

**MARINHA DO BRASIL  
DIRETORIA DE ENSINO DA MARINHA  
CENTRO DE INSTRUÇÃO ALMIRANTE ALEXANDRINO**

**CURSO DE APERFEIÇOAMENTO AVANÇADO EM  
SISTEMA DE CONTROLE E ELETRICIDADE DE NAVIOS**

**TRABALHO DE CONCLUSÃO DE CURSO**

**EVOLUÇÃO DOS SISTEMAS DE CONTROLE E MONITORAMENTO EM  
SISTEMAS NAVAIS: Soluções para veículo autônomo de superfície.**



**PRIMEIRO-TENENTE JOÃO FERNANDO RODRIGUES COSTA**

Rio de Janeiro  
2023

1º TENENTE JOÃO FERNANDO RODRIGUES COSTA

EVOLUÇÃO DOS SISTEMAS DE CONTROLE E MONITORAMENTO EM  
SISTEMAS NAVAIS: soluções para veículo autônomo de superfície.

Monografia apresentada ao Centro de Instrução  
Almirante Alexandrino como requisito parcial à  
conclusão do Curso de Aperfeiçoamento Avançado  
em Sistema de Controle e Eletricidade de Navios

Orientadores: Prof. Alessandro Jacoud Peixoto  
Capitão-Tenente Luiz Renato Tomelin

CIAA  
Rio de Janeiro  
2023

PRIMEIRO-TENENTE JOÃO FERNANDO RODRIGUES COSTA

Título: EVOLUÇÃO DOS SISTEMAS DE CONTROLE E MONITORAMENTO EM  
SISTEMAS NAVAIS: soluções para veículo autônomo de superfície.

Monografia apresentada ao Centro de Instrução Almirante Alexandrino como requisito parcial à conclusão do Curso de Aperfeiçoamento Avançado em Sistema de Controle e Eletricidade de Navios.

Aprovada em \_\_\_\_\_

Banca Examinadora:

Alessandro Jacoud Peixoto, DSc – UFRJ

\_\_\_\_\_

Luiz Renato Tomelin – Capitão-Tenente MB

\_\_\_\_\_

Lorran Santos Basílio – Primeiro-Tenente MB

\_\_\_\_\_

CIAA  
Rio de Janeiro  
2023

## **AGRADECIMENTOS**

Agradeço a minha esposa, cujo apoio incondicional durante todo esse ano, foi de imensurável importância para que eu conseguisse lograr bons resultados neste curso.

Aos professores da COPPE/UFRJ, que mesmo com o curto período disponibilizado, se dedicaram com afinco, para transmitir seus conhecimentos e expertises nas áreas de atuação.

A meus sinceros amigos, irmãos de armas, com quem pude dividir o espaço do aprendizado mais uma vez.

“Artificial Intelligence, deep learning, machine learning—whatever you are doing if you don’t understand it—learn it. Because otherwise you are going to be a dinosaur within 3 years.” Mark Cuban.

Título: Evolução dos Sistemas de Controle e Monitoramento em Sistemas Navais: Soluções para veículo autônomo de superfície.

## Resumo

Com a crescente expansão do uso do termo de Inteligência Artificial nos meios de comunicação, é importante educar corretamente a população sobre os prós, contras e desafios da chamada “nova revolução industrial”. Tangente à evolução dos sistemas de controles, este trabalho tem o intuito de fazer uma breve revisão sobre as soluções modernas encontradas para sistemas dinâmicos complexos e seus potenciais, investigando suas aplicações em um projeto embrionário de um veículo de superfície autônomo. O estudo apresenta as equações hidrodinâmicas de uma embarcação e direciona o estudo para resolver o problema particularmente difícil de cumprimento de um trajeto e evitar colisões.

**Palavras- chave:** Veículo de superfície autônomo, sistemas de controle, machine learning, redes neurais.

## LISTA DE FIGURAS

Figura 1: Função de Transferência malha fechada. ....	5
Figura 2: Sistema de malha fechada em loop. ....	7
Figura 3: Regulador LQR. ....	9
Figura 4: (a) Sinal analógico (b) Sinal discretizado com período $T=125\mu s$ . ....	11
Figura 5: Sinal analógico representado no Python pela função FFT. ....	12
Figura 6: Problema de Aliasing. ....	13
Figura 7: Neurônio.....	17
Figura 8: ANN com duas camadas.....	17
Figura 9: Diversos modelos de Redes Neurais.....	20
Figura 10: Rio brasileiro com batimetria realizada pelo CHN-9. Na imagem cada valor numérico representa a profundidade encontrada no local, dando uma ideia aos navegantes do caminho mais seguro a ser percorrido. ....	21
Figura 11:Subsistemas de uma embarcação inteligente. ....	22
Figura 13:Classificação de dados utilizando SVM. Determinar o vetor $w$ e a constante $b$ são o objetivo da otimização de um SVM. ....	24
Figura 14: Modelos de sensores óticos comercializados. ....	25
Figura 15: SDs e margens. ....	25
Figura 16: Nova trajetória gerada de maneira autônoma para evitar abalroamento.....	27
Figura 17: Estrutura básica de um MPC.....	28
Figura 18: Representação de um sistema de planejamento de derrota.....	29
Figura 19: Simulação de uma manobra de acordo com a COLREG perante um alvo inesperado.....	30
Figura 20: Navio e sistema de coordenadas. ....	32
Figura 21: Princípio da Navegação LOS. ....	34

## Sumário

<b>1. INTRODUÇÃO</b> .....	1
1.1. <b>Panorama Global</b> .....	1
1.2. <b>Motivação</b> .....	1
1.3. <b>Objetivos</b> .....	2
1.4. <b>Histórico</b> .....	2
<b>2. METODOLOGIA</b> .....	4
<b>3. REFERENCIAL TEÓRICO</b> .....	5
3.1. <b>Sistema de controle linear</b> .....	5
3.2. <b>Matriz Controlabilidade e Observabilidade</b> .....	7
3.3. <b>Linear Quadratic Regulator (LQR)</b> .....	8
3.4. <b>Filtro de Kalman</b> .....	10
3.5. <b>Sistemas Discretos</b> .....	10
3.6. <b>Filtros</b> .....	12
3.7. <b>Problema de <i>Aliasing</i></b> .....	13
<b>4. DATA DRIVEN CONTROL SYSTEMS</b> .....	15
4.1. <b>Conceitos</b> .....	15
4.2. <b>Machine Learning</b> .....	16
<b>5. ESTUDO DE CASO: IDEIAS PARA CONCEPÇÃO DE UM USV PARA A MB. (UNMANNED SURFACE VEHICLE)</b> .....	21
5.1. <b>Apresentação do caso</b> .....	21
5.1.1. <b>Subsistema de percepção</b> .....	23
5.1.2. <b>O subsistema de geração de derrota</b> .....	26
5.1.3. <b>O subsistema de execução</b> .....	30
5.1.4. <b>Movimentação da embarcação</b> .....	31
5.1.5. <b>Propulsão</b> .....	32
5.1.6. <b>Posicionamento</b> .....	33
<b>6. CONCLUSÃO</b> .....	36
6.1. <b>Trabalhos Futuros</b> .....	36



<b>REFERÊNCIAS .....</b>	<b>37</b>
--------------------------	-----------

# 1. INTRODUÇÃO

## 1.1. Panorama Global

O Departamento de Defesa Estratégica dos Estado Unidos da América tem um foco intenso em tecnologia e aponta que os EUA não conseguirão alcançar a segurança e prosperidade sem o aproveitamento destas tecnologias pelos soldados (Maritime Security Challenges, 2021).

O ambiente de segurança institucional também é afetado pelos rápidos avanços tecnológicos e mudanças no caráter da guerra. O Impulso para desenvolver novas tecnologias é incansável, expandindo para mais atores globais com barreiras de entrada cada vez menores. Novas tecnologias incluem computação avançada, inteligência artificial, veículos autônomos, robótica, energia direcionada, hipersônicos e biotecnologia — as mesmas tecnologias que garantem capacidade de lutar e vencer as guerras do futuro (Maritime Security Challenges, 2021).

Um *offset* estratégico é uma abordagem à competição militar que busca compensar assimetricamente uma posição desvantajosa. Ao invés de combater frente a frente em uma área que o adversário possa ter força suficiente, um *offset* estratégico muda o balanço de poder através da introdução de novos conceitos operacionais e tecnologias (Maritime Security Challenges, 2021).

Como exemplo, o primeiro *offset* estratégico aconteceu nos anos 50, quando o então Presidente Eisenhower buscar superar a vantagem numérica do novo pacto de Varsóvia, introduzindo potencial nuclear ao campo de batalha.

O segundo *offset* seria então, criar um sistema ISR (*intelligence, surveillance, reconnaissance*) em tempo real e de longa distância, capaz de apoiar ataques de alta precisão a tempo real.

Ao explicar os elementos tecnológicos do terceiro *offset*, o então Vice-Secretário de Defesa, Robert Work, enfatizou a importância das capacidades emergentes em sistemas não tripulados, inteligência artificial, aprendizagem automática e autonomia (Maritime Security Challenges, 2021).

## 1.2 Motivação

Enquanto possam existir vários pilares para garantir esta estratégia, um deles é, sem dúvidas, altamente dependente de dados e inteligência artificial.

Indubitavelmente, nota-se o crescente interesse das potências globais em aperfeiçoar e implementar o uso da Inteligência Artificial em suas cadeiras produtoras e executivas. A Marinha do Brasil não deve se distanciar das tendências globais, no que tange ao emprego e preparo da força naval. Com, aproximadamente, 4 anos servindo na região Amazônica, observei a árdua (e praticamente impossível) missão dos oficiais e praças da hidrografia de manter os anais hidrográficos atualizados.

São milhares de quilômetros que precisam ser cartografados, para garantir a segurança da navegação e a salvaguarda da vida humana nos rios. Alinhado com a falta de recursos humanos disponível para realizar a atividade de batimetria, é interessante que a Marinha invista em veículos totalmente autônomos de superfície que consigam realizar a atividade com a devida eficiência.

### **1.3. Objetivos**

Este trabalho tem por objetivo apresentar formulações básicas de engenharia de controle, elementos conceituais importantes, e suas evoluções até os dias atuais, bem como mostrar um exemplo da utilização desta tecnologia na concepção de um projeto de um veículo de superfície autônomo (USV), baseado em trabalhos e livros disponíveis.

### **1.4. Histórico**

Em uma embarcação, o Sistema de Posicionamento Dinâmico (S.D.P.) é um sistema que controla automaticamente a posição e aproamento de uma embarcação por meio de propulsão ativa (Wikipedia, 2020). Em linhas gerais, corresponde a um complexo sistema de controle de posição dinâmica, composto por várias variáveis capazes de tornar seu posicionamento mais preciso (GPS, DGPS, Anemômetros, Giroscópios etc.) Seus atuadores para mantê-los na posição são propulsores, thrustes azimutais ou fixos, com ou sem controles de velocidade e leme e um computador central é responsável pela execução das correções de posição e pela interface com o operador.

Controlar um Processo Industrial depende de seu Sistema de Controle. Os instrumentos estão ligados a um sistema de controle, o qual analisa a medição enviada pelo instrumento. A resposta programada no sistema de controle vai atuar nos dispositivos de controle inseridos no processo. Este ciclo de atualização dos valores das variáveis manipuladas, medida dos valores das variáveis controladas para se gerar a resposta adequada é a forma mais simples de descrever os conceitos associados ao controle de processos. Os dispositivos de controle utilizados são normalmente considerados como parte integrante da instrumentação, e podem ir desde os mais simples CLPs até aos já mais avançados computadores. As entradas nestes dispositivos podem variar desde um pequeno número de variáveis medidas, até à ordem dos milhares. O controlador recebe o sinal padrão do transmissor, executa cálculos com o sinal, set-point (valor desejado da variável medida), parâmetros como o clássico PID (Proporcional, Integral e Derivativo), ou até algoritmos complexos e inteligentes envolvendo redes neurais. Controladores, podem ser pneumáticos, elétricos/eletrônicos ou microprocessados utilizando redes de comunicação.

Durante a década de 1940, método de resposta em frequência (especialmente os métodos com base nos diagramas de Bode) tornaram possível o surgimento de sistemas de controle linear de malha fechada que satisfizessem o desempenho requerido. Muitos desses sistemas usavam controladores PID (Proporcional Integral Derivativo) no controle de pressão e temperatura.

À medida que os sistemas modernos se tornam mais e mais complexos, a sua descrição no campo da matemática, se tornava mais complexa, pois aumentava o uso de equações para representarem esses sistemas. (Medeiros, 2017)

A partir de 1960, com a disponibilidade dos computadores digitais, possibilitou-se a análise de sistemas complexos diretamente no domínio do tempo, o que permitiu lidar com a crescente dificuldade dos sistemas modernos. Entre 1980 e 1990, surgem desenvolvimentos na teoria do controle moderno, para lidar com incertezas na representação do modelo da planta, o controle robusto. (Medeiros, 2017)

Com a crescente complexidade dos sistemas de controle industriais e comerciais, surgem então novos estudos e metodologias para o controle eficiente de qualquer processo dinâmico: Uso da Inteligência Artificial (IA). Assim várias técnicas na área de IA são utilizadas para se obter diversos tipos de sistemas “inteligentes”, para a síntese de sistemas de controle ótimo do tipo LQR. (Medeiros, 2017)

## **2. METODOLOGIA**

Quanto aos objetivos, este trabalho está associado a uma pesquisa exploratória, com levantamento bibliográfico necessário para familiarizar o problema com a pesquisa realizada, visando respondê-los de forma hipotética.

Quanto à abordagem, trata-se de uma abordagem quali-quantitativa, buscando entender os fenômenos a partir de sua explicação e motivos, bem como utilizá-los em uma análise mais subjetiva.

Quanto à natureza, trata-se de uma pesquisa básica, com a intenção de condensar conhecimentos e autores necessários para a posterior exploração do assunto.

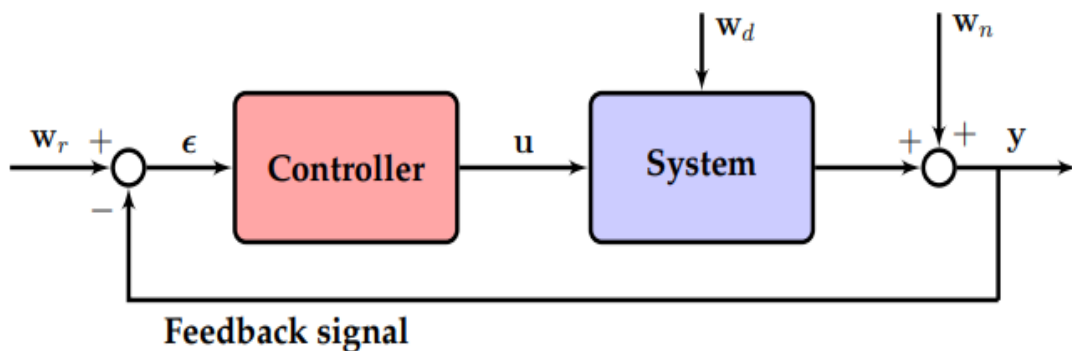
Quanto aos procedimentos, trata-se de uma pesquisa bibliográfica pautando-se por aqueles livros e artigos que precedem de modelos e experimentos; e estudo de caso, com a intenção de explorar hipóteses para a solução do problema de um veículo de superfície autônomo.

### 3. REFERENCIAL TEÓRICO

#### 3.1. Sistema de controle linear

Sistemas de controle de malha fechada usam sensores que medem o sistema diretamente e depois modelam a saída em resposta ao objetivo desejado. (Brunton; Kutz, 2019).

Figura 1: Função de Transferência malha fechada.



Fonte: Brunton; Kutz, 2019.

A Figura 1 representa um sistema em malha fechada, o sensor  $y$  realimenta a entrada e é subtraído do sinal de referência  $w_r$ , resultando em um erro  $\epsilon$ , no qual o controlador deve realizar a atuação correta  $u$  para gerar a resposta desejada. Sistemas de com retroalimentação são usados para estabilizar dinâmicas comumente instáveis enquanto rejeitam os distúrbios externos e ruídos gerados no sensor. (Brunton; Kutz, 2019)

Matematicamente essa função é dada por:

$$\dot{x} = f(x, u, w_d, t), \quad (1)$$

$$y = g(x, u, w_n, t). \quad (2)$$

O controle moderno tem por usa vez, a tendência de aumentar a complexidade, em virtude de realizar tarefas complexas e de alta precisão. (Ogata, 2010). Sistemas complexos possuem múltiplas saídas e múltiplas entradas (MIMO), podendo ser lineares ou não lineares, variantes ou não variantes no tempo. Se as equações 1 e 2 forem linearizadas em torno de um ponto de operação no tempo.

$$\dot{x}(t) = A(t)x(t) + B(t)u(t), \quad (3)$$

$$y = C(t)x(t) + Du(t). \quad (4)$$

Onde A será a matriz de estado, B é a matriz de entrada, C é a matriz de saída e D é a matriz de transmissão direta. Ainda segundo Ogata, se o sistema operar em torno de um ponto de equilíbrio e os sinais envolvidos forem pequenos, então é possível aproximar o sistema não linear por um sistema linear. Esse sistema linear é equivalente ao sistema não linear considerado dentro de um conjunto limitado de operações. Esse modelo linearizado (modelo linear, invariante no tempo) é muito importante na engenharia de controle.

O processo de linearização tem como base o desenvolvimento da função não linear em uma série de Taylor em torno do ponto de operação e a retenção somente do termo linear. Esses termos desprezados obrigatoriamente devem ser suficientemente pequenos; isto é, as variáveis devem se desviar apenas ligeiramente das condições de operação.

$$y = f(x),$$

$$y = f(\bar{x}) + \frac{df}{dx}(x - \bar{x}) + \frac{1}{(2!)} \frac{d^2f}{dx^2}(x - \bar{x})^2 + \dots \quad (5)$$

A primeira análise será feita em questão da ausência de controle, ou seja, quando  $u = 0$ .

O sistema linearizado, sendo  $y = x$  teremos que:

$$\dot{x} = Ax. \quad (6)$$

A solução é dada por:

$$x(t) = e^{At}x(0). \quad (7)$$

A matriz **A** é definida por:

$$e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \dots \quad (8)$$

A solução (ou os pólos da função de transferência) da equação (8) é dada pelos autovalores e autovetores da matriz **A**. Consideramos a autodecomposição de **A**:

$$AT = T\Lambda. \quad (9)$$

$\Lambda$  representa uma matriz diagonal onde as entradas são os autovalores da matriz A. É conveniente calcularmos a matriz exponencial  $e^{At}$  ao invés de  $e^{AT}$ .

$$e^{At} = \begin{bmatrix} e^{\lambda_1 t} & 0 & \dots & 0 \\ 0 & e^{\lambda_2 t} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & e^{\lambda_n t} \end{bmatrix}. \quad (10)$$

Podemos escrever a equação (9) de uma maneira que conseguimos representar potências de  $A$  (o que resolveria o problema da equação (8) em termos de:

$$\begin{aligned} A &= T\Lambda T^{-1}, \\ A^k &= (T\Lambda T^{-1})(T\Lambda T^{-1}) \dots (T\Lambda T^{-1}) = T\Lambda^k T^{-1}, \\ e^{At} &= e^{T\Lambda T^{-1}t} = T \left[ I + \Lambda t + \frac{\Lambda^2 t^2}{2!} + \frac{\Lambda^3 t^3}{3!} + \dots \right] T^{-1} = T e^{\Lambda t} T^{-1}. \end{aligned} \quad (11)$$

Combinando a equação (7) com a equação (11), é possível escrever a solução  $x(t)$  como:

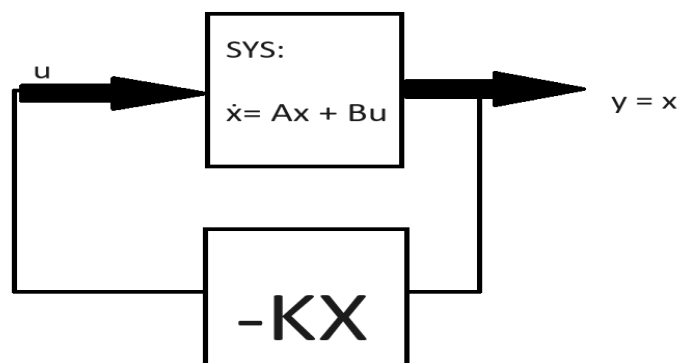
$$x(t) = T e^{\Lambda t} T^{-1} x(0). \quad (12)$$

### 3.2. Matriz Controlabilidade e Observabilidade

Um sistema é controlável se eu consigo escolher um  $u = Kx$  e colocar os autovalores do sistema em quaisquer lugares que eu quiser, com uma boa escolha de  $K$ . Com isso eu consigo deslocar o estado  $x$  para onde achar necessário (Brunton; Kutz, 2019).

Em um sistema de malha fechada como disposto abaixo:

Figura 2: Sistema de malha fechada em loop.



Fonte: Autor.

Se  $\dot{x} = Ax + Bu$  e  $u = -Kx$  podemos afirmar que  $\dot{x} = Ax - BKx$  ou simplesmente,  $\dot{x} = (A - BK)x$ . Isto gera uma nova matriz  $(A - BK)$  com autovalores estáveis.



A habilidade de encontrar autovalores em um sistema de loop fechado com a escolha de  $K$ , significa, em termos vagos que o sistema é controlável. A controlabilidade linear é determinada pela matriz  $C$ .

$$C = [B \ AB \ A^2B \ \dots \ A^{n-1}B] \in \mathbb{R}^n \quad (13)$$

Essa matriz pode ser obtida pelo código *python*, através da biblioteca *control*, através do comando *control.ctrb*.

Um sistema é observável quando for possível estimar qualquer estado  $\xi \in \mathbb{R}^n$  a partir de uma história temporal de medições  $y(t)$ .

$$O = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix} \in \mathbb{R}^n \quad (14)$$

A motivação para a estimado total de estado é relativamente direta. Com o feedback  $u = -Kx$  é possível modificar o comportamento do sistema controlável. Entretanto, se as medidas  $x$  não estiverem disponíveis, será necessário estimá-las e isto só é possível se o sistema for observável (Brunton; Kutz, 2019).

A matriz *observabilidade*  $O$  pode ser obtida através da biblioteca *python control*, através do comando *control.obsv*.

### 3.3. Linear Quadratic Regulator (LQR)

A utilização do LQR tem-se mostrado uma alternativa em diversas aplicações em sistemas modernos, como exemplo, nos controles de servomecanismo. Isto devido às suas excelentes características de desempenho bem como a sua simples exemplificação.

Partindo do suposto que as matrizes  $A$  e  $B$  são controláveis, é possível manipular a nova matriz  $(A-BK)$  através da escolha da lei de controle de feedback do estado completo  $u = -Kx$ . Esta otimização parte do princípio de que as matrizes de medições  $C$  e  $D$  estão disponíveis e são conhecidas, porém nem sempre isto se tornará verdade, ainda mais se tratando de sistemas com múltiplas entradas (Brunton; Kutz, 2019).

Dado um sistema de controle total de estado observável, existem muitas escolhas de controle para estabilização ( $u = -Kx$ ). É possível arbitrar a estabilidade

colocando os autovalores do sistema ( $A - BK$ ) mais à esquerda possível do plano complexo. (Brunton; Kutz, 2019). Esta é uma opção que irá resolver o problema da estabilidade, porém, autovalores muito estáveis requerem custos extremamente “caros” e podem resultar em sinais que ultrapassem os níveis aceitáveis pelos atuadores. Além disso, temos o problema de reações exageradas a ruídos e distúrbios, criando um problema de *jitter* (Brunton; Kutz, 2019).

Escolher a melhor matriz ganho  $K$  para estabilizar o sistema sem gastar muitos recursos de controle é importante em um controle otimizado. Um balanço deve ser alinhado entre a estabilidade e agressividade do controle. A função custo abaixo:

$$J(t) = \int_0^t \mathbf{x}(\tau)^* \mathbf{Q} \mathbf{x}(\tau) + \mathbf{u}(\tau)^* \mathbf{R} \mathbf{u}(\tau) d\tau. \quad (15)$$

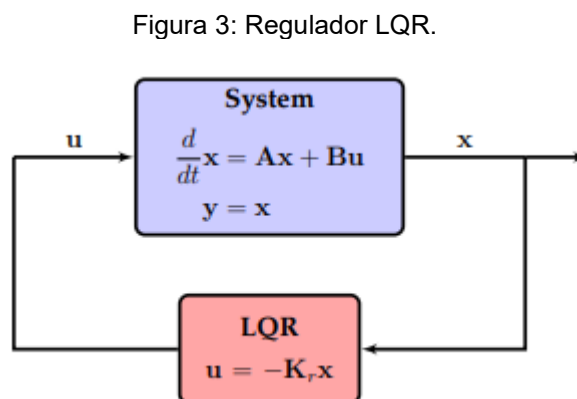
Realiza o balanço entre o custo de uma regulagem efetiva com o custo do controle. As matrizes  $\mathbf{Q}$  e  $\mathbf{R}$  pesam o custo dos desvios do estado de zero até o custo de atuação, respectivamente. Adicionando uma função custo, faz uma com que escolher um controle seja um problema com uma solução bem pensada. O Regulador Quadrático Linear (LQR) de controle  $\mathbf{u} = -\mathbf{K}_r \mathbf{x}$  é desenhado para minimizar  $J = \lim_{t \rightarrow \infty} J(t)$ .

O nome Quadrático vem pois o custo da função é uma função quadrática, analiticamente sendo resolvido achando  $\mathbf{X}$  e  $\mathbf{K}_r$  nas equações abaixo:

Estas equações são numericamente robustas e podem ser obtidas por meio de

$$\begin{aligned} \mathbf{K}_r &= \mathbf{R}^{-1} \mathbf{B}^* \mathbf{X}, \\ \mathbf{A}^* \mathbf{X} + \mathbf{X} \mathbf{A} - \mathbf{X} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^* \mathbf{X} + \mathbf{Q} &= \mathbf{0} \end{aligned} \quad (16)$$

*Python* através da biblioteca *control* por meio do código *control.lqr()*.



Fonte: Brunton; Kutz, 2019.

### 3.4. Filtro de Kalman

O controlador otimizado LQR depende de medições de estado total do sistema. Entretanto, estas medições podem ser tecnologicamente inviáveis de obter e nem sempre estarão à disposição.

É possível estimar o estado através de medições ruidosas  $y$ . A estimação do estado total é matematicamente possível desde que o par de matrizes  $(A, C)$  sejam observáveis. O filtro de Kalman é o estimador de estado total mais comumente usados (BRUNTON; KUTZ, 2019), por otimizar de maneira ideal os efeitos concorrentes de ruído, distúrbios e incertezas da modelagem.

A biblioteca *control* de *Python* possui o comando *control.lqe* que produz um filtro de Kalman para sistemas contínuos (*control.dlqe()* para sistemas discretos), a solução para a equação de Riccati e os autovalores dos polos de **(A-LC)**, dado o sistema:

$$\begin{aligned} dx/dt &= Ax + Bu + Gw, \\ y &= Cx + Du + v. \end{aligned} \tag{17}$$

de ruído de processo  $w$  e ruído de medição  $v$  com covariâncias:

$$E\{ww^T\} = QN, E\{vv^T\} = RN, E\{wv^T\} = NN. \tag{18}$$

A função *lqe()* calcula a matriz de ganho do observador  $L$  de modo que o filtro de Kalman estacionário não variante no tempo é:

$$\begin{aligned} \frac{dx_e}{dt} &= Ax_e + Bu + L(y - Cx_e - Du). \\ L, P, E &= lqe(A, G, C, QN, RN). \end{aligned} \tag{19}$$

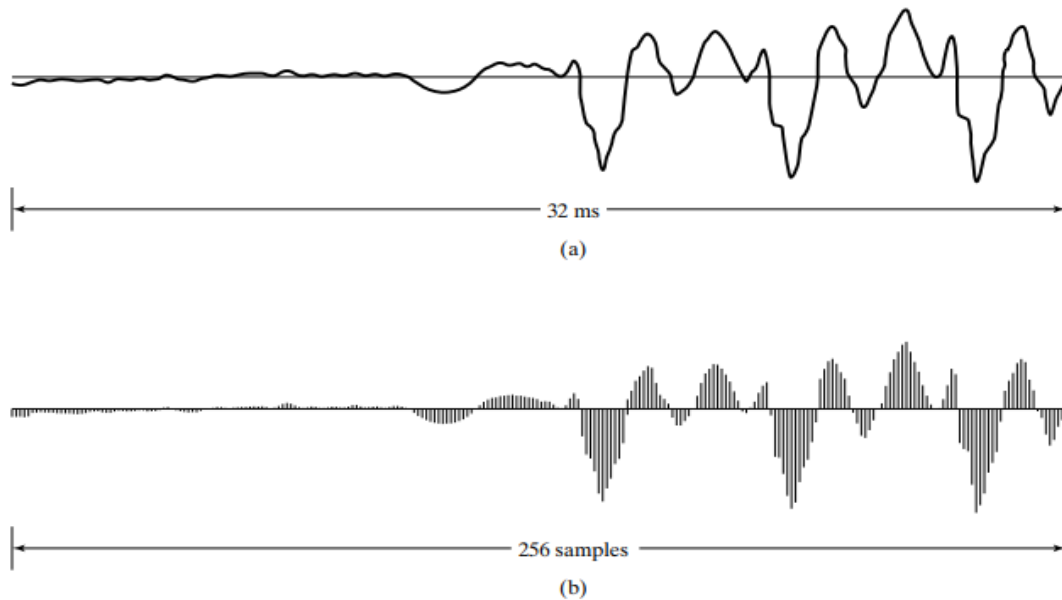
### 3.5. Sistemas Discretos

A instrumentação utilizada nas malhas de controle, medem, por vezes a amostragem do sinal em frequência. Diferente de um mostrador analógico, que sempre marcará a medição, instrumentos digitais trabalham com medições em intervalos igualmente espaçados.

Uma das aplicações de sistemas discretos são para Processamento Digital de Sinais (DSP), no qual os sinais são manipulados digitalmente em tempo discreto. Isso inclui aplicações em áudio, vídeo, comunicações, entre outras áreas.

Figura 4: (a) Sinal analógico (b) Sinal discretizado com período  $T=125\mu\text{s}$ .

Sinais analógicos são definidos por serem contínuos no tempo e são



Fonte: Oppenheim, 2014.

representados por uma única variável independente. Sinais de tempo discreto são definidos em tempos discretos e, com isso, a variável independente tem valores discretos; isto é, os sinais de tempo discreto são representados como sequências de números (Oppenheim, 2014).

A transformação bilinear, também conhecida como transformação de Tustin, é uma técnica importante na área de controle de sistemas dinâmicos. Ela é usada para converter um sistema contínuo em um sistema discreto, permitindo a análise e projeto de controladores para sistemas que operam em tempo discreto.

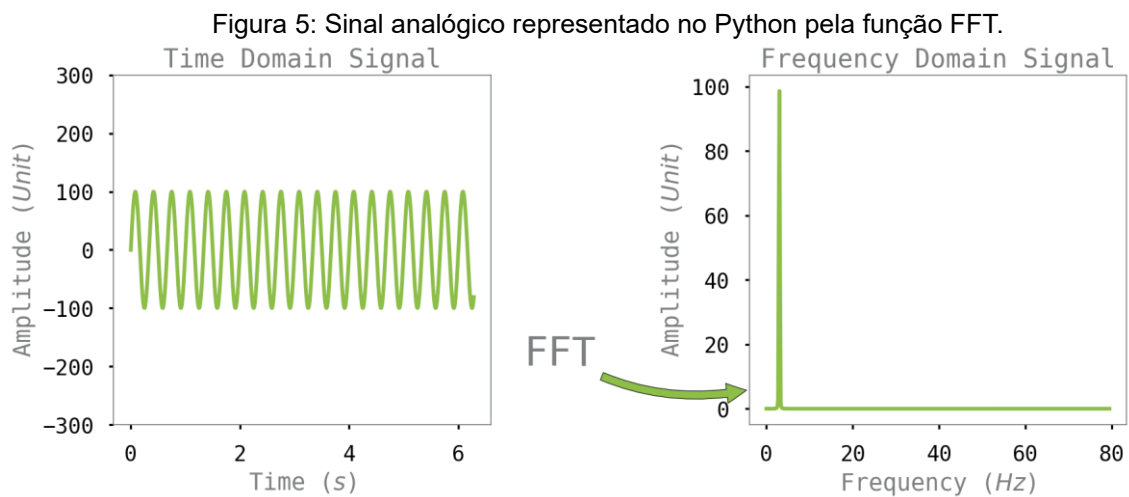
A transformação bilinear, corresponde a substituir o  $s$  pela expressão abaixo:

$$s = \frac{2}{T_d} \left( \frac{1 - z^{-1}}{1 + z^{-1}} \right). \quad (20)$$

A Transformada rápida de Fourier (FFT) é um algoritmo que calcula a Transformada Discreta de Fourier. É usada para algoritmos de filtragem, em compressão de áudio e vídeo, comunicação e no espectro geral para processamento de dados. O avanço da computação permitiu que o cálculo necessário para realizar esta operação pudesse ser feito em poucas linhas de códigos. Bibliotecas *Python*,

como *numpy* ou *scipy*, já possuem o próprio comando `fft` e `ifft` para se referir a transformada (equação DFT abaixo) e a sua operação inversa (Reducible, 2021).

$$A_k = \sum_{m=0}^{n-1} a_m \exp\left\{-2\pi i \frac{mk}{m}\right\} \quad k = 0, \dots, n-1. \quad (21)$$



Fonte: <http://balzer82.github.io/FFT-Python/>, 2023

### 3.6. Filtros

Um filtro tem como função remover partes não desejadas de um sistema de transmissão ou extrair partes que podem ser úteis do sinal. O filtro digital processa sinais digitais usando um processador digital para executar cálculos necessários para fazer a filtragem, multiplicando os valores da entrada por constantes e somando os produtos resultantes, além de utilizar técnicas matemáticas, como Transformadas de Fourier e Hilbert em valores amostrados do sinal de entrada. Em um processo de filtragem digital, o sinal analógico deve ser primeiramente digitalizado usando um ADC.

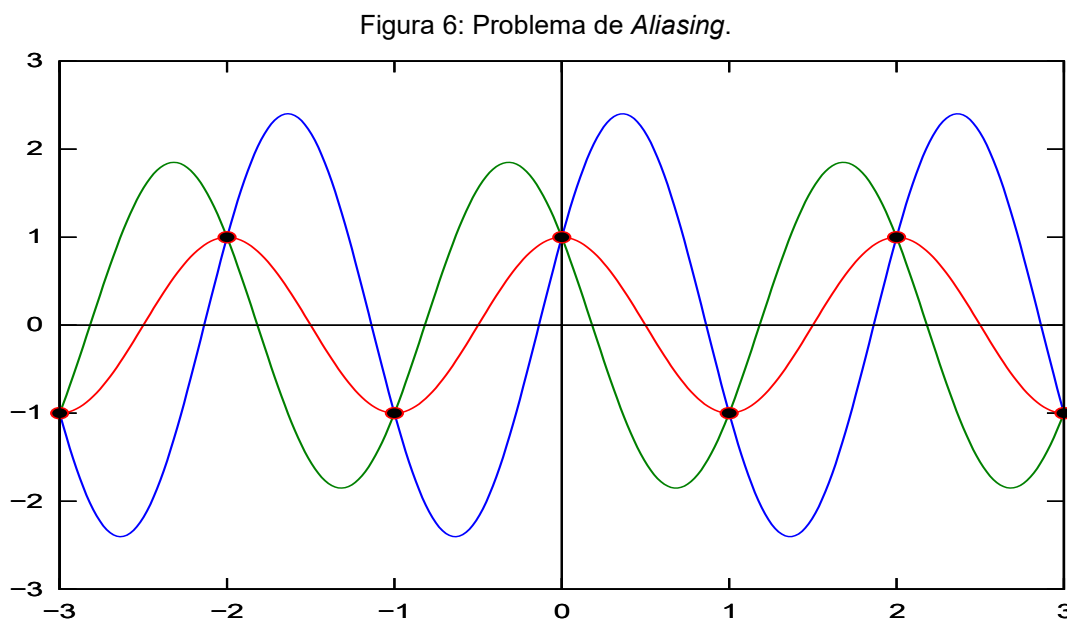
O design de filtros envolve seguir as seguintes etapas: especificar as propriedades do sistema, a aproximação das especificações usando tempo discreto e a realização do sistema. Embora essas três etapas pareçam dependentes, o foco deverá ser majoritariamente na segunda etapa. Em um cenário prático, o filtro desejado geralmente é implementado com *hardware* digital e frequentemente usado para filtrar um sinal derivado de um analógico por meio de amostragem periódica seguida de conversão A/D (Oppenheim, 2014).

Sistemas discretos implementados com equações diferenciais são definidos em duas categorias: Resposta de impulso infinito (IIR) e respostas de impulso finito (FIR). Projetar filtros IIR implica em obter a função de transferência aproximada que é uma função racional de  $z$ , enquanto projetar um filtro FIR implica em realizar aproximação polinomial (Oppenheim, 2014).

Como em todos os gêneros de filtros, o modelo típico é o filtro passa-baixas, que pode ser modificado para se tornar um filtro passa-altas, ou colocado em série com outros filtros para formar filtros passa-faixa ou rejeita-faixa, e versões de ordem mais elevadas destes (Wikipedia, 2021).

### 3.7. Problema de *Aliasing*

Na figura abaixo temos três funções senoidais no tempo representada em sua forma discreta por meio dos pontos pretos. Apesar das funções no tempo serem diferentes umas das outras, a discretização delas no dará uma única função, o que pode acarretar problemas quando for necessário trabalhar com estes dados.



Fonte: Wikipedia, 2023.

O teorema de amostragem de Nyquist-Shannon é o princípio essencial para processamento de sinais digitais ligando a frequência de um sinal e a *rate* de amostragem para evitar distorção (*aliasing*). O teorema diz que a *rate* de amostragem

deve ser pelo menos duas vezes a largura de banda do sinal para evitar esta distorção.  
(Wikipedia, 2023)

## 4. DATA DRIVEN CONTROL SYSTEMS

### 4.1. Conceitos

Segundo Brunton e Kutz (2019), os principais objetivos e desafios nos sistemas dinâmicos modernos que simulam fenômenos da vida real são:

**Design e otimização:** Melhorar os parâmetros do sistema para ter a performance ou estabilidade incrementada.

**Controle:** Com retroalimentação, é possível controlar ativamente um sistema dinâmico, usando medições para informar a atuação de forma a alterar o comportamento do sistema. É usual que tenha ou pelo menos estime-se o espaço de estado do sistema.

**Interpretação e entendimento físico:** Um dos objetivos mais fundamentais é prover percepção física e interpretação do comportamento do Sistema através de análise de trajetórias e soluções às equações de governo.

Sistemas reais geralmente são não-lineares e exibem diversos comportamentos no espaço-tempo, assumimos, ainda, incertezas nas equações, na especificidade dos parâmetros e nas medidas do sistema. Alguns sistemas são mais sensíveis a essas incertezas e abordagens probabilísticas devem ser usadas. Cada vez mais, também é o caso de que as equações básicas de movimento não são especificadas e podem ser intratáveis para derivar de primeiros princípios.

Identificar sistemas não lineares dinâmicos desconhecidos por dados e aprender coordenadas intrínsecas que permitem representá-los linearmente são os objetivos mais urgentes de sistemas dinâmicos modernos (Brunton; Kutz, 2019).

Sistemas de Controle baseados em dados (*Data-Driven Control Systems*) se referem a uma classe de metodologias de controle que utilizam análise de dados para melhorar as decisões. Estes sistemas utilizam, além da análise de dados; aprendizagem de máquinas (*Machine Learning*) e Inteligência Artificial (IA) para melhorar a eficiência dos sistemas de controle (Brunton; Kutz, 2019).

Podem adaptar-se em tempo real às condições em mudança, pois atualizam continuamente as decisões de controle com base nos dados mais recentes. E demonstra robustez e adaptabilidade pois aprendem com as respostas reais do sistema, potencialmente levando a um melhor desempenho em ambientes incertos.



Sistemas de controle baseados em dados se concentram na extração de padrões, tendências e correlações de grandes conjuntos de dados sem depender muito de modelos pré-existentes.

Sistemas de controle convencionais, dependem de modelos matemáticos derivados de princípios físicos ou já conhecidos. Esses modelos permitem que o design do controle Proporcional-Integral-Derivativo (PID) seja ajustado para uma melhor resposta desejada. O método de Ziegler-Nichols, por exemplo, é comumente aplicado em indústrias, devido a obter uma resposta de controle mais agressiva (IPCOS, 2020).

## 4.2. Machine Learning

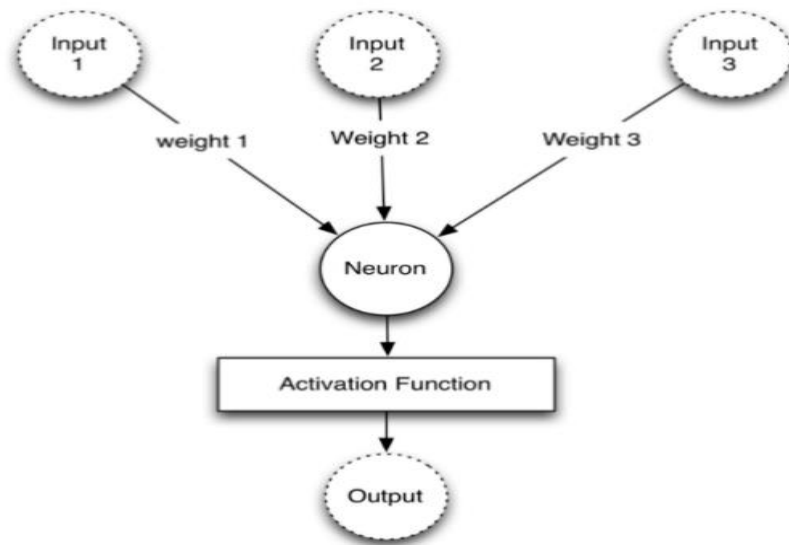
Estes métodos geralmente exigem conhecimento preciso da dinâmica do sistema real e podem se tornar extremamente não lineares e complexos de modelar.

*Machine learning* (ML) é a disciplina de ciência da computação que usa algoritmos computacionais e análise de dados para construir modelos de predição capazes de resolver problemas. Estes algoritmos aprendem e se atualizam através de programação intrínseca, sem necessidade de atualizar o código (Mohan, 2023).

O Perceptron é a forma mais simples de uma rede neural usada para a classificação de padrões ditos linearmente separáveis. Basicamente, ele consiste em um único neurônio com pesos sinápticos ajustáveis e bias (Heaton, 2022).

A habilidade para criar redes neurais profundas existe desde que Pitts em 1943, introduziu o Perceptron de múltiplas camadas. Entretanto, não se tem sido capaz de efetivamente treinar redes neurais profundas até Hinton (1984) tornar-se o primeiro pesquisador a treinar com sucesso estas redes neurais mais complexas (Heaton, 2022).

Figura 7: Neurônio.



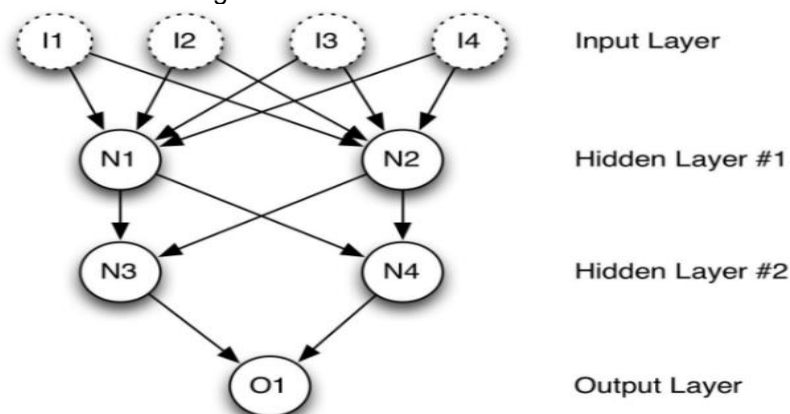
Fonte: <https://sites.wustl.edu/jeffheaton/t81-558/>, 2022.

O algoritmo de aprendizagem é composto de unidades individuais interconectadas chamadas neurônios, representados na Figura 8. Um neurônio multiplica cada uma destas entradas ( $x$ ) pelo peso ( $w$ ), adiciona as multiplicações e envia o dado para a função de ativação como mostrado a seguir (Heaton, 2022):

$$f(x_i, w_i) = \phi \left( \sum_i (w_i \cdot x_i) \right) \quad (22)$$

Encadeando vários neurônios, você pode criar uma rede neural artificial (ANN, ou simplesmente NN). Como demonstrado a seguir:

Figura 8: ANN com duas camadas.



Fonte: <https://sites.wustl.edu/jeffheaton/t81-558/>, 2022.

A Figura 8, representa uma rede neural com duas camadas. Note que a função de ativação foi propositalmente deixada de fora da imagem, pois essa é uma maneira de apresentar a rede neural, porém, é importante escolher uma função de ativação, pois ela pode afetar como deve-se formatar os dados da entrada.

Funções de ativação são conhecidas por funções de transferência, são usadas para calcular a saída de cada camada de uma NN. Historicamente, usaram tangente hiperbólica, sigmoid ou função de ativação linear. Porém, redes neurais modernas profundas, usam as seguintes funções de ativação (Heaton, 2022).

- Rectified Linear Unit (ReLU) – Usada para a saída de camadas ocultas,
- Softmax – Usada para a saída de NNs de classificação, e
- Linear – Usada para a saída de NNs de regressão.

A aprendizagem de ML pode ser dividida em 3 categorias, que mudam dependendo da natureza do sinal ou feedback disponível (Wikipedia, 2023)

Aprendizado supervisionado: constrói um modelo matemático a partir de um conjunto de dados que contém tanto a entrada quando a saída desejada. Neste modelo, cada exemplo de treino é representado por uma *array* ou vetor, e o dado do treino é representado por uma matriz. Através do processo iterativo de otimização de uma função objetivo, algoritmos supervisionados aprendem a função que pode ser usada para prever a saída associada a novas entradas. Exemplo: *Support Vector Machines*.

Aprendizado não supervisionado: Não são dados rótulos para o algoritmo, deixando-o achar por si próprio a estrutura da entrada. Ao invés de responder ao *feedback*, algoritmos não supervisionados identificam semelhanças nos dados e reagem baseados na presença (ou ausência) destas semelhanças. Durante a fase de aprendizado, a rede tenta imitar o dado que é disponibilizado como entrada e usa o erro gerado para se corrigir através de “pesos” e “vieses”. Algumas vezes o erro pode ser expresso como baixa probabilidade de que a saída errônea ocorra ou pode ser expresso como um estado instável. Exemplo: Máquina de Boltzmann.

Aprendizado por reforço: é a área de ML que concentra como os agente de software irão tomar ações em um ambiente que maximiza a noção de recompensa acumulada. Esse campo é estudado em diversas disciplinas, sendo, Teoria de controle, pesquisa de operações, teoria de informação, estatística e algoritmos genéticos e difere do aprendizado supervisionado em não necessitar rotular o par input/output e por não necessitar de ações sub-otimizadas para serem corrigidas. O

foco é achar um equilíbrio entre exploração de território não conhecido e exploração do conhecimento já adquirido. O programa é alimentado, enquanto funciona, de feedback que é análogo às recompensas, as quais tenta maximizar. Exemplo: *Q-Learning* e *Deep Q-Learning*.

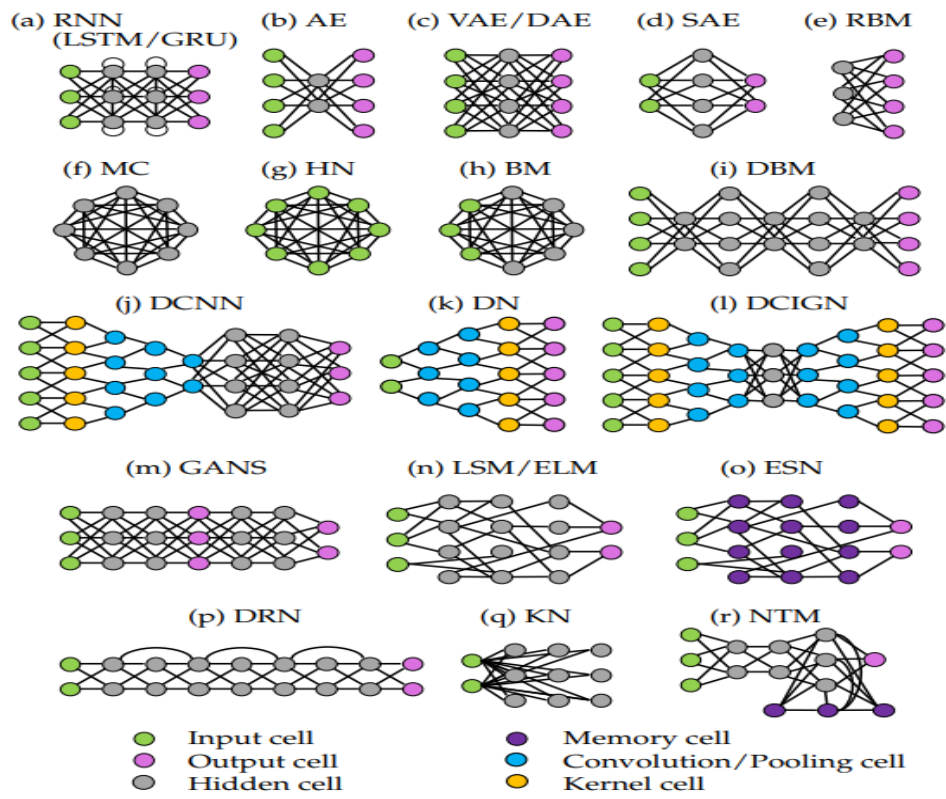
Aprendizagem Profunda (*Deep Learning*) é relativamente um novo avanço em programação de redes neurais artificiais e representa uma maneira para treinar redes neurais (NN) profundas.

Existem diversas poderosas bibliotecas de *Python*, tais como *PyTorch*, *TensorFlow* e *Theano*, que auxiliam na programação de NNs e Machine Learning

Se dados ricos suficientes forem adquiridos, NNs oferecem a habilidade de interrogar estes dados para uma gama de tarefas centradas na classificação e predição. Uma vez que a NN foi treinada no dado da trajetória, o modelo não linear mapeado  $x_k$  até  $x_{k+1}$  pode ser utilizado para prever o estado futuro do sistema em qualquer ponto (Brunton; Kutz, 2019).

Devido ao aprendizado online e à capacidade de aproximação de funções não lineares, NNs são extensivamente usadas para aproximar incertezas em modelos não lineares. Qualquer função pode ser aproximada com certo grau de precisão, com diversos estudos adotando a abordagem de ML para aproximar dinâmicas de sistemas de engenharia complexos para aplicações de controle.

Figura 9: Diversos modelos de Redes Neurais.



Fonte: Brunton; Kutz, 2019:

## 5. ESTUDO DE CASO: IDEIAS PARA CONCEPÇÃO DE UM USV PARA A MB. (UNMANNED SURFACE VEHICLE)

### 5.1. Apresentação do caso

A atividade de batimetria é o serviço que constitui a medição da profundidade de rios, mares e vias aquáticas, além de identificar a topografia de seus leitos. A Marinha do Brasil (MB), através da Diretoria de Hidrografia e Navegação (DHN) é o órgão brasileiro centralizador das habilidades técnicas e é através dele que são concedidas as autorizações para órgãos civis realizarem batimetria.

Um importante produto gerado pela DHN são as cartas náuticas, essenciais para o planejamento da navegação. Estas cartas possuem a representação hidrográfica da faixa oceânica e costeira adjacente ao litoral brasileiro, assim como a dos rios, canais e outras vias navegáveis do território nacional. sobre a região a ser navegada. Cabe à MB a competência privativa para elaboração e execução do Plano Cartográfico Náutico Brasileiro, conforme definido no Decreto-Lei nº 243, de 28 de fevereiro de 1967 (Marinha do Brasil, 2023).

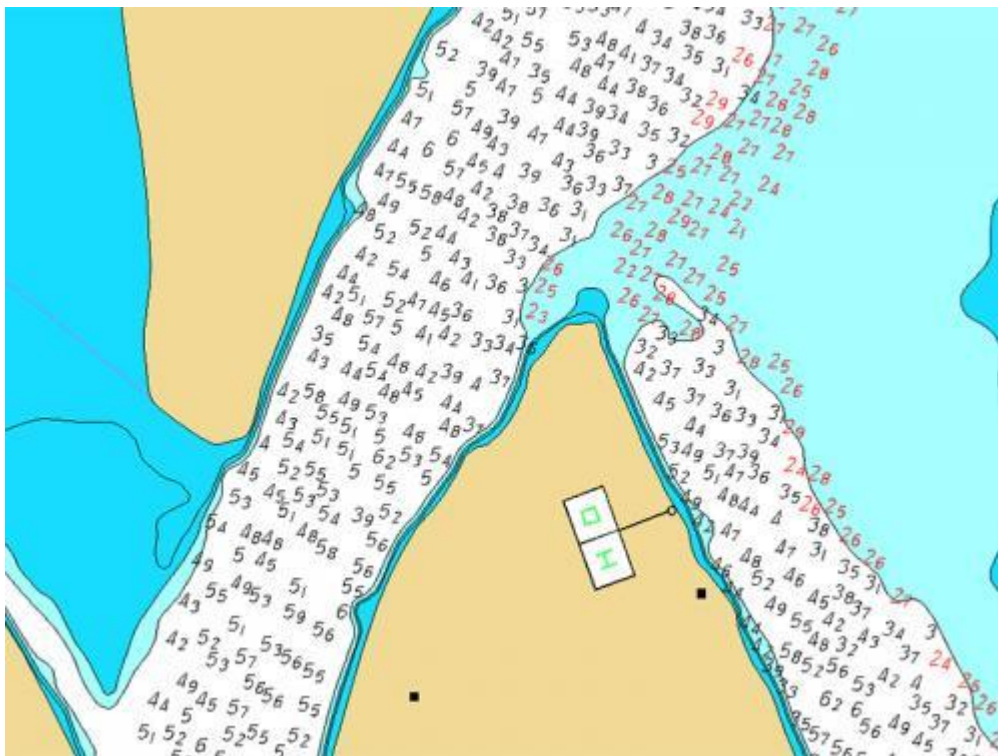


Figura 10: Rio brasileiro com batimetria realizada pelo CHN-9. Na imagem cada valor numérico representa a profundidade encontrada no local, dando uma ideia aos navegantes do caminho mais seguro a ser percorrido.

Consegue-se apenas imaginar a dificuldade em realizar a batimetria de uma extensão de 7,4 mil quilômetros de costa, além da bacia amazônica (Marinha do Brasil, 2019).

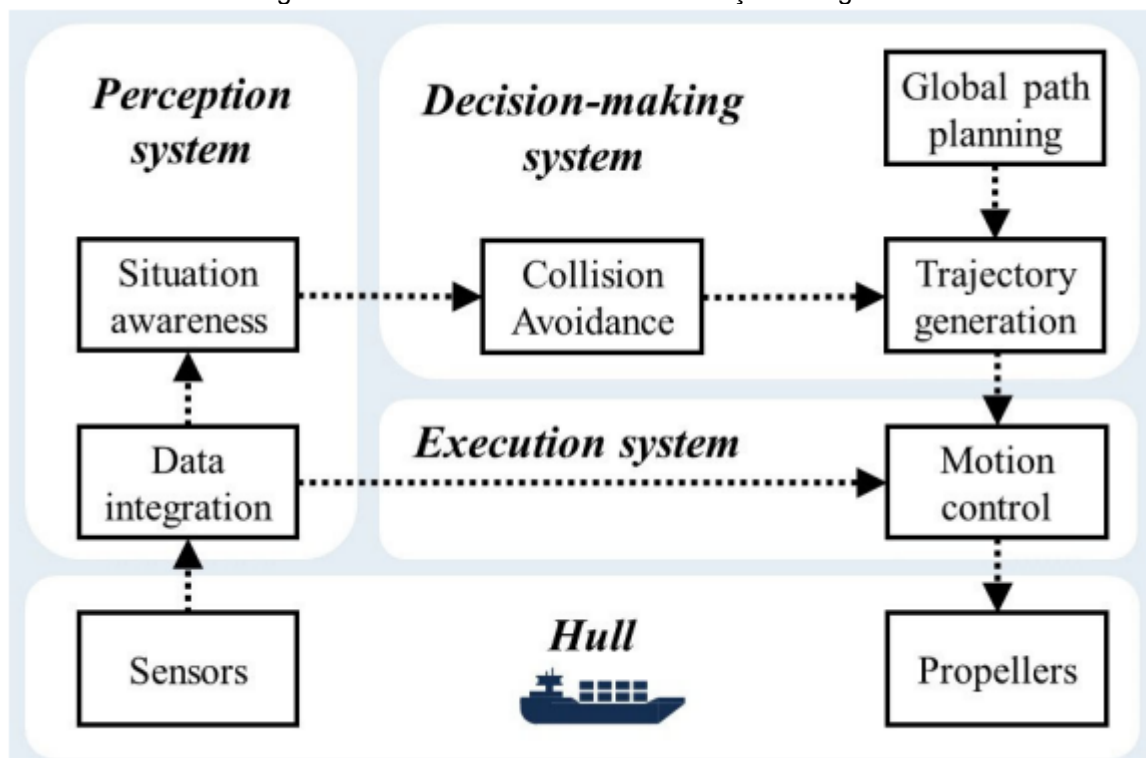
Com o advento da popularidade de tópicos como Inteligência Artificial e de Machine Learning, veículos não tripulados vêm ocupando espaço nas pesquisas e técnicas avançadas de controle de embarcações estão sendo desenvolvidas para garantir a independência de navios, carros e aeronaves.

Um navio inteligente é, como se pode esperar, um sistema de elevada complexidade que envolve teorias de diversos campos, como o design e fabricação do casco, propulsores e motores, aplicações de sensores, processo decisório inteligente, comunicação e fusão de informação (Tang et al., 2022).

Podemos definir o problema em 3 subsistemas: um sistema de percepção, um sistema para realizar o processo decisório e um sistema de execução das ações.

Esses módulos irão interagir entre si e a estabilidade e confiabilidade de cada módulo afetará a eficiência da missão e a segurança da navegação (Tang et al., 2022).

Figura 11: Subsistemas de uma embarcação inteligente.



Fonte: TANG et al., 2022.

### 5.1.1. Subsistema de percepção

O sistema de percepção coleta informações oriundas de vários sensores e resulta no conhecimento situacional da embarcação utilizando técnicas de fusão de informação. Um dos métodos mais usados para fusão de informação é o filtro de Kalman (Zaccaria; Rahman, 2019), como visto no capítulo 2.3 deste trabalho.

As Redes Neurais de Convolução (CNN), são uma técnica inovadora e altamente eficaz no campo do processamento e reconhecimento de imagens. Desenvolvidas para analisar dados visuais, elas imitam o modo como o cérebro humano processa informações visuais e têm sido amplamente aplicadas em uma variedade de tarefas, desde o reconhecimento facial até a análise de imagens médicas.

Em sua estrutura, as CNNs contam com camadas de convolução, pooling (agrupamento) e camadas totalmente conectadas. Na camada de convolução, a rede usa um conjunto de filtros para varrer a imagem e detectar padrões como bordas, cores e texturas, com cada filtro captando diferentes aspectos visuais. Em seguida, a camada de pooling reduz as dimensões dos dados capturados para tornar a rede mais eficiente e menos propensa a overfitting. Finalmente, as camadas totalmente conectadas interpretam esses recursos agrupados para realizar tarefas específicas, como classificação ou reconhecimento de padrões.

As CNNs são especialmente eficazes no reconhecimento de imagens devido à sua capacidade de preservar a relação espacial entre os pixels, aprendendo padrões visuais de maneira eficiente. Além disso, utilizam o compartilhamento de pesos, aplicando os mesmos pesos em diferentes partes da imagem, o que ajuda a identificar padrões similares em várias posições. Essa abordagem também contribui para a redução da complexidade computacional, pois as CNNs aprendem apenas os parâmetros mais importantes dos dados (Wikipedia, 2023).

As aplicações das CNNs são vastas e impactantes. No reconhecimento facial, elas são usadas para identificar ou verificar pessoas em imagens ou vídeos. Em veículos autônomos, contribuem para a visão computacional, permitindo que os veículos interpretem o ambiente ao seu redor.

De forma a ajudar a solucionar o subsistema de tomada de decisão, é proposto um algoritmo de *Support Vector Machine* (SVM) para assessorar no reconhecimento



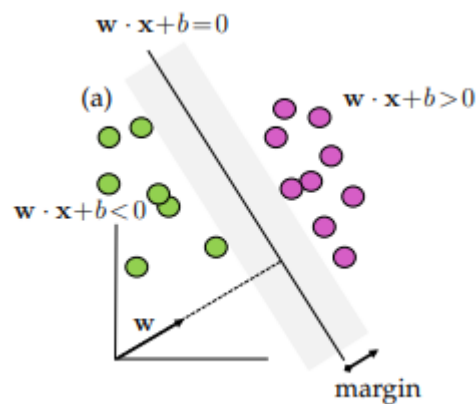
do risco de colisão. O SVM é um dos mais bem sucedidos métodos desenvolvidos para tratamento de dados até hoje (Brunton; Kutz, 2019).

A ideia principal de um SVM é construir um hiperplano caracterizado por:

$$w \cdot x + b = 0. \quad (23)$$

Onde o vetor  $w$  e a constante  $b$  parametrizam o hiperplano.

Figura 12: Classificação de dados utilizando SVM. Determinar o vetor  $w$  e a constante  $b$  são o objetivo da otimização de um SVM.



Fonte: Brunton; Kutz, 2019.

A otimização e o sucesso de um SVM são dados pela determinação de  $w$  e  $b$  de forma que sejam minimizados o número de dados classificados erroneamente e criando a maior margem possível (Brunton; Kutz, 2019).

O SVM pode ainda adotar formas não lineares onde  $x$  passa a ser mapeado em um espaço não linear e de dimensão superior. A habilidade de um SVM de ser conectado a espaços multidimensionais não lineares fazem deste algoritmo de ML um dos mais bem sucedidos.

Empresas como a sea.ai (<https://sea.ai>), já são capazes de produzir sistemas capazes de realizar a identificação de embarcações e objetos flutuantes utilizando sensores óticos.

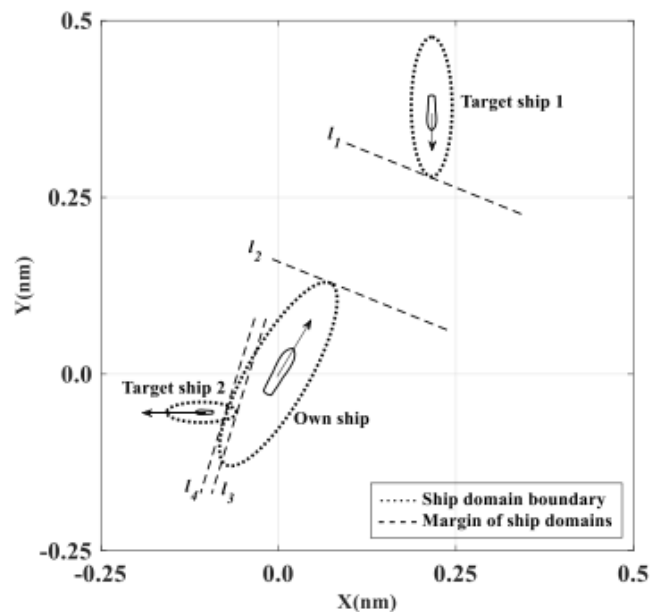
Figura 13: Modelos de sensores óticos comercializados.



Fonte: sea.ai, 2023.

Desta forma vamos utilizar o conceito proposto por Fujii e Tanaka em 1971 de “domínio do Navio” (*Ship’s Domain ou SD*), cuja se define como “uma área bidimensional ao redor de um navio que outros navios devem evitar”. Sendo assim, caso a área de dois navios se interpolarem, os sistemas de atuação deverão começar a atuar, de acordo com a área interpolada.

Figura 14: SDs e margens.



Fonte: Kai Zheng, 2020.

Do ponto de vista do assessoramento de risco, as margens entre os SDs são um fator importante para a medição do risco e podem ser obtidas resolvendo um SVM linear.

A distância e o tamanho as áreas sobrepostas de dois SDs refletem a probabilidade de colisão. Dois SDs se movendo cada vez mais perto e tendo mais área sobreposta implica no aumento da probabilidade de colisão (Kai Zheng, 2020).

A probabilidade de colisão fica definida (Kai Zheng, 2020) então por:

$$P_{o,k}(d_{o,k}) \begin{cases} \frac{1}{\lambda + \frac{\bar{d}_{o,k}}{m_{o,t}^k}}, & \text{se } D_{own}(x_o, y_o) \cap D_t^k(x_t^k, y_t^k) = \emptyset \\ \frac{1}{\lambda} + \left(1 - \frac{1}{\lambda}\right) \frac{\overleftrightarrow{d}_{o,k}}{m_{o,k}^k}, & \text{se } D_{own}(x_o, y_o) \cap D_t^k(x_t^k, y_t^k) \neq \emptyset \end{cases} \quad (24)$$

Onde  $\lambda > 1$  e é maior que 1.  $\overleftrightarrow{d}$  varia não linearmente conforme a interseção dos SDs. Estas equações independem do modelo de SD selecionado

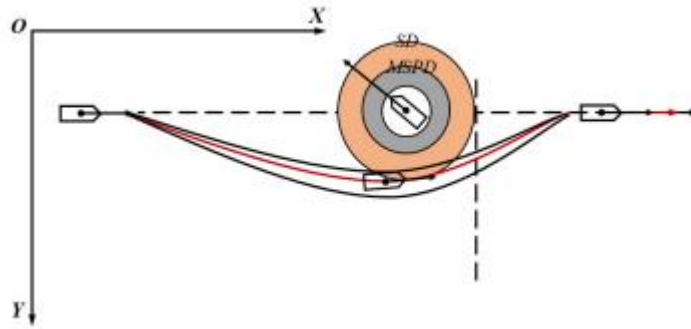
O uso de Redes Neurais de convolução (CNN) é um dos métodos de se analisar processamento de imagens. Estas imagens são oriundas de um vídeo em tempo real, câmera ou imagem arquivada (Heaton, 2022).

### 5.1.2.O subsistema de geração de derrota

Assumimos que no momento que o USV começa sua operação, este irá seguir uma derrota pré-determinada pelo operador. Uma derrota planejada pode ser feita de diversas maneiras, utilizando conhecimentos intrínsecos do operador, no qual este já conhece a região a ser navegada ou de posse de dados, usando redes neurais para criação de derrotas em áreas não navegadas anteriormente pelo operador.

De qualquer forma, uma vez que o USV encontra perigos a navegação, identificados pelo subsistema de percepção, esse deverá possuir a capacidade de se manter em segurança, prezando pela navegação de ambas as embarcações. Para que isto ocorra, é provável que o USV tenha que sair da derrota, sendo capaz de criar uma trajetória segura, que safe o perigo, que siga as convenções internacionais de prevenção de colisões (COLREGs) e use e que permita voltar a derrota planejada em um período futuro.

Figura 15: Nova trajetória gerada de maneira autônoma para evitar abalroamento.



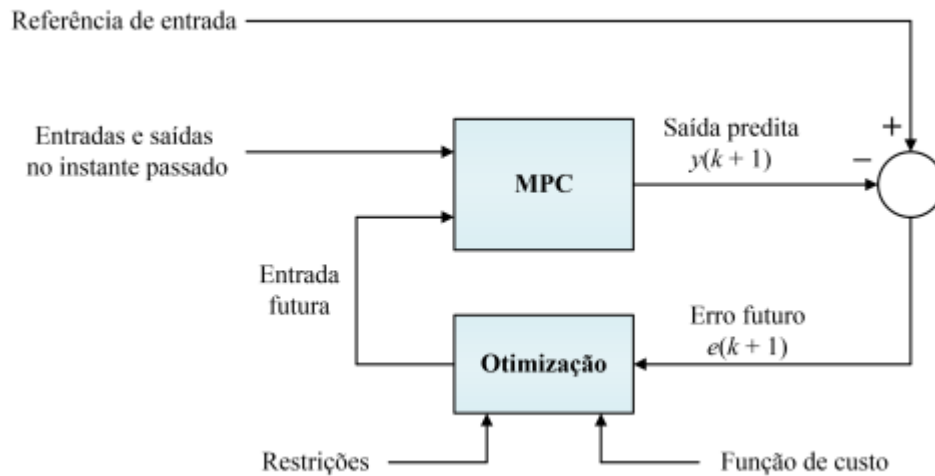
Fonte: Yingjie Tang, 2022.

Para propor uma resolução deste problema, este trabalho sugere utilizar Controle por Modelo Preditivo (MPC).

MPC é uma lógica de controle na qual um modelo matemático é usado para prever estados futuros do sistema e, baseado nestes, o conjunto futuro de sinais de é otimizado para aumentar a eficiência do controle. É como resolver um problema de otimização a cada intervalo de amostragem no futuro. MPC requer um modelo que necessita da capacidade de prever uma resposta futura de um sistema dinâmico com dados históricos. Isto faz com que redes neurais sejam respostas interessantes para implementar ao MPC (Dev et al, 2021).

O MPC desenvolveu-se consideravelmente nas últimas décadas, tanto na área de pesquisa quanto na indústria. Esse sucesso pode ser atribuído ao fato de o MPC lidar com o problema de controle de processo no domínio do tempo. Adicionalmente, é possível implementar restrições na formulação do MPC, enquanto também é possível aplicar esta técnica para o controle de sistemas não lineares (ESTRABIS, 2020).

Figura 16: Estrutura básica de um MPC.



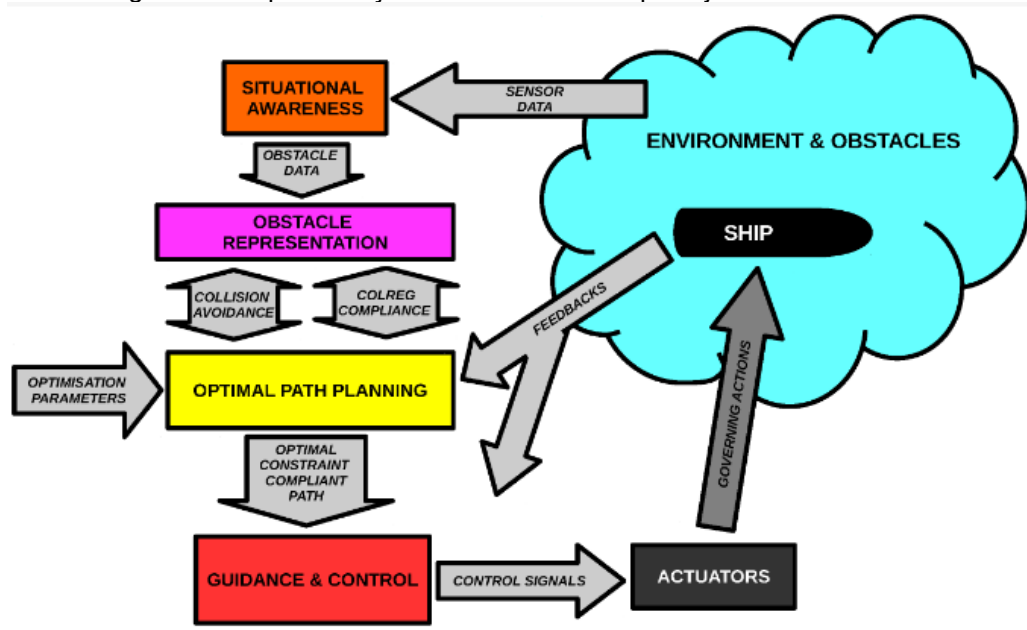
Fonte: Estrabis, 2020.

Ainda segundo Estrabis (2020), os controladores que deram início ao MPC foram o LQR e LQG. Mesmo sendo criado na década de 60, o MPC começou a ser realmente utilizado no final do período de sua primeira geração.

O Conceito de *Rapidly-exploring Random Tree* (RRT) é apresentado como uma estrutura de dados aleatória feito para resolver problemas de planejamento de rotas (Lavalle, 1998). RRTs são planejadas especificamente para não-holonômicos, ou seja, os sistemas com dimensão finita onde algum tipo de restrição é imposto a um ou mais estados do sistema. Estas limitações podem ser provocadas pela inércia do momento angular, condições impostas pela impossibilidade de deslocar em uma ou mais direções, como resultado da imposição de restrições durante o projeto do sistema de controle ou até pelo fato de o sistema não possuir atuadores em todas as direções (Murray apud Figueiredo, 2004).

A figura 16 representa uma arquitetura de um sistema de planejamento de derrota. Durante a navegação, o subsistema de percepção irá coletar informação via sensores, as informações e baseado nesta estrutura de dados, o subsistema de planejamento irá realizar as atuações nos controles do USV (Zaccone, 2021).

Figura 17: Representação de um sistema de planejamento de derrota.



Fonte: Zaccone, 2021.

O algoritmo RRT\* é uma variação otimizada do algoritmo RRT. O algoritmo RRT\* tenta minimizar a função custo do processo de geração do caminho. Os RRT e RRT\* são descritos como algoritmos de amostragem aleatória, à medida que eles amostram aleatoriamente o domínio espacial, criando uma árvore que liga as configurações viáveis.

A definição da função custo é crucial para o controle do formato da trajetória, enquanto o RRT cresce, a ação otimizada de RRT\* reestrutura localmente a árvore para minimizar o custo. Três principais efeitos desejados são identificados:

- Uma ação limitada de guinada,
- Uma margem razoável dos obstáculos, e
- Um alongamento do caminho mais curto.

$$c(x_p) = \gamma_s c_s(x_p) + \gamma_f c_f(x_p) + \gamma_m c_m(x_p) + \gamma_l c_l(x_p). \quad (25)$$

Onde  $\gamma$  representa o somatório dos coeficientes,  $c$  são as contribuições dos custos e  $s, f, m, l$  indicam a ação de direção, a margem para objetos fixos e móveis e o alongamento do trajeto, respectivamente (ZACCONE, 2021).

$$c_s(x_p) = \sum_{k=1}^{p-1} \theta_k^2, \quad (26)$$

$$c_f(x_p) = \sum_{k=0}^p (\min_{y_t \in F} |y_i - x_k|)^{-1}, \quad (27)$$

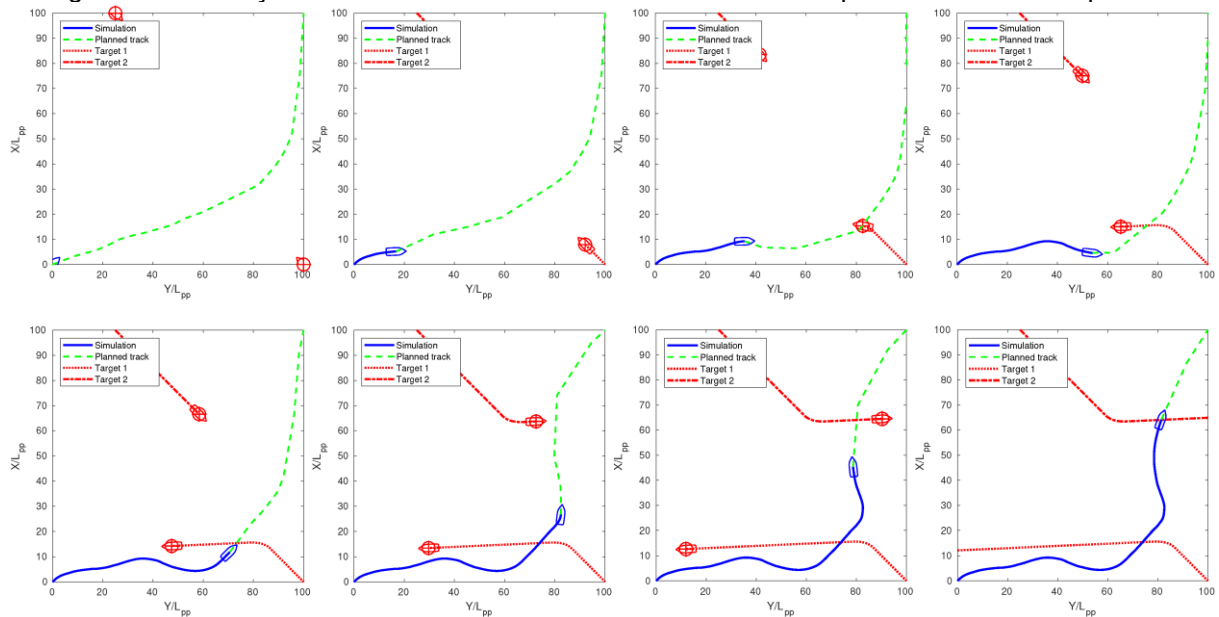
$$c_m(x_p) = \sum_{k=0}^p (\min_{y_t(t_k) \in M(t_k)} |y_i(t_k) - x_k|)^{-1}, \quad (28)$$

$$c_l(x_p) = \sum_{k=1}^p |x_k - x_{k-1}|. \quad (29)$$

Estas equações minimizam a função de custo, deixam a trajetória mais arredondada, maximizando a distância entre o obstáculo e o USV e gerando as menores trajetórias possíveis (Zaccone, 2021).

O trabalho de Zaccone (2021) ainda apresenta simulações que colocam em prática os códigos de RRT\* implementado como mostrado na Figura 17.

Figura 18: Simulação de uma manobra de acordo com a COLREG perante um alvo inesperado.



Fonte: Zaccone, 2021.

### 5.1.3.O subsistema de execução

O subsistema de execução tem como seu objetivo final, gerar entradas apropriadas nos atuadores do sistema de controle de governo e de propulsão, em conformidade com as informações recebidas pelos subsistemas de percepção e de tomada de decisão.

Sua função é então, garantir o controle do USV e que siga a derrota planejada, de uma forma que garanta a segurança da navegação, porém tenha a menor quantidade de alterações possíveis para que não haja prejuízo na tarefa (Tang, 2022).

#### 5.1.4. Movimentação da embarcação

A modelagem é importante para que se valide e desenvolvam controladores capazes de executar a ação desejada de forma precisa. No ambiente marítimo, isso pode ser de elevada complexidade não-linear e de complicada identificação (Wang et al., 2022).

Ainda, um modelo simples acurado do movimento de uma embarcação pode ser definido como:

$$(m + m_x)\dot{u} - (m + m_y)vr - x_G m r^2 = X_H + X_P + X_R, \quad (30)$$

$$(m + m_y)\dot{v} + (m + m_x)ur + x_G m \dot{r} = Y_H + Y_P + Y_R, \quad (31)$$

$$(I_Z + x_G^2 m + J_Z)\dot{r} + X_G m(\dot{v} + ur) = N_H + N_P + N_R. \quad (32)$$

Onde H,R e P representam o casco, o leme e o propulsor, respectivamente.  $m$ ,  $m_x$  e  $m_y$  são a massa do navio, massa adicionada no eixo x e massa adicionada no eixo y.  $u$  e  $v$  são a velocidade longitudinal e lateral a meio navio.  $I_z$  e  $J_z$  são a inércia e inércia adicionada em torno do eixo z.

Ainda, segundo Fossen (2021) o sistema de estados pode ser transformado no seguinte:

$$\dot{x} = u \cos \psi - v \sin \psi, \quad (33)$$

$$\dot{y} = u \sin \psi + v \cos \psi, \quad (34)$$

$$\dot{\psi} = r, \quad (35)$$

$$\dot{u} = \frac{m_v}{m_u} vr - \frac{f_u(\mathbf{v})}{m_u} + \frac{T_u(\cdot)}{m_u} |n|n + d_{wu}, \quad (36)$$

$$\dot{v} = -\frac{m_u}{m_v} ur - \frac{f_v(\mathbf{v})}{m_r} + d_{wv}, \quad (37)$$

$$\dot{r} = \frac{(m_u - m_v)}{m_r} uv - \frac{f_r(\mathbf{v})}{m_r} + \frac{F_r(\cdot)}{m_r} \delta + d_{wr}. \quad (38)$$

Onde:

$$m_u = m - X_{\dot{u}} = m + m_{xx}, \quad (39)$$

$$m_v = m - Y_{\dot{v}} = m + m_{yy}, \quad (40)$$



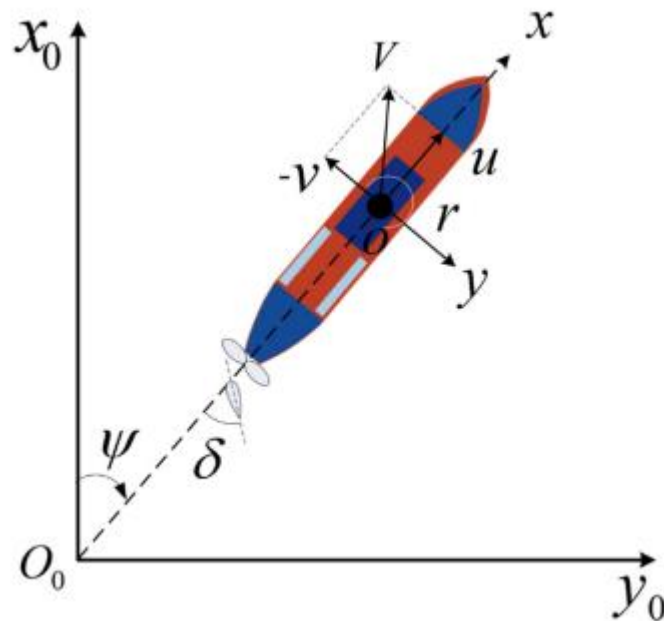
$$m_z = I_{zz} - N_{\dot{r}} = I_{zz} + J_{zz}. \quad (41)$$

Onde  $m_u$  e  $m_v$  são a soma da massa da embarcação e massa adicionada no eixo  $x$  e eixo  $y$ .  $m_r$  é a soma da inércia e inércia adicionada em volta do eixo  $z$ .  $f_u(v)$ ,  $f_r(v)$  e  $f_v(v)$  representam a dinâmica do fluido e  $F_r(\cdot)$  é o momento de guinada, este gerado pela variação do ângulo de leme  $\delta$ .  $d_{wu}$ ,  $d_{wr}$  e  $d_{wv}$  são as perturbações nos eixos.

Este modelo pode ser ainda mais complicado caso desejássemos adicionar o efeito de ondas, correntes, ventos para diferentes distribuições de pesos a bordo.

Com essas equações, conseguimos criar uma modelagem básica para realizar variações com ângulo do leme e variação na rotação do propulsor a ponto de seguirmos um caminho pré-determinado.

Figura 19: Navio e sistema de coordenadas.



Fonte: Wang, 2022.

### 5.1.5. Propulsão

Um navio, quando projetado, passa por diversos cálculos para que seja determinado o Empuxo necessário para a movimentação do peso da embarcação com as correções necessárias pelo número de Froude do modelo Holtrop (1984).

$$T_u(\cdot) = T = K_T D^4 n^2 \rho, \quad (42)$$

$$K_t = k_2 J^2 + k_1 J + k_0, \quad (43)$$

$$J = \frac{u(1-w)}{nD}. \quad (44)$$

Onde,  $K_t$  é o coeficiente de empuxo da embarcação,  $D$  é o diâmetro do propulsor,  $n$  é a rotação do propulsor em Hertz e  $\rho$  é a densidade da água local,  $J$  é o coeficiente de avanço do propulsor e  $w$  é o *wake*, coeficiente de efeito de fluido deixado pelo propulsor.

Em um problema de controle otimizado, o sinal de estado da embarcação, equação não linear da dinâmica do USV representada da forma abaixo, deve ser discretizado.

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}), \quad (45)$$

$$\mathbf{x} = [x, y, \psi, u, v, r]^T \in \mathbb{R}^6, \quad (46)$$

$$\mathbf{u} = [n, \delta]^T \in \mathbb{R}^2. \quad (47)$$

### 5.1.6. Posicionamento

Em uma navegação planejada, uma derrota (*track*) é o caminho previamente traçado pelos encarregados da navegação, de modo a otimizar o tempo ou a missão a ser executada pelo navio.

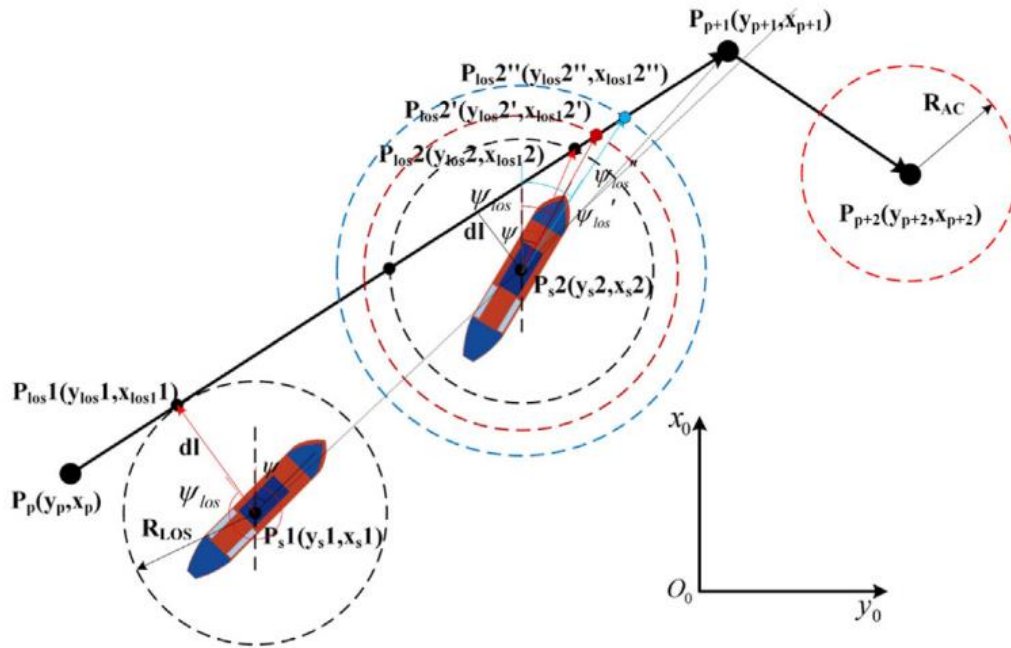
Por natureza, há uma dificuldade inerente da atividade de navegação de manter perfeitamente o navio sobre a derrota traçada. A imprevisível ação dos ventos e do ambiente marítimo faz com que haja a necessidade de correção do navio em relação ao ponto desejado. Navegadores mais experientes conseguem definir, através de cálculos na carta náutica, a partir da posição errônea, um novo rumo a ser adotado para chegar ao destino.

Uma vez que as equações da força dos propulsores e da resultante gerada pela ação do leme no navio, proporciona matematicamente uma solução ao problema mencionado. O sistema deve ser capaz de identificar erros com relação à derrota previamente traçada.

Sistemas de posicionamento de navios apresentaram evolução ao longo dos anos. Novas tecnologias de radar, Electronic Chart Display and Information System (ECDIS) e GPS são apresentadas com frequência e permitem aos navegantes feedback cada vez mais preciso da navegação.

Um controlador consegue ser projetado para imitar um algoritmo correção da navegação. *Line of Sight (LOS) navigation* é um método para simular a visão de marinheiros experientes que buscam o melhor caminho (Fossen apud Wang, 2015). De acordo com o princípio da navegação LOS, se um navio consegue rastrear o ângulo LOS, ele pode rastrear o caminho designado com sucesso.

Figura 20: Princípio da Navegação LOS.



Fonte: Wang, 2022.

É possível observar que existem diversos ângulos  $\psi$  que satisfazem colocar o navio de volta a sua derrota. É interessante treinar o controlador para que a resposta de menor custo seja escolhida. Esta resposta está baseada na distância  $R_{los}$ , dada por:

$$R_{los}(\lambda) = dl(\lambda) + \zeta Lpp. \quad (48)$$

Da figura podemos observar as equações:

$$P_{los} = \begin{cases} \frac{x_{p+1} - x_p}{y_{p+1} - y_p} \\ (y_{los}(\lambda) - y_s(\lambda))^2 + (x_{los}(\lambda) - x_s(\lambda))^2 = R_{AC}^2 \end{cases}, \quad (49)$$

$$\psi_{los} = \arctan \frac{y_{los}(\lambda) - y_s(\lambda)}{x_{los}(\lambda) - x_s(\lambda)}. \quad (50)$$

Quando o navio se aproxima do alvo  $P_{p+1}(y_{p+1}, x_{p+1})$ , o próximo ponto alvo da derrota precisa ser calculado, desta forma evitando delays desnecessários na guinada do navio. O atuador precisará entrar em ação para o ponto seguinte em uma distância  $R_{los}$  que o operador julgar adequada. (Wang ,2022).

Uma vez que a NN está treinada nos dados de trajetória, o modelo não linear do USV pode ser utilizado para prever o estado futuro  $x_{k+1}$  do a partir do estado inicial  $x_k$ . Este output pode ser reinserido no sistema para estimar uma solução  $x_{k+2}$ , sendo possível realizar iterações até quando for desejado.

## 6. CONCLUSÃO

Com o avanço das tecnologias e do poder de processamento dos meios de informática, houve uma facilidade ao acesso à disciplina de Inteligência Artificial, tópico este que vêm mostrando evolução acentuada nos últimos tempos.

Soluções para sistemas extremamente complexos começam a se tornar mais palpáveis. Em diversas áreas da ciência, encontramos uso de Inteligência Artificial e Aprendizagem em Máquinas para melhorar e auxiliar a performance do trabalho humano.

A Marinha do Brasil pode se beneficiar do uso de IA para auxiliar os trabalhos de varredura de minas, batimetria, previsão meteorológica, combate a incêndio, mergulho podendo até mesmo melhorar os sistemas de detecção e *target-tracking*. Este trabalho teve como propósito apresentar uma ideia a uma solução a um dos problemas acima, sendo ainda necessário melhorar a pesquisa na área.

### 6.1. Trabalhos Futuros

Para trabalhos futuros, de forma a manter o assunto de inteligência artificial e os conhecimentos ativos, sugere-se um trabalho dedicado analisando um dos subsistemas apresentados no capítulo 5, utilizando a linguagem de programação *Python* com a bibliotecas apropriadas e efetivamente treinar um modelo e apresentar os resultados.

## REFERÊNCIAS

BRUNTON, Steve; KUTZ, Jose Nathan. **Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control**. 1. ed. [S. l.]: Cambridge University Press, 2019. *E-book*.

Convolutional neural network. 1182204327. [S. l.]: Wikipedia, The Free Encyclopedia., 27 out 2023. Disponível em: [https://en.wikipedia.org/w/index.php?title=Convolutional\\_neural\\_network&oldid=1182204327](https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=1182204327) Acesso em: 05 dez. 2023.

DEV, Prabhat *et al.* Machine learning and its impact on control systems: A review. **Materials Today: Proceedings**, [s. l.], v. 43, p. 3744-3749, 2021. DOI <https://doi.org/10.1016/j.matpr.2021.02.281>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S2214785321013808?via%3Di> hub. Acesso em: 17 set. 2023.

ECONOMIA Azul: o desenvolvimento que vem do mar. [S. l.]: CCSM, 18 jul. 2019. Disponível em: <https://www.marinha.mil.br/economia-azul/noticias/economia-azul-o-desenvolvimento-que-vem-do-mar>. Acesso em: 18 jul. 2023.

ESTRABIS, Thyago. **Desenvolvimento e Implementação de um Controlador Preditivo Generalizado (GPC) para Rastreamento de Referência Rampa**. 2020. 78 p. Dissertação (Mestrado do curso de Engenharia Elétrica) - Universidade Federal de Mato Grosso do Sul, [S. l.], 2020. Disponível em: [https://ppgee.ufms.br/files/2021/04/Dissertação\\_Thyago\\_Estrabis.pdf](https://ppgee.ufms.br/files/2021/04/Dissertação_Thyago_Estrabis.pdf). Acesso em: 19 out. 2023.

Fossen, Thor. (2011). **Handbook of Marine Craft Hydrodynamics and Motion Control**. New Jersey: Wiley.

HEATON, Jeffrey. **Applications of Deep Neural Networks with Keras**. 1. ed. Missouri, EUA: Univesity of St. Louis, Maio 2022. 576 p. ISBN 9798416344269. Disponível em: <https://arxiv.org/pdf/2009.05673.pdf>. Acesso em: 1 out. 2023.

LAVALLE, Steven. **Rapidly-Exploring Random Trees: A New Tool for Path Planning**., Computer Science Department, Iowa State University, 11 out. 1998. Disponível em: <http://msl.cs.uiuc.edu/~lavalle/papers/Lav98c.pdf>. Acesso em: 19 out. 2023.

Filtro Digital. 62168637. [S. l.]: Wikipedia, The Free Encyclopedia., 03 out. 2021. Disponível em: [https://pt.wikipedia.org/w/index.php?title=Filtro\\_digital&oldid=62168637](https://pt.wikipedia.org/w/index.php?title=Filtro_digital&oldid=62168637) Acesso em: 20 out. 2023.

FIGUEIREDO, Luiz. **DESENVOLVIMENTO DE ESTRATÉGIAS DE CONTROLE PARA RASTREAMENTO E ESTABILIZAÇÃO DE SISTEMAS NÃO-HOLONÔMICOS**. 2004. 152 p. Tese (Doutorado em Engenharia Elétrica) -

Universidade Federal de Minas Gerais, Minas Gerais, 2004. Disponível em: <http://hdl.handle.net/1843/BUOS-9UNRNG>. Acesso em: 19 out. 2023.

FOSSSEN, Thor. **Handbook of Marine Craft Hydrodynamics and Motion Control**. 2. ed. New Jersey: Wiley, 2021. 707 p. ISBN 978-1119575054. *E-book*.

FUJII, Yahei; TANAKA, Kenichi. Traffic Capacity. **Journal of Navigation**, Cambridge, v. 24, p. 543-552, outubro 1971. DOI <https://doi.org/10.1017/S0373463300022384>. Disponível em: <https://www.cambridge.org/core/journals/journal-of-navigation/article/abs/traffic-capacity/B43EEF41D5FA4E6D48DF2C98FC3DEB14>. Acesso em: 7 out. 2023.

HOLTROP, J. A statistical re-analysis of resistance and propulsion data. **International shipbuilding progress**, Delft, Países Baixos, v. 31, p. 272-276, 1984. Disponível em: [https://www.academia.edu/42004143/A\\_statistical\\_re\\_analysis\\_of\\_resistance\\_and\\_propulsion\\_data\\_Holtrop\\_1984](https://www.academia.edu/42004143/A_statistical_re_analysis_of_resistance_and_propulsion_data_Holtrop_1984). Acesso em: 13 set. 2023.

IPCOS. Incatools (ed.). 3 PID Tuning methods. *In: 3 PID Tuning methods*. Bostel, Países Baixos, 28 maio 2020. Disponível em: <https://www.incatools.com/pid-tuning/pid-tuning-methods/>. Acesso em: 30 ago. 2023.

MACHINE learning. 1180348934. [S. l.]: Wikipedia, The Free Encyclopedia., 16 out. 2023. Disponível em: [https://en.wikipedia.org/wiki/Machine\\_learning#Approaches](https://en.wikipedia.org/wiki/Machine_learning#Approaches). Acesso em: 20 out. 2023.

MARINHA DO BRASIL. DHN. **Carta de serviços ao usuário**. Niterói, RJ: DHN, 2023. 24 p. Disponível em: <https://www.marinha.mil.br/dhn/sites/www.marinha.mil.br.dhn/files/downloads/Carta-de-Servicos-ao-Usuario-2023.pdf>. Acesso em: 10 out. 2023.

MARITIME SECURITY CHALLENGES, 2021, Victoria, British Columbia, Canada. **The Importance and Applications of Artificial Intelligence to Naval Operations** [...]. [S. l.: s. n.], 2021. Disponível em: <https://mscconference.com/wpp-content/uploads/2021/04/Galdorisi-Tangredi-Naval-Applications-of-AI.pdf>. Acesso em: 6 set. 2023.

MEDEIROS, José Charles. **Um Controlador Robusto Lqr/Lqg E Filtro De Kalman Em Um Manipulador Robótico Via Modelo Genético Neural**. 2017. 116 p. Dissertação (Pós-Graduação em engenharia de computação e sistemas) - Universidade Estadual do Maranhão, [S. l.], 2017. Disponível em: <http://www.pecs.uema.br/wp-content/uploads/2020/05/dissertacao-CHARLES.pdf>. Acesso em: 22 out. 2023.

MOHAN, Shruti. **AI vs Machine Learning vs Deep Learning: Know the Differences**. [S. l.], 23 fev. 2023. Disponível em: <https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/ai-vs-machine-learning-vs-deep-learning>. Acesso em: 20 ago. 2023.

Nyquist-Shannon sampling theorem. 1180085259. [S. l.]: Wikipedia, The Free Encyclopedia., 14 out. 2023. Disponível em: [https://en.wikipedia.org/wiki/Nyquist-Shannon\\_sampling\\_theorem#Critical\\_frequency](https://en.wikipedia.org/wiki/Nyquist-Shannon_sampling_theorem#Critical_frequency). Acesso em: 12 out. 2023.

OGATA, Katsuhiko. **Engenharia de Controle Moderno**. 5. ed. São Paulo: Pearson Education do Brasil, 2014. 822 p. ISBN 978-85-4301-375-6.

OPPENHEIM, Alan; SCHAFER, Ronald. **Discrete-Time Signal Processing**. 3. ed. Essex, England: Pearson Education Limited, 2014. 1042 p.

REDUBICLE. The Fast Fourier Transform (FFT): Most Ingenious Algorithm Ever?. YouTube, 27 de julho de 2023. Disponível em: [https://www.youtube.com/watch?v=h7apO7q16V0&ab\\_channel=Reducible](https://www.youtube.com/watch?v=h7apO7q16V0&ab_channel=Reducible). Acesso em: 28 de julho de 2023.

Reinforcement Learning. 1179499461. [S. l.]: Wikipedia, The Free Encyclopedia., 16 out. 2023. Disponível em: [https://en.wikipedia.org/w/index.php?title=Reinforcement\\_learning&oldid=1179499461](https://en.wikipedia.org/w/index.php?title=Reinforcement_learning&oldid=1179499461) Acesso em: 20 out. 2023.

TANG, Yingjie *et al.* Review of Ship Behavior Characteristics in Mixed Waterborne Traffic. **Journal of Marine Science and Engineering**, [s. l.], v. 10, fevereiro 2022. DOI <https://doi.org/10.3390/jmse10020139>. Disponível em: <https://www.mdpi.com/2077-1312/10/2/139>. Acesso em: 6 jul. 2023.

Work, Robert. Securing Tomorrow. [Entrevista concedida a] David Ignatius from The Washington Post, Washington DC, 30 Mar 2016

WANG, Le *et al.* Data-driven model identification and predictive control for path-following of underactuated ships with unknown dynamics. **International Journal of Naval Architecture and Ocean Engineering**, [s. l.], v. 14, 10 mar. 2022. Disponível em: <https://doi.org/10.1016/j.oceaneng.2020.107062>. Acesso em: 4 set. 2023.

ZACCARIA, Valentina; RAHMAN, Moksadur. A Review of Information Fusion Methods for Gas Turbine Diagnostics. **Sustainability**, [s. l.], v. 11, 6 nov. 2019. DOI <https://doi.org/10.3390/su11226202>. Disponível em: <https://www.mdpi.com/2071-1050/11/22/6202>. Acesso em: 11 out. 2023.

ZACCONE, Raphael. COLREG-Compliant Optimal Path Planning for Real-Time Guidance and Control of Autonomous Ships. **Journal of Marine Science and Engineering**, [s. l.], v. 9, p. 405, 11 abr. 2021. DOI <https://doi.org/10.3390/jmse9040405>. Disponível em: <https://www.mdpi.com/2077-1312/9/4/405>. Acesso em: 11 out. 2023.

ZHENG, Kai *et al.* A SVM based ship collision risk assessment algorithm. **Ocean Engineering**, [s. l.], v. 202, 15 abr. 2020. DOI <https://doi.org/10.1016/j.oceaneng.2020.107062>. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0029801820301359?via%3Di> hub. Acesso em: 7 set. 2023.



ZIEGLER, John; NICHOLS, Nathaniel. Optimum Settings for Automatic Controllers. **Transactions of the A.S.M.E.**, Rochester, NY, v. 64, p. 759-765, 1942. Disponível em:

[https://web.archive.org/web/20170918055307/http://staff.guilan.ac.ir/staff/users/chaibakhsh/fckeditor\\_repo/file/documents/Optimum%20Settings%20for%20Automatic%20Controllers%20\(Ziegler%20and%20Nichols,%201942\).pdf](https://web.archive.org/web/20170918055307/http://staff.guilan.ac.ir/staff/users/chaibakhsh/fckeditor_repo/file/documents/Optimum%20Settings%20for%20Automatic%20Controllers%20(Ziegler%20and%20Nichols,%201942).pdf). Acesso em: 13 set. 2023.