – Componentes do atuador mecânico**



Fonte: Autoria própria (2022).

O circuito de alimentação recebe duas fases e um neutro da rede elétrica já existente do local de trabalho. Sendo que duas fases serão direcionadas para alimentar o motor monofásico e o circuito de condicionamento de motor. Uma fase e um neutro para o circuito conversor CA/CC. Alguns componentes do circuito necessitam de diferentes tensões para funcionar, portanto o circuito de alimentação também é composto por um conversor CC/CC *buck*.

**Figura 7 – Diagrama de blocos do circuito eletrônico de um atuador mecânico**



Fonte: Autoria própria (2022).

O modo que o atuador trabalha, exige uma precisão na medição das distâncias de cabos que foram tracionadas ou distendidas. Uma forma de medir, é através da transladação de uma polia acoplada a um sensor rotativo. Quando essa polia é pressionada contra o cabo, a tração exercida entre ambos transforma o movimento retilíneo do cabo em um movimento de rotação da polia, criando uma razão entre o movimento angular com a distância de cabos tracionados ou soltados.

A comunicação entre os atuadores foi feita através de um *transceiver* de rádio com frequência de 2.4 GHz. Esse *transceiver* permitiu a comunicação entre o atuador mecânico

mestre e os demais atuadores escravos em até 600 metros de distâncias. Esse componente, atua dentro da banda *The industrial, scientific, and medical radio band* (ISM), na qual suporta altas transferências de dados e, é comumente utilizado em aparelhos comerciais de baixo consumo de energia, que necessitam de comunicação em longas distâncias. Pelo fato de ser muito utilizado, aparelhos que trabalham nessa frequência, acabam sendo vulneráveis a interferências.

O código fonte do sistema de controle e de comunicação, foi desenvolvido com a linguagem de programação C++. Por se tratar de uma linguagem muito utilizada em sistemas embarcados, proporciona maior desempenho, e também acesso a bibliotecas de código open source já desenvolvidos para alguns tipos de componentes.

O programa foi desenvolvido de forma que, pré-estabelecer alguns parâmetros específicos de cada área de trabalho como algumas entradas de cada usuário. Enfatizando assim a característica do sistema de ser genérico ao ponto de que, poucos parâmetros do sistema CDPR que possam mudar de acordo com a área de trabalho, seja alterada dentro do código.

### 3.1.3 Tensionador

O tensionador é composto por dois componentes: motor elétrico que gera o movimento de tração e liberação dos cabos; polia que faz o intermédio entre o EE e o motor. Nesse sistema, as três únicas características visadas no motor será o tipo de alimentação, ter controle de direção e ser resistente a local de trabalho aberto.

Os motores de corrente contínua, apesar de terem um controle de rotação muito preciso, são muito sensíveis e muito caros para esse tipo de aplicação, além de que, pode haver casos de não terem força o suficiente para mover o EE. Motores monofásicos, muitas vezes possuem um sistema de redução tipo, rosca sem fim, que facilita a manutenção e reduz os custos. Possuem maior resistência mecânica e são extremamente silenciosos. Conseguem entregar um torque muito mais alto, sendo o seu uso nessa aplicação superdimensionado para qualquer tipo de EE.

## 3.2 Materiais

Esta seção descreve os sensores e componentes eletrônicos e elétricos que serão utilizados para a implementação do dispositivo CDPR.

### 3.2.1 Microcontrolador Arduino Nano

O microcontrolador Arduino Nano é amplamente empregado em soluções *Internet of Things* (IoT) devido à sua versatilidade em atender vários requisitos de um microcontrolador em um uma placa de tamanho reduzido. Utiliza um processador ATmega328 de 8 bits de alta

performance e baixo consumo de energia. As características do microcontrolador pertinentes ao projeto informadas pela fabricante são:

- *Clock* de 16 MHz (ajustável);
- Tensão de trabalho: 1,8 até 5,0;
- Corrente de consumo típica: 0,2 mA;
- Temperatura de trabalho: -40 até 85°C;
- Portas: 14 portas digitais, 8 canais A/D (10 bits);
- Saídas de tensão regulada de 5,0 e 3,3 V.

### 3.2.2 Motor monofásico

Com o preço mais acessível e por exercer maior força no eixo através do seu sistema de redução, o motor monofásico que irá compor os quatro tensionadores é o motor de redução modelo VXRSA da marca VariVelox. Comumente usados em agitadores e resfriadores de leite, ele possui ótima resistência a locais de trabalhos extremos, e apresenta grau de proteção IP44 contra poeira e respingos de água.

**Figura 8 – Motor monofásico VariVelox, VXRSA**



**Fonte: Autoria própria (2022).**

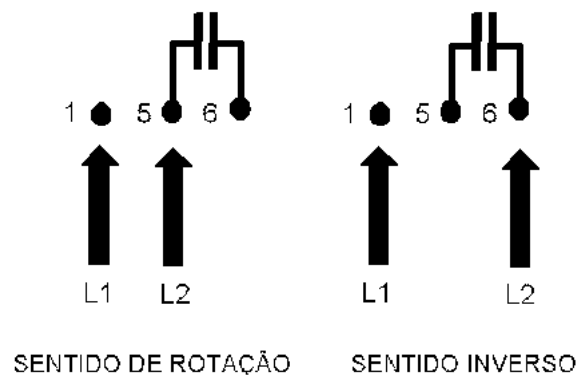
As características do motor pertinentes ao projeto e informadas pelo fabricante são:



- Potência de 1/20 CV;
- Tensão de 220/254 V, corrente de 0,5 A e 60 Hz;
- 1100 RPM com redução para 30 RPM;

O sentido da rotação do motor varia de acordo com a ligação de uma das fases, visto como L2 na Figura 9 com os terminais 5 e 6, que estão conectados com as bobinas do motor.

**Figura 9 – Esquema de ligação para cada sentido de rotação do motor monofásico**



**Fonte: Adaptado de VariVelox.**

### 3.2.3 Rotary encoder óptico digital

O sensor de rotação a ser utilizado é o LPD3806-600BM-G5-24C mostrado na Figura 10, sensor incremental de duas fases AB, com uma resolução de 600 pulsos por revolução, e velocidade máxima de 6000 revoluções por minuto. Atua em uma faixa de tensão entre 5 a 24 V e frequência de resposta 0 até 24 KHz. Ele é comumente utilizado para medir a velocidade de rotação de um objeto, ângulo, aceleração e medição de comprimento.

### 3.2.4 Transceptor de rádio NRF24L01

O módulo de rede sem fio fabricado pela Nordic é um módulo transceptor de rádio de um único chip, visto na Figura 11, com frequência de banda entre 2,4 – 2,5 GHz, atuando dentro da ISM *Band*, o componente é composto por um amplificador e sintetizador de sinal, cristal oscilador, modulador, demodulador e protocolo de comunicação *Enhanced ShockBurst* integrado. A comunicação entre o chip e o microcontrolador, é feita através da interface *Serial Peripheral Interface* (SPI).

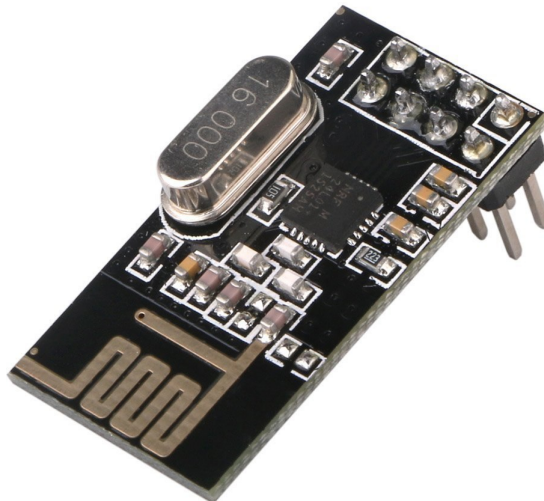
**Figura 10 – Encoder rotativo LPD3806-600BM-G5-24C**



**Fonte: Autoria própria (2022).**

O alcance de comunicação do módulo com antena embutida e em condições ideais, é de no máximo 1000 metros de distância, sendo que todos os quatro atuadores mecânicos recebem e enviam dados através desse módulo.

**Figura 11 – Transceptor NRF24101**



**Fonte: Nordic.**

### 3.2.5 Custos de implementação

Como o objetivo desse trabalho, é fornecer uma solução de baixo custo em comparação com o medidor de OD, se fez necessário fazer um orçamento de todos os componentes mais vitais do CDPR. Como pode ser observado na Tabela 1, alguns elementos como os postes de

suporte dos tensionadores e os cabos do sistema, não foram orçados, porém, levando em conta os valores de mercado de cada sensor de OD, pode se concluir que tais elementos mesmo se acrescentados aos custos finais, a rentabilidade do sistema ainda fica muito superior comparado a outra opção de instalar um medidor para cada viveiro.

**Tabela 1 – Orçamento dos materiais**

Material	Valor (R\$)	Quantidade	Total (R\$)
Motor VariVelox, VXRSA	635,35	04	2.541,40
Encoder rotativo LPD3806-600BM-G5-24C	153,90	04	615,6
Módulo NRF24L01 + Antena	38,97	04	155,8
Placa Arduino Nano V3	65,99	04	263,93
		Total (R\$)	3.576,8

**Fonte: Aatoria própria.**

## 4 RESULTADOS

Durante a primeira semana de junho até o último dia útil do mês de outubro de 2022, foi desenvolvido o primeiro protótipo funcional do sistema CDPR. As atividades foram desempenhadas de forma que, o robô conseguisse alcançar 6 pontos distintos posicionados dentro da sua área de trabalho, com uma taxa de erro relativamente baixa. Dessa forma, provando que o modelo matemático proposto, condizia com a realidade experimental.

### 4.1 Ambiente de testes

Com este trabalho espera-se que o EE consiga alcançar 6 pontos de medição estipulados de forma arbitrária com um erro máximo de até 20 centímetros de distância entre o ponto estipulado e o ponto alcançado na prática. Durante a implementação, para garantir precisão no posicionamento dos tensionadores, utilizaram-se equipamentos para determinar as dimensões dos quatro tensionadores. Nesse caso alguns equipamentos de medição de precisão, como trena a laser e hipsômetro foram utilizados para que todos os quatro atuadores tivessem os valores de altura e inclinação os mais exatos possíveis.

O sistema foi implementado em um ambiente de testes com dimensões reduzidas em relação à área de trabalho onde seria originalmente projetado para atuar. O ambiente de testes foi estabelecido dentro de uma sala comercial da PigMa - Bem estar animal, empresa parceira da UTFPR. O local forneceu espaço necessário para que a sonda se deslocasse em ambiente controlado sem as intempéries do ambiente aberto. As dimensões do sistema podem ser observadas no esquema resumido do ambiente de testes Figura 12.

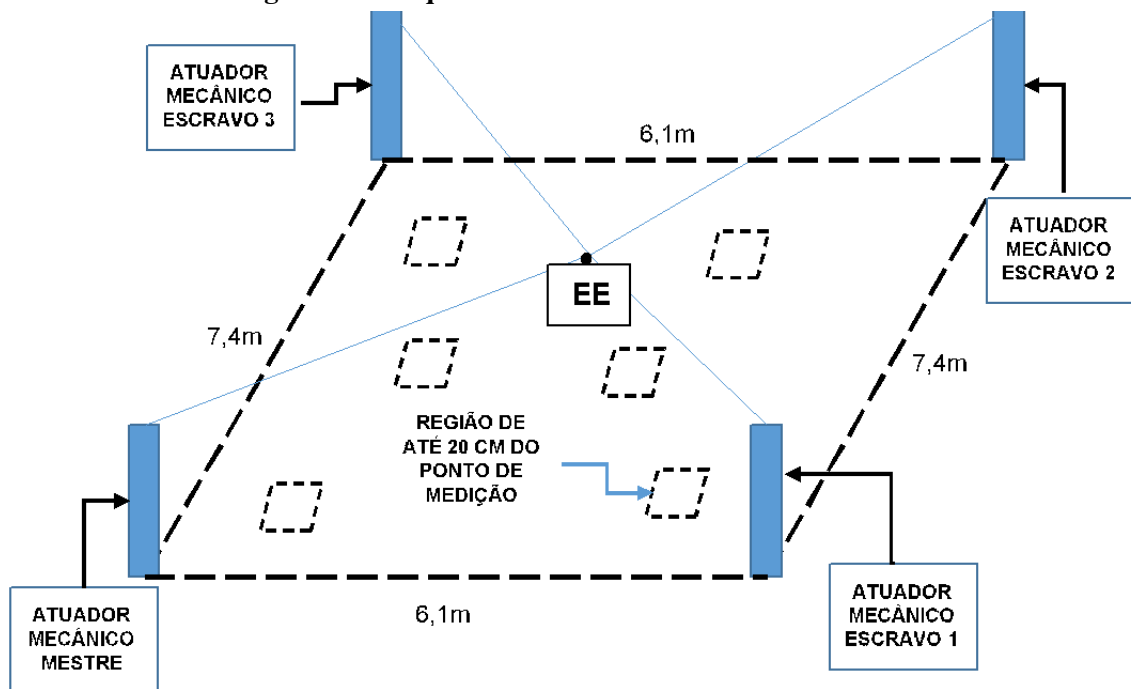
Foi necessário transformar quatro pilares de sustentação da sala comercial, nos quatro atuadores mecânicos. Dessa forma, a área de trabalho do sistema se limitou as dimensões dos pilares, como pode ser observado no Tabela 2. Nota-se que, esses valores foram medidos tomando de referência e origem do sistema, a qual foi determinada como sendo o atuador mecânico mestre. Todos os quatro atuadores foram instalados de forma que ficassem os mais equidistantes possíveis, e com a mesma altura, no intuito de que os quatro atuadores seguisse a mesma estrutura física. O PCP de todos os quatro atuadores mecânicos, foi estabelecido a 2,5m de altura.

**Tabela 2 – Posição dos atuadores mecânicos no ambiente de teste**

Atuador	Distância em x	Distância em y	altura
Mestre	0cm	0cm	250cm
Escravo 1	746cm	0cm	250cm
Escravo 2	0cm	542cm	250cm
Escravo 3	746cm	534cm	250cm

**Fonte: Autoria própria (2022).**

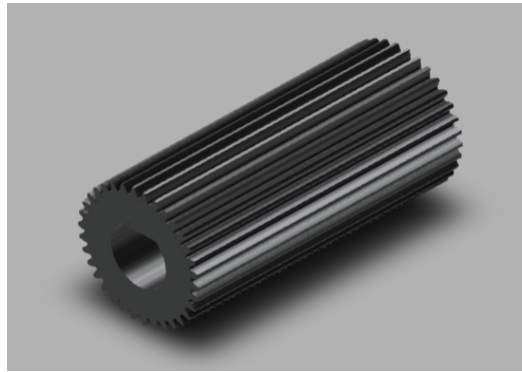
Figura 12 – Esquema resumido do ambiente de testes



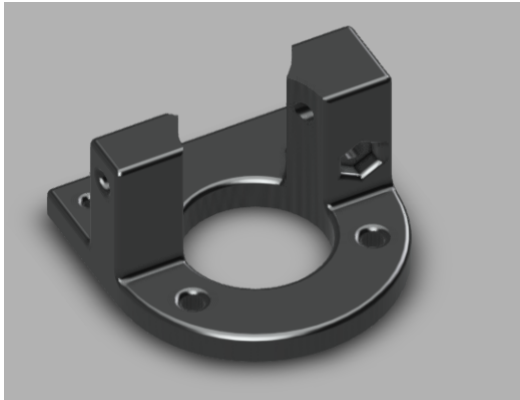
Fonte: Autoria própria (2022).

Durante o processo de montagem, projetou-se em 3D, três peças no intento de acoplar os *encoders* à estrutura física dos pilares, bem como manter pressionado o cabo contra o elemento rotativo responsável pelas medições de distâncias dos cabos (Figura 13). Algumas peças de suporte foram requeridas para acoplar o extruder junto ao pilar de sustentação. O resultado das peças montadas pode ser visto na Figura 14. Um teste das peças impressas foi conduzido para validar o desempenho delas. Para esse teste, o extruder foi acoplado em somente um dos tensionadores. Com a estrutura montada, o motor começou a movimentar o cabo, girando o encoder rotativo e o Arduino foi registrando esse movimento e convertendo para distância de cabos. Foi pré-determinado que em cada 2 metros de cabos tensionados, seria aceitável que a leitura tivesse uma imprecisão de até  $\pm 10$  cm, e o movimento deveria acontecer em até 11 segundos. No entanto, houve uma perda muito maior entre as distâncias medidas e as registradas, devido ao tipo e dimensões do material utilizado como cabo. Por serem fios de poliéster encerados com espessura de 1 mm, eles apresentavam pouco atrito com as peças de plástico impressas em 3D.

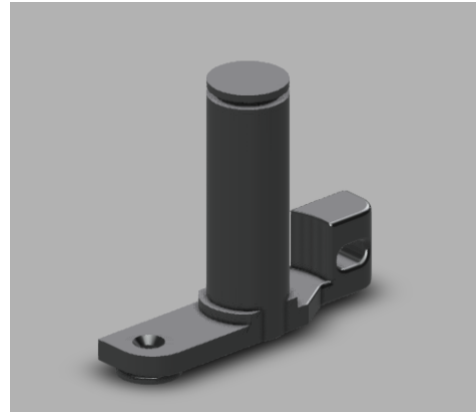
**Figura 13 – Peças impressas em 3D: (a) Rolete encoder, (b) Extruder peça 1, (c) Extruder peça 2**



**(b) Extruder peça 1**



**(c) Extruder peça 2**



**Fonte: Autoria própria (2022).**

**Figura 14 – Encoder rotativo montado**



**Fonte: Autoria própria (2021).**

Para analisar melhor esses resultados, quatro baterias de testes foram realizadas, no qual a cada bateria, os cabos, precisariam trabalhar tracionando e soltando 2 metros de cabos. E em seguida outras quatro baterias de teste em que essa mesma distância seria alcançada em intervalos de 2 passos de 20 cm. Esses testes foram feitos a fim de avaliar se seria possível prever ou evitar que os roletes escorregassem e perdessem a precisão na leitura das distâncias de cabos, mesmo utilizando esses fios encerados.

O resultado na quantidade final de cabos, através do método de trabalhar as distâncias de forma direta, pode ser visto na Tabela 3. O erro percentual, foi maior do que o aceito nessa rodada de testes, em perspectiva com o que foi proposto com este trabalho. Para conseguir movimentar o EE em 6 pontos distintos no espaço, esse erro seria carregado para cada sequência de movimento individual dos 4 tensionadores, isso encarrataria na posição final do EE estar muito mais distante do que foi estabelecido com aceitável para os resultados desses testes iniciais. As distâncias de cabos após o final das quatro baterias de testes através do método de iterações (Tabela 4), o erro ficou 12,35%, esse valor viabiliza que o EE consiga se deslocar para mais de 4 pontos sem ter um erro grande em sua posição final. Também foi observado, a tendência do erro aumentar em relação com o erro anterior, em um fator muito mais gradativamente mais elevado. Isso ocorreu por dois fatores: os fios eram envoltos em uma cera protetora e, conforme os roletes deslizavam com o movimento dos barbantes, a cera se acumulava nos sulcos da impressão 3D, diminuindo ainda mais o atrito com as peças; também notou-se que o estresse dos cabos constantemente se movimentando pelas peças, faziam com que a tensão nas cordas ficasse menor, conseqüentemente criando uma folga entre a peça e o cabo, gerando imprecisão nas leituras.

**Tabela 3 – Leituras das distâncias sem intervalos**

Numeração do Teste	Distância medida	Erro percentual
1	1,81	10,49
2	1,78	12,35
3	1,69	18,34
4	1,59	25,78

**Fonte: Autoria própria (2022).**

**Tabela 4 – Leituras das distâncias com intervalos a cada 20 cm**

Numeração do Teste	Distância medida	Erro percentual
1	1,89	5,82
2	1,87	6,95
3	1,82	9,89
4	1,78	12,35

**Fonte: Autoria própria (2022).**

## 4.2 Placa de circuito impresso

O circuito é dividido em quatro partes: circuito de chaveamento de motor; circuito do transceptor de rádio; implementação do Atmega328p; conversor CC/CC *Buck*. Esses quatro elementos foram desenvolvidos em duas versões de circuito impresso, sendo que uma delas foi feita somente para provar o conceito do robô, e a outra como um protótipo funcional e final, desenvolvido com uma abordagem mais prática e comercial do sistema.

A placa de testes foi montada por componentes encontrados no mercado, com encapsulamento dos componentes mais complexos e sensíveis como o transceptor de rádio e microcontrolador. Além disso, são fáceis de se encontrar em lojas especializadas, reduzindo os custos e tornando possível substituir facilmente componentes que por ventura apresentassem mau funcionamento. O circuito de testes foi impresso através de uma *Computer numerical control* (CNC) que entalhou as trilhas do circuito em placas de fenolite cobreada de duas faces, reunindo dessa forma todos os componentes que compõem o circuito controlador de cabos em um único lugar.

O resultado do atuador mecânico montado pode ser visto na figura Figura 16. Foi necessário que todos os quatro atuadores fossem exatamente iguais para obter uma correlação melhor do sistema previsto no referencial teórico do sistema implementado, pois qualquer diferença física nos atuadores mecânicos reflete no erro final da posição do EE.

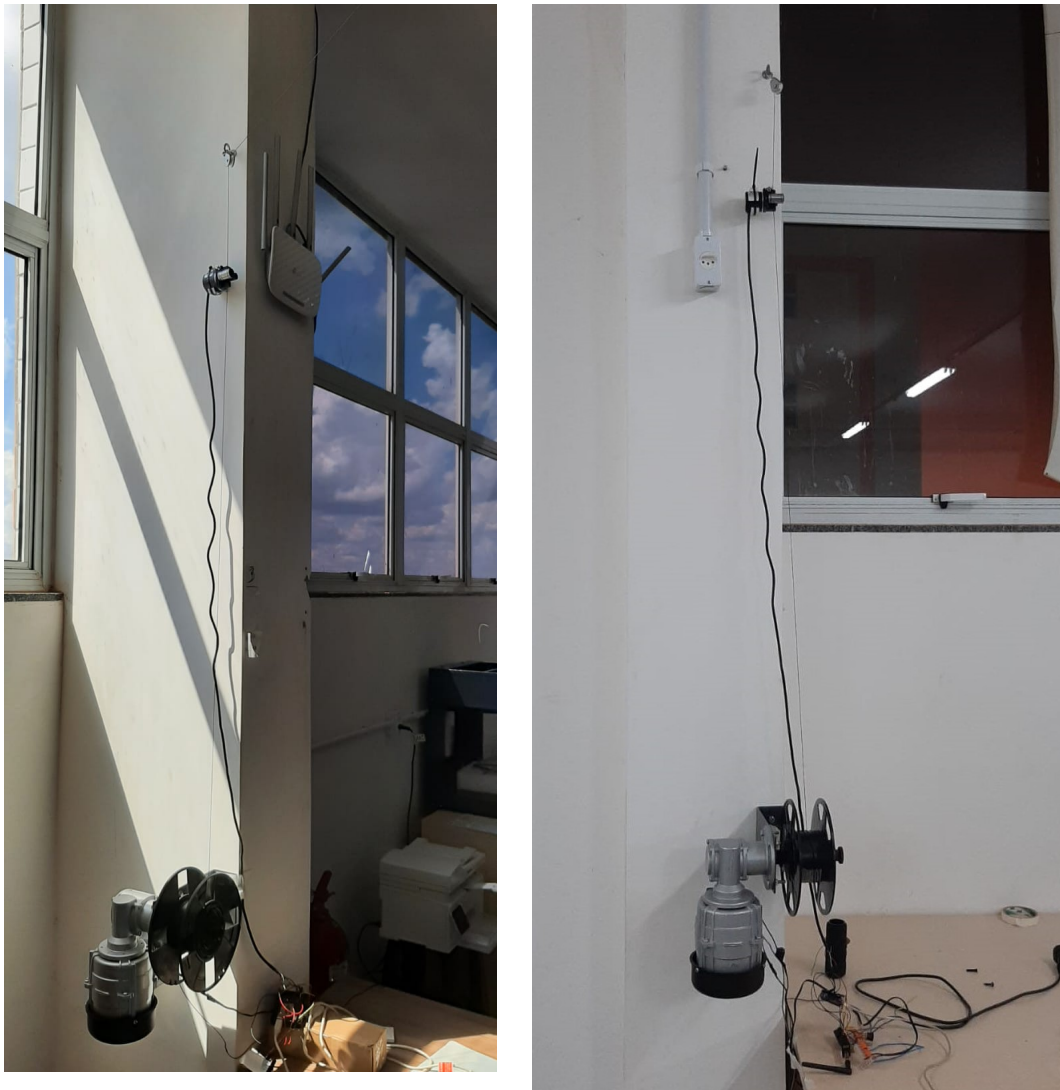
**Figura 15 – Placa de testes com componentes**



Fonte: Aatoria própria (2022).



**Figura 16 – Tensionadores Mecânicos: (a) Tensionador escravo 1, (b) Tensionador mestre**



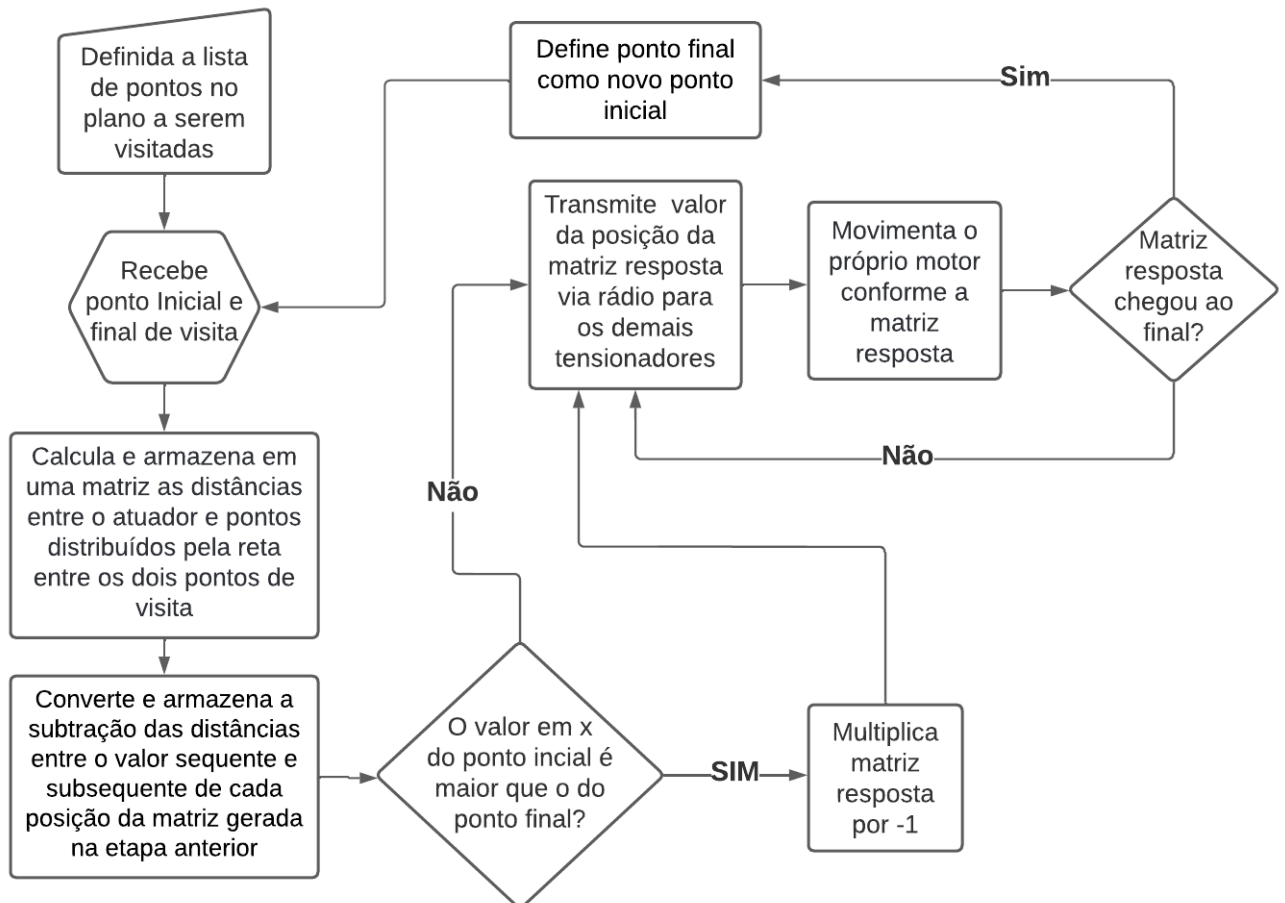
**Fonte: Autoria própria (2022).**

### **4.3 Software**

Foi desenvolvidos dois *softwares* para fazer o controle dos cabos: um para o atuador mecânico mestre, em que, é o maior e mais complexo de todos, cuja responsabilidade é de gerar os cálculos de cabos e transmitir por meio do rádio para os outros receptores, além de também movimentar os cabos referentes à sua parte como atuador; outro *software* para os demais atuadores mecânicos escravos, que só recebem valor do comprimento de cabos que devem tracionar ou soltar.

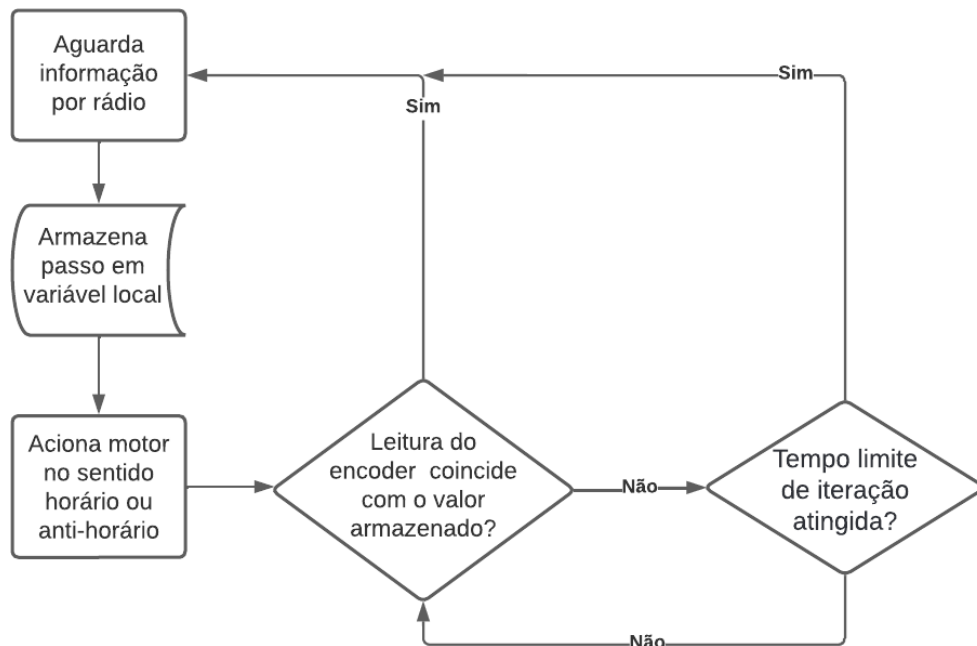
O desenvolvimento do código foi implementado conforme o que foi estruturado pelo diagrama de blocos, a Figura 17 para o controlador de cabos mestre e na Figura 18 para os demais controladores escravos. Elaborou-se primeiro a implementação dos cálculos de distâncias que foram abordados no referencial teórico Listagem 5, seguida das rotinas de como é tratado os resultados dos cálculos e, por fim, foi implementado a parte de comunicação via rádio.

**Figura 17 – Diagrama de blocos controlador de cabos mestre**



**Fonte: Autoria própria (2022).**

Figura 18 – Diagrama de blocos controlador de cabos escravo



Fonte: Autoria própria (2022).

#### Listagem 1 – Rotinas de calculos de distâncias no plano

```

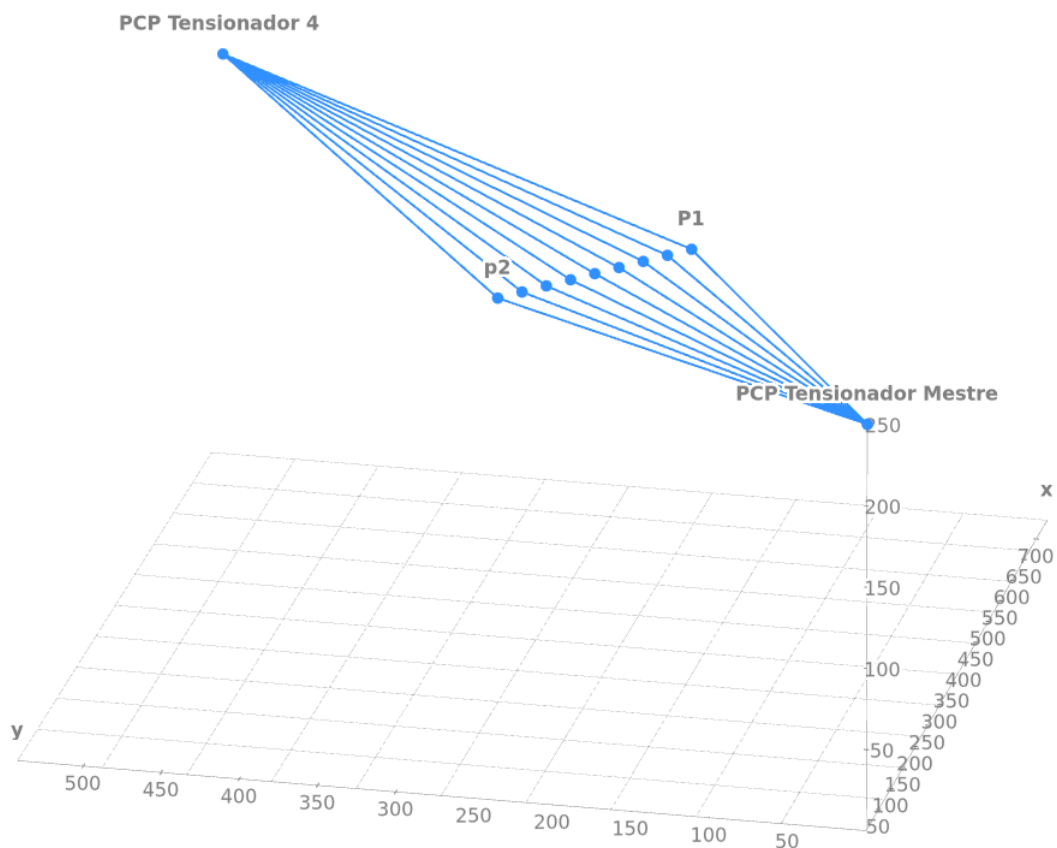
1  double distBetweenTwoPoints(Ponto BEG, Ponto END)
2  {
3      //Retorna a distancia entre dois pontos
4      return sqrt((pow((BEG.x - END.x), 2)) + (pow((BEG.y - END.y), 2)));
5  }
6
7  float angularCoeficent(Ponto BEG, Ponto END)
8  {
9      //Retorna o coeficiente angulas m entre dois pontos
10     return (END.y - BEG.y) * pow((END.x - BEG.x), -1);
11 }
12
13 float constOfflineEq(Ponto BEG, Ponto END)
14 {
15 //Retorna a constante c da eq. de reta
16 //Y=x.m + c -> (x,y) <-> c = y - x.m
17     return (END.y - END.x * angularCoeficent(BEG, END));
18 }
  
```

Fonte: Autoria própria (2022).

Nota-se a escolha de centralizar todos os cálculos do sistema no controlador de cabos do atuador mecânico mestre. Justifica-se essa escolha por motivo de todas as distâncias de cabos serem transmitidos via rádio, e para o EE se deslocar sem perdas e de forma mais eficiente, todos os quatro atuadores precisam atuar de forma conjunta e síncrona. Para isso, o controlador mestre precisa gerar os cálculos segundo os pontos definidos pelo software e transmitir para todos os quatro no mesmo instante.

Essa sequência de cálculos e transmissão dos dados, provou-se essencial conforme foram sendo realizadas as primeiras rodadas de testes do C DPR. Alguns recursos implementados em software foram desenvolvidos especificamente para mitigar as perdas geradas pelas partes mecânicas do sistema, como foi analisado nos testes anteriores, na qual foi obtido a relação das perdas de distâncias de cabos devido aos fios de barbantes e falta de tração dos cabos. Esses recursos foram cruciais na forma que o EE se deslocou. Portanto, foi implementado uma lógica de acionamento dos motores específica para esse projeto, no qual, o movimento dos cabos é dividido em iterações com intervalos entre elas. Como pode ser observado na Figura 19, a distância a ser percorrida entre os dois pontos de visita, P1 e P2, transformada em 8 pontos distintos e alinhadas por uma reta.

**Figura 19 – Movimento por iteração, com suas respectivas distâncias de cabos entre dois atuadores**



**Fonte: Autoria própria (2022).**

### Listagem 2 – Variáveis locais da função *sequenceOfIterations*

```

1 // variaveis locais
2 float m = angularCoeficent(BEG, END);
3 // coeficiente angular entre os dois pontos
4 float n = constOfflineEq(BEG, END);
5 // constante da equacao de reta
6 uint8_t divi = 4;
7 //define a quantidades de iteracoes
8 float aux = 0;
9 //variavel auxiliar
10 float MatrixA[divi + 1];
11 // define array de valores do tensionador A
12 float MatrixB[divi + 1];
13 // define array de valores do tensionador B
14 float MatrixC[divi + 1];
15 // define array de valores do tensionador C
16 float MatrixD[divi + 1];
17 // define array de valores do tensionador D
18 float Answers[4][divi];
19 // Matriz resposta , guarda todas as iteracoes dos tensionadores

```

Fonte: Autoria própria (2022).

A rotina principal do *software* (Listagem 2) criado para o tensionador mecânico mestre, *sequenceOfIterations*, é responsável por gerar as distâncias de cabos a partir das iterações necessárias, além de transmitir esses passos via rádio para os demais tensionadores escravos. Essa rotina recebe dois objetos pontos, nos quais tem como parâmetros, coordenadas em x e y. A ordem que os objetos são passados como parâmetro na rotina, indica a sequência que o EE, irá se deslocar, ou seja, o primeiro ponto passado como parâmetro, coincide com o ponto inicial do EE, e o último, como ponto final.

Algumas precauções para evitar erros de movimentação foram previstas dentro do código-fonte. Foram definidas algumas variáveis locais, dentro da rotina *sequenceOfIterations*, implementadas somente para fazer alguns testes (Listagem 2). Visando serem ajustadas conforme a sensibilidade que os mesmos exigiam. Um exemplo desses elementos, é o tamanho da matriz resposta, a qual, mantém a quantidade de passos que cada motor precisa efetuar. Ajustando o valor de uma variável, denominada "divi", é possível diminuir ou aumentar a quantidade de iterações por visitas do sistema, ou seja, aumenta o número de iterações por deslocamento do EE, consequentemente aumentando o tempo de deslocamento, e suavizando o movimento de todos os cabos do sistema. Esse elemento do código, entre outros, caso o robô trabalhe em condições reais, não seria uma variável genérica para ser alterada conforme a necessidade do usuário.

Dentro da rotina principal, existem 4 matrizes responsáveis por armazenar as distâncias de cabos necessárias para mover o EE, quatro delas abrigam as distâncias de cabos entre os pontos de iteração, e o respectivo tensionador. As duas variáveis fundamentais para gerar os pontos

**Listagem 3 – Loop para ponto inicial mais próximo que o final da função sequenceOfIterations**

```

1   if (BEG.x < END.x)
2   {
3       aux = fmin(BEG.x, END.x);
4       for (int i = 0; aux <= fmax(BEG.x, END.x); i++)
5       {
6           Ponto Z(aux, (aux * m + n));
7           // cria um novo ponto para medir os cabos de cada iteracao
8           MatrixA[i] = distBetweenTwoPoints(Z, Motor1);
9           MatrixB[i] = distBetweenTwoPoints(Z, Motor2);
10          MatrixC[i] = distBetweenTwoPoints(Z, Motor3);
11          MatrixD[i] = distBetweenTwoPoints(Z, Motor4);
12          aux += (fmax(BEG.x, END.x) - fmin(BEG.x, END.x)) / divi;
13          } //Retornar o valor de Y para cada X a partir da eq da reta
14          } //Ponto inicial em x seja menor que o final

```

**Fonte: Autoria própria (2022).**

de iteração distribuídas entre os pontos de visitas, os quais, foram passados como parâmetros, são as duas variáveis  $m$  e  $n$ . Essas variáveis recebem os dois elementos da equação da reta gerada através da função a partir dos dois pontos de visita. O primeiro elemento é o coeficiente angular, armazenado na variável  $m$ , valor esse sendo gerado através da rotina *angularCoeficent*. O segundo é coeficiente linear da equação da reta, armazenado na variável  $n$ , valor gerado através de *constOfflineEq*. A variável auxiliar "aux", se ajusta quando o ponto estiver se deslocando em sentido da origem ou se afastando da mesma, ela é utilizada dentro dos dois *loops* referentes ao armazenamento das distâncias entre pontos de iteração até os tensionadores.

O valor da coordenada em  $x$  referente aos dois pontos de visitas definem como a sequência de cálculos das iterações serão feitas, a partir de dois *loop* executados de acordo com esse valor. Caso o ponto final esteja localizado mais próximo do atuador mestre em comparação com o ponto inicial, significa que o EE se deslocará em sentido a origem do sistema, dessa forma a variável auxiliar, na qual recebeu valor em  $x$  do ponto mais próximo da origem, estará crescendo o seu valor conforme as iterações com os respectivos tensionadores estarão sendo registradas e guardadas dentro da respectiva matriz encarregada de guardar as distâncias. O *loop* será finalizado quando essa variável auxiliar se tornar maior que o valor em  $x$  do ponto de visita mais distante, como pode ser observado na Listagem 3. Quando o ponto final estiver mais distante da origem em relação ao ponto inicial, as condições serão opostas ao caso no qual o ponto final estiver mais perto. Portanto, nesse caso, a variável auxiliar recebe o valor em  $x$  do ponto mais distante, e o *loop* só será finalizado quando essa variável se tornar menor que o valor em  $x$  do ponto de visita mais próximo, como pode ser observado na Listagem 4.

**Listagem 4 – Loop para ponto inicial mais distante que o final da função *sequenceOfIterations***

```

1   if (BEG.x > END.x)
2   {
3       aux = fmax(BEG.x, END.x);
4       for (int i = 0; aux >= fmin(BEG.x, END.x); i++)
5       {
6           Ponto Z(aux, (aux * m + n));
7           // cria um novo ponto para medir os cabos de cada iteracao
8           MatrixA[i] = distBetweenTwoPoints(Z, Motor1);
9           MatrixB[i] = distBetweenTwoPoints(Z, Motor2);
10          MatrixC[i] = distBetweenTwoPoints(Z, Motor3);
11          MatrixD[i] = distBetweenTwoPoints(Z, Motor4);
12          aux -= (fmax(BEG.x, END.x) - fmin(BEG.x, END.x)) / divi;
13      }
14  } //Ponto inicial em x seja maior que o final

```

**Fonte: Aatoria própria (2022).**

No instante em que todas as iterações foram finalizadas, inicia-se o último estágio da rotina *sequenceOfIterations*, no qual a matriz *Answers* irá armazenar a diferença das distâncias de cabos entre as posições das matrizes cujas as distâncias entre os pontos de iterações foram guardadas. Para ajudar na sincronização entre cada motor, no mesmo *loop*, é enviado para os outros 3 tensionadores escravos, a quantidade de cabos que os mesmos precisam trabalhar, e em seguida o motor mestre trabalha a quantidade de cabos calculados para ele, Listagem 5. Observa-se que somente é enviado as distâncias de cabos da próxima iteração, quando o motor mestre terminar de se mover.

Um exemplo de como ajustando a variável "divi", melhora a resolução do movimento dos cabos e conseqüentemente melhora o movimento do EE. Foi determinado 4 pontos de visitas distribuídas pelo plano de movimento do EE, marcados como visita 1 a visita 4 nas Figura 20 e Figura 21. Em seguida o sistema gerou as distâncias a serem trabalhadas em centímetros do atuador mecânico mestre, para 4 valores de iterações por visita do EE, sendo eles: sem iterações, 4 iterações, 8 iterações e 16 iterações. O resultado da quantidade de cabos a serem trabalhadas ao todo, conforme a variável "divi" é dobrada, pode ser observada no eixo y da Figura 20 e Figura 21, de tal forma que conforme o EE se aproximava do tensionador, a quantidade de cabos se torna negativos, e conforme se afasta se torna positiva. Esses quatro gráficos evidenciam a forma linear que os tensionadores precisam trabalhar para que o EE se desloque com o mínimo de perda, e para atingir esse comportamento mais linear do EE, é necessário, que seja definido mais iterações por pontos de visitas.

**Listagem 5 – Loop referênte ao cálculo das variações de distâncias e movimento do motor 4 da função `sequenceOfIterations`**

```

1 // variacoes de dist ncias e movimento do motor 4 da
2 for (int i = 0, x = 1; i < divi; i++, x++)
3 {
4     Answers[0][i] = (MatrixA[x] - MatrixA[i]);
5     // Iterações do motor 1
6     Answers[1][i] = (MatrixB[x] - MatrixB[i]);
7     // Iterações do motor 2
8     Answers[2][i] = (MatrixC[x] - MatrixC[i]);
9     // Iterações do motor 3
10    Answers[3][i] = (MatrixD[x] - MatrixD[i]);
11    // Iterações do motor 4
12
13    //Enviando dados da Iterações do motor 1
14    radio.openWritingPipe(address1);
15    radio.write(&Answers[0][i], sizeof(Answers[0][i]));
16    //Enviando dados da Iterações do motor 2
17    radio.openWritingPipe(address2);
18    radio.write(&Answers[1][i], sizeof(Answers[1][i]));
19    //Enviando dados da Iterações do motor 3
20    radio.openWritingPipe(address3);
21    radio.write(&Answers[2][i], sizeof(Answers[2][i]));
22    //Motor 4, os dados são tratados aqui
23
24    if (Answers[3][i] < 0 && Answers[3][i] != 0)
25    {
26        clockwise(Answers[3][i]);
27        Answers[3][i] = 0;
28    }
29    if (Answers[3][i] > 0 && Answers[3][i] != 0)
30    {
31        counterClockwise(Answers[3][i]);
32        Answers[3][i] = 0;
33    }
34    Serial.println(Answers[3][i]);
35    delay(5000);
36 }

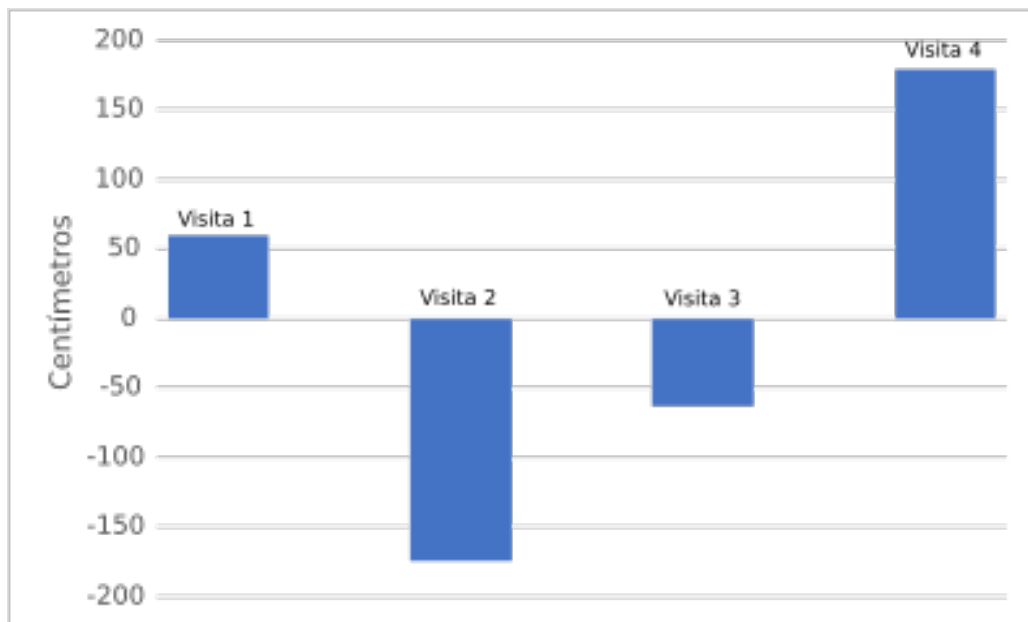
```

**Fonte: Autoria própria (2022).**

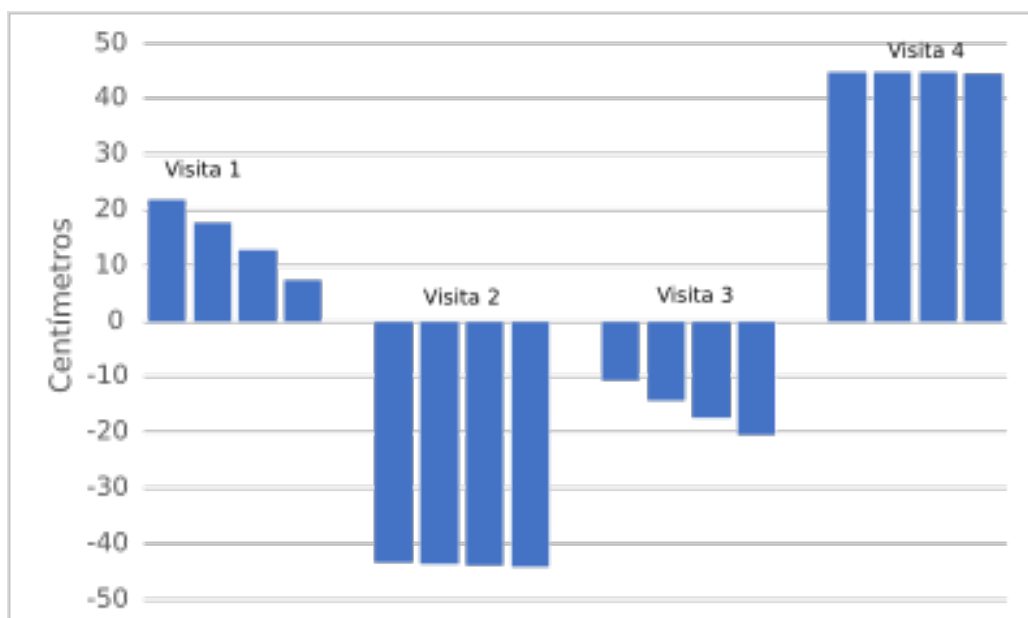


**Figura 20 – Distâncias de cabos para deslocar o EE entre quatro pontos (a) Sem iterações, (b) 4 iterações**

**(a) Sem iterações**



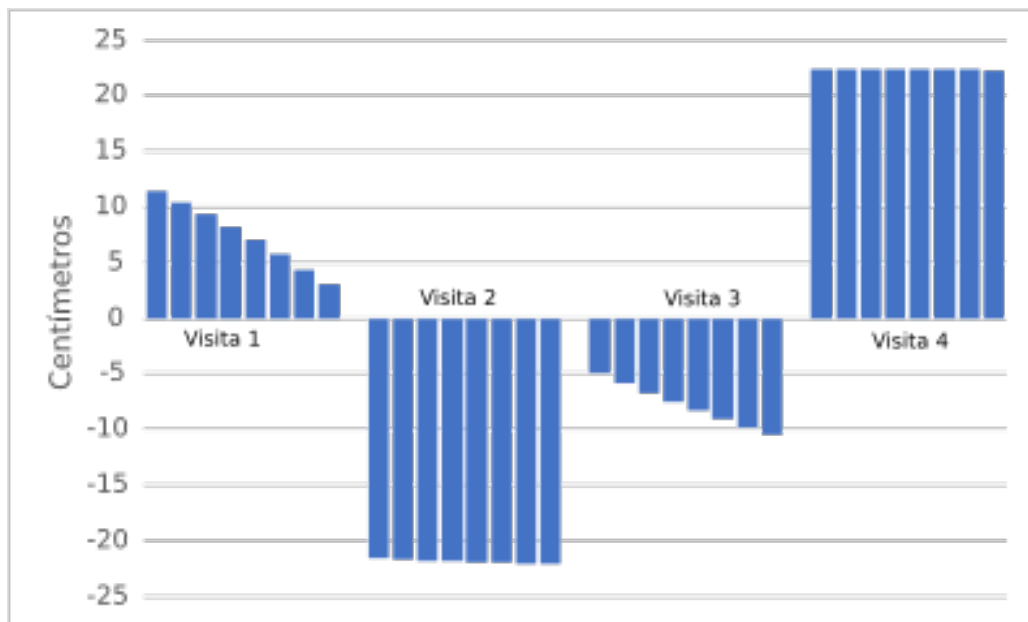
**(b) 4 iterações**



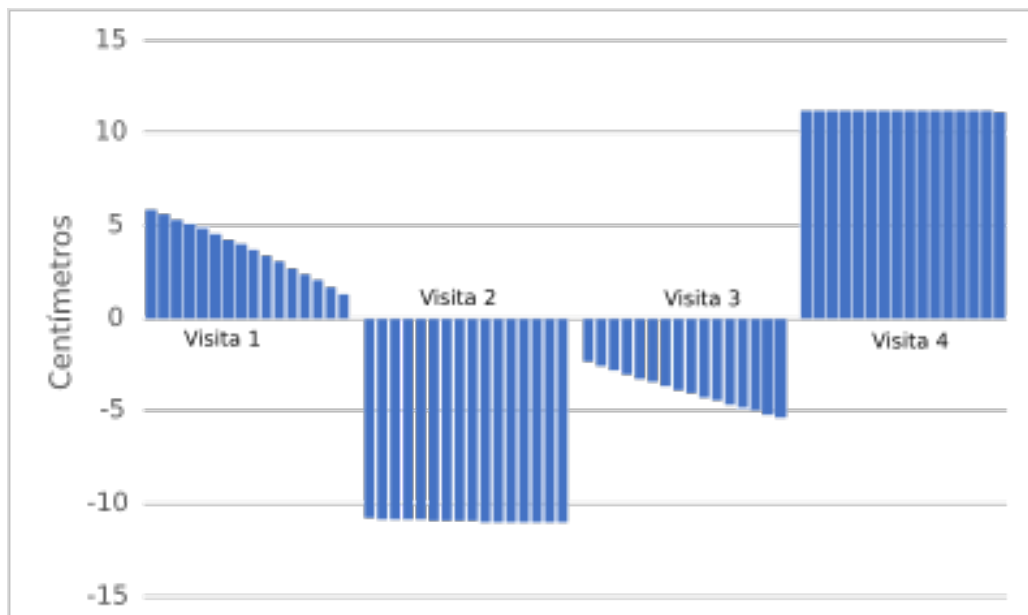
**Fonte: Autoria própria (2022).**

**Figura 21 – Distâncias de cabos para deslocar o EE entre quatro pontos: (a) 8 iterações, (b) 16 iterações**

**(a) 8 iterações**



**(b) 16 iterações**



**Fonte: Autoria própria (2022).**

#### 4.4 Resultados na prática

Após todos os tensionadores estarem conectados uns aos outros, através do EE, foi medido o ponto central entre os quatro tensionadores, e em seguida posicionado o EE acima do mesmo. Os quatro cabos foram todos sendo ajustados para que ficassem totalmente tracionados e, sem nenhuma folga no cabo entre o EE e os tensionadores. Com isso preparado, os três primeiros pontos de visita do EE, ficaram limitados a somente um reta na diagonal entre o EE e o tensionador escravo 3, os pontos estavam localizados somente no plano e a suas coordenadas em x,y eram:  $P_0(350,250)$ ,  $P_1(450,350)$ ,  $P_2(550,450)$ .

Foram feitas 3 viagens consecutivas a esses três pontos. Foi possível notar, que, conforme o EE se aproximava de um tensionador e se afasta dos outros posicionados nas extremidades mais distâtes, o valor de perdas de distância de cabos diminuía consideravelmente e, quando o EE se aproximava do centro do sistema, maior era essa perda de cabos. Isso acontece, devido à tração causada pelo peso do EE estar melhor distribuídas entre os 4 cabos quando ele estiver localizado no centro. Dessa forma, quando o mesmo se aproximar gradativamente do ponto central do sistema, dois tensionadores estarão trabalhando soltando e outros dois tracionando praticamente a mesma distâncias de cabos delimitada pela iteração, porém a tração do EE é menor no centro do sistema, então acontece uma folga no cabo entre o EE e os tensionadores mais distantes. Assim, por alguns instantes, não existe tração do cabo necessária para girar o rolete do *encoder*, fazendo com que o cabo deslize, gerando alguns centímetros de erro.

Esses primeiros resultados, pautaram a segunda etapa de testes. Estabeleceu-se o ponto de partida do EE, não mais no centro do CDPR, mas um pouco mais próximo de um dos tensionadores escravos. Optou-se por limitar o movimento do EE a pontos de visitas, localizados, em lugares que não seria necessário que o mesmo se movimente através do centro do sistema. O intuito desse teste, era provar que, evitando posições que o CDPR não trabalhe de forma ideal, é possível fazer com que o sistema consiga caminhar, além de seis pontos sem ter perdas consideráveis de precisão de posição. A partir disso, estabeleceu 6 novas coordenadas de visitas, e configurou-se o movimento do sistema com uma resolução de 4 iterações por ponto de visita. O resultado dos pontos previstos com os pontos medidos em centímetros após o movimento do EE ser finalizado, podem ser observados na Tabela 5. Os pontos medidos com seus respectivos pontos previstos foram plotados na Figura 22.

É possível observar com os resultados das distâncias medias com as previstas, que o sistema acaba subestimando a altura em sua posição final. Então fez-se necessário quantizar esse erro de posição, através do valor das distâncias euclidianas (Equação 5) entre os pontos referenciados  $P_{ref}(x_{ref},y_{ref},z_{ref})$ , com os medidos  $P_{med}(x_{med},y_{med},z_{med})$ .

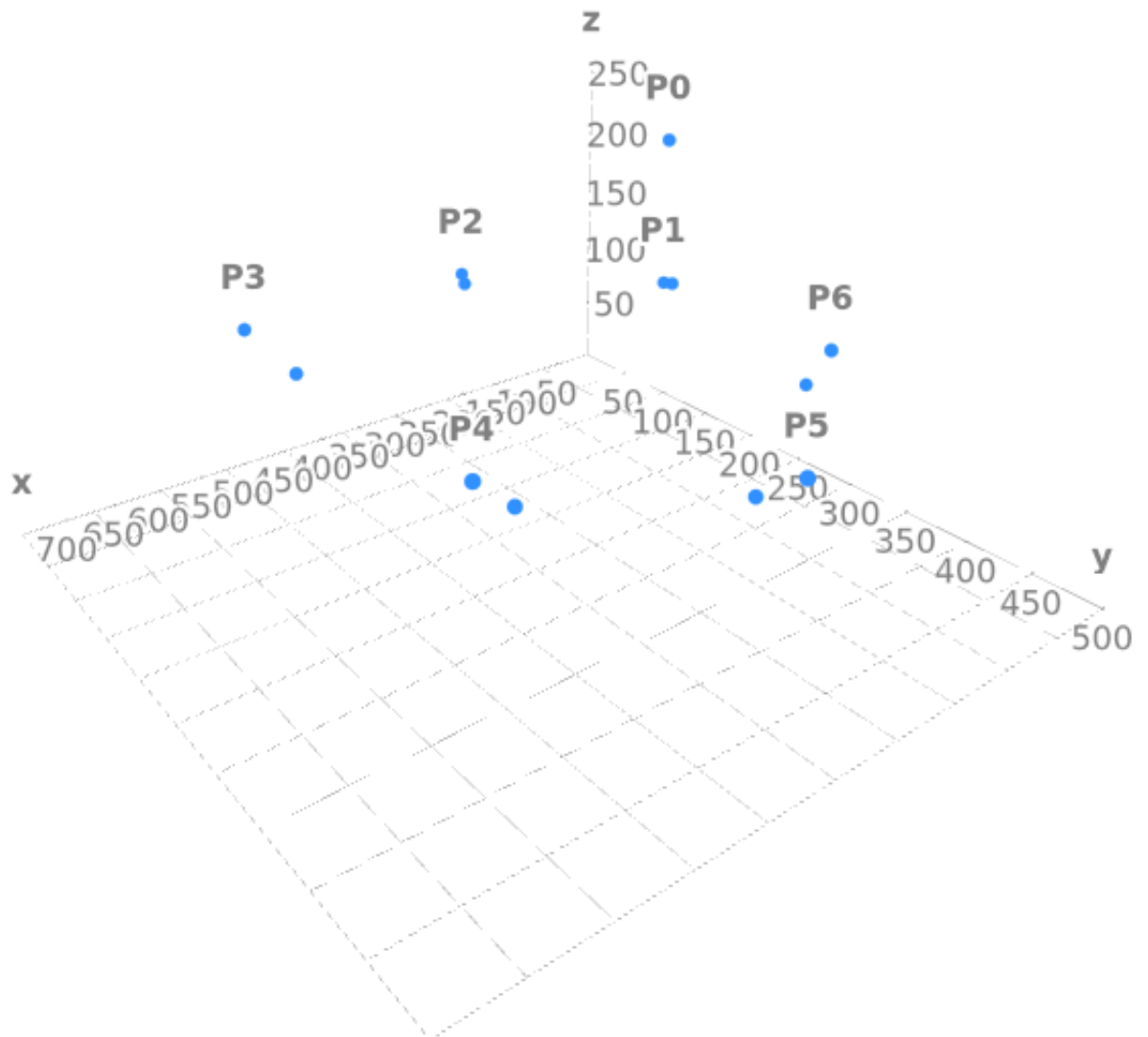
$$D_{erro} = \sqrt{(x_{med} - x_{ref})^2 + (y_{med} - y_{ref})^2 + (z_{med} - z_{ref})^2}, \quad (5)$$

**Tabela 5 – 6 pontos de visitas com 4 iterações de resolução entre pontos de visita**

Ponto de visita	Ponto de referência ( $P_{ref}$ )	Ponto Medido ( $P_{med}$ )	Erro( $D_{erro}$ )
P1	(151,200,150)	(145,205,150)	7,810
P2	(300,100,150)	(310,113,148)	16,523
P3	(550,100,150)	(530,140,130)	45,858
P4	(550,350,150)	(500,340,119)	59,674
P5	(350,450,150)	(321,400,110)	70,292
P6	(150,350,150)	(110,311,105)	71,736

Fonte: Autoria própria..

Figura 22 – Pontos de visitas (rotulados) e pontos medidos (não rotulados) com 4 iterações de resolução entre pontos de visita



Fonte: Autoria própria (2022).

Quando o EE terminou de se movimentar, a coordenada final do mesmo ficou a 71,736 centímetros do que foi estabelecido. Como o objetivo desse trabalho é de alcançar 6 pontos de visitas com um erro de até 20 centímetros, uma terceira etapa de testes foi feita. Dessa vez para os mesmos pontos de visitas, foi configurado uma resolução de 16 iterações entre deslocamentos. O resultado dos pontos previstos com os pontos medidos, após o movimento do EE ser finalizado, podem ser observados na Tabela 6. Os pontos medidos com seus respectivos pontos previstos foram plotados na Figura 23.

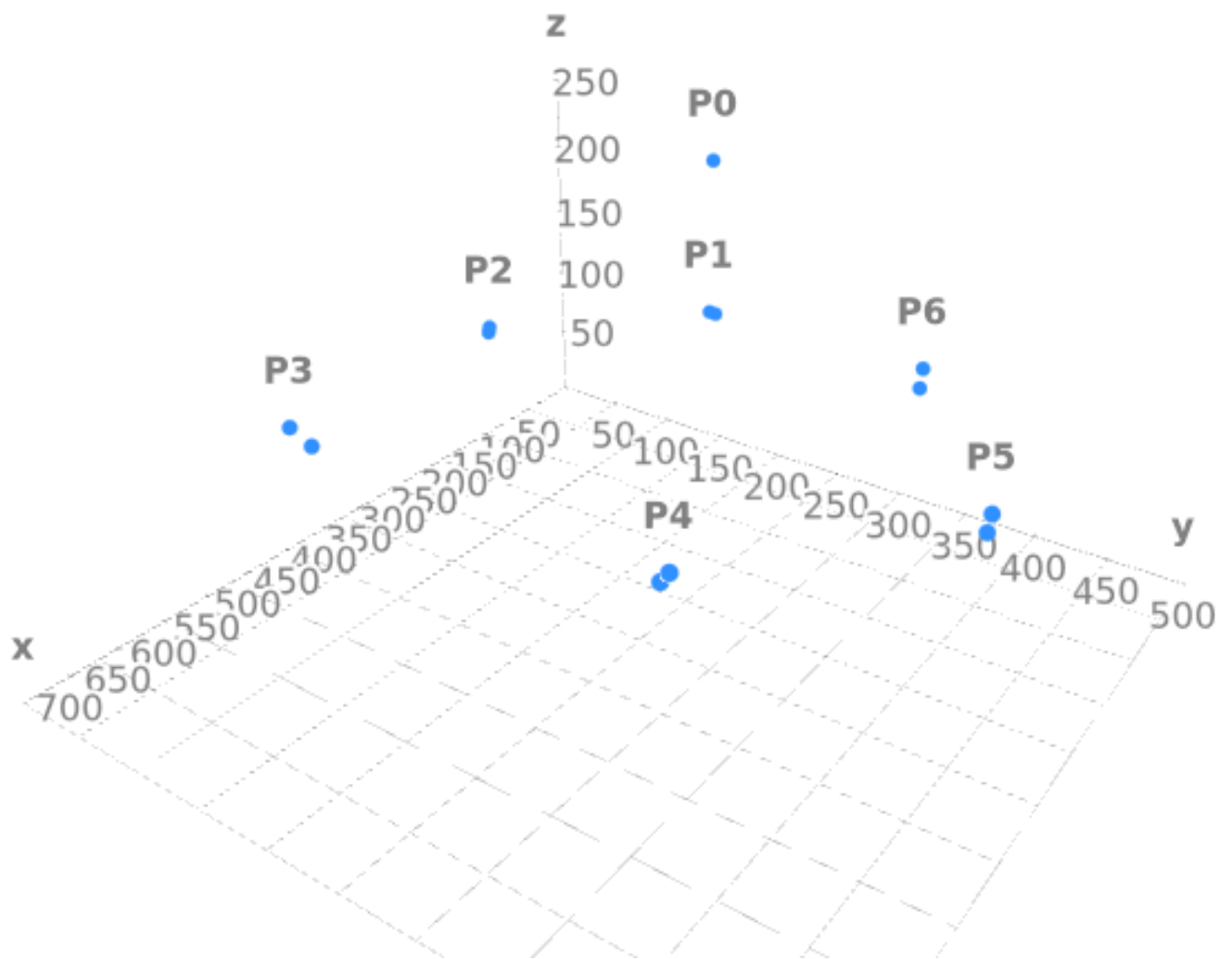
Os pontos medidos ficaram muito próximos dos pontos esperados, provando que o modelo matemático de relacionar as distâncias de cabos no espaço com as distâncias a serem movimentadas pelos tensionadores, condiz com o resultado do local que o EE se encontra no espaço. Porém, apesar do sistema apresentar um erro relativamente baixo após a visita de todos os seis pontos, não foi previsto um sistema de auto-calibração do robô, dessa forma, se o CDPR continua-se movimentando o EE para além dos 6 pontos estipulados, o erro seria carregado para todos os próximos pontos de visita subsequentes. É possível concluir que esses 6 pontos só são atingíveis uma vez, com essa taxa de erro baixa. Caso o EE precise atingir outros pontos distribuídos no espaço, precisa-se recalibrar as coordenadas iniciais, medindo o ponto final do EE e estabelecendo-o como um novo ponto inicial com suas coordenadas medidas manualmente e redefinidas no *software*.

**Tabela 6 – 6 pontos de visitas com 16 iterações de resolução entre pontos de visita**

Ponto de visita	Ponto de referência ( $P_{ref}$ )	Ponto Medido ( $P_{med}$ )	Erro( $D_{erro}$ )
P1	(151,200,150)	(151,205,150)	2,236
P2	(300,100,150)	(305,102,148)	5,745
P3	(550,100,150)	(555,120,146)	12,570
P4	(550,350,150)	(550,345,143)	10,295
P5	(350,450,150)	(355,450,141)	10,230
P6	(150,350,150)	(156,351,139)	12,570

**Fonte: Autoria própria..**

**Figura 23 – Pontos de visitas (rotulados) e pontos medidos (não rotulados) com 16 iterações de resolução entre pontos de visita**



Fonte: Autoria própria (2022).

## 5 CONCLUSÃO

É possível concluir a partir dos testes iniciais do CDPR, que apesar do método de deslocamento por pontos de iterações reduzir muito o erro relativo do EE em sua posição final, apesar de pequeno, ele se mantém e, se caso fosse determinado pontos além daqueles estabelecidos e testados, esse erro ira ser carregado e conseqüentemente ira fazer com que o EE fique muito fora do que pode ser considerado aceitável. Portanto, se faz necessário criar um método de calibração do robô, após esses 6 pontos de visitas atingidos, para conseguir manter uma qualidade de precisão em seus pontos de deslocamentos seguintes.

Durante o processo de desenvolvimento do CDPR vários métodos de calibração foram avaliados, e dois deles foram considerados; o método de calibração via GPS, no qual consegue manter uma precisão nas coordenadas em x e y, porém, não consegue entregar a posição referente a altura do EE, além disso, seria necessário implementar novos componentes que consigam entregar uma boa precisão nessas coordenadas, conseqüentemente aumentando os custos do robô. Outro método estudado, e mais promissor, seria ter 4 secções transversais posicionados em locais conhecidos dos cabos, nos quais seriam compostos por materiais ferromagnéticos, que interagiriam com um sensor indutivo posicionado no PCP de cada tensionador. Dessa forma, cria-se uma espécie de *checkpoint* nos cabos, que quando fossem ativados, recalibrariam o robô devido essas distâncias de cabos já serem conhecidas, e que por representarem as distâncias do EE e o PCP, definem a posição da sonda no espaço.

O desenvolvimento deste trabalho resultou em um dispositivo dotado de equipamentos de baixo custo, trabalhando para gerar um movimento em um EE, de forma, a fornecer três graus de liberdades no mesmo. O dispositivo disponibiliza que o EE, para a piscicultura, seja a partir de uma sonda, capaz de medir o nível de OD na água dos criadouros. A implementação do circuito eletrônico, sistema embarcado, o *firmware* e a estrutura mecânica atenderam as exigências para que o projeto seja facilmente escalado para locais de trabalhos maiores. O movimento do EE em local controlado, foi satisfatório, porém caso seja aplicado em campo, com escalas na ordem de grandeza muitas vezes maiores com o que foi testado em ambiente reduzido, é possível que os erros medidos em laboratório seja proporcionalmente escalados com suas novas dimensões de trabalho. Portanto, ficou explicito que para viabilizar o robô nessa aplicação, o sistema precisa implementar algum dos métodos de calibração citados anteriormente conforme os pontos de visita são alcançados pelo EE, somente assim o robô consegue ultrapassar a barreira dos 6 pontos de medição com confiança em sua posição final.

## REFERÊNCIAS

- BRUCKMANN, A. P. T. **Cable-Driven Parallel Robots**. [S.l.]: Springer, 2013.
- MARLET, J. P. **Parallel Robots. Solid Mechanics and Its Applications**. [S.l.]: Springer, 2006.
- MERSI SINA VALI, M. S. h. G. A.-M. T. M. R. Design and Control of a Suspended Cable-Driven Parallel Robot with Four Cables. **International Conference on Robotics and Mechatronics**, out. 2018.
- NEWMAN., M. Design and experimentation of cable-driven platform stabilization and control systems. ago. 2017.
- PIAO XUEJUN JIN, J. J. E. C. J.-O. P. C.-S. K. J. Development of a high payload cable-driven parallel robot. *In: 17th International Conference on Control, Automation and Systems (ICCAS 2017)*. [S.l.: s.n.], 2017.
- QUEIROZ, R. C. B. Julio Ferraz de. Boas práticas de manejo para manter concentrações adequadas de oxigênio dissolvido em viveiros de piscicultura. **Embrapa**, v. 54, p. 1–9, dez. 2016.
- WANG, S. C. B. Exit Point, Initial Length and Pose Self-calibration Method for Cable-Driven Parallel Robots. **HAL science ouverte**, out. 2021.