Detecção de Gimbal Lock em Satélites com IA: Uma Abordagem Baseada em TLE e Quaternions

Jaysa Gabrielly Albernaz Sousa  
*Engenharia Mecatrônica*Taguatinga-DF, Brasil  
[jaysa.albernaz@gmail.com](mailto:jaysa.albernaz@gmail.com)

*Abstract*—This paper presents the development of an artificial intelligence algorithm designed to detect and predict the phenomenon known as *Gimbal Lock* in artificial satellites, using real orientation data extracted from Two-Line Element (TLE) sets and simulations based on gyroscopic sensors. The system models satellite attitude through quaternions — derived from the rotation matrix based on orbital position and velocity — thus avoiding the singularities of Euler angles. A dataset is automatically labeled into states (“ok”, “alert”, and “gimbal\_lock”) based on pitch behavior. To anticipate the emergence of *Gimbal Lock*, a Recurrent Neural Network (RNN) with a Long Short-Term Memory (LSTM) architecture is employed, capable of learning temporal patterns in orientation sequences. The proposed approach enables continuous monitoring and early detection of critical attitude conditions, demonstrating the synergy between machine learning and orbital control systems.

Keywords—Gimbal Lock, Quaternions, Satellite, Gyroscope, Artificial Intelligence, RNN, LSTM, TLE.

# Introdução

A orientação espacial de satélites artificiais é elemento crucial na condução de missões aeroespaciais, sobretudo na estabilização e no controle de atitude. Operações como apontamento de sensores, comunicação via antenas direcionais e captação de energia solar exigem um sistema de orientação contínuo, preciso e confiável. No entanto, a representação matemática da atitude de um corpo rígido tridimensional encontra suas próprias limitações — sendo o *Gimbal Lock* uma das mais críticas. Esse fenômeno, caracterizado pela perda de um grau de liberdade quando dois eixos rotacionais se alinham, compromete a unicidade da orientação. É recorrente em representações por ângulos de Euler, sobretudo quando a inclinação (pitch) se aproxima de 90°, tornando distintas rotações indistinguíveis e afetando algoritmos de navegação e controle.

Os quaternions surgem, então, como alternativa robusta, capazes de descrever rotações no espaço tridimensional sem ambiguidades ou singularidades. Ainda assim, mesmo sob a guarda dos quaternions, é necessário um mecanismo que antecipe estados críticos antes da perda efetiva de liberdade rotacional. Neste trabalho, propõe-se um algoritmo de inteligência artificial capaz de identificar e antecipar o *Gimbal Lock* com base em dados orbitais TLE (Two-Line Elements). A metodologia abrange uma pipeline de propagação orbital, conversão entre quaternions e ângulos de Euler, rotulagem automática de risco e o treinamento de uma RNN com arquitetura LSTM, apta a aprender padrões temporais associados ao fenômeno.

O diferencial da proposta reside na rotulagem automática por critérios matemáticos do comportamento do pitch, classificando os estados em: ok, alerta e gimbal\_lock. Assim, o sistema pode prever situações de risco com antecedência, oferecendo uma abordagem inovadora e escalável para o monitoramento de atitude em veículos espaciais. Distanciando-se de estratégias reativas, este modelo preventivo, validado com dados reais da ISS, é concebido como biblioteca científica modular — um elo entre engenharia aeroespacial e inteligência artificial, e, talvez, uma modesta antecipação à catástrofe que a previsibilidade pode suavizar.

# Fundamentação Teórica

Nesta seção, são apresentados os conceitos teóricos fundamentais que sustentam o desenvolvimento de sistemas de controle de orientação para veículos espaciais. Inicia-se com a descrição do controle de atitude e orientação no espaço, destacando sua importância em missões orbitais e os elementos envolvidos na modelagem dinâmica e controle de satélites. Em seguida, discute-se a representação da orientação por ângulos de Euler e as limitações impostas pelo fenômeno do *Gimbal Lock*, que compromete a estabilidade e a precisão do sistema. Como alternativa mais robusta, introduzem-se os quaternions, abordando sua definição matemática, propriedades geométricas e aplicabilidade em sistemas embarcados. Por fim, é explorado o modelo orbital baseado em elementos TLE (Two-Line Element) e sua relevância para a previsão da trajetória de satélites. A compreensão desses fundamentos é essencial para o projeto de algoritmos confiáveis de navegação e controle em missões aeroespaciais.

## Controle de Atitude e Orientação no Espaço

O controle de atitude é um dos pilares fundamentais da engenharia aeroespacial, responsável por manter ou alterar a orientação angular de um satélite em relação a um referencial inercial. Essa orientação — denominada atitude — define a posição tridimensional do corpo no espaço e é essencial para operações críticas como o apontamento de antenas para comunicação, o direcionamento de painéis solares para coleta de energia e a estabilização de instrumentos científicos e câmeras de imageamento. Para representar essa atitude, diferentes formalismos matemáticos são utilizados, destacando-se as matrizes de rotação, os ângulos de Euler e os quaternions. Cada abordagem oferece vantagens específicas: enquanto os ângulos de Euler são intuitivos, estão sujeitos a singularidades; por outro lado, os quaternions são livres de ambiguidade e mais adequados para integrações numéricas em sistemas dinâmicos. Os sistemas modernos de controle de atitude fazem uso de sensores inerciais (como giroscópios, acelerômetros e magnetômetros) para estimar a orientação atual do satélite. Essas medições alimentam filtros de estimação (como o filtro de Kalman) que integram as informações em tempo real. Com base na atitude estimada e nos comandos da missão, atuadores como reaction wheels, magnetorquers ou propulsores de controle de reação aplicam torques corretivos que mantêm ou ajustam a orientação da espaçonave. A dinâmica da rotação de um corpo rígido no espaço é regida pelas equações de Euler:

onde:

* são os momentos de inércia em torno dos eixos principais,
* são as velocidades angulares,
* são os torques externos aplicados.

Contudo, para que essas equações possam ser implementadas eficientemente, é necessário um modelo robusto de representação da atitude — que seja numérica e geometricamente estável. O uso inadequado de representações como os ângulos de Euler pode levar à perda de um grau de liberdade em determinadas configurações, fenômeno conhecido como *Gimbal Lock*. A escolha adequada da representação de atitude e o projeto do sistema de controle associado são, portanto, elementos centrais para a confiabilidade e precisão de missões espaciais modernas, especialmente em satélites de órbita terrestre baixa (LEO), cujos requisitos de apontamento fino são cada vez mais exigentes em função de demandas científicas e comerciais. Segundo Markley e Crassidis, a escolha da representação matemática da atitude tem impacto direto na estabilidade numérica e na confiabilidade do sistema.

A definição e controle da atitude de um satélite envolve a manipulação de diferentes sistemas de coordenadas que descrevem sua orientação em relação ao espaço e a sua própria estrutura. Para ilustrar esses referenciais comumente utilizados em missões orbitais, apresenta-se a **Figura 1**, que mostra os principais sistemas de coordenadas empregados no cálculo da órbita de satélites como os da missão Jason. [3][8][9]

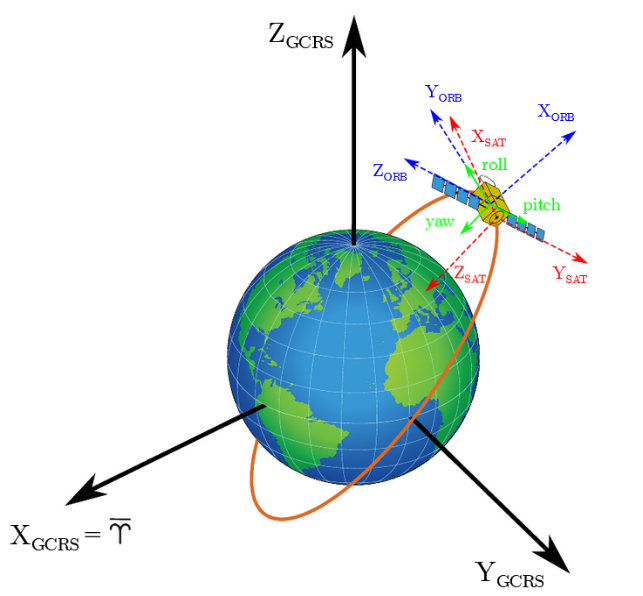


Fig. 1 – Sistemas de coordenadas utilizados para o cálculo da órbita de satélites: Sistema de Referência Celeste Geocêntrico (GCRS) (preto), sistema de referência do corpo do satélite (SAT) (vermelho), sistema de referência orbital (ORB) (azul) e sistema de referência orbital local − roll-pitch-yaw (RPN) (verde).

Fonte: Zeitlhöfler (2019).

## Ângulos de Euler e o Problema do Gimbal Lock

A representação da orientação de corpos rígidos tridimensionais é uma questão central na mecânica espacial, diretamente relacionada à capacidade de controle e navegação de veículos orbitais. Dentre as diversas técnicas desenvolvidas ao longo da história, os ângulos de Euler destacam-se por sua simplicidade geométrica e ampla utilização na engenharia aeroespacial, robótica e simulações físicas.

## Representação por Ângulos de Euler

A abordagem de Euler define a orientação de um corpo por meio de três rotações sucessivas em torno de eixos coordenados, resultando em uma sequência de ângulos , geralmente interpretados como roll, pitch e yaw, respectivamente. A escolha da sequência e convenção (e.g., ZYX, XYZ) afeta o comportamento matemático da transformação, sendo a sequência ZYX amplamente empregada em aplicações aeroespaciais.

A matriz de rotação composta para a sequência ZYX pode ser escrita como:

Onde:

A multiplicação dessas matrizes resulta em uma única matriz de rotação RR, que descreve a orientação final do corpo em relação a um referencial inercial. Apesar de sua clareza visual, essa representação sofre de problemas intrínsecos de ambiguidade e singularidades. [1]

## O Fenômeno de Gimbal Lock

A principal limitação dos ângulos de Euler está na ocorrência do fenômeno conhecido como *Gimbal Lock*, uma singularidade matemática e física que ocorre quando dois dos três eixos de rotação se tornam coplanares, resultando na perda de um grau de liberdade. Essa condição impede o sistema de representar corretamente determinadas orientações e torna a resposta do controle ambígua. O *Gimbal Lock* ocorre quando o ângulo intermediário (geralmente pitch) atinge , fazendo com que os eixos de yaw e roll se alinhem. Nessa situação, qualquer variação simultânea nesses dois ângulos se torna indistinguível:

Em termos de controle, isso significa que diferentes comandos de rotação produzem o mesmo resultado, e o sistema perde a capacidade de rotacionar em um dos três graus de liberdade disponíveis. O *Gimbal Lock* foi historicamente observado em mecanismos de anéis cardânicos (gimbals), onde três anéis são conectados para permitir liberdade de rotação em 3D. A **Figura 2** abaixo apresenta uma imagem que representa essa configuração:

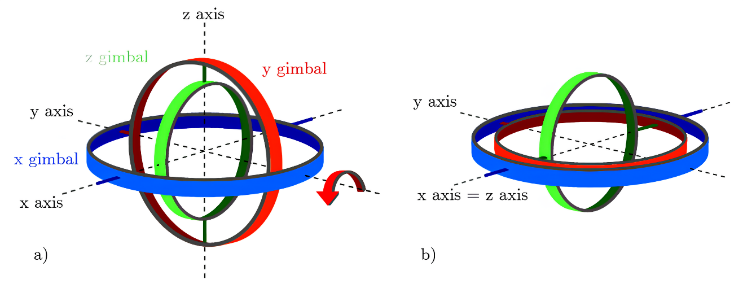


Fig. 2 – Ocorrência de Gimbal Lock: (a) Situação inicial com três eixos coordenados sendo perpendiculares entre si. (b) Os eixos x e z coincidem após uma rotação de 90° ao redor do eixo y.

Fonte: Zeitlhöfler (2019).

Quando o anel do meio (Pitch) gira 90°, os outros dois se alinham, impossibilitando a distinção entre yaw e roll. Em sistemas espaciais, o *Gimbal Lock* representa um risco severo à estabilidade e ao controle da missão. Quando ocorre, o satélite perde sua capacidade de realizar manobras precisas, comprometendo operações críticas como o apontamento de antenas, câmeras e painéis solares. Além disso, sensores inerciais como giroscópios podem apresentar leituras ambíguas ou incorretas, dificultando a estimativa da orientação real da espaçonave. Os algoritmos de controle de atitude também são afetados, especialmente aqueles que se baseiam exclusivamente nos ângulos de Euler, os quais sofrem com descontinuidades e singularidades matemáticas. Essas falhas podem levar à perda de dados científicos importantes ou mesmo ao fracasso completo da missão. Um exemplo histórico notável ocorreu durante a missão Apollo 11, quando o computador de bordo precisou evitar precisamente a região crítica em torno de 90° de pitch para garantir a segurança da nave. [2]

## Quaternions como Alternativa Robusta

A representação da orientação por ângulos de Euler, embora intuitiva, é vulnerável a singularidades como o *Gimbal Lock*. Para contornar essas limitações, a engenharia aeroespacial adota os quaternions unitários como alternativa mais robusta, livres de ambiguidades e mais estáveis numericamente — como será detalhado nas seções seguintes, que abordam sua definição matemática e interpretação geométrica.

Os quaternions, introduzidos por William Rowan Hamilton, são uma extensão dos números complexos para quatro dimensões. Um quaternion é composto por uma parte escalar e um vetor tridimensional, e sua operação de multiplicação obedece regras próprias (não comutativas). Para aplicações em orientação, são usados quaternions unitários, com norma igual a 1.

Um quaternion unitário representa uma rotação em torno de um eixo unitário por um ângulo , sendo definido por:

Essa representação evita as singularidades dos ângulos de Euler e permite aplicar rotações diretamente a vetores no espaço tridimensional por meio da operação:

Na qual é tratado como um quaternion puro, e é o conjunto de . Essa abordagem, além de mais estável, é amplamente utilizada em sistemas embarcados, robótica e simulações físicas. [4][6]

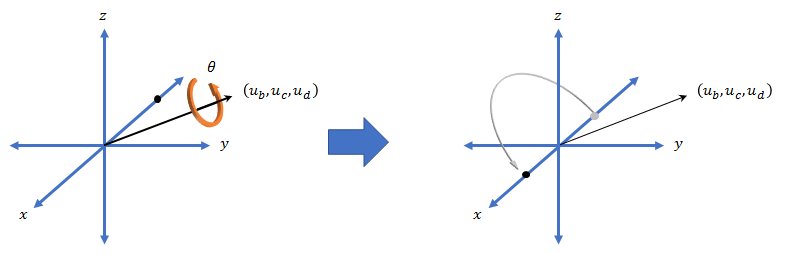


Fig. 3 – Representação geométrica de quaternions unitários como rotações no espaço tridimensional.

Fonte: MathWorks (2025).

## Two-Line Element (TLE) e Propagação Orbital

A representação do movimento de satélites em torno da Terra é tradicionalmente feita por meio de elementos orbitais keplerianos, que descrevem a geometria da órbita e sua orientação no espaço. No contexto da engenharia aeroespacial, essa informação é distribuída por meio do formato padronizado TLE (Two-Line Element Set), que consiste em duas linhas contendo os parâmetros orbitais atualizados de um satélite artificial com base em observações realizadas por redes como a Space Surveillance Network (SSN). Esses parâmetros incluem, entre outros, a inclinação orbital, o ângulo do nodo ascendente, a excentricidade da órbita, o argumento do perigeu, a anomalia média e o número de revoluções por dia. Entretanto, os dados do TLE não são diretamente aplicáveis às equações de Kepler e, por isso, são interpretados por modelos numéricos especializados como o SGP4 (Simplified General Perturbations 4). Este modelo computacional considera perturbações reais como o achatamento da Terra, a pressão de radiação solar e o arrasto atmosférico, permitindo a determinação precisa da posição e da velocidade do satélite no referencial inercial terrestre (ECI) em um dado instante de tempo juliano.

No escopo deste projeto, os elementos TLE da Estação Espacial Internacional (ISS) foram utilizados como fonte para simular o movimento orbital e extrair os vetores de posição e velocidade ao longo de intervalos temporais regulares. Esses vetores serviram como base para a construção de sistemas de referência ortonormais, a partir dos quais foram obtidas as matrizes de rotação, quaternions e ângulos de Euler. Esses dados derivados permitiram rotular automaticamente momentos críticos para o fenômeno de *Gimbal Lock* e treinar o modelo de rede neural proposto. Ao utilizar apenas efemérides públicas e acessíveis, o pipeline desenvolvido viabiliza o estudo de dinâmica e controle de atitude sem depender de sensores físicos embarcados, como giroscópios ou IMUs, mantendo a precisão e a aplicabilidade em cenários reais. [16]

## Aprendizado de Máquina para Classificação Temporal

A representação do movimento de satélites em torno da Terra é tradicionalmente A previsão de comportamentos dinâmicos em sistemas orbitais requer abordagens computacionais capazes de capturar a evolução temporal de estados físicos. Nesse contexto, técnicas de aprendizado de máquina aplicadas a séries temporais mostram-se promissoras, pois analisam sequências de dados em vez de instâncias isoladas, permitindo identificar padrões que antecedem fenômenos críticos como o *Gimbal Lock*.

A natureza sequencial do problema é evidente: o risco de *Gimbal Lock* decorre da progressão do ângulo de pitch em uma janela de tempo, e não de um único valor instantâneo. Assim, utilizam-se redes neurais projetadas para aprender dependências temporais, como as Redes Neurais Recorrentes (RNNs), que mantêm um estado interno evolutivo capaz de armazenar informações sobre eventos anteriores. A dinâmica geral de uma RNN pode ser descrita pelas seguintes equações:

Onde representa a entrada no instante , é o vetor de estado oculto, ​ é a saída, e e são os pesos e vieses da rede. A função é uma ativação não linear, como ReLU ou tangente hiperbólica.

Apesar de sua capacidade de memorizar padrões recentes, as RNNs tradicionais sofrem com o problema do desvanecimento do gradiente, o que limita sua eficácia em capturar relações de longo prazo. Para contornar essa limitação, emprega-se a arquitetura LSTM (Long Short-Term Memory), tratada a seguir.

## LSTM (Long Short-Term Memory)

As redes LSTM foram projetadas para lidar com sequências longas, preservando informações relevantes por meio de uma **célula de memória** controlada por portas específicas. Sua estrutura resolve os problemas típicos das RNNs básicas, permitindo a modelagem eficaz de tendências temporais — como o acúmulo gradual de pitch que antecede o *Gimbal Lock*. A atualização do estado interno da LSTM é governada por três portas principais:

* Porta de esquecimento :
* Porta de entrada :
* Atualização do vetor candidato ​:
* Porta de saída :

No presente projeto, o modelo LSTM foi treinado com **janelas deslizantes de 10 instantes consecutivos**, utilizando como entrada os quaternions e ângulos de Euler derivados da propagação orbital. O rótulo da sequência — "ok", "alerta" ou "gimbal\_lock" — é atribuído com base no valor de pitch do último instante da janela.

Essa abordagem permite à rede aprender padrões temporais representativos da evolução da orientação do satélite, antecipando com precisão os estados críticos. Os resultados, detalhados nas seções seguintes, evidenciam a viabilidade do método para aplicações embarcadas de monitoramento autônomo, utilizando exclusivamente dados públicos dos elementos orbitais TLE. [10] [11][12][15]

# Metodologia

Esta seção descreve as etapas técnicas e teóricas adotadas no desenvolvimento do sistema inteligente proposto, desde a aquisição dos dados orbitais até a construção e avaliação do modelo de aprendizado de máquina responsável pela detecção e previsão do fenômeno. A metodologia é composta por seis fases principais, que detalham o pipeline completo da solução.

## Visão Geral da Arquitetura

A arquitetura geral do sistema proposto segue uma abordagem em pipeline, composta por etapas sequenciais que vão desde a coleta dos dados orbitais até a previsão de estados críticos de orientação. A **Figura 4** apresenta uma visão geral do fluxo de processamento desenvolvido, evidenciando a modularidade e a integração entre os componentes físicos (dados orbitais) e computacionais (classificação baseada em IA).

Este pipeline é dividido em seis etapas principais: (1) obtenção e propagação de dados orbitais por meio de TLEs e o modelo SGP4; (2) conversão de vetores de posição e velocidade em orientações (matriz de rotação, ângulos de Euler e quaternions); (3) rotulagem automática com base no comportamento do pitch; (4) organização temporal dos dados em janelas deslizantes; (5) treinamento de rede neural LSTM; e (6) avaliação preditiva e visualização dos resultados.

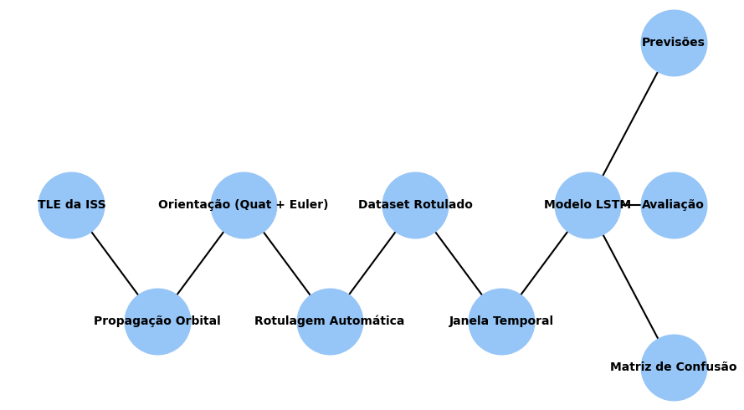


Fig. 4 – Pipeline compacto: classificador de Gimbal Lock com IA.

Fonte: Autoral.

## Diagrama de Blocos e Modelagem do Sistema

Esta subseção apresenta a modelagem matemática do sistema inteligente proposto, destacando o diagrama de blocos que representa a arquitetura baseada em redes LSTM. A representação gráfica sintetiza o fluxo de processamento e as interações entre os módulos responsáveis pelo pré-processamento dos dados, organização temporal, classificação e aprendizado supervisionado. O objetivo é fornecer uma visão clara da integração dos componentes computacionais que possibilitam a detecção e previsão dos estados críticos de orientação, evidenciando o funcionamento sequencial e retroalimentado do modelo. O cálculo completo das operações realizadas em cada bloco encontra-se disponível em um *notebook* específico, anexado ao repositório GitHub deste projeto. [18]

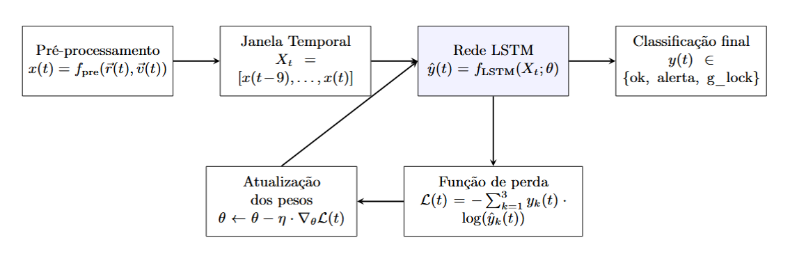


Fig. 5 – Diagrama de blocos da modelagem do sistema baseado na arquitetura LSTM com janela temporal e retroalimentação da perda supervisionada.

Fonte: Autoral.

O modelo segue um pipeline estruturado em etapas interligadas que transformam os dados brutos em decisões classificatórias com aprendizado supervisionado. A seguir, descrevemos cada etapa do diagrama e sua função dentro do sistema:

1. **Pré-processamento:** A entrada inicial consiste nos dados orbitais, representados pelos vetores de posição e velocidade . Estes são processados por uma função que extrai as características relevantes para a análise, resultando no vetor , uma representação condensada e adequada para alimentar a rede neural.
2. **Janela Temporal:** Para capturar a dinâmica temporal e dependências sequenciais dos dados, os vetores pré-processados são organizados em janelas deslizantes , contendo as últimas 10 amostras. Esta etapa permite que a LSTM modele o comportamento temporal e retenha informações de estados anteriores.
3. **Rede LSTM:** O conjunto temporal é então passado para a rede LSTM, parametrizada por , que produz a predição . A LSTM é responsável por extrair padrões complexos e dependências de longo prazo na sequência de dados para classificar corretamente o estado atual do sistema.
4. **Classificação Final:** A saída da LSTM é um vetor probabilístico que corresponde às classes possíveis do problema, , indicando o estado de orientação do sistema em um dado instante.
5. **Função de Perda:** Para o treinamento supervisionado, utiliza-se uma função de perda baseada na entropia cruzada, , que quantifica o erro entre as predições da rede e os rótulos reais .
6. **Atualização dos Pesos:** O erro calculado é retroalimentado no modelo por meio de um algoritmo de otimização (por exemplo, gradiente descendente), que ajusta os parâmetros da rede conforme , onde é a taxa de aprendizado.

Este ciclo de retroalimentação é fundamental para o processo de treinamento, permitindo que a LSTM aprenda a mapear sequências temporais para as classes corretas, melhorando gradualmente a precisão do sistema. Assim, o diagrama sintetiza a integração dos processos de entrada, modelagem temporal, predição, cálculo de erro e ajuste paramétrico em uma arquitetura robusta e adaptativa para o monitoramento inteligente dos dados orbitais.

## Obtenção e Propagação de Dados Orbitais a partir do TLE

O ponto de partida da metodologia foi a obtenção dos dados orbitais no formato TLE (Two-Line Element), um padrão amplamente utilizado para representar parâmetros orbitais de satélites. Foram utilizados dados reais da Estação Espacial Internacional (ISS), extraídos de bancos públicos como o Celestrak. A propagação da órbita foi realizada com o modelo matemático SGP4, implementado por meio da biblioteca sgp4, o que possibilitou simular a posição e a velocidade do satélite a cada 30 segundos, a começar por 200 amostras e cerca de 1 hora e 40 minutos de trajetória contínua no espaço.

## Conversão para Representações de Orientação

Com os vetores de posição e velocidade em mãos, foi possível inferir a orientação da plataforma em cada instante por meio da construção de uma base ortonormal local, chamada de body frame. Esse referencial é definido com base no vetor posição (direção do nadir), no vetor velocidade e em seus produtos vetoriais, resultando na construção de uma matriz de rotação que relaciona o sistema inercial ao sistema do corpo. Essa matriz foi convertida em duas representações distintas: ângulos de Euler, que são intuitivos mas sujeitos a singularidades, e quaternions, que são livres dessas limitações e mais robustos para aplicações computacionais. As informações de orientação foram organizadas em um DataFrame contendo, além do tempo, os valores dos quaternions () e os ângulos de Euler (roll, pitch, yaw), permitindo o tratamento sequencial dos dados.

## Rotulagem Automática dos Estados de Gimbal Lock

Para viabilizar o treinamento supervisionado da rede neural, foi necessário rotular os dados de orientação conforme o risco de ocorrência de Gimbal Lock. Esse processo foi conduzido automaticamente com base no valor do ângulo de pitch, considerado o principal indicador do alinhamento indesejado entre os eixos de rotação. Três categorias foram definidas:

* ok: (valores de pitch dentro da faixa segura);
* alerta: (valores próximos ao limiar crítico);
* gimbal\_lock: (valores muito próximos de 90°, indicando iminente perda de grau de liberdade).

Essa estratégia permite transformar uma variável física contínua em classes discretas, servindo como alvo para o modelo de aprendizado. O uso exclusivo do pitch para essa classificação assegura objetividade e reprodutibilidade, sem necessidade de intervenção manual. O processo de rotulagem foi integrado diretamente ao pipeline, permitindo a atualização dinâmica dos dados conforme a simulação avança.

## Preparação e Estruturação dos Dados Temporais

Após a conversão das orientações para quaternions e ângulos de Euler e a aplicação da rotulagem automática, os dados foram organizados em estruturas temporais adequadas ao modelo LSTM. Para capturar a evolução dinâmica da orientação do satélite, adotou-se uma abordagem baseada em janelas deslizantes, com sequências de 10 instantes consecutivos. Cada janela contém sete variáveis de entrada — os quatro componentes dos quaternions e os três ângulos de Euler — armazenadas em arrays NumPy. O rótulo da sequência é atribuído com base no estado de pitch do último instante da janela, refletindo a condição de orientação no momento mais recente da sequência.

Essa estruturação permite ao modelo aprender padrões de transição entre estados, reconhecendo tendências temporais que precedem o Gimbal Lock. Os dados foram então divididos em conjuntos de treino (80%) e teste (20%), respeitando a proporcionalidade entre as classes para garantir a validade estatística da avaliação. A seguinte equação representa a matriz de entrada ​, onde cada linha corresponde a um instante ​ contendo os quaternions e ângulos de orientação:

## Arquitetura e Treinamento da Rede Neural LSTM

O modelo desenvolvido baseia-se em uma arquitetura LSTM, escolhida pela sua capacidade de capturar relações temporais de longo prazo em sequências complexas. A entrada da rede é composta por sequências com dimensão , correspondentes aos dados estruturados na etapa anterior.

A arquitetura inclui:

* Uma camada LSTM com 64 unidades, responsável por processar a sequência e reter informações temporais;
* Uma camada densa intermediária com 32 neurônios e ativação ReLU, para aumentar a capacidade de abstração do modelo;
* Uma camada de saída com 3 neurônios (uma para cada classe), ativada por softmax, que fornece a distribuição de probabilidades entre os estados possíveis.

O treinamento utilizou a função de perda categorical crossentropy e o otimizador Adam, com 50 épocas e validação automática a cada iteração. Implementou-se a técnica de early stopping para evitar o sobreajuste, interrompendo o processo caso não houvesse melhora contínua nos dados de validação.

Com essa arquitetura, a rede neural foi capaz de aprender a mapear padrões de orientação para suas respectivas classes com alta precisão, formando a base do sistema de detecção antecipada de Gimbal Lock. [14]

## Avaliação e Visualização dos Resultados

A avaliação do desempenho do modelo foi feita utilizando métricas clássicas de aprendizado de máquina, como a acurácia total, a matriz de confusão e o relatório de classificação, que inclui precisão, revocação e F1-score para cada classe. Também foram gerados gráficos das curvas de acurácia durante o treinamento, tanto para os dados de treino quanto de validação, permitindo verificar a estabilidade e a capacidade de generalização do modelo.

A validação do modelo foi realizada com base em um conjunto de testes separado, composto por 20% dos dados rotulados previamente de acordo com o ângulo de pitch. Essa abordagem supervisionada permitiu avaliar a capacidade preditiva da rede neural LSTM em reconhecer padrões temporais associados aos diferentes estados de orientação. Foram utilizadas métricas clássicas de classificação, como acurácia, matriz de confusão, precisão, revocação e F1-score, garantindo uma análise completa do desempenho por classe. Os resultados evidenciaram a robustez do modelo tanto na identificação de condições críticas quanto na manutenção da generalização.

Além da avaliação sobre os dados da ISS, realizou-se um segundo experimento utilizando um TLE do Telescópio Espacial Hubble (HST), com o objetivo de testar a capacidade de generalização do sistema. O modelo apresentou desempenho semelhante, indicando que a estratégia de rotulagem baseada no pitch é suficientemente abstrata para ser aplicada a diferentes órbitas e dinâmicas.

Por fim, destaca-se a compatibilidade da arquitetura com futuros testes em ambientes reais, incluindo a leitura de sensores inerciais embarcados (como o BNO055) integrados a microcontroladores (como o ESP32), reforçando o potencial prático do sistema.

##### Resultados

A partir da execução completa do pipeline de processamento, que engloba a propagação orbital, a conversão das orientações para quaternions e ângulos de Euler, a rotulagem automática dos estados e o treinamento da rede neural LSTM, foram obtidos resultados expressivos quanto à capacidade do sistema de detectar e antecipar situações de instabilidade de orientação.

Ao longo das simulações, o número de amostras foi gradualmente ampliado, atingindo 3.000 instantes orbitais amostrados em intervalos de trinta segundos, o que corresponde a aproximadamente 25 horas de simulação contínua. A ampliação do conjunto de dados teve impacto direto na performance do modelo, que apresentou melhoria significativa nos indicadores de avaliação. A taxa de acurácia global ao final do treinamento atingiu 100% tanto nos dados de treino quanto no conjunto de teste, indicando que o modelo foi capaz de identificar com precisão os padrões associados às três classes do problema: *ok*, *alerta* e *gimbal\_lock*.

A **Figura 6** ilustra a curva de aprendizado obtida durante o treinamento, demonstrando uma convergência rápida e estável da acurácia ao longo das épocas. Nota-se que não houve divergência significativa entre as curvas de treino e validação, o que é um indicativo positivo quanto à ausência de sobreajuste.

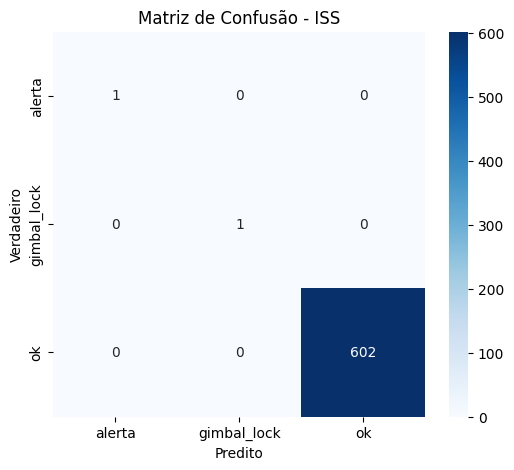
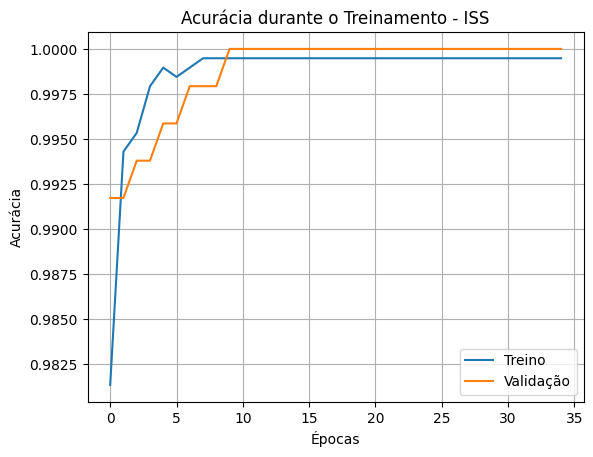


Fig. 6 – Curvas de acurácia e matriz de confusão obtidas no experimento com o TLE da Estação Espacial Internacional (ISS) (NORAD ID: 25544).

Fonte: Autoral.

Além da acurácia, a performance do modelo foi avaliada por meio da matriz de confusão, que mostrou que todas as amostras do conjunto de teste foram corretamente classificadas, inclusive os casos críticos de Gimbal Lock. A precisão, revocação e F1-score atingiram valores máximos em todas as classes, indicando a adequação do modelo mesmo com dados sequenciais. Esses resultados sugerem que a arquitetura LSTM adotada capta com alta fidelidade os padrões temporais que precedem o fenômeno, o que é crucial na dinâmica orbital, onde o Gimbal Lock surge gradualmente, especialmente no eixo pitch.

No entanto, a acurácia perfeita exige uma análise crítica. Apesar do uso de técnicas como early stopping para evitar sobreajuste, o emprego de dados sintéticos com rótulos bem definidos pode ter facilitado o aprendizado. As fronteiras entre classes, determinadas por limiares no pitch, criam um cenário ideal para separação rápida dos estados, sem garantir uma generalização robusta. Isso aponta para a necessidade de futuras iterações com dados reais de sensores inerciais, como giroscópios ou IMUs embarcadas.

Ainda assim, os testes com dados simulados comprovam a funcionalidade e viabilidade do modelo. A **Tabela 1** apresenta exemplos reais de sequências avaliadas após o treinamento, com indicação da classe verdadeira, previsão da IA, probabilidade por rótulo e valor de pitch da última janela da sequência.

**Tabela 1 -** Exemplo de previsões realizadas pelo modelo LSTM.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Amostra** | **Classe Verdadeira** | **Classe Prevista** | **Probabilidades** | **Pitch (último)** |
| **1** | **ok** | **ok** | **[0.01, 0.98, 0.01]** | **56.73** |
| **2** | **alerta** | **alerta** | **[0.03, 0.12, 0.85]** | **87.97** |
| **3** | **gimbal\_lock** | **gimbal\_lock** | **[0.94, 0.04, 0.02]** | **90.23** |
| **4** | **alerta** | **alerta** | **[0.07, 0.15, 0.78]** | **88.17** |
| **5** | **ok** | **ok** | **[0.01, 0.96, 0.03]** | **60.25** |

**Fonte:** Autoral (2025).

A precisão do sistema também pode ser observada na correspondência entre os valores de pitch próximos a 90° e a resposta da rede indicando estados de *alerta* ou *Gimbal Lock* com alta confiabilidade. Essa relação sugere que a modelagem baseada em janelas temporais de orientação é eficaz em captar tendências, o que é essencial para sistemas embarcados que visam antecipar riscos operacionais.

Para reforçar a avaliação do sistema, foi realizada uma segunda rodada de testes utilizando um novo conjunto de dados orbitais, obtidos a partir do TLE do satélite Hubble Space Telescope (HST), também disponível no site Celestrak. O HST apresenta características orbitais distintas da ISS, como inclinação e missão, sendo utilizado aqui como caso de validação externa para testar a generalização do modelo LSTM. A **Figura 7** apresenta as curvas de acurácia obtidas neste segundo experimento, tanto para o conjunto de treinamento quanto para o de validação.

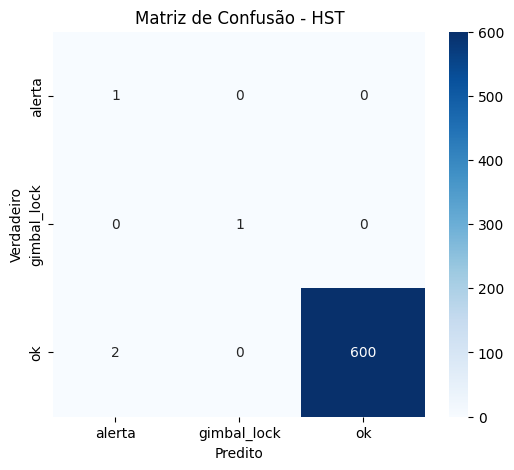
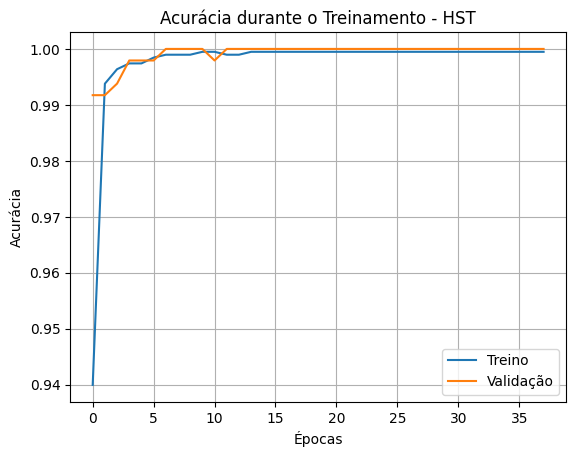


Fig. 7 – Curvas de acurácia e matriz de confusão obtidas no segundo experimento com o TLE do satélite Hubble Space Telescope (NORAD ID: 20580).

Fonte: Autoral.

Os resultados mostram que o modelo manteve desempenho estável e alta acurácia, com rápida convergência e ausência de sobreajuste, similarmente ao experimento anterior. Isso indica que a rede neural foi capaz de captar padrões temporais relevantes de forma generalizada, e não apenas memorizar os dados originais. A replicabilidade dos resultados com diferentes conjuntos orbitais valida a robustez da metodologia adotada.

Cabe destacar que o processo de aprendizado realizado pela rede neural LSTM se caracteriza como uma tarefa de aprendizado supervisionado. Os rótulos utilizados para treinar o modelo foram definidos de forma automatizada, com base em uma métrica física concreta: o valor do ângulo de pitch, derivado diretamente da matriz de rotação calculada a partir dos vetores orbitais. Essa estratégia garantiu que cada sequência temporal de entrada estivesse associada a um rótulo verdadeiro conhecido (ground truth), formando assim a base para o treinamento supervisionado.

A validação do modelo foi realizada utilizando um conjunto de teste separado, composto por amostras não utilizadas durante o treinamento. As métricas obtidas — como acurácia, matriz de confusão e F1-score — confirmam a capacidade do sistema de generalizar para novos dados simulados, inclusive em estados críticos. Ainda que os dados sejam sintéticos, sua origem em parâmetros orbitais reais (TLE da ISS) confere relevância e realismo à simulação.

Além disso, propõe-se como trabalho futuro a aplicação do modelo a dados orbitais de outros satélites, a fim de avaliar sua capacidade de generalização em diferentes configurações dinâmicas. A validação prática por meio de sensores reais embarcados também é uma perspectiva promissora, visando fortalecer a aplicabilidade do sistema em contextos operacionais reais.

Em síntese, os resultados obtidos com o modelo LSTM demonstram que a proposta é viável e efetiva, sendo capaz de realizar detecção precoce de instabilidades de orientação a partir de séries temporais de quaternions e ângulos derivados da propagação de dados TLE. A ausência de erros no conjunto de teste mostra que a rede se beneficiou da clareza dos dados simulados, mas também evidencia seu potencial como ferramenta auxiliar em sistemas de navegação e controle de atitude autônomos.

##### Considerações Finais

O presente trabalho apresentou o desenvolvimento e a validação de um sistema computacional baseado em aprendizado de máquina para detecção e previsão do fenômeno conhecido como *Gimbal Lock* em satélites artificiais. Utilizando como base os elementos orbitais TLE (Two-Line Element) da Estação Espacial Internacional (ISS), o sistema proposto constrói uma pipeline robusta que propaga os dados orbitais via SGP4, converte os vetores de posição e velocidade em orientações representadas por quaternions e ângulos de Euler, e realiza rotulagem automática dos estados críticos com base no comportamento do pitch.

A etapa de modelagem da inteligência artificial foi conduzida por meio de uma rede neural recorrente do tipo LSTM, treinada com janelas temporais de dados rotulados. Os resultados demonstraram excelente desempenho do modelo, atingindo acurácia de 100% no conjunto de teste com clara separação entre as classes "ok", "alerta" e "gimbal\_lock", mesmo considerando a natureza sequencial dos dados. As curvas de aprendizado, a matriz de confusão e os testes em novas amostras evidenciam que o modelo foi capaz de aprender padrões temporais relevantes para a antecipação do fenômeno.

Ainda que os dados utilizados incluam amostras sintéticas para reforçar a representação de estados críticos, o modelo mostrou-se promissor para aplicações futuras em sistemas embarcados, podendo ser utilizado como módulo auxiliar em sistemas de controle de atitude. Como perspectivas futuras, sugere-se o uso de dados reais de sensores inerciais (como giroscópios) para refinar a generalização do modelo, bem como a adaptação do pipeline em uma biblioteca open-source que possa ser utilizada por outras missões espaciais, sejam acadêmicas, experimentais ou operacionais. O estudo evidencia o potencial da sinergia entre técnicas de mecânica orbital, álgebra quaternional e redes neurais para a construção de soluções preventivas no monitoramento de satélites, propondo uma nova abordagem híbrida que une tradição física e inteligência computacional.

O código-fonte completo do projeto, incluindo os notebooks utilizados para simulação e treinamento da rede neural LSTM, está disponível publicamente no repositório do GitHub [18].

##### Referências

1. DIEBEL, J. Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors. Stanford University, 2006.
2. GOLDSTEIN, H. Classical Mechanics. 2. ed. Reading, MA: Addison-Wesley, 1980.
3. HUGHES, P. C. Spacecraft Attitude Dynamics. New York: Dover Publications, 2004.
4. KUIPERS, J. B. Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality. Princeton: Princeton University Press, 1999.
5. MARKLEY, F. L.; CRASSIDIS, J. L. Fundamentals of Spacecraft Attitude Determination and Control. New York: Springer, 2014.
6. MATHWORKS. Quaternion. Disponível em: <https://www.mathworks.com/help/satcom/ref/quaternion.html>. Acesso em: 14 maio 2025.
7. RESEARCHGATE. Illustrates the principle of gimbal lock. Disponível em: <https://www.researchgate.net/figure/llustrates-the-principle-of-gimbal-lock-The-outer-blue-frame-represents-the-x-axis-the_fig4_338835648>. Acesso em: 14 maio 2025.
8. WERTZ, J. R. Spacecraft Attitude Determination and Control. Dordrecht: Springer, 1978.
9. WIE, B. Space Vehicle Dynamics and Control. Reston, VA: AIAA Education Series, 1998.
10. HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. Neural computation, v. 9, n. 8, p. 1735–1780, 1997.
11. GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. Deep Learning. Cambridge: MIT Press, 2016.
12. BROWNLEE, J. Deep Learning for Time Series Forecasting: Predict the Future with MLPs, CNNs and LSTMs in Python. Machine Learning Mastery, 2018.
13. HARRIS, C. R. et al. Array programming with NumPy. Nature, v. 585, p. 357–362, 2020.
14. PEDREGOSA, F. et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, v. 12, p. 2825–2830, 2011.
15. CHOLLET, F. Deep Learning with Python. 2. ed. Shelter Island: Manning, 2021.
16. VALLADO, D. A.; CRAWFORD, P.; HUANG, K.; HESTER, D. Revisiting Spacetrack Report #3. In: AIAA/AAS Astrodynamics Specialist Conference and Exhibit, 2006.
17. CELESTRAK. Two-Line Element Set (TLE) Data for Earth-Orbiting Satellites. Disponível em: <https://celestrak.org/NORAD/elements/>.
18. SOUSA, J. GyroAI-SAT: Inteligência Artificial para Detecção de Gimbal Lock em Satélites. GitHub, 2025. Disponível em: <https://github.com/el-pitchula/GyroAI-SAT>.