**Safety ArtISt – Método para a Garantia de Segurança Crítica (Safety) de Sistemas com Inteligência Artificial: Desenvolvimento e Análise de Estudos de Caso**

**Relatório Técnico de Pesquisa**

**Protótipo de Sistema de Controle de Frenagem de Veículo por Aprendizado Supervisionado: Relatório Técnico para Estudos de Caso do Projeto de Pesquisa “Safety ArtISt: Um Método para a Garantia de Segurança Crítica de Sistemas com Inteligência Artificial”**

Autor(es): Henrique Lefundes da Silva

Revisor: Antonio Vieira da Silva Neto

Revisão **1** – **28/03/2022**

Protótipo de Sistema de Controle de Frenagem de

Veículo por Aprendizado Supervisionado: Modelo de Relatório Técnico para Estudos de Caso do Projeto de Pesquisa “Safety ArtISt: Um Método para a Garantia de Segurança Crítica de Sistemas com Inteligência Artificial”

[1 Introdução 4](#_Toc99390718)

[1.1 Siglas e Abreviações 4](#_Toc99390719)

[2 Descrição do Sistema do Estudo de Caso 5](#_Toc99390720)

[3 Primeira iteração do Método Safety ArtISt 6](#_Toc99390721)

[3.1 Etapa 1: Especificação de Requisitos e Definição Preliminar da Arquitetura 6](#_Toc99390722)

[3.2 Etapa 2: Seleção e Análise de Bases de Dados de Entrada 6](#_Toc99390723)

[3.3 Etapa 3: Projeto Preliminar da IA 7](#_Toc99390724)

[3.4 Etapa 4: Projeto dos Modelos Base da IA 7](#_Toc99390725)

[3.5 Etapa 5: Treinamento e V&V Preliminar da IA 8](#_Toc99390726)

[3.6 Etapa 6: V&V Avançada da IA 8](#_Toc99390727)

[3.7 Etapa 7: Monitoramento durante Operação e Manutenção 9](#_Toc99390728)

[3.8 Conclusões da Iteração 9](#_Toc99390729)

[4 Enésima iteração do Método Safety ArtISt 10](#_Toc99390730)

[4.1 Etapa 1: Especificação de Requisitos e Definição Preliminar da Arquitetura 10](#_Toc99390731)

[4.2 Etapa 2: Seleção e Análise de Bases de Dados de Entrada 10](#_Toc99390732)

[4.3 Etapa 3: Projeto Preliminar da IA 11](#_Toc99390733)

[4.4 Etapa 4: Projeto dos Modelos Base da IA 11](#_Toc99390734)

[4.5 Etapa 5: Treinamento e V&V Preliminar da IA 12](#_Toc99390735)

[4.6 Etapa 6: V&V Avançada da IA 12](#_Toc99390736)

[4.7 Etapa 7: Monitoramento durante Operação e Manutenção 13](#_Toc99390737)

[4.8 Conclusões da Iteração 13](#_Toc99390738)

[5 Conclusões 14](#_Toc99390739)

[6 Referências Bibliográficas 15](#_Toc99390740)

**CONTROLE DE REVISÕES DO RELATÓRIO**

| **Revisão** | **Data** | **Autor da Revisão** | **Descrição da Revisão** |
| --- | --- | --- | --- |
| 1 | 28/03/2022 | Antonio V. da S. Neto | Versão inicial de modelo para a documentação técnica dos estudos de caso do projeto de pesquisa Safety ArISt |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Introdução

O objetivo desta seção do documento é apresentar uma breve descrição do estudo de caso e da estratégia utilizada para realizá-lo, desde a especificação inicial de requisitos até as etapas de operação e manutenção do protótipo correlato. Deve-se descrever que todo o ciclo de vida do estudo de caso foi realizado aplicando-se o método Safety ArtISt (*Artificial Intelligence Structure*).

## Siglas e Abreviações

As siglas e abreviações utilizadas na elaboração deste documento são apresentadas a seguir.

<Instrução: uma sigla por linha da tabela, ordenando em ordem alfabética. Sempre que uma sigla aparecer no texto, deve ser definida por extenso na primeira ocorrência, apenas.>

|  |  |
| --- | --- |
| **ArtISt** | *Artificial Intelligence Structure* |
| **FPGA** | *Field-Programmable Gate Array* |
| **HDL** | *Hardware Description Language* |
| **HLS** | *High Level Synthesis* |
| **ML** | *Machine Learning* |
| **SVM** | *Support Vector Machine* |
| **VHDL** | *VHSIC Hardware Description Language* |

# Descrição do Sistema do Estudo de Caso

O objetivo deste estudo de caso é implementar um sistema de controle de frenagem de um veículo que utilizando apenas inteligência articial, isto é, sem intervação humana no laço de controle, consiga avaliar e impedir uma possível colisão entre o veículo e o objeto detectado. Devido a complexidade do problema algumas simplificações devem ser feitas, dadas pelas seguintes hipóteses:

1. O veículo move-se em um plano não inclinado;
2. A movimentação ocorre ao longo de um único eixo e em um único sentido.

Por consequência, o controle de frenagem será feito a partir de 3 dados de entrada: (i.) distância medida para o obstáculo à frente, (ii.) velocidade atual do veículo. É evidente que será necessário o uso de dois sensores acoplados ao veículo, um para aferir sua velocidade e outro, na frente, para medir a distância do objeto à frente. Durante o projeto será utilizado apenas processos de aprendizado supervisionado, que, por questões de segurança, precisa ser IA explicável. O diagrama da arquitetura de alto nível do projeto pode ser visto abaixo:



Figura 1 - Arquitetura do Sistema de Controle de Frenagem do Estudo de Caso 4

# Pesquisa Bibliográfica – Descrição de Atividades e Resultados

O objetivo desta seção é apresentar um panorama das pesquisas realizadas para dar subsídios à realização dos estudos de caso, destacando-se o método de pesquisa, os principais resultados obtidos e como eles contribuem com a execução dos estudos de caso de interesse.

## Tema 1 – Bibliotecas de Subsídio ao Estudo de Caso

### Objetivos

O intuito é realizar a pesquisa de possíveis pacotes, bibliotecas ou projetos realizados que empregam qualquer tipo de aprendizado supervisionado, considera-se também alguns algoritmos de ML (Machine Learning) que fogem deste escopo, como redes neurais. Diante todos os requisitos de projeto, determinados algoritmos serão priorizados, tais como árvores de decisão e k-NN (k-Nearest Neighbor), por serem mais facilmente explicáveis.

Entretanto, deve-se considerar também os seguintes algoritmos: SVM (Support Vector Machine), florestas aleatórias e redes neurais especificas, que, em união com os algoritmos supracitados, formam o conjunto dos principais algoritmos de aprendizado supervisionado.

### Principais Resultados Obtidos

Em comparação com linguagens de programação usuais como C++ e Python, é notável a escassez de bibliotecas de ML para FPGAs (Field-Programmable Gate Array). Porém foram encontrados programas escritos em Python que, após escrever o algoritmo desejado, convertem o código para uma linguagem de alto nível, seja ela Verilog, HLS (High Level Synthesis), ou VHDL (VHSIC Hardware Description Language). Alguns projetos que implementam tais algoritmos também foram encontrados, todos os resultados discutidos podem ser acessados em:

1. **LeFlow**  
   Essa ferramenta permite escrever códigos em Python com o uso da biblioteca TensorFlow e, em seguida gera um código sintetizável que pode ser utilizado em um FPGA. Mais detalhes em: (NORONHA; SALEHPOUR; WILTON, [S.d.]).   
   A documentação, download e exemplos estão no repositório da ferramenta: https://github.com/danielholanda/LeFlow
2. **CNN-VHDL**  
   Biblioteca de componentes para redes neurais, implementada em VHDL.  
   Disponível em: https://github.com/SalvatoreBarone/CNN-VHDL
3. **Decision Tree**Esse código é capaz de treinar uma árvore de decisão e transformá-la em código de descrição de hardware, neste caso, em VHDL. O código disponível está programado para detecção de humanos.  
   Disponível em: https://github.com/PUTvision/decision\_tree
4. **k-Nearest Neighbors (KNN) Iris flower classification algorithm in VHDL**Implementação do algoritmo de k-NN integralmente em VHDL, o projeto foi realizado especificamente para classifição de flores utilizando a base de dados Iris Flower.

Disponível em: https://github.com/dgarigali/IrisFlowerClassification\_KNN

1. **HLS for ML**

Pacote em Python que implementa algoritmos de ML em FPGAs, a conversão é realizada através de HLS, que então pode ser implementado em uma FPGA. O programa responsável pela compilação de uma linguagem para outra é o Xilinx Vivado.

Dois exemplos de utilização estão documentados, sendo um utilizado em redes neurais (AARRESTAD *et al.*, 2021) e outro para árvores de decisão (SUMMERS *et al.*, 2020)  
Dispoinível em: https://github.com/fastmachinelearning/hls4ml

1. **SVM to VHDL**Implementação parcial de um projeto que tinha como objetivo escrever uma SVM treinada em VHDL, o projeto foi finalizado, porém os arquivos não estão completamente reunidos.

Disponível em:https://github.com/philtomson/svm2vhdl

### Contribuição da Pesquisa Bibliográfica para o Estudo de Caso

O levamento feito foi de suma importância para o andamento do estudo de caso, não somente para direcionar qual método de implementação deverá ser feito, mas também para oferecer alternativas de projeto. A descoberta de códigos que convertem códigos em linguagem de progração convencional como Python para HDL (Hardware Description Language) facilitaram a criação de tais algoritmos, cabendo ao programador, ao final, realizar as devidas alterações e otimizações do código que descreve o hardware.

# Primeira iteração do Método Safety ArtISt

O objetivo desta seção é apresentar os resultados da primeira iteração do método Safety ArtISt, evidenciando as decisões e os resultados de cada uma de suas etapas, atividades e subatividades.

## Etapa 1: Especificação de Requisitos e Definição Preliminar da Arquitetura

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 1 do método Safety ArtISt:

* Detalhar restrições iniciais ao projeto (exemplo: deve ser feito para FPGA, deve ser feito em C/C++, etc.);
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Ao final:
  + Detalhar os requisitos funcionais e não funcionais (especialmente de segurança) do sistema;
  + Detalhar a arquitetura preliminar do sistema.

## Etapa 2: Seleção e Análise de Bases de Dados de Entrada

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 2 do método Safety ArtISt:

* Detalhar quais dados foram selecionados para o projeto, de onde foram obtidos e, se foram gerados especificamente para o estudo de caso, descrever como essa geração foi realizada;
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Ao final:
  + Detalhar as bases de dados geradas ao final da etapa 2, incluindo sua estrutura e os testes estatísticos realizados para argumentar que tratam-se de versão apropriada para o seguimento do projeto;
  + Rastrear os resultados obtidos nos requisitos de segurança da etapa 1;

## Etapa 3: Projeto Preliminar da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 3 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os blocos funcionais da arquitetura, com seus respectivos componentes básicos de HW e SW. O detalhamento deve prever diagramas de arquitetura e descrições dos blocos e de suas interfaces;
* Detalhar as ferramentas que serão empregadas no projeto e, quando necessário, as configurações que serão adotadas;
* Detalhar e justificar as características de tolerância a falhas previstas no projeto, identificando o porquê de cada mecanismo e como eles se relacionam com os requisitos de segurança do sistema.
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 1.

## Etapa 4: Projeto dos Modelos Base da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 4 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar quais modelos base de bibliotecas de terceiros serão reaproveitados e quais serão desenvolvidos especificamente para o estudo de caso, justificando com base nas decisões globais da etapa 3;
* Detalhar as características dos modelos base, como elas foram escolhidas e por que foram selecionadas;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 1.

## Etapa 5: Treinamento e V&V Preliminar da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 5 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar como foi realizado o treinamento dos modelos de aprendizado supervisionado;
* Detalhar os resultados de cada atividade / subatividade de V&V, identificando os problemas de segurança e categorizando-os como “verdadeiro positivo”, “falso positivo” e “falso negativo” (se introduzido deliberadamente pelo projetista);
* Se houver pelo menos um “verdadeiro positivo”, identificar causa raiz e iniciar nova iteração a partir da etapa da causa raiz da etapa mais inicial do projeto;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 5.

## Etapa 6: V&V Avançada da IA

Esta seção é aplicável apenas se a etapa 5 da iteração não tiver nenhum problema “verdadeiro positivo” ou “falso negativo”. Devem ser apresentadas as seguintes características sobre a aplicação da etapa 6 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os resultados de cada atividade / subatividade de V&V, identificando os problemas de segurança e categorizando-os como “verdadeiro positivo”, “falso positivo” e “falso negativo” (se introduzido deliberadamente pelo projetista);
* Se houver pelo menos um “verdadeiro positivo”, identificar causa raiz e iniciar nova iteração a partir da etapa da causa raiz da etapa mais inicial do projeto;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 6.

## Etapa 7: Monitoramento durante Operação e Manutenção

Esta seção é aplicável apenas se a etapa 6 da iteração não tiver nenhum problema “verdadeiro positivo” ou “falso negativo”. Devem ser apresentadas as seguintes características sobre a aplicação da etapa 6 do método Safety ArtISt:

* Definir qual a estratégia de prototipação do sistema para operação e manutenção;
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os resultados de cada atividade, com as respectivas reentrâncias em etapas pregressas durante a operação ou em eventuais manutenções;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 7.

## Conclusões da Iteração

Apresentar um breve resumo dos principais resultados da iteração (projeto, problemas de segurança e potenciais correções).

# Enésima iteração do Método Safety ArtISt

Esta seção deve ser instanciada tantas vezes quanto necessário sempre que as etapas 5 ou 6 da iteração anterior permitirem identificar problemas de segurança das categorias “verdadeiro positivo” ou “falso negativo”. A seção deve ser iniciada a partir da etapa do método Safety ArtISt rastreada na causa mais pregressa dos problemas constatados.

O objetivo desta seção é apresentar os resultados da enésima iteração do método Safety ArtISt, evidenciando as decisões e os resultados de cada uma de suas etapas, atividades e subatividades.

## Etapa 1: Especificação de Requisitos e Definição Preliminar da Arquitetura

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 1 do método Safety ArtISt:

* Detalhar restrições iniciais ao projeto (exemplo: deve ser feito para FPGA, deve ser feito em C/C++, etc.);
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Ao final:
  + Detalhar os requisitos funcionais e não funcionais (especialmente de segurança) do sistema;
  + Detalhar a arquitetura preliminar do sistema.

## Etapa 2: Seleção e Análise de Bases de Dados de Entrada

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 2 do método Safety ArtISt:

* Detalhar quais dados foram selecionados para o projeto, de onde foram obtidos e, se foram gerados especificamente para o estudo de caso, descrever como essa geração foi realizada;
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Ao final:
  + Detalhar as bases de dados geradas ao final da etapa 2, incluindo sua estrutura e os testes estatísticos realizados para argumentar que tratam-se de versão apropriada para o seguimento do projeto;
  + Rastrear os resultados obtidos nos requisitos de segurança da etapa 1;

## Etapa 3: Projeto Preliminar da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 3 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os blocos funcionais da arquitetura, com seus respectivos componentes básicos de HW e SW. O detalhamento deve prever diagramas de arquitetura e descrições dos blocos e de suas interfaces;
* Detalhar as ferramentas que serão empregadas no projeto e, quando necessário, as configurações que serão adotadas;
* Detalhar e justificar as características de tolerância a falhas previstas no projeto, identificando o porquê de cada mecanismo e como eles se relacionam com os requisitos de segurança do sistema.
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 1.

## Etapa 4: Projeto dos Modelos Base da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 4 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar quais modelos base de bibliotecas de terceiros serão reaproveitados e quais serão desenvolvidos especificamente para o estudo de caso, justificando com base nas decisões globais da etapa 3;
* Detalhar as características dos modelos base, como elas foram escolhidas e por que foram selecionadas;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 1.

## Etapa 5: Treinamento e V&V Preliminar da IA

Nesta seção, devem ser apresentadas as seguintes características sobre a aplicação da etapa 5 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar como foi realizado o treinamento dos modelos de aprendizado supervisionado;
* Detalhar os resultados de cada atividade / subatividade de V&V, identificando os problemas de segurança e categorizando-os como “verdadeiro positivo”, “falso positivo” e “falso negativo” (se introduzido deliberadamente pelo projetista);
* Se houver pelo menos um “verdadeiro positivo”, identificar causa raiz e iniciar nova iteração a partir da etapa da causa raiz da etapa mais inicial do projeto;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 5.

## Etapa 6: V&V Avançada da IA

Esta seção é aplicável apenas se a etapa 5 da iteração não tiver nenhum problema “verdadeiro positivo” ou “falso negativo”. Devem ser apresentadas as seguintes características sobre a aplicação da etapa 6 do método Safety ArtISt:

* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os resultados de cada atividade / subatividade de V&V, identificando os problemas de segurança e categorizando-os como “verdadeiro positivo”, “falso positivo” e “falso negativo” (se introduzido deliberadamente pelo projetista);
* Se houver pelo menos um “verdadeiro positivo”, identificar causa raiz e iniciar nova iteração a partir da etapa da causa raiz da etapa mais inicial do projeto;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 6.

## Etapa 7: Monitoramento durante Operação e Manutenção

Esta seção é aplicável apenas se a etapa 6 da iteração não tiver nenhum problema “verdadeiro positivo” ou “falso negativo”. Devem ser apresentadas as seguintes características sobre a aplicação da etapa 6 do método Safety ArtISt:

* Definir qual a estratégia de prototipação do sistema para operação e manutenção;
* Para cada atividade / subatividade:
  + Avaliar se é aplicável ao desenvolvimento e justificar a decisão;
  + Descrever quais os resultados obtidos e justifica-los.
* Detalhar os resultados de cada atividade, com as respectivas reentrâncias em etapas pregressas durante a operação ou em eventuais manutenções;
* Rastrear os resultados obtidos nos requisitos de segurança da etapa 7.

## Conclusões da Iteração

Apresentar um breve resumo dos principais resultados da iteração (projeto, problemas de segurança e potenciais correções).

# Conclusões

O objetivo desta seção é apresentar as conclusões do estudo de caso, contemplando minimamente os seguintes aspectos:

* Arquitetura e técnicas de tolerância a falhas envolvendo IA;
* Resultado global do estudo de caso (sistema “seguro” ou com problemas insolúveis);
* Velocidade de convergência para uma solução aceitável (em função do número de iterações do método) ou para problema insolúvel;
* Eficácia da etapa 5 em filtrar antecipadamente problemas da etapa 6. É possível antecipá-los, ou há mais chance de “exclusão mútua” entre ambas?;
* Principais problemas, desafios e resultados positivos encontrados durante o estudo de caso;
* Potenciais alterações e ajustes ao método Safety ArtISt para contornar os problemas e desafios verificados.

# Referências Bibliográficas

AARRESTAD, Thea *et al.* Fast convolutional neural networks on FPGAs with hls4ml. 13 jan. 2021. Disponível em: <http://arxiv.org/abs/2101.05108>.

NORONHA, Daniel H; SALEHPOUR, Bahar; WILTON, Steven J E. *LeFlow: Enabling Flexible FPGA High-Level Synthesis of Tensorflow Deep Neural Networks*. . [S.l: s.n.], [S.d.].

SUMMERS, Sioni *et al.* Fast inference of Boosted Decision Trees in FPGAs for particle physics. 5 fev. 2020. Disponível em: <http://arxiv.org/abs/2002.02534>.