

PoKER: a probability of kill estimation rate model for air-to-air missiles using machine learning on stochastic targets

Journal of Defense Modeling and Simulation: Applications, Methodology, Technology
1–21

© The Author(s) 2025
DOI: 10.1177/15485129241309675
journals.sagepub.com/home/dms



Joao PA Dantas¹ , Andre N Costa¹, Diego Geraldo¹, Marcos ROA Maximo², and Takashi Yoneyama³

Abstract

This work introduces PoKER, a novel probabilistic model engineered to optimize missile launch effectiveness in air-to-air scenarios, specifically within Beyond Visual Range (BVR) air combat. Unlike conventional Weapon Engagement Zone (WEZ) models that delineate zones based on static distances such as maximum, minimum, and no-escape ranges, PoKER applies machine learning to predict kill probabilities more accurately by integrating the stochastic behaviors of targets and missile miss distances. This model dynamically evaluates target behavior, greatly expanding the predictive capabilities of engagement analysis. By factoring in elements such as warhead lethality, target and shooter orientations, and the specific conditions of engagement, PoKER provides important insights into engagement dynamics and quantifies success probabilities. Consequently, it can potentially be an important tool for BVR air combat pilots, improving operational decision-making within this specialized combat domain.

Keywords

Air combat, decision support systems, machine learning, military simulation, weapon engagement zone

1. Introduction

Missile systems hold a critical position in contemporary military operations,¹ providing an effective means to engage aerial targets over extensive distances, particularly significant in the context of Beyond Visual Range (BVR) air combat. BVR air combat refers to aerial engagements that occur at distances beyond the visual sight of the pilots, typically over tens of kilometers. In these scenarios, aircraft rely on advanced radar, sensors, and long-range missiles to detect, track, and engage opponents without the need for visual identification.² The evolving landscape of BVR air combat, driven by recent advancements in detection and missile technology, underscores the necessity for sophisticated Weapon Engagement Zone (WEZ) models.³ These models, essential for evaluating the operational efficacy of air-to-air missiles, bring about complex computations of maximum and minimum engagement ranges alongside the identification of No Escape Zones (NEZ).⁴ Notably, these computations are usually dynamically executed in real-time during flight, factoring in the orientations of both the target and the shooter.⁵

WEZs are not simply geometric boundaries determined by orientations and positions; they also take into account the dynamics and maneuverability of both the target and shooter, along with their weapons systems.² The insights derived from WEZ calculations may be relayed to pilots through heads-up or heads-down displays,⁶ which may empower them to make more informed decisions regarding missile launches with increased precision and confidence.⁷

¹Decision Support Systems Subdivision, Institute for Advanced Studies, Brazil

²Computer Science Division, Aeronautics Institute of Technology, Brazil

³Electronic Engineering Division, Aeronautics Institute of Technology, Brazil

Corresponding author:

Joao PA Dantas, Decision Support Systems Subdivision, Institute for Advanced Studies, Trevo Coronel Aviator José Alberto Albano do Amarante, n° 1, Sao Jose dos Campos, Sao Paulo 12.228-001, Brazil.
Email: jpdantas@ita.br

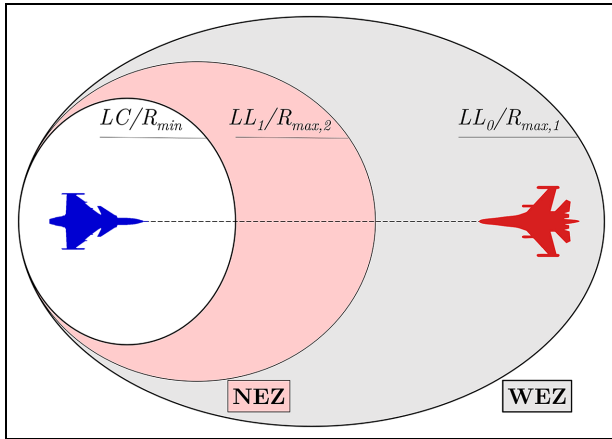


Figure 1. Diagram illustrating aerial combat WEZ, detailing ranges for missile engagement from maximum kinematic reach to minimum arming distance, including the airspace where the target has limited evasion options.

The definition of WEZ can incorporate concepts such as long and short limits or maximum and minimum ranges, whose interpretations might slightly vary between countries.⁸ To exemplify, we present two common nomenclatures, as depicted in Figure 1, to evidence their nuances, mostly with respect to enemy maneuverability. Some armed forces use Long Limit 0 (LL_0) to delineate the furthest kinematic reach of a missile, where it uses up all its energy just to activate its proximity fuse upon reaching the target. Similarly, Maximum Range ($R_{max,1}$) indicates the maximum distance a missile can effectively engage a passive target, with its success probability decreasing if the target maneuvers to increase flight distance. Long Limit 1 (LL_1) specifies the range within which a missile can effectively counteract a target's ultimate evasion efforts with maximum performance. Alternatively, $R_{max,2}$ introduces the beginning of the NEZ, which represents a range within which the missile is expected to hit, barring specific defensive actions by the target, though escape is still possible through engine maneuvers or missile countermeasures. Finally, the Short Limit (LC)/ R_{min} marks the closest distance at which a missile's fuze can arm.

Even though these concepts provide a general idea of where to engage a target, depending on rules of engagement and shot philosophies,^{9,10} these regions may still be quite extreme, not accounting for what happens in between them, which makes it difficult for pilots to choose the best moment to launch a missile in order to maximize the chances of success. This is aggravated by the fact that the opposing aircraft may maneuver unpredictably. Amid these complex considerations, it becomes essential to conduct detailed simulations to improve operational readiness and tactical accuracy, analyzing missile efficacy in diverse

engagement scenarios, which include different adversarial reactions to missile threats. In this research, we introduce a novel WEZ model tailored to improve the assessment of missile performance after launch, specifically in the case of an Active Radar Homing (ARH) weapon system utilized in BVR air combat. Leveraging insights from prior work, we present an R-based simulator based on a 5-degree-of-freedom (5-DOF) model.¹¹

ARH, or Active Radar Homing, refers to a guidance method used in missile systems, where the missile itself is equipped with a radar seeker that actively emits radar signals to detect and track its target independently. This type of homing provides the missile with greater autonomy after launch, reducing the need for continuous guidance from the launching platform and enabling effective engagement of targets in BVR air combat scenarios.

Our simulator encapsulates the complexities of an ARH missile, including its radar seeker functionality and autonomous target-tracking capability, enabling rigorous analysis and prediction of its performance across various operational parameters. By exploring the nuances of missile-target interactions with an active radar seeker, our simulator provides valuable insights into BVR air combat dynamics, aiding in the formulation of strategic and tactical decisions.

Furthermore, we explore the domain of warhead lethality, offering insights into the calculation of the proximity-fuzed warhead's conditional probability of target destruction, a critical component in assessing missile effectiveness. In addition, we focus on generating and preprocessing simulation data, proposing a supervised machine learning methodology to develop regression models capable of predicting missile kill probabilities. We carefully evaluate various algorithms' efficacy and computational efficiency, illuminating their suitability for real-world applications.

The main contribution of this work is as follows:

Development of the Probability of Kill Estimation Rate (PoKER) model: Designed to provide enhanced situational awareness in dynamic air-to-air combat scenarios, particularly in BVR air combat settings.

The specific contributions that support the development and application of the PoKER model are as follows:

- **Complementation of WEZ with kill probability estimation:** Addressing the gap in traditional weapon engagement analysis by providing an estimation of the probability of kill when launching air-to-air missiles, incorporating real-time conditions of both the launcher and the target for a more comprehensive understanding of engagement success.

- Modeling target decisions: Integration of potential adversary maneuvers using high-fidelity models of aircraft and missiles, allowing for flexibility rather than relying on fixed target profiles and predefined launcher strategies.
- Simulation and machine learning integration: Utilization of well-designed simulation tools and machine learning to deepen the understanding of missile engagement dynamics.
- Warhead lethality model: Inclusion of a warhead lethality model that estimates the probability of kill (P_{kill}) based on missile miss distance and a dynamic model of the opponent's behavior, substantially improving tactical decision-making and enhancing operational capabilities in BVR air combat.

In summary, the primary focus of the PoKER model is to enhance situational awareness in dynamic BVR engagements, marking an initial step toward the development of probabilistic models for weapon engagement analysis. This work offers a novel methodological contribution, advancing beyond traditional WEZ analysis by incorporating real-time probabilistic assessments and accounting for the dynamic interactions between the launcher and the target. Furthermore, the insights gained from this model could be leveraged in the future to support decision-making processes, enabling more informed tactical decisions in complex air combat scenarios.

The remainder of this document is organized as follows. The “Related work” section provides an overview of the literature pertinent to our study and introduces critical terminology. The “Methodology” section delineates our approach, outlining each of its steps and the employed tools and techniques. The “Results and analysis” section showcases and discusses the findings derived from the application of our methodology. The “Conclusion and future work” section summarizes our approach and its outcomes while also highlighting potential avenues for future research.

2. Related work

The success of BVR air combat heavily relies on the seamless integration of offensive and defensive tactics supported by accurate and timely information. Consequently, numerous research efforts have focused on enhancing tactical decision-making through advanced air combat simulations, aiming to improve the effectiveness of engagements in dynamic scenarios.^{12–16} These studies explore various aspects of BVR air combat, including weapon engagement, evasion strategies, and cooperative tactics, all of which contribute to a more comprehensive understanding of air combat dynamics.

2.1. Engagement and escape zone strategies

The current study models predominantly integrate missile information to define Dynamic Escape Zones (DEZ), which refers to the region where the aircraft's survivability can be guaranteed through kinetic evasion maneuvers,¹⁷ and WEZ, which are also referred to as Dynamic Launch Zones,¹⁸ Missile Attack Zones,¹⁹ or Launch Acceptability Regions.²⁰ The primary aim of WEZ models is to aid in decision-making processes concerning weapon launch and guidance support—through uplink updates—by outlining the boundaries and operational effectiveness of air-to-air missiles during target engagement. Two main methods are traditionally used to calculate WEZ for various missile configurations. The first method conducts ongoing flyout simulations of the missile model during the engagement period. The second, more widely used method performs simulated flyouts in an offline setting, creating a lookup table that incorporates the pre-simulated data. Upon employing the WEZ model, interpolation is applied to this data to produce outcomes that are not directly available in the table.²¹

Several techniques have been developed for both collecting the predefined data and for its subsequent interpolation. However, notable advancements have also been made in fields such as post-launch WEZ analysis, cooperative modeling of WEZ, and algorithms for WEZ estimation.^{4,22–24} These studies tailor WEZ concepts to specific combat scenarios and maneuver types, often integrating them with 3-degree-of-freedom (3-DOF) or 6-degree-of-freedom (6-DOF) models.²⁵ Solution methods for WEZ calculations range from offline simulation algorithms to polynomial, interpolation, and neural network fitting algorithms.²⁶

However, the DEZ definition aims to encompass essential elements of tactical decision-making from the perspective of missile avoidance. Techniques such as differential game theory and principles like Minimum Evasive Range²⁷ or NEZ²⁸ play a central role in differentiating between aggressive and defensive tactics, as well as in identifying the best timing for evasive actions.^{29,30} This can be done either by analyzing missile miss distances in one-on-one engagements or by considering multiple incoming threats.³¹

2.2. Cooperative engagement models

While most of the earlier works focused on duel engagements, there has been a shift toward modeling cooperative engagement among multiple aircraft, particularly in scenarios involving Unmanned Combat Aerial Vehicles (UCAVs).¹ Cooperative engagement models extend WEZ and DEZ strategies by introducing coordination between multiple aircraft to enhance engagement effectiveness and

survivability. These models aim to optimize WEZ and NEZ calculations using cooperative strategies, leveraging algorithmic approaches to increase effectiveness in group engagements.³ Furthermore, optimization methodologies for attack positioning and cooperative tactics have been proposed, incorporating offensive and defensive factors for assessing the best maneuvers.^{32–34}

2.3. Probabilistic modeling in BVR air combat

In recent years, there has been growing interest in incorporating probabilistic elements into BVR combat modeling, moving beyond traditional deterministic approaches. Some works have used Bayesian Networks to create probabilistic graphical models representing variables and their conditional dependencies through a directed acyclic graph.³⁵ BVR combat can be divided into four phases: position occupy maneuver, launch maneuver, midcourse guidance, and terminal guidance. Utilizing Dynamic Bayesian Networks (DBN), these studies encapsulate the probabilistic relationships and uncertainties in each phase, predicting the likelihood of a successful missile hit by considering aircraft positions, velocities, maneuvers, missile specifications, and environmental conditions. Integrating expert knowledge and data from air combat simulations creates a temporal representation of the combat scenario. Through DBN analysis, key determinants of missile success probability are identified, and the Dynamic Attack Zone (DAZ) is delineated—representing the region where a missile launch is likely to result in a hit. This probabilistic framework facilitates simulation and tactical analysis, providing insights for optimizing missile deployment and supporting decision-making under uncertainty in BVR engagements.

2.4. Limitations in existing work

While the existing literature explores various dimensions of WEZ determination, a noticeable gap exists regarding incorporating the target's maneuvering capabilities into WEZ creation. Even some works that include probabilities through Bayesian Networks do not show clear variability in the target maneuvering, as the target makes a turn as soon as it detects the threat of the incoming missile.³⁵ Our approach distinguishes itself by incorporating the concept of maneuver probability into the process of determining the WEZ, a feature absent in the studies we reviewed. This innovative method extends beyond only considering the initial dynamics and physical attributes of both the missile and the aircraft. It proactively forecasts the adversary's potential maneuvers, transforming the WEZ from a simple geometric entity into a dynamic probability distribution that reflects the likelihood of the missile's successful engagement with the target. Our model furthermore

enhances this by incorporating the target's behavior in a probabilistic manner and providing a probability of success through studying warhead lethality.

3. Methodology

This section examines the methodology encompassing the missile launch simulator, the acquisition of simulation data, and the preprocessing techniques applied to ensure high-quality data sets. We thoroughly describe the input variables and the steps taken to analyze and undersample the data. We then elucidate the methods used to calculate warhead lethality, which is integral to determining the missile's probability of kill based on its miss distance. The development, training, and evaluation of supervised machine learning models are discussed in depth, concluding with an assessment of their performance. Finally, we explore the estimation of turn degree and delay, key factors in computing the probability of hitting a target with the missile.

3.1. Missile launch simulator

In this research, we have developed an enhanced version of the missile launch simulator, primarily based on our previous work.⁴ Our simulator, implemented in R, models the behavior of a Fox 3 missile, following a 5-DOF framework as outlined in the Missile Handbook.¹¹ This type of missile, as defined in military terminology,³⁶ is an ARH (Active Radar Homing) missile, meaning it is equipped with an autonomous seeker capable of tracking a target after activation at a specific range.⁴ Through this simulator, we are able to analyze and predict missile behavior and performance under a variety of operational conditions, providing valuable insights into missile-target interactions in BVR air combat. Due to confidentiality and sovereignty concerns, the missile model is under restricted access.

3.1.1. Guidance and navigation. The model's core feature is its guidance system, designed to emulate proportional navigation. This ensures the missile maintains an optimal trajectory relative to the moving target, adjusting its course to align precisely according to its guidance law. Furthermore, the simulator incorporates the possibility of a loft maneuver—a sharp, upward trajectory immediately after launch. This maneuver is critical in extending the missile's range and enhancing its probability of intercepting distant targets.⁴

3.1.2. Target interaction. The simulation considers various target behaviors, including both stationary and maneuvering targets. One key aspect of the model is its ability to simulate high-performance maneuvers of the target, such

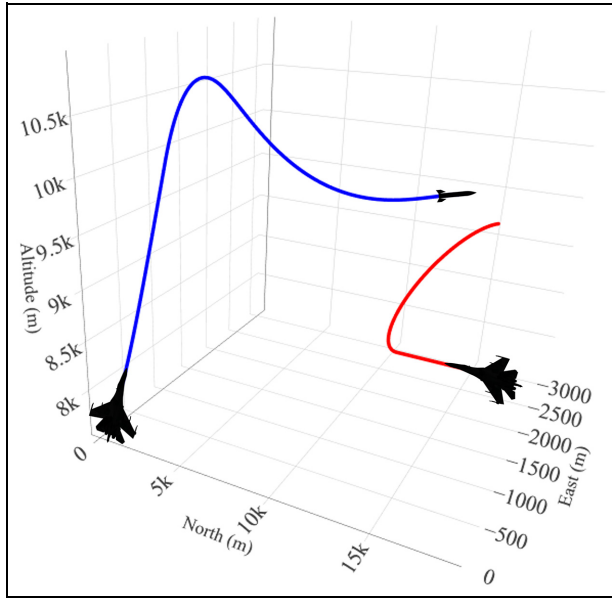


Figure 2. Missile engaging a target performing a +5 G maneuver, illustrating the missile's adaptive trajectory in response to the target's high-G evasive actions.

as the +5 G maneuver depicted in Figure 2. The timing of these maneuvers can be varied, allowing for a comprehensive analysis of missile performance under different engagement scenarios. The data in Figure 3 offer insights into the target's behavior, illustrating sample metrics from the simulation, including velocity, acceleration, altitude, heading, and pitch, and showing how these maneuvers impact the engagement.

3.1.3. Missile trajectory metrics. The simulation generates a comprehensive analysis of missile trajectory metrics. One notable aspect is the missile's mass, which shows a consistent decrease almost linearly during the boost phase. This pattern aligns with the expected behavior of a dual-thrust rocket motor, a critical feature in missile design.^{11,37} In addition, the simulation reveals significant variations in the missile's angle of attack and pitch angle. These variations are especially pronounced during complex maneuvers such as the loft maneuver.³⁸ The data depicted in Figure 4 highlight these dynamics, providing a comprehensive view of the missile's performance metrics throughout the simulation.

3.1.4. Flight dynamics. In the domain of flight dynamics, the simulation includes detailed heading adjustments. These adjustments are fine-tuned to simulate responses to potential evasion tactics employed by a target, thereby evaluating the missile's adaptability in dynamic combat

environments. The model also tracks accelerations along the East and North axes, adhering to the North-East-Down (NED) coordinate system. This aspect of the simulation is instrumental in assessing the missile's response to both loft maneuvers and evasive actions by the target. Another key aspect studied is the velocity profile of the missile. This profile demonstrates a characteristic increase during the boost phase followed by a gradual decrease during the sustain phase.³⁹ Figure 4 also illustrates these aspects, providing additional insights into the missile's adaptability and performance in various dynamic scenarios.

3.1.5. Seeker behavior. The final aspect covered by the simulation is the missile's seeker behavior. The seeker angle is particularly important, as it directly correlates with the missile's guidance system performance. This angle reflects the adjustments made by the missile's seeker in following a proportional navigation path. Analyzing deviations, especially from the off-boresight angle, provides critical insights into the missile's effectiveness, especially in scenarios where the target employs advanced evasive maneuvers.⁴⁰

3.1.6. Visualization and debugging platform. The Aerospace Simulation Environment, or *Ambiente de Simulação Aeroespacial (ASA)* in Portuguese, serves as a critical resource for the Brazilian Air Force, facilitating enhanced missile launch visualization and debugging processes.⁴¹ This comprehensive C++ simulation framework, tailored for Brazilian military applications, supports detailed examinations and visualizations of various military operations, including missile launches. More than a tool for basic training, ASA is key for executing complex, integrated combat scenarios and is essential in testing and validating tactical strategies. It provides a platform where analysts and engineers can perform comprehensive assessments and refine aerospace mechanisms and missions.

The missile model used in this research follows the same structural framework as the model used in ASA. This demonstrates the level of fidelity aimed for in this work, ensuring consistency with established simulation models in ASA. The ASA platform plays an essential role in validating and debugging this missile model, as it incorporates other elements and behaviors into the scenario, providing a more comprehensive environment for testing and refinement.

The advanced functionalities of ASA lead to improved accuracy in mission analysis and bolster reliability in operational execution.⁴² Looking forward, the insights gained from WEZ analyses in this research could be integrated into more complex simulations within ASA, where agents could make decisions based on a variety of

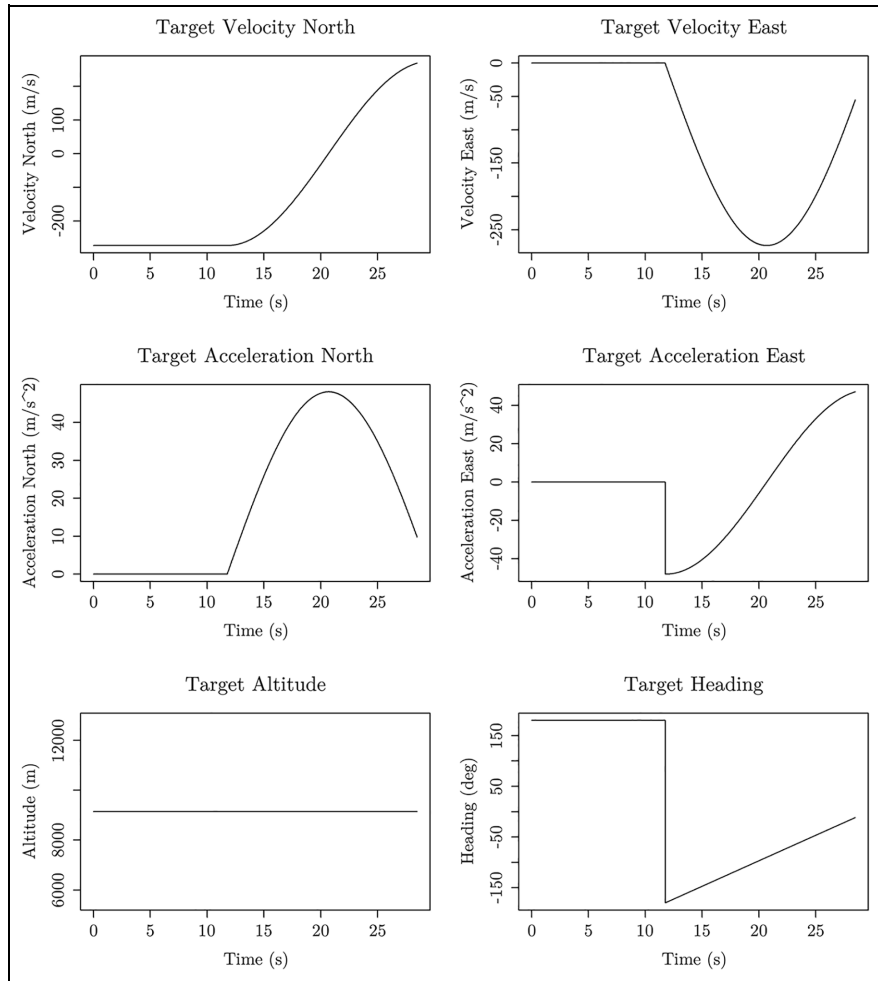


Figure 3. Metrics from the simulation include target velocity (North, East), acceleration (North, East), altitude, and heading over time. Note that in this case, the target maneuver was considered to be leveled, and the target kept a constant altitude while evading the missile, resulting in a flat altitude profile. There is also a delay time in the pilot's reaction, causing all of the charts to present a flat profile for the first few seconds.

available models, thus enhancing the overall operational readiness and tactical efficiency of the aerospace forces.

3.2. Simulation data

Initially, to generate the simulation data, we created simulator input files using Latin Hypercube Sampling (LHS), an efficient statistical method for designing experiments. This method divides the space into a prespecified number of sections and randomly samples one point from each.^{43,44} Unlike factorial designs, LHS offers a superior alternative for populating the sample space more efficiently than purely random methods.^{9,45} We employed LHS using the AsaPy Library,⁴⁶ a custom-made Python library associated with ASA, specifically designed to optimize the analysis of simulation data.

Both the shooter's altitude (alt_sht) and velocity (vel_sht) significantly impact the energy transferred to the launched missile.⁵ The launch altitude (alt_sht) influences the missile's drag, thus affecting flight performance.⁴ The shooter's pitch (pit_sht), which is defined with respect to the North-East plane, is crucial for determining the missile's initial angle post-launch, aiding in the loft maneuver, which is the missile's initial maneuver after launch. The target's velocity (vel_tgt) is significant, as it determines the defensibility of the launched missile. Two aspects are fundamental in terms of the angles between the shooter and the target: heading and off-boresight (Figure 5). The target's heading (hdg_tgt) indicates the direction taken to evade the missile with respect to the North, while the off-boresight angle (rgt_tgt), defined as the angle between the shooter's

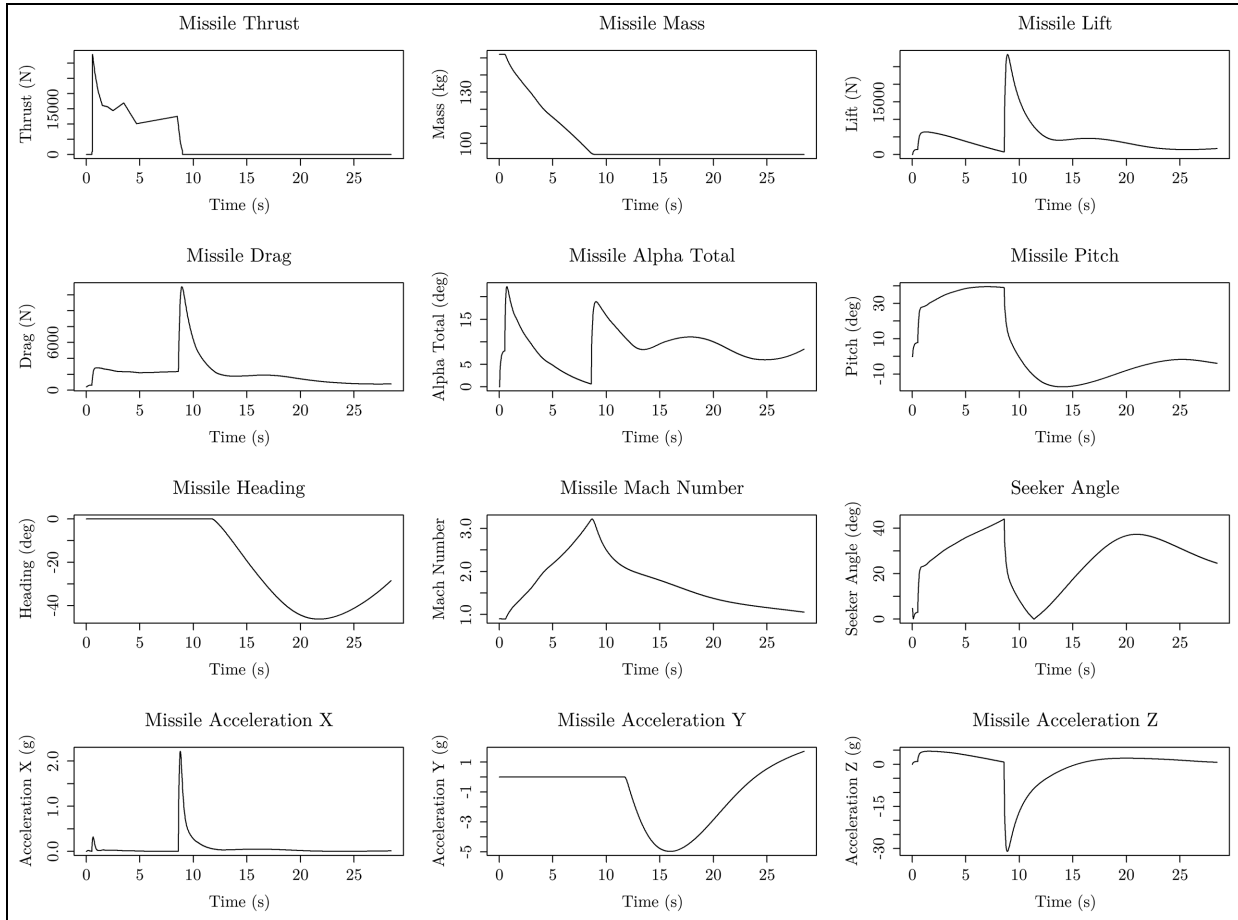


Figure 4. Simulation metrics for the missile, showing thrust, mass, lift, drag, total alpha, pitch, heading, Mach number, seeker angle, and acceleration (X, Y, Z) over time.

nose and the target, along with the target's heading, reveals whether the shooter aircraft is approaching the target and thus the missile.^{47,48} In addition, the distance between the shooter and the target (*dist*) is critical as it determines the missile's flight time and significantly influences the hit probability. The target's response time (*delay*) represents the time before initiating a defensive maneuver. A longer response time may force the target to execute more aggressive maneuvers, which are characterized by a higher turn degree (*turn_dg*) and greater acceleration (*load_factor*). All measurements are taken with respect to the center of mass of either the aircraft or the missile.

The target's response time (*delay*) represents the time before initiating a defensive maneuver. Depending on how long the target delays the response, it may need to perform more aggressive maneuvers, characterized by a higher turn degree (*turn_dg*) and greater acceleration (*load_factor*). All measurements are taken with respect to the center of mass of either aircraft or missile.

Table 1 illustrates the range for each parameter considered during the generation of samples. These parameters were established based on the expertise of military fighter pilots, serving as Subject Matter Experts (SMEs) in military tactics, who identified critical values pertinent to the scenario. This approach aligns with methodologies previously reported in similar studies focusing on input variables within this domain.^{4,21}

Since the LHS method allows the user to specify the desired sample points, we created 10 million different input files. The authors used this work as an opportunity to thoroughly test the simulator and work with a high-fidelity model, considering the large number of variations in input parameters and the potential for non-realistic combinations to arise. In addition, previous works by the authors using the same simulator provided indications of the necessity of a large volume of data to achieve reliable and meaningful results. These 10 million input files generated an equal number of missile launch simulations, which were executed using an Intel Xeon Silver 4210R CPU with

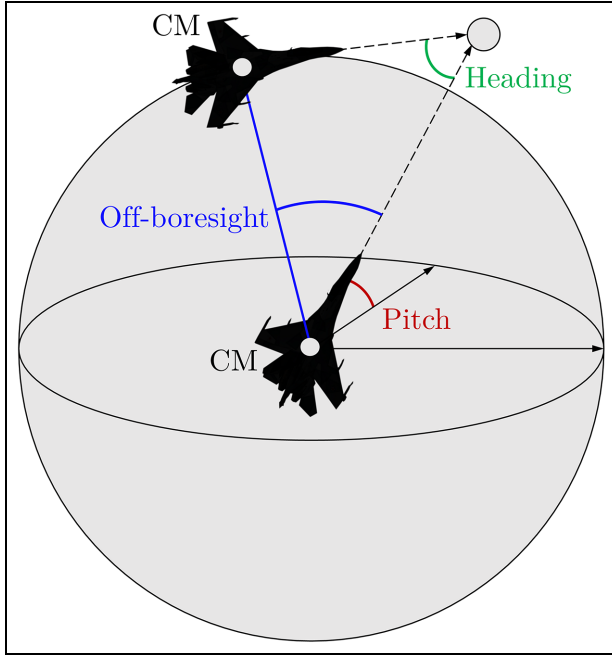


Figure 5. Diagram illustrating the heading, off-boresight, and pitch angles between two aircraft in aerial maneuvering with respect to their centers of mass (CM).

Table 1. Input simulation data with the respective intervals.

Parameter	Min	Max	Unit
Shooter altitude (<code>alt_sht</code>)	1,000	45,000	ft
Shooter velocity (<code>vel_sht</code>)	300	700	kt
Shooter pitch (<code>pit_sht</code>)	-30	30	deg
Target altitude (<code>alt_tgt</code>)	1,000	45,000	ft
Target velocity (<code>vel_tgt</code>)	300	700	kt
Target heading (<code>hdg_tgt</code>)	-180	180	deg
Target off-boresight (<code>rgt_tgt</code>)	-60	60	deg
Distance shooter-target (<code>dist</code>)	5	45	NM
Target response time (<code>delay</code>)	15	30	s
Target turn degree (<code>turn_dg</code>)	0	180	deg
Target acceleration (<code>load_factor</code>)	1.5	5	G (m/s ²)

2.40 GHz and 128 GB of RAM. It took approximately 40 days to execute all the simulation runs. Each run generated an output file containing the missile's miss distance (`miss_dist`), measured in meters by the simulator, under the respective input conditions. The miss distance is measured at the closest point of approach to the target aircraft and represents the minimum separation distance between the missile and its target.⁴⁹ The large number of simulations also allows for an effective downsampling process, as described in the subsection, ensuring that a representative subset can be selected while maintaining the fidelity of the original data.

While the LHS method ensures a well-distributed sample across the input parameter space, the simulation data generated in this study is subject to certain limitations due to assumptions and simplifications inherent in the model. One such limitation is the idealized behavior of both the shooter and target, which may not fully capture the complexity of real-world engagements, such as pilot decision-making under uncertainty or environmental factors like varying weather conditions. In addition, some parameters, such as missile dynamics, were modeled based on a consistent structure, but without accounting for potential variations in performance due to system degradation or other operational constraints.

The simplifications in target response and maneuvering behavior might also limit the accuracy of the model when applied to diverse combat scenarios. Moreover, the computational resources limited us to certain ranges and resolutions of parameters, which might not entirely cover all possible operational scenarios. These factors must be taken into account when interpreting the results, as they could impact the generalizability of the findings to different environments and conditions. Despite these limitations, the insights gained from this analysis provide a valuable foundation, which could be further tested and validated using more complex scenarios and comprehensive simulation environments.

3.3. Preprocessing

Concerning the preprocessing step, we initially performed an Exploratory Data Analysis (EDA) to identify the data set's general behaviors and obtain an introductory understanding of the data. This understanding allowed us to formulate hypotheses and assess the need for new data collections. The methods employed for this primary analysis included descriptive statistics, histograms, box plots, correlation, and outlier analysis.

In our study, we employed histograms and box plots to understand the output variable's distribution and behavior across simulations and identify initial outliers in the samples. A crucial aspect of our analysis was the investigation of correlations among variables to detect multicollinearity, which can significantly impair the efficacy of supervised machine learning algorithms.⁵⁰ Notably, we observed a substantial correlation between the Target turn degree (`turn_dg`) and Target acceleration (`load_factor`). This finding is particularly relevant, suggesting the redundancy of one of these variables. Therefore, we propose eliminating `load_factor` from the set of input variables to optimize the performance of our future machine learning models.

Furthermore, we employed a downsampling technique on the initial LHS design to examine potential outliers in the data collected. Identifying outliers and inadequate data

in a data set is arguably one of the most challenging parts of the preprocessing stage, and it is always a topic to be explored cautiously.⁵¹ Based on subject matter expert knowledge, we defined the variable intervals but did not control the combination of values generated for each parameter. As a result, we could produce some improbable input variables. For instance, an aircraft at 1,000 ft launching a missile at a target at 45,000 ft is extremely rare from an operational perspective since a pilot would most likely increase altitude before shooting.⁴ We manually removed these disfavored samples, such as the one presented, to prevent confusion in the model. For more details, check the code associated with the EDA methods (EDA code: <https://github.com/jpadantas/poker/tree/main/eda>).

3.4. Warhead lethality

The lethality of a proximity-fuzed warhead, represented by the probability of kill (P_{kill}), depends on the conditions at the time of detonation and the relative position between the warhead and the target aircraft. The P_{kill} function estimates the likelihood that an aircraft is destroyed given the detonation of the warhead.⁴⁹ This probability is influenced by several factors, including fragment velocity, density of fragment dispersion, and the miss distance.

Figure 6 illustrates how increasing the detonation distance reduces the effectiveness of the warhead. The ideal lethal radius represents the optimal distance for maximum lethality, while the declining effectiveness curve shows how the kill probability decreases as the detonation distance deviates from this optimal value.⁴⁹

3.4.1. Factors affecting warhead lethality. The lethality of a proximity-fuzed warhead is determined by various factors that influence how effectively the warhead fragments impact the target. Key factors include the following:

- **Miss Distance:** The proximity of the warhead at detonation to the target is the primary factor affecting the probability of kill. A smaller miss distance generally leads to a higher likelihood of target destruction, while a larger miss distance reduces effectiveness.
- **Fragment Distribution and Density:** The arrangement and density of the fragments around the explosive affect how many fragments are likely to impact the target. In this study, the pre-formed fragments are arranged in a brickwork pattern, which optimizes coverage and ensures a consistent fragment distribution.
- **Fragment Initial Velocity:** The initial velocity of the fragments post-detonation, which depends on the Gurney constant and the explosive characteristics,

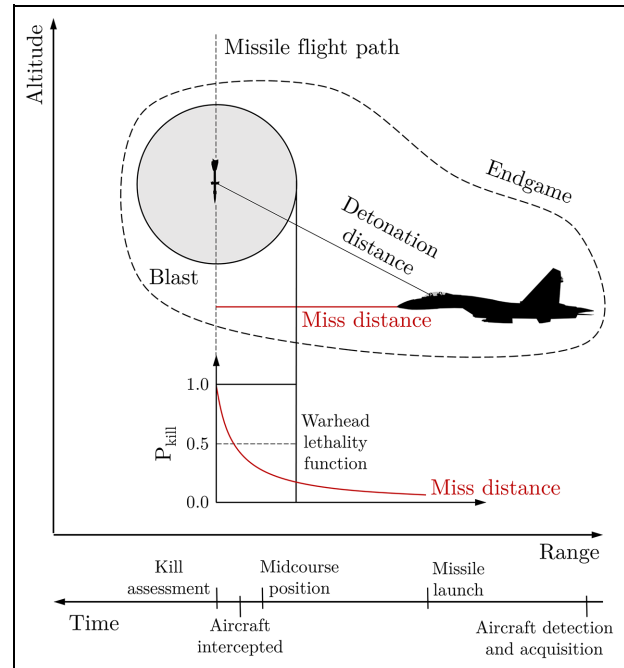


Figure 6. Impact of detonation distance on warhead lethality. The figure shows the relationship between detonation distance and P_{kill} for a proximity-fuzed warhead. The lethal radius represents the optimal distance for maximum lethality, contrasted with the decreasing kill probability as the distance increases.

determines how far the fragments travel and their ability to penetrate the target structure.

- **Warhead and Explosive Properties:** The material composition, dimensions, and explosive characteristics (e.g. detonation velocity and explosive mass) collectively impact the lethality.

3.4.2. Modeling the physical properties of the warhead. To calculate P_{kill} , it is necessary to accurately model the warhead's physical properties. Table 2 presents the structural, fragment, and explosive properties of the warhead used in this study.

The properties presented in Table 2 form the foundation for understanding the warhead's behavior during detonation, which is essential for modeling its effectiveness. Each key parameter contributes to the warhead's ability to distribute fragments and ultimately destroy the target. The data are organized as follows:

- **Explosive Data:** The warhead uses PBX-9404, a high-energy explosive with a density (ρ_e) and detonation velocity (D). The explosive mass (M_e), Gurney constant (G), and the initiation point (X_2) are key factors in determining the velocity and energy imparted to the fragments upon detonation.

Table 2. Missile's warhead data including structural, fragment, and explosive properties.

Variable	Value	Unit
Explosive Data		
Explosive Type	PBX-9404	-
Explosive Density (ρ_e)	1,710	kg/m ³
Explosive Mass (M_e)	8.94	kg
Detonation Velocity	9,500	m/s
Gurney Constant (G)	2,895	m/s
Initiation Point (X_2)	100	mm
Charge-to-Mass Ratio (C/M)	0.65	-
Fragment Initial Velocity (V_0)	2,028	m/s
Material Data		
Material	Mild Steel	-
Density (ρ)	7,850	kg/m ³
Geometric Data		
External Diameter	178	mm
Skin Thickness	2	mm
Structural Diameter (D_s)	154	mm
Internal Diameter (D_i)	144	mm
Structural Thickness (T_s)	5	mm
Effective Length (L_e)	321	mm
Structural Mass (M_s)	5.90	kg
Pre-formed Fragment Data		
Type	Sphere	-
Pattern	Brickwork	-
Diameter or Distance (\overline{AO})	10	mm
Distance (\overline{OB})	8.66	mm
Sphere Mass (m_{sph})	0.0041	kg
Number of Spheres		
Total Number (N_{sph})	1,909	-
Total Mass of Spheres (M_{sph})	7.85	kg

- **Material Data:** The warhead casing is made from mild steel, providing sufficient strength to contain the explosive until detonation. The density (ρ) of the steel is used to calculate structural mass and other characteristics relevant to energy transfer during detonation.
- **Geometric Data:** Geometric properties such as structural diameter (D_s), internal diameter (D_i), and effective length (L_e) determine the size and volume of both the warhead structure and the explosive content. These dimensions influence explosive power and fragment distribution.
- **Pre-formed Fragment Data:** The fragments are pre-formed as spherical mild steel pieces and arranged in a brickwork pattern. This arrangement ensures uniform fragment dispersion. Parameters include

fragment diameter (\overline{AO}), distance between fragment centers (\overline{OB}), and sphere mass (m_{sph}). The total number of fragments (N_{sph}) and their combined mass (M_{sph}) are essential for assessing lethality.

- **Fragment Initial Velocity (V_0):** Using the Gurney equation, the initial velocity of the fragments (V_0) is estimated. This velocity is critical for determining how far and how fast the fragments will travel toward the target, which directly impacts P_{kill} .

In summary, the combination of explosive, material, geometric, and fragment properties forms the foundation of the warhead's lethality model. These data points will be integrated into subsequent probabilistic models to evaluate P_{kill} under varying combat scenarios, providing a comprehensive assessment of the warhead's effectiveness.

3.4.3. Warhead performance metrics. The initial velocity (V_0) of the warhead fragments post-detonation is determined using the Gurney equation

$$V_0 = G \cdot \sqrt{\frac{C/M}{1 + 0.5 \cdot C/M}} \quad (1)$$

where G is the Gurney constant of the explosive, and C/M is the ratio of the explosive mass to the total mass of the warhead's structural components and fragments.

In addition, static ejection angles are necessary for determining the initial dispersion of the fragments. These angles represent the directional pattern of fragment ejection relative to the warhead, influenced by the warhead's design and explosive-structure interaction. Understanding these angles helps optimize fragment spread for maximum target coverage.

3.4.4. Endgame condition determination. In this stage, the model considers operational flight conditions at the moment of detonation. The conditions include the following:

- **Missile Velocity (V_m):** Determines the additional kinetic energy imparted to the fragments, influencing their relative velocity to the target.
- **Target Velocity (V_t):** Affects the relative speed and encounter geometry, impacting the likelihood of fragment impact.
- **Miss Distance:** The predicted closest distance between the missile and the target at detonation, directly affecting P_{kill} by determining how many fragments can reach the target.
- **Altitude:** Influences air density, affecting drag on the fragments and their ability to penetrate the target.

3.4.5. Endgame dynamics computation. The endgame dynamics involve computing dynamic ejection angles and fragment velocities relative to the missile's trajectory to predict the interaction with the target. Unlike static ejection angles, dynamic ejection angles consider missile velocity at detonation, modifying the trajectory of each fragment.

The relative velocity of each fragment with respect to the moving target is calculated to assess the likelihood of impact. This velocity depends on the initial velocity of the fragment (V_0), missile velocity (V_m), and target velocity (V_t). The computed relative velocity and ejection angles allow for the estimation of the distance each fragment will travel post-detonation, which is crucial for assessing their ability to reach and penetrate the target.

3.4.6. Fragment spray density and lethality assessment. The fragment spray density represents the concentration of fragments within the target zone, a key factor in determining lethality. The density of fragment saturation directly influences the number of impacts on the target, affecting P_{kill} .

Several factors are combined to evaluate lethality, including the following:

- Drag Coefficient (C_D) and Air Density (ρ): These parameters affect how quickly fragment velocity decreases due to air resistance. A higher air density or drag coefficient results in greater velocity reduction, reducing penetration likelihood.
- Impact Velocity: The impact velocity of the fragments determines their ability to inflict damage upon reaching the target. This velocity is influenced by the drag experienced during flight.

The lethality assessment combines these factors to estimate the probability of kill per fragment hit, ultimately leading to the overall kill probability for each fragment.

3.4.7. Computation of final probability of kill. For the calculation of P_{kill} in this work, it was assumed that all missile launches used consistent values for key parameters. The intrinsic missile parameters, which are assumed to remain constant across different simulations, include fragment initial velocity $V_0 = 2,028$ m/s, effective length $L_e = 321$ mm, internal diameter $D_i = 144$ mm, structural diameter $D_s = 154$ mm, and fragment diameter $\overline{AO} = 10$ mm.

However, the parameters that may vary depending on the simulation context are the missile velocity $V_m = 592$ m/s, target velocity $V_t = 250$ m/s, target wingspan of 8.13 m, and the angle between trajectories, which was set to 0 radians for this study. This distinction helps clarify which parameters are intrinsic to the missile type and

which are subject to change depending on the specific conditions of the simulation.

Since these parameters remain constant for all launches, the miss distance becomes the primary factor influencing the variation in P_{kill} . This assumption simplifies the model, allowing us to focus on optimizing warhead lethality based on detonation distance.

The final P_{kill} integrates the structural and explosive characteristics of the warhead with dynamic ejection analysis and fragment lethality assessment. This probability provides a statistical measure of the warhead's effectiveness and its likelihood of achieving its intended destructive outcome upon the target.

This model integrates principles from materials science, ballistics, aerodynamics, and probability theory to predict warhead performance, from detonation to target impact. By incorporating the miss distance obtained from the simulation, we transform this distance into a P_{kill} value, enhancing the model's predictive capacity. Such an approach is important for optimizing warhead configurations to maximize their efficacy. Interested readers can refer to the code for lethality calculations available at the following repository (Lethality code: <https://github.com/jpadantas/poker/tree/main/lethality>).

3.5. Supervised machine learning models

We created supervised machine learning models using a subset of the initially proposed input variables, excluding the `load_factor` variable, as mentioned previously, to develop a regression model capable of predicting the missile P_{kill} , which was calculated from the data set based on the miss distance from the target, using the lethality theory described in the previous subsection. Notably, variables such as `turn_dg` and `delay` are not directly available to the launcher and require assumptions for their estimation. These variables will be addressed in detail in a later subsection. We employed three distinct algorithms to build the models: Polynomial Regression (PR), Artificial Neural Networks (ANN), and Extreme Gradient Boosting (XGBoost).

Before starting creating and training the models, we performed a stratified train-validate-test split for all supervised machine learning algorithms regarding 34 different probability blocks (bins) using Doane's formula,⁵² allocating 80% for training and validation using a 5-fold cross-validation technique and 20% for testing, following the approach proposed in previous works.^{4,26} This data set separation will allow the evaluation of the machine learning models later.

PR extends the linear model by adding extra predictors obtained by raising each original predictor to a determined degree. Adding polynomial terms to the linear model can effectively allow the model to identify nonlinear

patterns.⁵³ The degree of the polynomial controls the number of features added. Using a degree greater than four is unusual because the polynomial curve can become overly flexible and take on some unusual shapes.⁵⁴ We proposed three different models with degrees from one to three using the Scikit-learn library.⁵⁵ For additional information, refer to the code associated with the PR models (PR code: <https://github.com/jpadantas/poker/tree/main/pr>).

ANN seeks to approximate the function represented by the data set, calculating the error between the predicted outputs and the expected outputs and minimizing this error during the training process, working as function approximation machines that are designed to achieve statistical generalization.⁵⁶ In this work, we used a multi-layer perceptron using the backpropagation algorithm, adjusting the weights of the connections between ANN neurons to minimize the mean-squared error loss.⁵⁷ We performed data scaling to equally distribute the importance of each input in the ANN learning process.⁵⁸ We used a min-max scaler, transforming all data features to a range from 0 to 1.⁵⁹ For the ANN model, we utilized TensorFlow⁶⁰ to run 50 different hyperparameters configurations, changing the number of hidden layers in {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} and the number of units in {16, 32, 64, 128, 256}. All nodes have a rectified linear activation function (ReLU).⁵⁶ In addition, we included the Adaptive Moment Estimation (Adam) optimizer, a well-known training algorithm for ANN.⁶¹ Adam is a stochastic gradient descent method based on adaptive estimation of first- and second-order moments function.⁶² We typically should choose a batch size between one and a few hundred. For a given computational cost, small batch sizes achieve the best training stability and generalization performance across diverse experiments.⁶³ We selected 32 as the model's batch size, considered a reasonable default value.⁶⁴

XGBoost, a leading gradient boosting framework, is engineered to construct a robust ensemble of decision trees by iteratively correcting errors from previous trees, thereby enhancing the model's accuracy for regression, classification, and ranking tasks and is considered the state-of-the-art for tabular data, as evidenced by its widespread adoption and superior performance in various competitions and applications.⁶⁵ In our study, we leveraged an extensive array of hyperparameters, including `n_estimators` set to an ambitious 1,000,000 trees for comprehensive data exploration, and `max_depth` options ranging in {10, 12, 14, 16, 18, 20}, enabling the model to uncover complex patterns without excessively increasing complexity. The learning rate was varied across {0.01, 0.1, 0.2, 0.3, 0.4}, balancing the convergence speed with the solution's accuracy. Alongside this, subsampling and feature sampling rates were set using `subsample`, `colsample_bytree`, and `colsample_bylevel`, which were varied in the

range from 0.5 to 1.0 with a step size of 0.1, to enhance model generalization. In addition, `min_child_weight` and `gamma` parameters were finely adjusted to regulate the model's growth and complexity, with `min_child_weight` in {1, 3, 5} and `gamma` in {0.0, 0.1, 0.2, 0.3, 0.4}. For further insights, please consult the code for the XGBoost models (XGBoost code: <https://github.com/jpadantas/poker/tree/main/xgboost>).

For both the ANN and XGBoost models, we implemented an early-stopping mechanism to oversee the training phase efficiently. This strategy entailed continuous monitoring of performance metrics on the validation set to detect any stagnation in improvement. We defined "patience" as the maximum number of epochs to continue training without observing any advancement in the validation set's metrics, setting this threshold at 10 epochs. This specific interval proved to be balanced for mitigating the effects of noise within the model optimization processes, ensuring that both models ceased training at a juncture that prevented overfitting while maximizing performance.⁵⁶

The choice of machine learning models in this study was made to provide a balanced comparison between simplicity and performance. We included PR as a simpler baseline model, which we anticipated would be faster to train but likely result in lower performance compared to more complex models. This choice allowed us to understand the trade-offs between model complexity and accuracy in our context. The ANN and XGBoost models were selected as they are well-known for their effectiveness with tabular data and are frequently used for achieving high predictive metrics in such scenarios. This combination of models ensured a comprehensive evaluation of different approaches, ranging from simpler to more sophisticated techniques, thereby providing valuable insights into model performance under different levels of complexity.

3.6. Models evaluation

Since this work analyzed a regression problem, we evaluated all the supervised machine learning models using the following well-known metrics: Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the Coefficient of Determination (R^2). In addition, training and inference time are two critical aspects of a model's assessment. We analyzed each cross-validation fold's training time and calculated each model's inference time on the test data set. While training time is important, inference time is even more important as it directly impacts the model's practical applicability in real-time scenarios. We can estimate which model offers the best trade-off between performance and computational cost by examining these factors.

3.7. Estimating turn degree and delay

Estimating the turn_dg and delay for a model presents significant challenges due to the inherent unpredictability of a target's response to an incoming missile. This unpredictability is largely attributed to the unique characteristics and decision-making processes of the target, which can vary widely in real-world scenarios. Understanding and predicting how a target will maneuver in response to a perceived threat is critical for accurate simulation and assessment of missile engagement outcomes.

To address this challenge, we employ the expertise of SMEs in military tactics and missile defense systems to estimate the potential turn_dg and delay of a target upon missile detection. The approach leverages the deep understanding of SMEs regarding the behavior of various targets under threat, allowing for a more informed and realistic set of estimations.

These parameters are essential for modeling the target's evasive maneuvers and the subsequent trajectory adjustments of the missile. We represent the target's response variability using normal distributions for both turn_dg and delay . The choice of a normal distribution is based on the assumption that the target's response will vary around a mean value (median) with a certain degree of predictability, as influenced by the target's characteristics and situation. These distributions' median and standard deviation are determined through a collaborative process with SMEs, ensuring that the estimations reflect a realistic range of target behaviors.

The estimation process involves the identification of specific scenarios in which a target might find itself when a missile is launched toward it. Each scenario is carefully analyzed to determine the likely response of the target, focusing on two key parameters: the turn_dg and delay before initiating the turn.

To estimate the turn_dg of the target, we assume that the target will adopt the most defensive posture possible, always attempting to move in the direction opposite to the missile launcher. Regardless of the distance between the missile launcher and the target, the target aircraft will aim to maximize the separation by performing a defensive turn of up to 180° , whenever feasible. This approach ensures that the target consistently prioritizes evasion by moving away from the threat to the greatest extent possible.

The initial value of the target's turn degree is defined by the following function

$$\text{turn_dg}_0 = ((\text{rgt_tgt} + 180) \bmod 360) - 180. \quad (2)$$

To illustrate the initial value of the defensive behavior of the targets, Figure 7 presents 10 diverse and well-spread scenarios, where each target attempts to perform a defensive maneuver aimed at maximizing separation from the

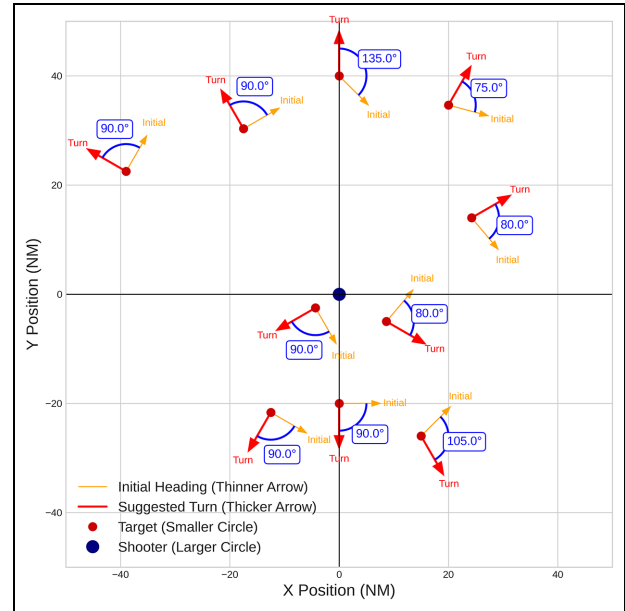


Figure 7. Illustration of 10 diverse scenarios of defensive turn maneuvers. Each scenario shows a target attempting to maximize distance from the launcher. The dark red points (smaller) indicate the initial position of the target, while the orange (thinner) and red arrows (thicker) represent the initial heading and the suggested turn direction, respectively. The dark circle (larger) represents the shooter. The blue arc indicates the turn degree (turn_dg) performed by the target, which varies between 0° and 180° .

launcher. The distances, initial headings, and off-boresight angles were chosen to ensure a good distribution in space, avoiding overlap and facilitating visualization of the maneuvers. Each target, represented by the red point, performs a turn to move away from the launcher (the blue point at the center) according to the suggested direction indicated by the red arrow, in an effort to maximize evasion. The turn degree (turn_dg_0) is represented by the blue arc and varies according to the relative position of the target to the launcher

$$\text{turn_dg} = \text{turn_dg}_0(1 + 0.05 \cdot w), \quad (3)$$

where turn_dg_0 is the value calculated from Equation 2, and $w \sim \mathcal{N}(0, 1)$ represents a standard normal random variable. This introduces a 5% variation, which accounts for natural fluctuations in the target's maneuver.

Moreover, regarding the variable delay , we anticipate specific behaviors based on the target's orientation relative to the missile launcher. While we initially assumed that shorter reaction times would occur when the target directly faces the launcher due to the immediate perception of the threat, we acknowledge that this is not always the case.

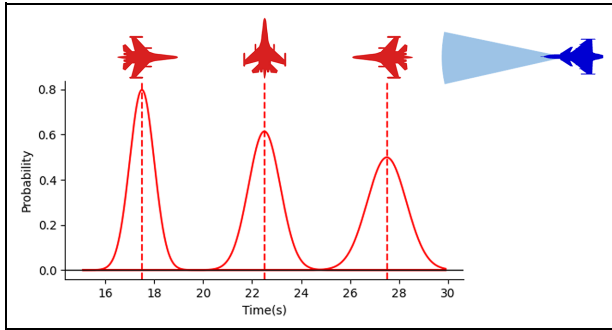


Figure 8. Graph illustrating the target's delay time as a function of its orientation with respect to the shooter.

The reaction time depends on various factors, including the target's sensor capabilities and engagement strategy. For instance, even when the launcher is positioned behind the target, a Radar Warning Receiver (RWR) may immediately detect the radar, leading to a prompt response.

To simplify this context, we made certain assumptions about the target's behavior. Specifically, we structured the estimated reaction times within a range of 15 to 30 s. This range is further segmented into three distinct intervals, each spanning 5 s, to reflect varying degrees of threat awareness and response times. These intervals are divided into 120° each to cover the full 360° range, ensuring that all possible orientations are accounted for. Each interval is associated with a standard deviation value of approximately 5%, which ensures the three curves are almost connected, allowing for comprehensive sampling of all potential values. Please refer to Figure 8 for a detailed visualization of the considered scenarios and the corresponding distributions for *delay*.

To validate the chosen values for the median and standard deviation, we employ the face validation method.⁶⁶ This method involves a review by an independent group of experts who assess the plausibility and accuracy of the estimations based on their own experience and knowledge. The face validation process ensures that the estimations are not only based on a solid theoretical foundation but also resonate with the practical insights of seasoned professionals in the field.

It is important to note that the parameters and distributions outlined in this section adhere to the intervals and assumptions previously described in this work. By leveraging SME knowledge and employing rigorous validation techniques, we aim to provide a robust framework for estimating *turn_dg* and *delay*, thereby enhancing the accuracy of missile engagement simulations. The validation process was conducted through face validation by SMEs with extensive experience in missile engagements. Specifically, several simulated scenarios were presented to the SMEs, covering a range of engagement distances,

target maneuvers, and missile launches. The SMEs evaluated whether the estimated *turn_dg* and *delay* values aligned with their operational experience and expectations under different combat situations. This process ensured that the estimated values were realistic and consistent with real-world operational behavior.

3.8. The probability of hitting a target

Determining the final probability of hitting a target is intricately linked to the analysis performed by machine learning algorithms. These algorithms are designed to process a comprehensive set of input variables derived from the aircraft's subsystems, providing the pilot with actionable intelligence to enhance missile targeting accuracy. However, a critical component of this targeting process is the estimation of the target's *turn_dg* and *delay*, which are not directly observable and must be inferred through the methodology outlined previously.

This estimation plays an essential role in calculating the hit probability, accounting for the variability in the target's response behavior. By incorporating the target's expected maneuvers, the machine learning model can adjust its predictions to better reflect the dynamic nature of aerial engagements. The aim is to achieve a conservative estimation of the target's performance, ensuring that the probability of a hit considers the best possible maneuvers the target could execute to evade the missile.

To this end, the model leverages the variability in the target's behavior as an essential factor, allowing for a better understanding of potential evasive actions. This approach is grounded in the principle of anticipating optimal target performance, where the target utilizes its capabilities to the maximum extent to evade incoming threats. By assuming a high level of performance from the target, the model inherently considers the target to perform well on average, thereby minimizing the risk of underestimating the target's evasive capabilities.

Integrating this realistic estimation into the machine learning algorithm enhances the reliability of the hit probability calculation. It enables the algorithm to account for the widest possible range of target behaviors, including those that may occur under extreme conditions. This approach ensures the pilot has a probability of hit that reflects the most challenging engagement scenarios, facilitating informed decision-making during missile launch sequences.

In summary, the probability of hitting a target is not merely a static figure derived from straightforward inputs; it is a dynamic estimation that incorporates the complexity and unpredictability of aerial combat. By integrating conservative estimations of the target's *turn_dg* and *delay* into the machine learning response, we aim to improve the accuracy and realism of missile engagement outcomes.

Table 3. Comparative evaluation of machine learning models based on the proposed metrics.

Model	PR	ANN	XGBoost
Hyperparameters	degree = 4	hidden_layers = 3, units = 128	max_depth = 14, learning_rate = 0.01, subsample = 0.9, colsample_bytree = 0.9, colsample_bylevel = 0.6, min_child_weight = 5, gamma = 0.1
MAE	11.52	2.33	5.04
MSE	296.26	51.90	117.14
RMSE	17.21	7.20	10.82
R^2	67.94	94.38	87.31
Training Time (s)	532.23	4,722.17	6,098.33
Inference Time (s)	9.57	6.65	8.20

The best results for each metric are highlighted in bold red text.

However, we acknowledge the limitations of the current approach, and further empirical validation is required to fully demonstrate the practical benefits of these estimations. This ongoing work lays the foundation for future studies that will more comprehensively assess the impact of these estimations on aerial defense strategies.

4. Results and analysis

This section examines the metrics of the test data set for the proposed machine learning models. In addition, we provide a Multi-Function Display (MFD) representation, incorporating the traditional maximum range and no-escape zone indicators along with the probability of kill in the WEZ indication based on the proposed probabilistic model.

4.1. Comparative analysis of the machine learning models

Table 3 summarizes the mean performance metrics for each model across the five folds of the three distinct proposed models: PR, ANN, and XGBoost. These metrics include MAE, MSE, RMSE, and R^2 , reflecting their respective best hyperparameter settings, chosen based on performance on the validation data set during training. It is important to note that the predicted variable, P_{kill} , is expressed as a percentage. Consequently, all the error metrics (MAE, MSE, RMSE) are dimensionless, as they are derived from percentage values. In addition, R^2 is also dimensionless. The analysis further includes computational efficiency metrics, such as training and inference time, both measured in seconds, using the training/validation and test data sets, respectively.

Table 4. Friedman test results.

Metric	Friedman statistic	p -value
MAE	10.0	0.0067
MSE	10.0	0.0067
RMSE	10.0	0.0067
R^2	10.0	0.0067
Training Time	8.4	0.0150
Inference Time	4.8	0.0907

Significant p -values are highlighted in bold red text.

4.2. Statistical analysis

To determine whether the observed differences in model performance are statistically significant, we performed the Friedman test⁶⁷ followed by the Nemenyi post hoc test⁶⁸ for multiple comparisons. Interested readers can refer to the code for the statistical analysis at the following repository (Statistics code: <https://github.com/jpadantas/poker/tree/main/statistics>).

4.2.1. Friedman test. The Friedman test was used to detect differences in performance across the three models using the results from 5 folds of each model. The null hypothesis is that there are no differences in the performance metrics among the models. Table 4 presents the results of the Friedman test for each metric.

The results indicate that for all metrics except inference time, there are statistically significant differences among the models (p -value < 0.05). Notably, the p -values for several metrics (MAE, MSE, RMSE, and R^2) are identical, suggesting that the models perform similarly across these related metrics. This similarity could be due to the fact that these metrics are strongly correlated, and the relatively

Table 5. Nemenyi post hoc test p -values for pairwise comparisons among models.

Metric	Comparison	p -value
MAE	PR vs ANN	0.0045
	XGBoost vs PR	0.2541
	XGBoost vs ANN	0.2541
MSE	PR vs ANN	0.0045
	XGBoost vs PR	0.2541
	XGBoost vs ANN	0.2541
RMSE	PR vs ANN	0.0045
	XGBoost vs PR	0.2541
	XGBoost vs ANN	0.2541
R^2	PR vs ANN	0.0045
	XGBoost vs PR	0.2541
	XGBoost vs ANN	0.2541
Training Time	XGBoost vs PR	0.0123
	PR vs ANN	0.1394
	XGBoost vs ANN	0.6008

Significant p -values are highlighted in bold red text.

small sample size (5 folds) may limit the ability of the Friedman test to distinguish nuanced differences between models. In addition, the identical Friedman statistics for these metrics imply that the rankings across models were consistent, reflecting similar performance trends.

4.2.2. Nemenyi post hoc test. To identify which models differ significantly from each other, we conducted the Nemenyi post hoc test. Table 5 shows the p -values for pairwise comparisons among the models for the metrics where the Friedman test was significant. Note that the inference time metric did not show significant differences in the Friedman test; therefore, the Nemenyi test was not conducted for this metric.

The Nemenyi post hoc test results, shown in Table 5, indicate that significant differences primarily occurred between PR and ANN. For XGBoost, its comparisons with both PR and ANN resulted in similar p -values across multiple metrics. This suggests that XGBoost's performance was relatively stable and consistently fell between the other two models. This pattern might reflect limited differentiation capability given the inherent relationships among the metrics and the modest sample size, highlighting that larger samples might be needed to detect finer performance differences.

5. Discussion

The statistical analysis confirms that the choice of model has a significant impact on performance metrics and training time. The Friedman test indicated statistically significant differences among the models for MAE, MSE, RMSE, R^2 , and Training Time (p -value < 0.05). The Nemenyi post hoc test further identified that

- For MAE, MSE, RMSE, and R^2 , ANN significantly outperformed PR (p -value = 0.0045).
- No significant differences were found between XGBoost and PR, or XGBoost and ANN for these metrics.
- For Training Time, PR had a significantly shorter training time than XGBoost (p -value = 0.0123).
- No significant differences were found between PR and ANN, or XGBoost and ANN for Training Time.

These results suggest that ANN provides superior predictive accuracy compared to PR, with a statistically significant improvement in error metrics. Although ANN's training time was longer than PR's, the difference was not statistically significant between ANN and PR, indicating that the increased computational cost is acceptable given the performance gains.

Among the models tested, ANN exhibited the lowest error metrics across all considered measures. Specifically, ANN achieved the lowest MAE (2.33), suggesting its predictions were closer to the actual values on average compared to PR and XGBoost. Similarly, ANN recorded the lowest MSE (51.90) and RMSE (7.20), indicating a strong capability to minimize both variance and bias in its predictions. The R^2 value for ANN stood at 94.38%, demonstrating that it could explain a substantial portion of the variance in the data set, which indicates strong predictive power.

Regarding computational efficiency, PR had the shortest training time (532.23 s), which was significantly shorter than that of XGBoost (6,098.33 s) as indicated by the Nemenyi test (p -value = 0.0123). ANN's training time (4,722.17 s) was not significantly different from either PR or XGBoost. In terms of inference time, ANN was the fastest (6.65 s), slightly quicker than XGBoost (8.20 s) and PR (9.57 s).

Specifically for the ANN, Figure 9 illustrates a comparative analysis of the neural network's performance metrics, demonstrating the best results among the evaluated models. The three-dimensional surface plots capture how various configurations of model units and layers affect both model accuracy, measured by the mean R^2 value, and the inference time required for predictions. Plot (a) demonstrates that the relationship between the number of units and layers significantly impacts accuracy, indicating an interplay between model depth and performance. Plot (b) shows an increase in inference time with more complex models, emphasizing the need to consider computational efficiency alongside model precision. These visualizations highlight the balance that must be struck in neural network architecture design to optimize both performance and efficiency.

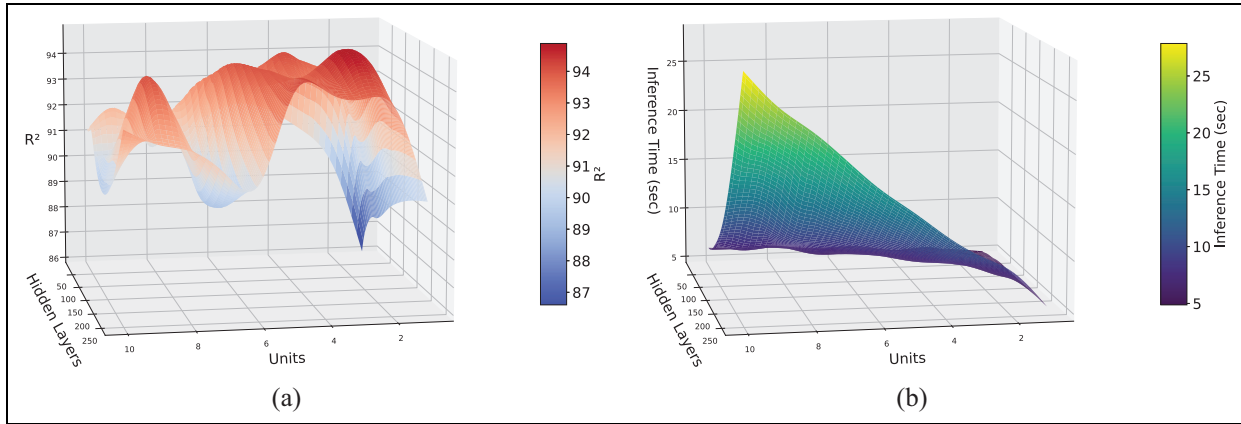


Figure 9. Comparative analysis of neural network performance metrics. (a) illustrates how model configurations, varying in units and hidden layers, impact accuracy through the mean R^2 value. (b) delineates the effect of model complexity on inference time, measured in seconds.

XGBoost achieved moderate accuracy metrics with an MAE of 5.04, MSE of 117.14, RMSE of 10.82, and an R^2 of 87.31%. While its error metrics were higher than those of ANN, the differences were not statistically significant according to the Nemenyi test (p -value = 0.2541 for XGBoost vs PR and p -value = 0.2541 for XGBoost vs ANN). XGBoost's performance was acceptable and it may serve as a robust alternative, especially in contexts where ensemble methods are preferred. Its inference time (8.20 s) was shorter than PR's but slightly longer than ANN's, with no significant differences detected.

The simplicity of PR resulted in lower predictive performance compared to more advanced models. Its error metrics were significantly higher than those of ANN, as confirmed by the Nemenyi test (p -value = 0.0045 for PR vs ANN). PR lagged behind, with the highest MAE (11.52), MSE (296.26), RMSE (17.21), and the lowest R^2 value (67.94%). These results highlight the limitations of PR in capturing complex patterns within the data set, especially when compared to models like ANN and XGBoost, which are better suited for such complexity.

Note that discrepancies in computational times may be due to differences in library implementations, as PR and ANN were implemented using different libraries, which could impact their computational efficiency.

In summary, the statistical analysis confirms that ANN provides significantly better predictive accuracy compared to PR, without a significant increase in training time. While XGBoost did not show significant differences compared to the other models, it offers a balance between performance and computational efficiency. These findings can guide practitioners in selecting the appropriate model based on the specific requirements of accuracy and computational efficiency, considering both statistical significance and practical performance metrics.

5.1. Enhanced weapon engagement zone representation

We introduce an enhanced MFD visualization for the WEZ, integrating a probabilistic model that encompasses the traditional maximum range and no-escape zone indicators and a novel probability of kill aspect within the WEZ indications. By applying the proposed probabilistic model, we adjusted the target's off-boresight angles from -60° to $+60^\circ$. The enhanced WEZ visualization, depicted in Figure 10, now incorporates P_{kill} , offering a more

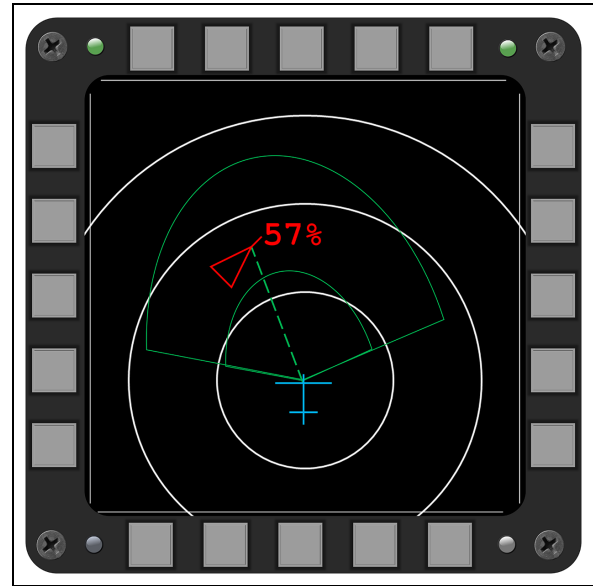


Figure 10. Enhanced MFD representation incorporating the integration of P_{kill} alongside the maximum range and no-escape zones, providing a multifaceted view of the engagement zone to support informed decision-making in missile launch scenarios.

comprehensive understanding of the engagement zone. Incorporating this metric alongside traditional WEZ metrics may improve a pilot's situational awareness and decision-making capabilities, particularly in controlled scenarios. However, this potential benefit is heavily dependent on model assumptions and requires further validation for broader applicability. This integration can possibly support pilots in making more informed decisions regarding missile launches against potential targets by providing a clearer assessment of the likelihood of successfully engaging a target under various conditions. Including a probabilistic kill probability metric may aid in optimizing the use of onboard weaponry by evaluating the tactical viability of a missile launch, thus enhancing operational effectiveness and minimizing resource loss.

6. Conclusion and future work

In conclusion, this work presents PoKER, an innovative probabilistic model for WEZ analysis in BVR air combat scenarios. By extending beyond traditional WEZ metrics to include a probability of kill calculation, PoKER enhances the precision and relevance of missile engagement assessments. This integration may improve the pilot's situational awareness and decision-making capabilities by providing a better understanding of engagement outcomes. The capability of PoKER to offer real-time, accurate estimations of missile effectiveness based on dynamic combat conditions and target behavior represents a notable advancement in air combat strategy and planning. Its application not only can enhance operational efficiency and effectiveness but also contributes to the optimal use of onboard weaponry, thus minimizing resource loss and maximizing engagement success.

For future work, we advocate the utilization of data derived from real training missile launches, whether from virtual simulations or live exercises, to refine the modeling of target behavior. The introduction of human-in-the-loop simulations could significantly enhance the model's practical application by aligning theoretical predictions more closely with real-world decision-making processes. Furthermore, the adoption of advanced machine learning techniques promises to make WEZ modeling more adaptive to changing combat conditions. Enhancements in missile simulation accuracy, alongside investigations into the impact of environmental variables and electronic warfare, could offer deeper insights into engagement strategies. Moreover, developing methods to predict adversary tactics and countermeasures accurately will greatly improve PoKER's strategic utility. In addition, creating a representation of how to maneuver into regions of higher

probability could inform pilots on effective maneuvers before a missile launch.

We also agree that evaluating the data set size is an important contribution, especially for the scalability of future work. While 10 million simulations were feasible for this study due to available computational resources, we acknowledge that this may not be practical for all scenarios. Therefore, we suggest exploring methods to reduce the data set size without compromising model fidelity, such as using efficient sampling techniques or selecting a subset of representative points. This could help balance computational costs while maintaining the quality of the analyses.


Furthermore, we plan to conduct empirical studies to validate the overall effectiveness of the proposed model across different scenarios. These studies will provide valuable insights into the model's strengths and limitations in varied operational contexts. While these studies will help assess the model's applicability, incorporating such estimations into real-world systems will require further investigation, which we suggest as a direction for future work as well. The ultimate goal is to demonstrate the utility of PoKER in enhancing situational awareness and decision-making for both autonomous agents and human pilots, particularly in scenarios involving AI-driven adversaries.

By integrating such predictive capabilities, pilots can make more informed decisions, enhancing their engagement strategies. Such efforts can solidify the model's foundational strength and expand its applicability in addressing the complexities of modern air combat. By pushing the boundaries of current WEZ models, this study contributes to academic understanding and offers practical insights for military strategists and defense technology developers.

Funding

This work was supported by Finep (Reference no. 2824/20). T.Y. and M.R.O.A.M. are partially funded by CNPq—National Research Council of Brazil through the grants 304134/2-18-0 and 307525/2022-8, respectively.

ORCID iD

Joao PA Dantas  <https://orcid.org/0000-0003-0300-8027>

Supplemental material

The code and instructions to demonstrate the methodology and calculations used in this study are available at <https://github.com/jpadantas/poker>. Due to the classified nature of the data, it is not possible to fully reproduce the results. However, the shared code provides valuable insight into the methods used in this study.

References

1. Li A, Meng Y and He Z. Simulation research on new model of air-to-air missile attack zone. In: *2020 IEEE 4th information technology, networking, electronic and automation control conference (ITNEC)*, Chongqing, China, 12–14 June 2020, pp. 1998–2002. New York: IEEE.
2. Dantas JPA, Costa AN, Geraldo D, et al. Engagement decision support for beyond visual range air combat. In: *Proceedings of the 2021 Latin American robotics symposium, 2021 Brazilian symposium on robotics, and 2021 workshop on robotics in education*, Natal, Brazil, 11–15 October 2021, pp. 96–101. New York: IEEE.
3. Li WH, Shi JP, Wu YY, et al. A multi-UCAV cooperative occupation method based on weapon engagement zones for beyond-visual-range air combat. *Def Technol* 2022; 18(6): 1006–1022.
4. Dantas JPA, Costa AN, Geraldo D, et al. Weapon engagement zone maximum launch range estimation using a deep neural network. In: Britto A and Valdivia Delgado K (eds) *Intelligent systems*. Cham: Springer, pp. 193–207.
5. Herrmann J. Air-to-air missile engagement analysis using the USAF Trajectory Analysis Program (TRAP). In: *Proceedings of the AIAA Flight Simulation Technologies Conference*, p. 3489. San Diego, CA, USA: American Institute of Aeronautics and Astronautics.
6. Yoon KS, Park JH, Kim IG, et al. New modeling algorithm for improving accuracy of weapon launch acceptability region. In: *Proceedings of the 29th Digital Avionics Systems Conference (DASC)*, pp. 6.D.4–1–6.D.4–6. Salt Lake City, UT, USA: IEEE.
7. Ling B, Bo L, Bingzheng S, et al. A cognitive study of multicolour coding for the head-up display (HUD) of fighter aircraft in multiple flight environments. *J Phys Conf Ser* 2019; 1215: 012032.
8. van Hoorn M. *Optimizing air-to-air missile guidance using reinforcement learning*. Master Thesis, Delft University of Technology, Delft, 2019, <http://resolver.tudelft.nl/uuid:6b26d24a-780c-47ee-9456-7af07490a317>
9. Dantas JPA, Costa AN, Medeiros FLL, et al. Supervised machine learning for effective missile launch based on beyond visual range air combat simulations. In: *Proceedings of the winter simulation conference: WSC '22*. Singapore: IEEE.
10. Kuroswiski AR, Medeiros FLL, De Marchi MM, et al. Beyond visual range air combat simulations: validation methods and analysis using agent-based models. *J Def Model Simul*. Epub ahead of print 27 November 2023. DOI: 10.1177/15485129231211915.
11. Department of Defense. Military handbook: missile flight simulation part one: surface-to-air missiles (MIL-HDBK-1211), 1995. https://quicksearch.dla.mil/qsDocDetails.aspx?ident_number=11687
12. Dantas JPA, Maximo MROA and Yoneyama T. Autonomous agent for beyond visual range air combat: A deep reinforcement learning approach. In: *Proceedings of the 2023 ACM SIGSIM Conference on Principles of Advanced Discrete Simulation (SIGSIM-PADS'23)*, pp. 13–24. New York, NY: Association for Computing Machinery.
13. Dantas JPA, Maximo MROA, Costa AN, et al. Machine learning to improve situational awareness in beyond visual range air combat. *IEEE Lat Am Trans* 2022; 20(8): 2039–2045, <https://latam.ieeer9.org/index.php/transactions/article/view/6530>
14. Reinisch F, Strohal M and Stütz P. Behaviour modelling of computer-generated-forces in beyond-visual-range air combat. In: *Proceedings of the 12th international conference on simulation and modeling methodologies, technologies and applications*, pp. 327–335. Lisbon: SCITEPRESS—Science and Technology Publications, <https://www.scitepress.org/DigitalLibrary/Link.aspx?doi=10.5220/0011306600003274>
15. Ummah K, Setiadi H, Pasaribu HM, et al. A simple fight decision support system for BVR air combat using fuzzy logic algorithm. *Int J Aviat Sci Eng* 2019; 1(1), <https://avia.ftmd.itb.ac.id/index.php/jav/article/view/8>. Number: 1
16. Kang Y, Liu Z, Pu Z, et al. Beyond-visual-range tactical game strategy for multiple UAVs. In: *2019 Chinese automation congress (CAC)*, Hangzhou, China, 22–24 November 2019, pp. 5231–5236. New York: IEEE.
17. Yagci O and Nikbay M. Evasive Maneuver Trajectory Optimization of an UCAV Against an Air to Air Missile. In: *AIAA AVIATION 2022 Forum*, p. 3791. Chicago, IL: American Institute of Aeronautics and Astronautics (AIAA).
18. Katukuri M. Target-interception feasibility assessment methodology for air-launched weapon systems. In: *2023 3rd international conference on range technology (ICORT)*, Balasore, India, 23–25 February 2023, pp. 1–5. New York: IEEE.
19. Qian C, Han B, Tang Y, et al. Missile attack zone fitting based on K-SAE-SVM. In: *2022 2nd international conference on networking systems of AI (INSAI)*, Shanghai, China, 14–15 October 2022, pp. 209–215. New York: IEEE.
20. Ozdemir MR, Cevher L and Ertekin S. AI-based air-to-surface mission planning using predictive launch acceptability region approach. In: *2021 international conference on military technologies (ICMT)*, Brno, Czech Republic, 8–11 June 2021, pp. 1–8. New York: IEEE.
21. Birkmire BM. *Weapon engagement zone maximum launch range approximation using a multilayer perceptron*. Master's Thesis, Wright State University, Dayton, OH, 2011.
22. Hui Y, Nan Y, Chen S, et al. Dynamic attack zone of air-to-air missile after being launched in random wind field. *Chin J Aeronaut* 2015; 28(5): 1519–1528.
23. Shi Z, Liang X, Zhang J, et al. Modeling and simulation analysis on three-dimensional air-to-air missile attack zone of two aircrafts. In *2018 IEEE CSAA guidance, navigation and control conference (CGNCC)*, Xiamen, China, 10–12 August 2018, pp. 1–5. New York: IEEE.
24. Yue L, Xiaohui Q, Xiaodong L, et al. Deep reinforcement learning and its application in autonomous fitting optimization for attack areas of UCAVs. *J Syst Eng Electron* 2020; 31(4): 734–742.
25. Harini Venkatesan R and Sinha NK. A new guidance law for the defense missile of nonmaneuverable aircraft. *IEEE Trans Control Syst Technol* 2015; 23(6): 2424–2431.
26. Dantas JPA, Geraldo D, Medeiros FLL, et al. Real-time surface-to-air missile engagement zone prediction using simulation and machine learning. In: *Proceedings of the Interservice/Industry Training, Simulation and Education*

- Conference (IITSEC). Orlando, FL: National Training and Simulation Association (NTSA).
27. Kung CC and Chiang FL. A study of missile maximum capture area and fighter minimum evasive range for negotiation team air combat. In: *2015 15th international conference on control, automation and systems (ICCAS)*, Busan, South Korea, 13–16 October 2015, pp. 207–212. New York: IEEE.
 28. Scukins E, Klein M, Kroon L, et al. Monte Carlo tree search and convex optimization for decision support in beyond-visual-range air combat. In: *2023 international conference on unmanned aircraft systems (ICUAS)*, Warsaw, 6–9 June 2023, pp. 48–55. New York: IEEE.
 29. Alkahrer D and Moshaiov A. Dynamic-escape-zone to avoid energy-bleeding coasting missile. *J Guid Control Dyn* 2015; 38(10): 1908–1921.
 30. Ryu H, Bai JH, Tahk MJ, et al. No-escape envelope with field of regard constraint using gradient-based direct method for pursuit-evasion games. *Int J Aeronaut Space Sci* 2018; 19: 675–684.
 31. Scukins E, Klein M, Kroon L, et al. Deep learning based situation awareness for multiple missiles evasion. arXiv:2402.10101 2024.
 32. Yang R, Huang FM and Gong HJ. Best attack position model for BVR multi-target air combat. *Adv Mater Res* 2014; 1016: 511–515.
 33. Gao W, Yang Z, Sun Z, et al. Real-time calculation of tactical control range in beyond visual range air combat. In: *2022 IEEE international conference on unmanned systems (ICUS)*, Guangzhou, China, 28–30 October 2022, pp. 76–80. New York: IEEE.
 34. Gao W, Yang Z, Huang J, et al. Design of dynamic tactical control range in air combat based on three-party game. In: *2023 35th Chinese control and decision conference (CCDC)*, Yichang, China, 20–22 May 2023, pp. 3946–3951. New York: IEEE.
 35. Sun Y, Wang X, Wang T, et al. Modeling of air-to-air missile dynamic attack zone based on Bayesian networks. In: *2020 Chinese automation congress (CAC)*, Shanghai, China, 6–8 November 2020, pp. 5596–5601. New York: IEEE.
 36. Air Land Sea Application Center. Brevity: multi-service tactics, techniques, and procedures for multi-service brevity codes, 2020. https://armypubs.army.mil/epubs/DR_pubs/DR_a/ARN37654-ATP_1-02.1-000-WEB-1.pdf
 37. Aamir Raza M and Liang W. Robust performance optimization of dual thrust rocket motor. *Aircr Eng Aerosp Technol* 2012; 84(4): 244–251.
 38. Ma Y, Guo J and Tang S. High angle of attack command generation technique and tracking control for agile missiles. *Aerosp Sci Technol* 2015; 45: 324–334.
 39. Kumar VRS, Raghunandan BN and Kawakami T. Studies on internal ballistics of dual-thrust motors for nozzleless propulsion. In: *46th AIAA/ASME/SAE/ASEE joint propulsion conference & exhibit*, Nashville, TN, 25–28 July 2010. Reston, VA: American Institute of Aeronautics and Astronautics (AIAA). <https://arc.aiaa.org/doi/10.2514/6.2010-6842>
 40. Park BG, Kim TH and Tahk MJ. Range-to-go weighted optimal guidance with impact angle constraint and seeker's look angle limits. *IEEE Trans Aerosp Electron Syst* 2016; 52(3): 1241–1256.
 41. Dantas JPA, Costa AN, Gomes VCF, et al. ASA: a simulation environment for evaluating military operational scenarios, 2022. <https://arxiv.org/abs/2207.12084>
 42. Dantas JPA, Geraldo D, Costa AN, et al. ASA-SimaaS: advancing digital transformation through simulation services in the Brazilian air force. In: *Simpósio de Aplicações Operacionais em Áreas de Defesa (SIGE2023)*, p. 6. São José dos Campos, Brazil: Instituto Tecnológico de Aeronáutica (ITA). https://www.sige.ita.br/edicoes-antiores/2023/st/235455_1.pdf
 43. McKay MD, Beckman RJ and Conover WJ. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics* 2000; 42(1): 55–61.
 44. Helton JC and Davis FJ. Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. *Reliab Eng Syst Saf* 2003; 81(1): 23–69.
 45. Husslage BG, Rennen G, Van Dam ER, et al. Space-filling Latin hypercube designs for computer experiments. *Optim Eng* 2011; 12: 611–630.
 46. Dantas JPA, Silva SR, Gomes VCF, et al. AsaPy: a Python library for aerospace simulation analysis. In: *Proceedings of the 2024 ACM SIGSIM conference on principles of advanced discrete simulation: SIGSIM-PADS'24*, Atlanta, GA, 24–26 June 2024, pp. 15–24. New York: Association for Computing Machinery.
 47. Vergez PL and McClendon JR. Optimal control and estimation for strapdown seeker guidance of tactical missiles. *J Guid Control Dyn* 1982; 5(3): 225–226.
 48. Vergez PL. Tactical missile guidance with passive seekers under high off-boresight launch conditions. *J Guid Control Dyn* 1998; 21(3): 465–470.
 49. Ball RE. *The fundamentals of aircraft combat survivability analysis and design*. Reston, VA: American Institute of Aeronautics and Astronautics, 2003.
 50. Alin A. Multicollinearity. *WIREs Comput Stat* 2010; 2(3): 370–374. <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.84>
 51. Kazil J and Jarmul K. *Data wrangling with Python: tips and tools to make your life easier*. Sebastopol, CA: O'Reilly Media, 2016. <https://books.google.com.br/books?id=OmeDCwAAQBAJ>
 52. Doane DP. Aesthetic frequency classifications. *Am Stat* 1976; 30(4): 181–183.
 53. Kuhn M and Johnson K. *Feature engineering and selection: a practical approach for predictive models*. Boca Raton, FL: CRC Press, 2019.
 54. James G, Witten D, Hastie T, et al. *An introduction to statistical learning* (Vol. 112). Cham: Springer, 2013.
 55. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: machine learning in Python. *J Mach Learn Res* 2011; 12: 2825–2830.

56. Goodfellow I, Bengio Y and Courville A. *Deep learning*. Cambridge, MA: MIT Press, 2016, <http://www.deeplearningbook.org>
57. Reed R and Marks II RJ. *Neural smithing: supervised learning in feedforward artificial neural networks*. Cambridge, MA: MIT Press, 1999.
58. Priddy KL and Keller PE. *Artificial neural networks: an introduction* (Vol. 68). Bellingham, WA: SPIE Press, 2005.
59. Bonaccorso G. *Machine learning algorithms*. Birmingham: Packt Publishing Ltd, 2017.
60. Abadi M, Agarwal A, Barham P, et al. TensorFlow: large-scale machine learning on heterogeneous systems, 2015, <https://www.tensorflow.org/>
61. Bock S and Weiß M. A proof of local convergence for the Adam optimizer. In: *2019 international joint conference on neural networks (IJCNN)*, Budapest, Hungary, 14–19 July 2019, pp. 1–8. New York: IEEE.
62. Kingma D and Ba J. Adam: a method for stochastic optimization. In: *International conference on learning representations (ICLR)*, San Diego, CA, 7–9 May 2015.
63. Masters D and Luschi C. Revisiting small batch training for deep neural networks, 2018. <https://arxiv.org/abs/1804.07612>.
64. Bengio Y. Practical recommendations for gradient-based training of deep architectures. In: Montavon G, Orr GB and Müller KR (eds) *Neural networks: tricks of the trade*. Cham: Springer, 2012, pp. 437–478.
65. Chen T and Guestrin C. XGBoost: a scalable tree boosting system. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining: KDD '16*, pp. 785–794. New York, NY: Association for Computing Machinery. DOI: 10.1145/2939672.2939785.
66. Balci O. Verification, validation, and testing. *Handb Simul* 1998; 10(8): 335–393.
67. Friedman M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *J Am Stat Assoc* 1937; 32(200): 675–701.
68. Nemenyi PB. *Distribution-free multiple comparisons*. PhD Thesis, Princeton University, Princeton, NJ, USA, 1963.

Author Biographies

Joao PA Dantas received his BSc degree in Mechanical-Aeronautical Engineering (2015) and his MSc degree in Electronic and Computer Engineering (2018) from the Aeronautics Institute of Technology (ITA), Brazil.

Currently, he is pursuing his PhD degree at ITA and working as a researcher for the Brazilian Air Force at the Institute for Advanced Studies. His research interests include artificial intelligence, machine learning, robotics, and simulation.

Andre N Costa received his BSc degree in Mechanical-Aeronautical Engineering (2014) and his MSc degree in Electronic and Computer Engineering (2019) from the Aeronautics Institute of Technology (ITA), Brazil. Currently, he is a researcher for the Brazilian Air Force at the Institute for Advanced Studies, with research interests in modeling and simulation, artificial intelligence, and machine learning.

Diego Geraldo received his BSc degree from the Brazilian Air Force Academy (2005) and became a fighter pilot (2006). He earned his MSc degree in Aeronautical and Mechanical Engineering (2012) from the Aeronautics Institute of Technology (ITA), Brazil. Since then, he has been a researcher for the Brazilian Air Force at the Institute for Advanced Studies, specializing in modeling, simulation, and artificial intelligence.

Marcos ROA Maximo received his BSc degree (Hons.) (summa cum laude) in Computer Engineering (2012), and his MSc (2015) and PhD (2017) degrees in Electronic and Computer Engineering from the Aeronautics Institute of Technology (ITA), Brazil. He is currently a Professor at ITA, being a member of the Autonomous Computational Systems Laboratory (LAB-SCA) with research interests in robotics and artificial intelligence.

Takashi Yoneyama received his BSc degree in Electronic Engineering (1975) from the Aeronautics Institute of Technology (ITA), Brazil, his MD degree in Medicine (1993) from Taubate University, Brazil, and his PhD degree in Electrical Engineering (1983) from Imperial College London, United Kingdom. He is a Professor of Control Theory with the Department of Electronics at ITA. His research is focused on stochastic optimal control theory.