9. ANKARA INTERNATIONAL AEROSPACE CONFERENCE **AIAC-2017-127**
20-22 September 2017 - METU, Ankara TURKEY

AIR COMBAT RULE MINING FROM MOVEMENT SEQUENCE OF FLIGHTS

Mustafa Karli[1], Mehmet Önder Efe[2] and Hayri Sever[3]
Hacettepe University
Ankara Turkey

ABSTRACT

Decomposing real flight route into movement sequences allows learning air combat among multiple aircrafts. Machine learning techniques or control systems can be designed to choose a maneuver for any given relative condition between to opponents. But each relative condition forces different maneuver logic that requires specific domain information. This paper shows classifying movement decisions of real flight under certain relative variables and generates a rule base using decision trees.

INTRODUCTION

Basic Fighter Maneuvers [Shaw,1985] are composed of movement sequences in three dimensional air space that results moving to advantageous position to shoot opponent. Keeping the enemy in the effective weapon zone results a success.

Rule based systems [Burgin and Sidor, 1988] for jet fighters addresses some set of BFMs. It is not new to use artificial intelligence methods [McManus, 1990 and McManus and Goodrich, 1989 and Rodin and Amin, 1992] to support human pilots. A simple close range logic is implemented in [Karli, Efe and Sever, 2015] to preserve energy during combat. Using ANFIS as auto-pilot is studied in [Konakoğlu , Kaynak, 2006 and Çetin, Kaynak, 2010] for static routes of autonomous flight control. [Karli, Efe and Sever, April 2017] defines a method to extract movement sequence from a real F-16 flight information and proposes an ANFIS for air combat learning in another study in [Karli, Efe and Sever, July 2017] utilizing movement sequence learning without enough learning corpus. We neglect the latency of information about the opponent for high level decisions and assume that we have enough positional data to calculate relative geometry.

Artificial neural network systems perform a good learning mechanism based on previous experience while rule based systems determine learning path based on domain information. ANFIS merges fuzzy classification into neural architecture and results a perfect learning. But study in [Karli, Efe and Sever, July 2017] handles only 13 rules obtained from domain information in [Burgin and Sidor, 1988]. Without enough rules, ANFIS may not result enough learning either. This study shows how to derive new rules from F-16 combat scenarios. These rules can be merged into ANFIS architecture later on.

Our work advances air combat learning in terms of decomposing flight into movement sequences, generating relative geometry, grouping relative variables and angular changes, creating decision tree for move selection and extracting BFM rules composed of 3D movement sequences. The decision tree is generated and tested using real F-16 flight information. We assume that aircrafts have a low level robust control system, so flight control system is out of our scope. Also agile aircraft characteristics resulting non-linear equations is not validated because data is already obtained from real F-16 aircrafts.

---

[1] PhD. student in Computer Engineering Email: mustafa.karli@hacettepe.edu.tr

[2] Prof. in Computer Engineering, Department Chief Email: onderefe@gmail.com

[3] Prof. in Computer Engineering Email: hayri.sever@gmail.com

This paper has following sections: In the next section we define the problem. The third section defines how learning corpus structured and obtained. The forth section proposes designing decision tree and generating rule sets. The fifth section discusses the results of rule generation and last section includes future work and conclusion.

PROBLEM DEFINITION

Domain information about air combat proposes executing basic fighter maneuvers (BFM) under certain conditions. These conditions include own state variables and relative geometry between opponents. BFMs are also composed of sequence of simple moves. These simple moves and composing a BFM is studied in [Karli, Efe and Sever, April 2017] where an abstraction stack is proposed as well. Using such an abstraction stack allows us to focus on BFM intelligence instead of dealing with low level control systems.

BFM intelligence for each BFM definition should be studied separately. But choosing the right BFM is another intelligence and requires a decision logic. A simple approach is to say "under such conditions, choose this maneuver". This intelligence can be obtained from combat scenarios of experienced pilots.

The problem is divided into three steps. First step is obtaining combat data for learning and testing. Second step is transformation of combat data into understandable format and third step is generation of rules to select best BFM.

CORPUS

Data

Getting a number of sensor information, F-16 flight computer extracts a huge set of status information every 40 milliseconds. We retrieved positional and angular information which includes north position, east position and altitude, air-speed, roll angle, pitch angle, heading angle. The angular difference between body and velocity directions and momentum of roll, pitch and heading angles are calculated later. These state variable are;

$$X = \{n, e, h, v, \varphi, \theta, \psi, \alpha, \beta, P, Q, R\} \quad (1)$$

Movement Sequence

The columnar data retrieved every 40 milliseconds is not easy to understand, learn or process. This data should be converted into a more understandable and meaningful format with less footprint. So instead of handling data with repeating state variables, data is converted into series of movements. The method for decomposition is expressed in [Karli, Efe and Sever, April 2017] more detail. The final data format of aircraft movement σ and flight information F is converted to data format below;

$$\sigma=\{t, q, v, \theta, \psi\} \quad (2)$$

$$F=\{ \sigma_1, \sigma_2, .. \sigma_n \} \quad (3)$$

Relative Geometry

In air combat, there are minimum 2 aircrafts involved. So there should be state variables of the two. The combat status of both depends on the relative geometry to each other. Relative geometry and advantage function is discussed in [Karli, Efe and Sever, 2015] and includes range, closure velocity, altitude delta, heading crossing angle and antenna train angles of two aircrafts.

$$R=\{r, V_c, h, \lambda, \eta_1, \eta_2\} \quad (4)$$

METHOD

Air combat learning is a type of machine learning problem. The classification or learning process varies according to standpoint of the mission. In air combat pilots cannot follow a pre-defined path. They have to check the relative position to the opponent and make instant movement decisions. The agility of combat aircraft limits the planning horizon to millisecond level. A method used in [Karli, Efe and Sever, July 2017] is to design an ANFIS model. The model uses movement sequences of both and relative geometry to each other as input, define membership functions for each input, based on the input membership follow some rules which are derived from domain information, finally fine-tune the control signals of the chosen movement at the de-fuzzification step.

ANFIS model combined both neural and fuzzy learning system into single architecture. Generic sugeno [Sugeno, 1985] style rules implementation is shown in Figure 1.

$$If\ x_1\ is\ M_1,\ x_2\ is\ M_2,\ ..\ x_n\ is\ M_n\ then\ y=x_0+k_1x_1+...+k_nx_n \quad (5)$$

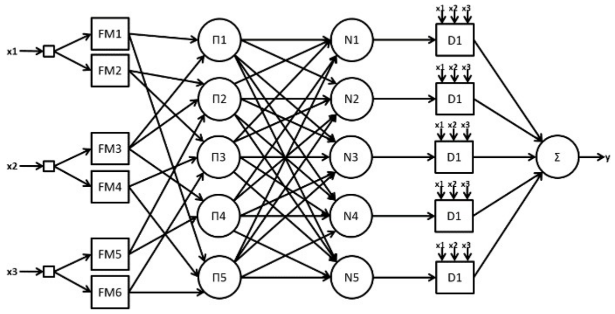

Figure 1 : ANFIS Template

The above method is shown to perform well for rules derived from domain information. But success in air combat is hidden behind the philosophy of unpredictability. The method implies to execute a small set of predicted maneuver each time. The war fighters claim that the books are to learn the basics. But in real world scenarios, pilots improve their own experience and hidden tactics. This experience is recorded every 40 milliseconds and may allow us to derive more combat rules that cannot be learned from domain information.

Decision Tree

Decision tree is a good method to analyze and group a large set of data with multiple variables. The method divides the input data into sets for each variable value and the leaf nodes include the resulting classification. For our case; the tree will include resulting maneuver following value sets of state and relative geometry. A sample tree with branch and leaf nodes is show in Figure 2.

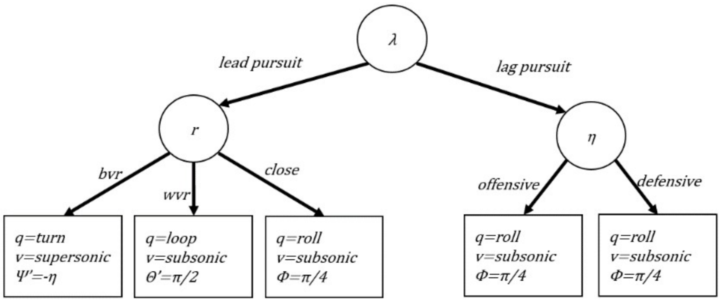

Figure 2 : Decision Tree Template

The C4.5 algorithm by [Quinlan, 1993] is used as tree model in this work. An implementation of the algorithm can be found in [Coenen, 2007]. The algorithm defines a binary tree and has a left and right branch for each node. Each node compares a single value. Algorithm automatically weighs and balances the tree structure while adding new values to the tree.

Feature Set

The feature set defines the input value types. Each feature may have a value from a pre-defined set. The values are discrete and any fuzzy membership is not used. The feature sets are explained below.

Heading Crossing Angle (Angle Off) (λ): Heading crossing is the angle between two velocity vectors at the point of intersection. This angle determines the pursuit rate of the aircrafts. Angle off lying behind the aircrafts shows neutral and greater than 60 degrees shows a kamikaze position. Table 1 lists the value ranges for classification of heading crossing angle.

Table 1 : Heading Crossing Angle Classification

Label	Value Range	Index
Low	0..30 degrees	1
High	30..60 degrees	2
Kamikaze	60..180 degrees	3
Neutral	-180..0 degrees	4

Antenna Train Angle (Aspect Angle) (η): Antenna train angle is the angle between the aircraft velocity and range vectors. Each aircraft has its own aspect angle. This angle shows how close the aircraft is pointing the opponent. Thus lower values of this angle has better advantage. Since cosine of 0 is 1 and cosine of 180 degrees is -1, cosine of this angle is assumed to be the angular advantage function over the other. Table 2 lists the value ranges for classification of aspect angle.

Table 2 : Aspect Angle Classification

Label	Value Range	Index
Low	0..30 degrees	5
Medium	30..60 degrees	6
High	60..90 degrees	7
Defensive	90..180 degrees	8

Range (r): Range is the Euclidian distance between two aircrafts. This value is divided into 3 intervals. Beyond visual range has the case of initial setup positions and requires approaching to each other. Within visual range is where air combat maneuvering occurs. Collusion should be avoided unless kamikaze maneuver is allowed. Table 3 lists the value ranges for classification of range.

Table 3 : Range Classification

Label	Value Range	Index
Collusion	0..300 meters	9
Within Visual Range	300..10000 meters	10
Beyond Visual Range	10000+ meters	11

Velocity (v): Velocity is arranged according to the speed of sound. Speed below 0.8 Mach is subsonic, where over 1.2 Mach is supersonic. Table 4 lists the value ranges for classification of velocity.

Table 4 : Closure Velocity Classification

Label	Value Range	Index
Subsonic	0..280 meters/second	12
Sonic	280..400 meters/second	13
Supersonic	400..10000 meters/second	14

Altitude Difference (Δh): Altitude difference shows how higher the aircraft is from the opponent. Higher altitude has more energy advantage. Values less than 100 meters is ignored and assumed to be close to each other. Table 5 lists the value ranges for classification of altitude difference.

Table 5 : Altitude Difference Classification

Label	Value Range	Index
Above	100..30000 meters	15
Close	-100..100 meters	16
Below	-30000..-100 meters	17

Pitch Angle (θ): Pitch is the angle between the velocity vector and the earth surface. Being parallel to the earth either upside down or level flight is assumed to be straight with ignoring -+10 degrees. Positive angles are climbing and negative angles are descending. Table 6 lists the value ranges for classification of pitch angle.

Table 6 : Pitch Angle Classification

Label	Value Range	Index
Straight	[-10..10] and [170..-170] degrees	18
Climb	+10..170 degrees	19
Descend	-170..-10 degrees	20

Move (q): Move is the resulting decision based on the input feature set. There are 6 different movement types with inputs. Table 7 defines movement modes selected from [Karli, Efe and Sever, April 2017] which are initially defined in [Üre and İnalhan, 2009] and modified later in [Üre and İnalhan, 2012].

Table 7 : Movement Modes

Mode	Name	Inputs	Index
SP	Straight Path	$\Delta t, v, \theta$	21
TU	Turn	$\Delta t, v, \theta, \dot{\psi}$	22
LO	Loop	$\Delta t, v, \dot{\theta}, \dot{\psi}'$	23
PY	Pitching Yaw	$\Delta t, v, \dot{\theta}, \dot{\psi}$	24
RO	Roll	$\Delta t, v, \dot{\varphi}$	25
RP	Rolling Pitch	$\Delta t, \dot{\varphi}, \dot{\theta}$	26

RESULTS

We run the C4.5 decision tree algorithm on 6688 records with 52.06% accuracy, 20% support and 80% confidence values. Input has 6 attributes and output has only 1 attribute as movement mode. The rule extraction algorithm extracted total 170 rules from the resulting tree. Speed attribute is neglected after execution since it was seen that pilots are not allowed to fly at supersonic speeds during training. Also 52 rules beyond visual range are ignored since BVR is out of scope for this work. Looping rules where aircraft is already climbing or descending is also ignored since it means to keep doing the same move. Some rules are combined into one rule since they produce the same result in whatever condition one of the input feature is set to. About 60% of the rules were resulting the same maneuver of the opponent because average %40 of a flight was combat engagement while 60% was the neutral flight to the training area. After eliminating irrelevant results, final rule set has 8 elements listed in Table 8. Out of these rules, 7 of them comply with current domain information since they match with the rules in [Burgin and Sidor, 1988]. Last two rules address the same pointing algorithm in close ranges where any action to point the opponent is valid.

Table 8 : List of Derived Rules

λ	η	r	Δh	θ	q	
High	Defensive	Collusion	Above	Climb	Pitching turn	π/2
High	Defensive	WVR	Below	-	Turn	π
High	Defensive	WVR	-	-	Roll	π/2
Neutral	-	Collusion	-	-	Straight path	-
Not neutral	-	Collusion	-	-	Rolling Pitch	π/4
Medium	Offensive	WVR	-	-	Turn	π/4
Low	Offensive	WVR	-	-	Loop	π/4
Low	Offensive	WVR	-	-	Rolling Pitch	π/8

This result seems to be disappointing since even domain rules are not completely found among the results though it was expected to result more rules including complete set of domain information. We had 10 sorties of F-16 flights which include 2 of 1x2 and 2 of 1x1 ACM scenarios. This data set is not enough to achieve desired results. Also we are not sure that every air combat maneuver has been flown or correctly sampled in the data set. But the results show that used method outputs 7/8 accurate rules which are validated by air combat authorities. Using more training and test information, we can get more accurate and hidden air combat maneuvers with the proposed method.

CONCLUSION AND FUTURE WORK

The ANFIS model in [Karli, Efe and Sever, July 2017] utilized 13 pre-defined air combat rules from [Burgin and Sidor, 1988]. Using fixed set of rules yields the system to memorize the maneuvers and result the same predictable action in every air combat. This work shows that we can also merge additional rules derived from real world experience into the learning process.

Preparing the corpus data is a challenge for air combat learning. It is very difficult or nearly impossible to obtain such information from military authorities. We used only 10 sorties of F-16 flight and it gives a very limited view of the process. Generating more accurate rules requires much more than this data set. A practical way can be using simulation software. In any case advanced combat pilots are the rare resource because they are the data source for the combat experience.

Another way of obtaining data is using multiple radar systems which are commercially cheap and available in the market. They get echoes from every object in the air. The intervals may be to large but using lots of radar systems, streaming data into a big data environment and merging and unifying the echoes of the same object can result more reliable and short intervals of position information for different kinds of aerial objects. This can be another future work and data collection method as well.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the facilities of the Autonomous Systems Laboratory at the Department of Computer Engineering of Hacettepe University.

References

Burgin, G., Sidor, L. (1988) *Rule-Based Air Combat Simulation*, Tech. rep., NASA, CR-4160.

Coenen F. (2007) *The lucs-kdd decision tree classifier software*, Department of Computer Science, The University of Liverpool, UK.

Çetin Ö., Kaynak O. (2010) *Adaptive neuro-fuzzy inference system based autonomous flight control of unmanned air vehicles*, Expert Systems with Applications vol 37, p 1229-1234, 2010.

Karli M., Efe M.Ö., Sever H. (2015) *Close range one to one air combat maneuvering for autonomous UAV*, 8th Ankara International Aerospace Conference, 10-12 September 2015, Ankara, Turkey (AIAC2015-046).

Karli M., Efe M.Ö., Sever H. (2017) *Extracting Basic Fighter Maneuvers from Actual Flight Data*, International Conference on Computer and IT Technologies, 12-14 April 2017, Kusadasi, Turkey.

Karli M., Efe M.Ö., Sever H. (2017) *Air Combat Learning from F-16 Flight Information*, Fuzz-IEEE 2017 IEEE Conference on Fuzzy systems, 9-12 July 2017, Naples, Italy.

Konakoğlu E., Kaynak O. (2006) *Autonomous Flight Control of Unmanned Air Vehicles by use of ANFIS*, MSc Thesis, ASTIN, 2006.

McManus J., Goodrich K. (1989) *Application of Artificial Intelligence (AI) Programming Techniques to Tactical Guidance for Fighter Aircraft*, AIAA Guidance, Navigation and Control Conference, August 14-16 1989.

Quinlan J. R. (1993) *C4.5 programs for machine learning*, San Mateo, Morgan Kaufmann.

Rodin E.Y., Amin S.M. (1992) *Maneuver prediction in air combat via artificial neural networks*, Computers & Mathematics with Applications, vol. 24, p.95-112.

Shaw, R. (1985) *Fighter Combat Tactics and Maneuvering*, Naval Institute Press, Annapolis, Maryland.

Sugeno, M. (1985) *Industrial applications of fuzzy control*, Elsevier Science Pub. Co.

Üre N.K., İnalhan G. (2009) *Feasible agile maneuver identification and generation algorithms on multi modal control framework*, AIAA Guidance, Navigation, and Control Conference 10-13 August 2009, Chicago, IL.

Üre N.K., İnalhan G. (2012) *Autonomous Control of Unmanned Combat Air Vehicles*, IEEE Control System Magazine, 74-95, October 2012.