ISSN: 2508-7894
KJAI website: http://accesson.kr/kjai
doi: http://dx.doi.org/10.24225/kjai.2024.12.2.17

Air-Launched Weapon Engagement Zone Development Utilizing SCG (Scaled Conjugate Gradient) Algorithm

Hansang JO[1], Rho Shin MYOUNG[2]

Received: April 02, 2024. Revised May 11, 2024. Accepted: June 05, 2024.

Abstract

Various methods have been developed to predict the flight path of an air-launched weapon to intercept a fast-moving target in the air. However, it is also getting more challenging to predict the optimal firing zone and provide it to a pilot in real-time during engagements for advanced weapons having new complicated guidance and thrust control. In this study, a method is proposed to develop an optimized weapon engagement zone by the SCG (Scaled Conjugate Gradient) algorithm to achieve both accurate and fast estimates and provide an optimized launch display to a pilot during combat engagement. SCG algorithm is fully automated, includes no critical user-dependent parameters, and avoids an exhaustive search used repeatedly to determine the appropriate stage and size of machine learning. Compared with real data, this study showed that the development of a machine learning-based weapon aiming algorithm can provide proper output for optimum weapon launch zones that can be used for operational fighters. This study also established a process to develop one of the critical aircraft-weapon integration software, which can be commonly used for aircraft integration of air-launched weapons.

Keywords: Air-launched Weapon, Mission System, Machine Learning, Weapon Engagement Zone, SCG, Scaled Conjugate Gradient

Major Classification Code: Artificial Intelligence, Aerospace Engineering, etc.

1. Introduction

With the F-16 and F-18, which represent the 4th generation fighters, the development of fighter jets prioritized their maneuverability; however, current technologies are focused on avionics, sensors, and weapons integration to improve the mission capabilities of fighter jets. The representative standard capability that distinguishes general aircraft from fighter jets is combat power, and the most important factor in evaluating combat power can be represented by the weapons that each fighter can operate. Until World War II, the installation of more weapons showed combat capabilities, and recently, how actively smart weapons can be used has become a standard for evaluating a fighter's combat effectiveness (Jung, 1998). This can also be seen through the birth and development history of the world's best-selling F-16, which started with the YF-16. The F-16, which was created as a light fighter, started with short-range air-to-air missions and was confirmed to have mid-range air-to-air capabilities, and its air-to-ground attack capabilities were also gradually expanded.

Air-launched weapons integration should be considered in three categories: physical integration, electrical

1 First Author. Graduate Student, Aerospace Convergence Engineering, Gyeongsang National University, South Korea. Email: hansang.jo@gnu.ac.kr
2 Corresponding Author. Professor, Aerospace Engineering, Gyeongsang National University, South Korea. Email: myong@gnu.ac.kr

integration, and logical integration. The first physical integration is in the areas of typical structures and aerodynamics for carrying and flying with those weapons. Therefore, physical integration is the basis of all weapons integration and is a representative field of aircraft development for weapons operations. The second electrical integration has been specified through the U.S. military standard MIL-STD-1760E. Air-launched weapons developed before this standard was established had no standards for interface with aircraft, so each weapon used its own interface. To mount weapons on an aircraft, modification of the physical interface of the aircraft or weapon was required. The interoperability standard established to overcome these problems in system integration between existing aircraft and weapons and to minimize the time and cost required for integration between aircraft and weapons is MIL-STD-1760E (Lee, 2018). Interoperability between aircraft and weapons has already been ensured in the United States and most European countries and all new weapons are being developed in compliance with these specifications.

On the other hand, logical integration is representative of guided weapons integration technology that is different from existing unguided weapons integration. The difference between weapons is large, so standardization is difficult, and the aircraft system integration schedule is also the longest of all weapons integration periods. This field is a technology that defines the data flow between aircraft and weapons in chronological order to adjust each function of the weapons. Logical integration generally refers to software integration between aircraft and weapons and is achieved through interfaces universalized by the U.S. military standard MIL-STD-1553B. Through the previously mentioned electrical interface MIL-STD-1760E, representative messages between the aircraft and weapons for logical integration have been defined, but the detailed interface has been defined in each Weapon ICD individually in consideration of the characteristics of the weapons.

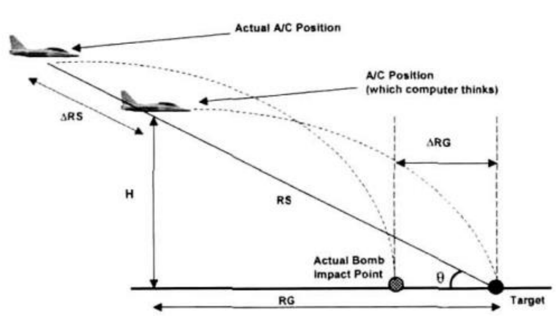

Figure 1: Ballistic Trajectory and Impact Point Estimation

For traditional unguided bomb operation, aircraft calculate the bomb's accurate free-fall trajectory (ballistic trajectory) and provide the pilot with the optimal release point based on this (Jo, 2007). However, recently, the accuracy of guided weapons has been confirmed as an expanded launch zone rather than the estimated bomb impact point, and the guidance algorithm of the weapon itself and the navigation synchronization (transfer alignment) of the aircraft and the weapon before release are getting more important.

There is one more thing to consider weapons integration on the aircraft. Technology for predicting the optimal weapon launch area for guided weapons and providing relevant information to pilots in real-time also falls under the category of logical integration, and the applicable technologies and fields for this are different from the previously mentioned fields. The technologies that have been used so far are divided into three types. The first is the previously mentioned free fall trajectory prediction, which has been mainly used for unguided bombs. Second, in the case of guided bombs or missiles, the launch area of the weapon is organized in the form of a table (look-up table) that is divided by combining the approximate speed, altitude, and distance between the target and the aircraft, and is executed in real-time on the aircraft. However, this table should be simplified for real-time conditions (Ryoo, 2014). The third technique that has been used is mainly for high-mobility missiles that are difficult to predict and whose various paths are difficult to standardize. It is a method of simulating a guidance algorithm similar to an actual weapon in real time on an aircraft. Although the pilot can predict the optimal launch time of the weapon, existing methods force the algorithms to be simplified to ensure real-time for increasingly complex navigation guidance characteristics of the weapons, which inevitably tends to reduce the accuracy of the predicted trajectory (Yoon, 2010).

In this paper, we use machine learning technology used in artificial intelligence to develop an algorithm that can satisfy both accuracy and real-time (Warwick, 2016) and will be quickly integrated into aircraft when developing many newly developed air-launched weapons in the future. We also would like to present a method to formalize the weapon launch zone algorithm integration process.

2. Development of Machine Learning-based Weapon Aiming Algorithm

2.1 Data Generation for Machine Learning

Due to the increasingly complex nature of the combat system, existing repetitive calculations have limitations in securing the real-time performance of the mission computer in battlefield situations. Due to the nature of embedded software, resources are limited, but the number of calculations continues to increase, causing problems in which calculations cannot be completed within a limited period, or the time that should be allocated to other calculations cannot be sufficiently allocated. To solve this problem, the number of cases of embedded software development that secures real-time performance by using pre-learned results is increasing. In the case of US fighter aircraft, the F-16, and F-15, the efficiency of the development of highly complex weapon targeting algorithms previously developed by each aircraft manufacturer has recently been improved through specialized companies. FACC company learned each situation in advance using data and simulation models acquired from weapons development companies and built a ZAP (Zone Acquisition Program) solution that can quickly predict the optimal firing area in actual battlefield situations. Compared to the existing method that reflects the characteristics of the weapon through specific coefficients, it boasts prediction accuracy and fast calculation speed and is installed in the mission computer operation software (OFP) of the latest US military fighter jets such as the F/A-18, F-35, and F-22. It is used as a standard for air-launched weapons cueing algorithms. In this way, the development of a weapon-aiming algorithm through machine learning has the advantage of providing a faster and more accurate solution than existing methods. However, for this to happen, it is important to secure actual weapon data (TDS: Truth Data Set) that can be used as learning and verification targets. In general, TDS is a process of creating and formalizing the footprint of a weapon under each condition using a six-degree-of-freedom model of the weapon.

In this study, the fly-out model of a virtual air-to-ground bomb was used to compare how well the footprint and launchable area of the weapon generated through learned data matched the fly-out model used for creation. For machine learning, it is necessary to secure sufficient data based on many inputs, such as the conditions of the launching aircraft and the location of the target, as well as the target impact parameters of the weapon. However, since it is practically impossible to secure relevant valid data through the actual launch of the weapon, the 6-degree-of-freedom model used in M&S (Modeling and Simulation) must be used, or a corresponding model (Truth Model) must be used. However, these models are tools that can directly evaluate the performance of weapons, are strictly controlled to protect their technology, and are generally not provided to other countries. Therefore, the development of a weapon-targeting algorithm through machine learning can be used as the most efficient method when integrating domestically developed air-launched weapons into domestic fighter jets.

2.2. Setting Valid Conditions for Data Generation

To build a weapon footprint database, three values must be set. First, the input values of the aircraft are the altitude, speed, and descent angle of the aircraft at the time of launch. Aircraft conditions must be set to applicable categories for each aircraft and each armament combination. Second, the target's position value can be defined as a relative position based on the aircraft or can be created based on absolute coordinates (latitude, longitude). The third is the collision conditions that must be achieved when the weapon reaches the target, including impact angle, impact azimuth, and impact velocity. This third condition can only be considered with the latest guided weapons. In the case of existing general-purpose bombs, they were launched only by checking how far the bomb could fly. However, the latest guided armament aims to maximize the destruction of the target (In-Zone) in addition to the maximum distance the armament can reach (In-Range) based on current aircraft conditions. To improve the mission success rate, pilots tend to use the In Zone as the optimal launch area. Accordingly, the machine learning-based weapon aiming algorithm must provide the pilot with two possible weapon launch areas.

Table 1: Weapon Delivery Conditions

Category	Parameters	Test range
Delivery condition	Altitude	2,000~50,000ft
	Velocity	160KCAS ~ M0.9
Impact condition	Angle	15~90deg
	Azimuth	-135~135deg

Table 1 shows the conditions to create data for learning in this study. Altitude and speed were generally based on 2,000~50,000ft, 160KCAS to M1.3, at which an aircraft can launch a guided bomb, and the release angle of the aircraft was within 5 degrees of horizontal flight to limit the range of data to be studied. This has the advantage of improving accuracy only in the case of general delivery conditions but has the limitation that the launch area of the armament cannot be presented if it is outside the applicable range. Therefore, when extracting algorithm learning target data, it is important to generate a data set by considering these strengths, weaknesses, and characteristics of weapons.

Additionally, in cases where the weapon launch area cannot be displayed to the pilot due to data limitations, the role of the algorithm is to inform the pilot of the reason and guide the pilot so that he can fly within stable weapon release conditions. Similarly, the selection of target impact conditions is also important. As mentioned earlier, the latest guided weapons can improve their target probability kill rate

(Pk) and mission success rate (Ps) by setting the conditions for when the weapon reaches the target. For example, to destroy a target surrounded by a solid concrete wall, the armament must fall perpendicularly to the target and fly along a flight path to maximize the impact speed; however, which conditions limit the launchable distance of the armament. In addition, when it is necessary to attack the rear of a target like North Korea's long-range artillery, the glide length of the weapon must be longer, but the weapon must be capable of setting a flight path that can strike from the rear. Therefore, in the armament targeting algorithm, it is necessary to set learning data for the impact conditions between the armament and the target according to the characteristics of the armament.

In this study, the scope was limited to the level of general GPS/inertial guided bombs. This was limited to an impact angle of 15 to 90 degrees and a collision azimuth angle of -135 to 135 degrees. The reason for limiting the impact angle to 15 degrees as the minimum angle is because, unlike a missile, in the case of a regular bomb, when it collides with a target at a low impact angle, it has the effect of slipping instead of exploding and causing a misfire. The reason for limiting the azimuth angle is that in the case of regular bombs without thrust, the armament capable of attacking the target from behind is limited. Additionally, collision speed was not considered in this study because the warhead used in this study was targeted at the 500-pound bomb class, which is the easiest to obtain in Korea. In the future, impact speed is a factor that must be considered when creating algorithms for penetrating bombs such as BLU-109.

3. Algorithm Generation

3.1. Prediction Using SCG algorithm

The algorithm creation software consists of a total of four parts: the algorithm creation user interface, the algorithm creation main function, the footprint validity learning function, and the footprint learning function. The user interface was configured as shown in the left picture of Figure 2, and neural network learning provided by MATLAB was selected and applied. The artificial intelligence algorithm to be learned, number of repetitions, time limit, and data sampling are received through user input, and learning can be performed as a whole or individually.

Using the Gaussian process technique has the advantage of using a small amount of data and being able to check the accuracy of the data through the algorithm itself. However, taking real-time into account and considering that it is a domestically developed weapon, it is possible to secure a large amount of data, in the study, machine learning using the neural network algorithm was used. As shown in the right window of Fig. 2, when learning is performed, the Neural Network Toolbox is executed and learning progresses. The detailed algorithm used is a Scaled Conjugate Gradient (SCG) among the supervised learning provided by MATLAB. SCG consists of the Conjugate Gradient algorithm with Line search (CGL) and Broyden-Fletche-Goldfarb-Shanno (BFGS) and is famous for fast convergence speed which can avoid exhaustive line searches used repeatedly to determine the appropriate stage and size of machine learning [Meller, 1990]. SCG algorithm is also fully automated, includes no critical user-dependent parameters, and avoids a time-consuming line search. This phenomenon is beneficial for applying this process to any other weapon systems.

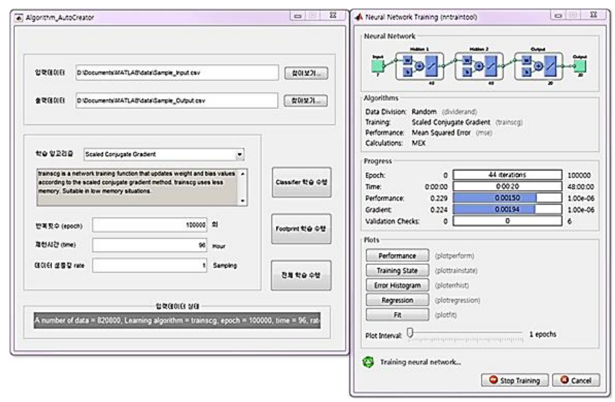

Figure 2: Algorithm Generation

3.2. Algorithm Optimization For No-overfitting

The problem that must be considered first in data learning through machine learning methods is overfitting. Overfitting is a problem in which a model shows a certain level of prediction accuracy within the training data, but its accuracy drops significantly when new data is applied. Since this study needs to predict the weapons delivery area during actual combat, a decrease in accuracy due to overfitting can be very fatal. Overfitting occurs in two main cases. The first is when the variables are too complex, and the second usually occurs when the dataset used for learning and the dataset that verifies the learned results overlap. In this study, to prevent overfitting in the first case, data generation was optimized with aircraft release conditions and target impact conditions that can actually occur and be practically used for real weapons drop, as shown in Table 1. Additionally, to avoid overfitting during the data learning and verification process, only a portion of the total dataset was used for learning, and the remaining non-redundant dataset was used to evaluate the learning results. If you evaluate the learning results with the test dataset used for learning, you can quickly identify and modify model performance, and accuracy can be easily improved just by increasing the number of executions. However, the same tendency doesn't occur even when using separately classified data, this is evidence that the learned results are overfitted.

Even when the training dataset and the evaluation data set are separated, the prediction rate must be compared to determine when to stop learning as the number of learning iterations and the number of hidden layers increases. If the prediction rate of the training data increases as the number of hidden layers increases, but there is a point where the prediction rate of the test set begins to decrease, this should be judged as overfitting. In other words, while removing under-fitting parts of the machine learning model, learning must be stopped just before over-fitting occurs. In this study, the model was optimized through iterative learning and judgment of overfitting as shown below.

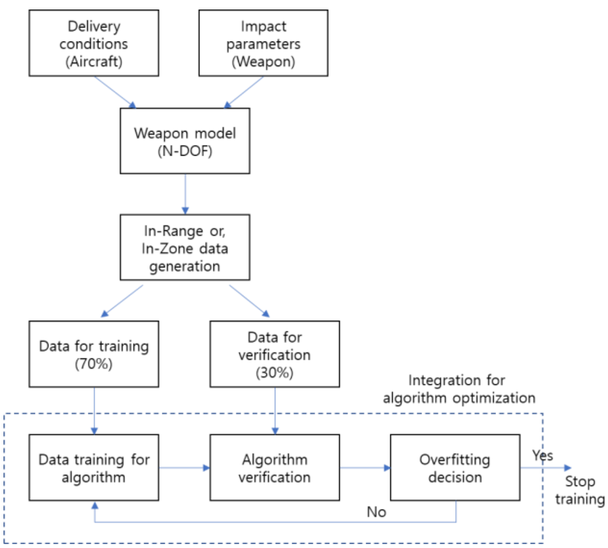

Figure 3: Training Model Optimization

Fig. 3 is the actual use of the data set created in this paper. First, a database using the M&S model of armament was constructed for each aircraft release condition and armament impact condition for In-Range and In-Zone. Only 70% of the total data was used for learning, and the remaining 30% was used for evaluation of the learned model, and optimization was performed to determine whether the entire model was overfitted.

4. Algorithm Development and Verification

4.1. Conservative Extraction of Learned Data

Contrary to overfitting, a conservative approach to the results extracted through learned data is a factor to consider in aircraft integration. The learned data contains error elements, and in particular, the accuracy of determining the true value tends to be relatively low when predicting the weapon release area of the learned edge. Therefore, to utilize the relevant data as an actual weapons release area and present it to pilots in flight as a standard for judging launch timing, the learned data must be approached conservatively. Additionally, the data needs to be simplified to take into account limited mission computer resources and real-time. Taking this into consideration, the weapon release area provided to the aircraft is generally represented as a limited number of polygons, and this area is more conservative than the data generated by actual learning, so the pilot be confident that he/she would get the results you want dropping weapons within this area.

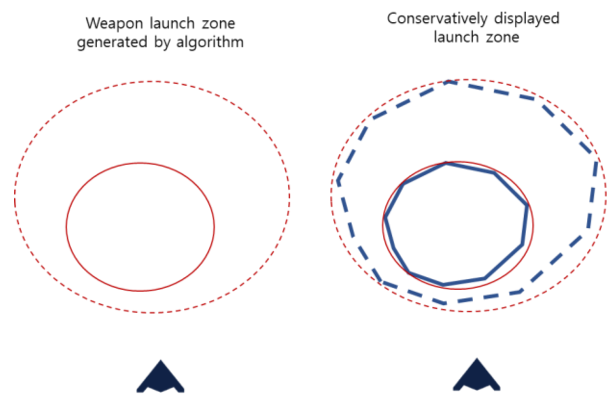

Figure 4: Display Data Conservatism

In Fig. 4, the outer dotted line on the left shows the In Range area created through machine learning, and the inner solid line area is the In Zone. In general, when an impact condition is added as an additional condition, the area tends to become smaller. The second picture on the right side shows a method of extracting the relevant areas into polygons of n points. Polygons must be created with a conservative approach to improve data reliability, and considering this conservatism and calculation speed, the weapons launch acceptable area provided to pilots in actual flight uses a polygonal shape as shown in the picture on the right. Since there is uncertainty in the learning data at the boundary of the data generated through learning, the learned data is extracted by simplifying the data into a smaller area than the machine learning model result to ensure a sufficient mission success rate and perform the mission in that area. This is because it is presented that the success rate can be guaranteed.

4.2. Generated Algorithm Rresults and Verification

The algorithm was also verified by comparing the footprint of the algorithm and the weapon truth data used for this study. Figure 5 is the result of comparing the truth data of weapons generated under the same delivery conditions (left) and the results learned through this study (right) on the same scale. The dotted line represents the In Range, and the internal solid line represents the In Zone area.

The second graph is the result of comparing the same results between In Zone and In Range, respectively. The red line shows algorithm algorithm-generated weapon launchable area and the blue line is the truth data that was used for machine learning. The reason why the comparison of the In Range on the right appears relatively accurate is because the actual footprint size is large, and in the case of the In Zone, not only the distance the bullet reaches but also more parameters are required compared to the In Range to set the impact condition. In Zone, the learned data tended to be somewhat biased to the left, and in In Range, it can be seen that overall, the learned data converged well into the actual data (Truth Data).

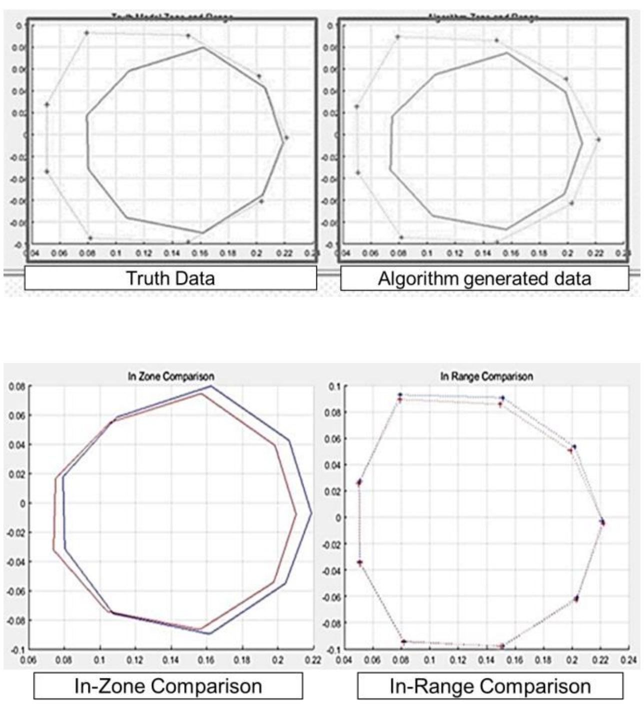

Figure 5: Output Data Verification

With the technological development of the latest weapons, it is not only difficult to accurately predict the weapon impact point using existing methods, but it is also becoming more difficult to provide optimal launch conditions to pilots in real time by considering massive amounts of data. This limitation is evident in aerial weapons where shooter aircraft and targets move at high speeds. To solve this problem, this study proposed a method that can provide pilots with a fast and accurate solution through machine learning based on a neural network algorithm (Dantas, 2021). Through the results of this study, the development of a weapon aiming algorithm using a SCG algorithm showed improved results in both accuracy and real-time assurance compared to previously existing methods.

5. Conclusion

Integration of new weapons into aircraft is divided into physical integration, electrical integration, and logical integration. Among these, there is no standardization, and the most time-consuming part is logical integration, which includes software integration between weapons and aircraft. For this software integration, the development of weapons aiming algorithms is an area related to the direct performance of weapons, which means that technology transfer and data acquisition are limited when integrating non-domestic weapons. This study selected the SCG method to develop an air-launched weapon aiming algorithm because it is fully automated, and includes no critical user-dependent parameters. In summary, this study established a process to develop one of the critical aircraft-weapon integration software, a fast and accurate weapon aiming algorithm that can be commonly used for aircraft integration of domestic air-launched weapons that will be continuously developed in the future.

References

Jung, K. J. (1998). A study on the mission effectiveness analysis for air-to-air/air-to-surface of a light fighter. *Journal of the KIMST, 1*(1), 92-113.

Lee, S. G. (2018). A software update technique for aircraft missiles based on MIL-STD-1760. *Journal of the KIMST, 21*(5), 649-657.

Jo, H. S. (2007). A study on the accuracy analysis for air-to-ground weapon. *Journal of The Korean Society for Aeronautical and Space Sciences, 35*(8), 741-746.

Ryoo, C. K. (2014). Computation algorithm for launch acceptability region of air-to-surface missiles. *Journal of The Korean Society for Aeronautical and Space Sciences, 42*(9), 762-772.

Yoon, K. S., Kim, I. G., & Ryu, K. S. (2010). New modeling algorithm for improving accuracy of weapon launch. *29th Digital Avionics Systems Conference*.

Warwick, G., & DiMascio, J. (2016). Targeted learning: Artificial-intelligence techniques are being explored for fast, accurate targeting in complex combat environments. *Aviation Week & Space Technology*, 45.

Meller, M. F. (1990). A scaled conjugate gradient algorithm for fast supervised learning (ISSN 0105-8517). University of Aarhus.
Dantas, J. P., & Costa, A. N. (2021). Weapon engagement zone maximum launch range estimation using a deep neural network.