Calculated Moves: Generating Air Combat Behaviour

Toubman, A.

Author: Toubman, A.
Title: Calculated Moves: Generating Air Combat Behaviour
Issue Date: 2020-02-05

1 Introduction

The military philosopher Sun Tzu once said, "Now the general who wins a battle makes many calculations in his temple ere the battle is fought" (translation by Giles, 1994). Today, we have access to a new type of calculations, which is called *machine learning*. In this thesis, we use machine learning to improve training simulations for air forces.

Air forces are an essential part of modern defence forces. However, air forces worldwide struggle to maintain the combat readiness of their pilots (cf. Ausink, Taylor, Bigelow and Brancato, 2011; Chapman and Colegrove, 2013; Doyle and Portrey, 2014; Church, 2015). In the last decades, shrinking budgets have led to dwindling numbers of operational aircraft, including the aircraft available for training. At the same time, a steady stream of air force deployments has called for pilots at maximum readiness. Maintaining a high level of readiness with smaller numbers of aircraft requires efficient use of alternative means of training, such as simulators (Foster and Fletcher, 2013; Mattingly, Bolton, Walwanis and Priest, 2014; McLean, Lambeth and Mavin, 2016; Bruzzone and Massei, 2017).

Simulators can provide a flexible training environment with access to a wide variety of virtual opponents to train with. Such an opponent is often called a computer generated force (CGF, plural: CGFS), i.e., a computer representation of a real-world force that displays human-like behaviour (cf. Lu and Gong, 2014; Kamrani, Luotsinen and Løvlid, 2016). By training with virtual opponents, trainees can build the experience necessary for air combat operations, at a fraction of the cost of training with real aircraft. In practice however, the variety of virtual opponents is not as wide as it can be. This is largely due to a lack of behaviour models, i.e., computational models used to govern the behaviour that the virtual opponents display (cf. Lu and Gong, 2014; Pelosi and Brown, 2016). The goal of the thesis is to investigate to what extent behaviour models for the virtual opponents in air combat training simulations can be automatically generated, by the use of machine learning.

This chapter is organised as follows. In Section 1.1, we describe the process by which behaviour models are created today. In Section 1.2, we briefly look at the possibility of automatically generating behaviour models. In Section 1.3 we present our problem statement and five research questions. The research methodology is given in Section 1.4. Finally, in Section 1.5 we outline the structure of the thesis.

1.1 The behaviour modelling process

The behaviour modelling process is the process by which behaviour models are created today. For the remainder of this thesis, we define behaviour models as follows.

Definition 1.1 (Behaviour model)**.** A behaviour model is a model that maps (a) the observations made by some entity to (b) the actions that the entity should perform.

We define three roles that take part in the behaviour modelling process: (1) the training specialist, (2) the subject matter expert, and (3) the programmer. We refer to the people that fill in the roles as the professionals[1].

We divide the behaviour modelling process into four steps. The four steps of the behaviour modelling process are shown in Figure 1.1. First, the training specialist writes a behaviour specification for a new CGF. Second, the subject matter expert refines the behaviour specification, using knowledge on the real-world forces that the CGF represents. Third, the programmer creates an executable behaviour model based on the refined behaviour specification. Fourth, the training specialist, subject matter expert, and programmer validate the behaviour model and then make improvements to the model as needed. The four steps are explained in more detail in Chapter 2.

In its current form, the behaviour modelling process brings about two obstacles (see Subsection 1.1.1). Furthermore, we discuss the consequences that the obstacles in the process have for the effectiveness of simulator training (Subsection 1.1.2).

1.1.1 Obstacles in the process

We identify two main obstacles in the behaviour modelling process. We describe them below.

The first obstacle is the observation that the process is *labour-intensive*. However, the professionals capable of performing the four steps in the process come in limited numbers and are not always available. Furthermore, the four steps in the behaviour modelling process must be performed in the order given, and the professionals depend on each other to complete Step 4. Therefore, unavailability of the professionals hampers the completion of the behaviour modelling process.

The second obstacle is the *complexity* of human(-like) behaviour modelling (cf. Banks and Stytz, 2003; Stytz and Banks, 2003a; Stytz and Banks, 2003b). The complexity stems in part from the fact that a CGF's behaviour model has to be able to react as much as possible in a proper way (i.e., as a human would) to all situations that may occur in the simulation (Bourassa, Abdellaoui and Parkinson, 2011). Failure to react appropriately or to react at all to these situations (1) makes the behaviour model brittle, and (2) makes the behaviour produced by the behaviour model to be considered as lacking realism (Bourassa and Massey, 2012). However, eliciting the

[1] See Darken and Blais (2017) for a discussion of the responsibilities of modelling and simulation professionals in the military.

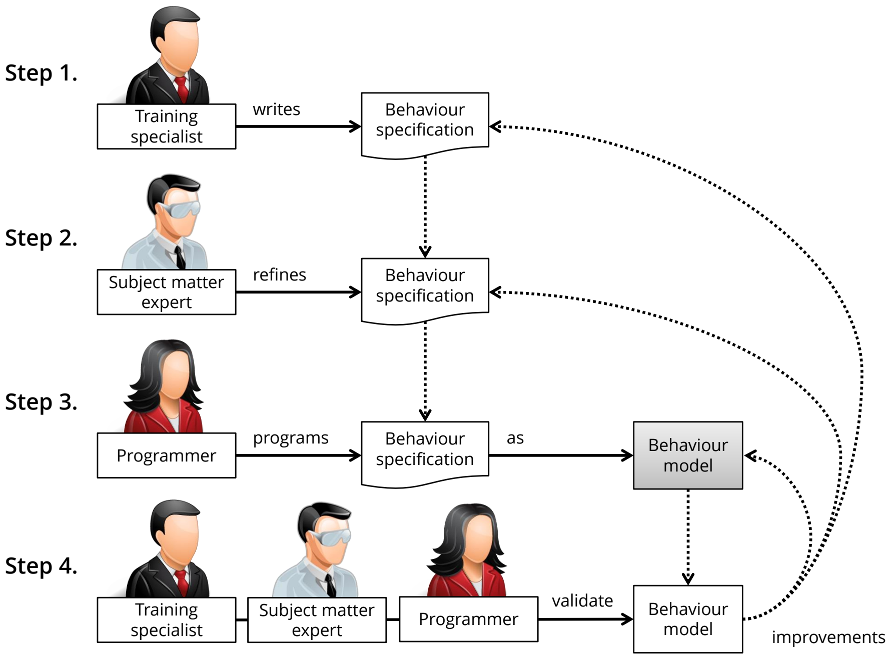

Figure 1.1 The four steps in the behaviour modelling process (adapted from the process described by Gerretsen, Van Oijen, Ferdinandus and Kerbusch, 2017). Step 1: the training specialist writes a behaviour specification for a new CGF. Step 2: the subject matter expert refines the behaviour specification, using knowledge on the real-world forces that the CGF represents. Step 3: the programmer creates an executable behaviour model based on the refined behaviour specification. Step 4: together, the training specialist, subject matter expert, and programmer validate the behaviour model and then make improvements to the model as needed.

knowledge required to model the proper reactions is not a straightforward task (cf. Marcus, 2013; Hoffman, 2014). As a result, behaviour that should have been specified in Step 1 and Step 2 of the behaviour modelling process will only transpire in Step 4. The unspecified behaviour then has to be implemented as an improvement to the model, requiring further work by the professionals.

The two obstacles described above render completing the behaviour modelling process a slow and difficult task. The duration of the process leads to a relatively low number of behaviour models being created. However, at the same time, real-world developments such as (1) new strategies, (2) new tactics, and (3) new equipment are introduced at a high pace. Trainees need to gain experience with these developments in simulations. Therefore, the behaviour modelling process must be reiterated frequently to keep up with the demand for new behaviour models. Furthermore, because of the high pace of real-world developments, the rate of model reuse is low. Lu and Gong (2014) state a reuse rate of behaviour models as low as 10 to 15%.

1.1.2 Consequences for training effectiveness

Because of the low number of behaviour models available for use by CGFs, training specialists are limited in the range of training simulations they can create. A limited range of training simulations has two closely related negative consequences for the effectiveness of the training given by means of these training simulations. We discuss the two consequences below.

The first consequence is that the trainees *miss out on the proven benefits* of variation in training tasks (such as training simulations). For instance, recent studies show that variation in training tasks improves the cognitive and motor skills of trainees (Taylor and Rohrer, 2010; Vakil and Heled, 2016). Furthermore, variation helps trainees to develop the capability to (1) recognise patterns across situations, (2) adapt their mindset to their situation, and (3) come up with creative solutions (Fletcher and Wind, 2014).

The second consequence is that the behaviour of the CGFs becomes *predictable* by the trainees. The predictable behaviour may lead to boredom which transpires in the trainees' behaviour. Furthermore, predictable behaviour may cause the trainees to try to exploit the behaviour of the CGFs, rather than to focus on achieving the learning objectives of the simulations (Lopes and Bidarra, 2011; Silva, do Nascimento Silva and Chaimowicz, 2015).

1.2 Generating air combat behaviour models

The field of artificial intelligence (AI) may offer an alternative to the behaviour modelling process, and improve the effectiveness of training simulations by remedying the two consequences mentioned in the previous section. The alternative is generating behaviour models by means of machine learning. Machine learning programs outperform humans in a variety of tasks (Jordan and Mitchell, 2015), such as credit card fraud detection (Dal Pozzolo, Caelen, Le Borgne, Waterschoot and Bontempi, 2014), cloud computing resource allocation (Hameed, Khoshkbarforoushha, Ranjan, Jayaraman, Kolodziej et al., 2016), and playing games like poker (Bowling, Burch, Johanson and Tammelin, 2015) and Go (Silver, Huang, Maddison, Guez, Sifre et al., 2016; Silver, Schrittwieser, Simonyan, Antonoglou, Huang et al., 2017b). For such tasks, machine learning programs are able to produce creative solutions through a combination of three properties: (1) computational speed, (2) precise constraint satisfaction abilities, and (3) clever learning algorithms. By taking advantage of these three properties and applying the properties to the development of behaviour models, we gain the ability to develop (1) behaviour models at a higher pace, and (2) models with more variation in the behaviour than is currently possible. As a result, the use of machine learning programs to develop behaviour models has the potential to lift the two consequences that the current behaviour modelling process has on training effectiveness.

However, before we apply machine learning to air combat simulations, it is essential to consider the domain of air combat. The domain of air combat is complex, and machine learning methods that operate within this domain must be suited to the challenges posed by the domain. Below, we list five challenges that emerge when generating behaviour models for use in air combat simulations (Subsection 1.2.1). Next, because air combat is a broad concept, we establish the scope of the thesis (Subsection 1.2.2).

1.2.1 Challenges

Below, we identify and describe five challenges: (A) producing teamwork, (B) computationally evaluating CGF behaviour, (C) efficient reuse of acquired knowledge, (D) validating generated behaviour models, and (E) generating accessible behaviour models. The five challenges are not unique to the air combat domain. However, the challenges require solutions that will fit to the domain.

Challenge A: Producing team coordination. Nowadays, the smallest unit that carries out air combat missions consists of two. Flying in pairs has major advantages over flying alone, such as (1) improved situational awareness, and (2) the ability for one teammate to apply offensive pressure on opponents while the other teammate is forced to make defensive manoeuvres (cf. Shaw, 1985; Stillion, 2015). The challenge is to *make optimal use of these advantages* in simulations. It requires a form of coordination between the teammates. Thus challenge A is to let the machine learning method also generate the required team coordination between the CGFs that use the models.

Challenge B: Computationally evaluating CGF behaviour. Machine learning methods require the ability to evaluate the behaviour produced by the behaviour models they generate. In reinforcement learning, which is the family of machine learning methods that we focus on in this thesis (see Chapter 2), the evaluation is performed computationally by the reward function. The reward function is named so because it rewards CGFs for good behaviour, with the intent to stimulate that behaviour (viz. produce better behaviour models). However, the evaluation of air combat behaviour suffers from two issues. First, the concept of *good air combat behaviour* remains ill-defined. Second, non-deterministic factors influence the success of the behaviour of air combat CGFs. We expand on these two issues in Chapter 4. Reward functions that are used in the air combat domain therefore must take into account the two issues in order to stimulate good behaviour with rewards. Thus challenge B is the computational evaluation of the behaviour displayed by air combat CGFs, with the goal of improving the behaviour models generated for these CGFs.

Challenge C: Efficient reuse of acquired knowledge. During the automated generation and testing of behaviour models for a CGF, the machine learning method learns which actions of the CGF are effective in which situations. Therefore, it can be said that the machine learning method acquires and stores knowledge about air combat. It is imaginable that some of this knowledge will be applicable to multiple scenarios in which the CGF may be active. Reuse of air combat knowledge will save computational resources in the search for effective behaviour models for the CGF across different scenarios. Challenge C is *enabling the machine learning method* used to generate behaviour models *to efficiently reuse air combat knowledge that has been acquired previously*.

Challenge D: Validating generated behaviour models. Just like behaviour models that are manually developed by professionals, behaviour models that are generated by a machine learning method have to be validated. Validation of behaviour models ensures that the behaviour models are fit for their intended purpose. Challenge D is *validating behaviour models* that have been generated by means of machine learning, to prove that the models are fit for use in training simulations.

Challenge E: Generating accessible behaviour models. The creative capabilities that machine learning methods possess have a drawback. In brief, the solutions created by these methods may become too clever, and take on forms that are too difficult for humans to comprehend and validate. This is especially problematic for CGF behaviour models, as these models must represent the behaviour of real-world forces at all times. Furthermore, the professionals may wish to inspect and revise behaviour models that have been generated, e.g., in order to slightly adjust the model to better support a learning objective. These professionals are only able to do so if the generated models are constructed in a way that is easily understandable (see, e.g., Luotsinen, Kamrani, Hammar, Jändel and Løvlid, 2016). For this reason, challenge E is *generating accessible behaviour models* by means of machine learning. We define this accessibility as follows.

Definition 1.2 (Accessible behaviour model)**.** A behaviour model is accessible if it is directly interpretable by the professionals (i.e., training instructors, subject matter experts, and programmers) who develop and apply the model.[2]

On various occasions, research has already specifically been focused on the use of machine learning to generate air combat behaviour models. However, as we will show in Chapter 2, the machine learning methods that have been used so far have produced inaccessible behaviour models. In Chapter 2, we will review a machine learning method called dynamic scripting, that was introduced in the previous decade (Spronck, Ponsen, Sprinkhuizen-Kuyper and Postma, 2006). Dynamic scripting was designed to directly address the issue of accessibility as we have described it here. We will investigate dynamic scripting's applicability to air combat simulations.

1.2.2 Scope of the thesis

Both air combat and training simulations are complex domains. It means that in our research we will not take the full domains into account. Below, we restrict the scope of this thesis regarding three areas: (1) the mode of air combat that we study, (2) the specific type of training simulations that we consider, and (3) the width of our view on training simulations.

First, air combat is often divided into two modes (Shaw, 1985). Mode (a) is within-visual-range (WVR) air combat, also known as dog-fighting. In a WVR air combat situation, the opposing aircraft engage each other in the visual arena using on-board cannons and short-range missiles. Mode (b) is beyond-visual-range (BVR) air combat. In BVR air combat, opposing aircraft engage each other using medium-range to long-range missiles, while sensing each other using radar and other instruments. Simulations of the two modes of combat require CGFS with different behaviour. Today, the majority of air combat engagements are BVR engagements (Stillion, 2015; Floyd, Karneeb, Moore and Aha, 2017). For this reason, we restrict the scope of this thesis to BVR engagements, i.e., mode (b).

Second, because our main goal is generating behaviour models for use in air combat training simulations, we specify the particular type of training simulations that we consider in our research. The CGFS that use the generated behaviour models will need to support the learning objectives of this type of training simulations. In this thesis, we restrict ourselves to tactical training at the unit (squadron) level. In tactical training, the objective of the trainees is to defeat all opposing CGFS. The CGFS in tactical training simulations require behaviour models that are capable of handling the most common elements of air combat (e.g., acquiring and pursuing targets, firing missiles, and evading incoming missiles).

Third, training simulations are highly complex systems. The study of training simulations lies at the crossroads of multiple fields of research, e.g., knowledge representation, instructional

[2]Doyle and Portrey (2014) take the definition of accessibility a step further, and pose that the behaviour produced by the models should be "transparent to users not involved in the core modeling process."

theory, human factors, interaction design, competency development, and the modelling of systems, organisations, and behaviours. There are many interactions between these fields. For instance, (a) the development of competencies by the trainees depends on the interaction with the CGFs, (b) the interaction of trainees with CGFs depends on the behaviour models of the CGFs, (c) the behaviour models of the CGFs depend on the modelling technique, knowledge representation, and so on. To study such a chain of interactions as a whole is a complex task that is to be considered intractable with the current means of research and the time allotted to our research project. Therefore, in the thesis, we restrict ourselves to modelling air combat behaviours (see Chapters 3 to 5) and evaluating the perception of the modelled behaviours by training specialists (see Chapters 6 and 7).

1.3 Problem statement and research questions

The two consequences that the current behaviour modelling process has for training effectiveness (Section 1.1), and the prospect of automatic and fast generation of varied behaviour models (Section 1.2) lead us to the following problem statement.

Problem statement: *To what extent can we use dynamic scripting to generate air combat behaviour models for use in training simulations, in such a way that the five challenges of generating air combat behaviour models are met?*

The use of dynamic scripting would bring us underway to meet challenge E. However, challenges A–D remain to be met. Below, five research questions are formulated based on the remaining challenges. In combination, the answers to the five research questions form the answer to the problem statement.

Meeting the first challenge requires investigating the possibility of using dynamic scripting to generate behaviour models that (1) take into account the presence of teammates, and (2) are able to coordinate their observations and actions with these teammates in some manner. This leads us to the first research question.

Research question 1: *To what extent can we generate air combat behaviour models that produce team coordination?*

Dynamic scripting uses a reward function to evaluate the behaviour displayed by the air combat CGFs that use the generated behaviour models. The rewards produced by the reward function are used to adjust newly generated behaviour models in the search for an optimal model. As mentioned (see challenge B), the evaluation of air combat behaviour suffers from two issues. In the literature, these two issues are known as *sparse rewards* and *unstable rewards*, respectively (see Chapter 4). Still, reward functions for air combat behaviour that have been presented in the literature do not always take these two issues into account. However, doing so might lead to behaviour models that produce a more desirable behaviour. This leads us to the second research question.

Research question 2: *To what extent can we improve the reward function for air combat CGFs?*

Dynamic scripting stores the knowledge that a CGF builds throughout the CGF's learning process in the form of weight values that are attached to the rules in the rulebase. The weight value of each rule indicates the rule's importance relative to the other rules in the rulebase. In terms of reuse, it may be possible that the knowledge that is built in one air combat scenario, may also be applied effectively in another air combat scenario. We place the reuse of knowledge in the context of *transfer learning*, i.e., letting a CGF learn in one scenario, and then *transferring* its knowledge to a CGF in a new, unseen scenario. This leads us to the third research question.

Research question 3: *To what extent can knowledge built with dynamic scripting be transferred successfully between CGFs in different scenarios?*

We aim for the generated behaviour models to be used in training simulations. Validating the models is an important step in achieving a productive use of the models. The importance of validation is illustrated by Step 4 in the behaviour modelling process. However, since there is no *one-size-fits-all* solution to the validation of behaviour models, we first have to determine the proper way to do so. This leads us to the fourth research question.

Research question 4: *How should we validate machine-generated air combat behaviour models for use in training simulations?*

The answer to research question 4 is a validation procedure. By means of the procedure, we are able to determine the validity of the behaviour models that we generate in our research. The chosen research approach leads us to the fifth research question.

Research question 5: *To what extent are air combat behaviour models generated by means of dynamic scripting valid for use in training simulations?*

Answering these five research questions will allow us to answer the problem statement. The next section describes the methods that will be used in our research to answer the five research questions.

1.4 Research methodology

In our research, we use four methods to answer the research questions: (1) literature review, (2) automated simulations, (3) questionnaires, and (4) human-in-the-loop simulations. We describe the four methods briefly below.

Literature review. We review scientific articles, books, and technical reports that are related to (1) air combat training simulations (see Section 1.1), (2) the use of machine learning in these simulations (Chapter 2), and (3) the five research questions (Chapters 3–7).

Automated simulations. By automated simulations we mean software simulations in which one team of CGFs engages another team of CGFs in air combat encounters. In this case, both teams of CGFs employ behaviour models for their behaviour. Because such simulations are implemented purely in software, they have the advantages of (a) being able to run faster than real-time and (b) not being dependent on the presence of human participants.

Questionnaires. While reward functions are capable of evaluating the behaviour of a CGF to a certain extent, the final word on the desirability of a CGF's behaviour comes from the training specialist. In the end, it is the training specialist who has to use the CGF in training simulations. Measuring the desirability of a CGF's behaviour is therefore an essential part in the validation of behaviour models. We will develop a novel questionnaire, which we call the Assessment Tool for Air Combat CGFs (ATACC). The ATACC aims to capture the opinions of training specialists observing the behaviour of air combat CGFs, in such a way that we are able to draw conclusions on the desirability of the behaviour.

Human-in-the-loop simulations. By human-in-the-loop simulations we mean simulations in which a team of CGFs using behaviour models engages a team of CGFs controlled by human participants. Training simulations are a prime example of human-in-the-loop simulations. In human-in-the-loop simulations, we are able to observe (1) the behaviour of human pilots, when confronted with CGFs using generated behaviour models, and (2) the behaviour of CGFs using generated behaviour models, when confronted with human pilots.

Below, we briefly describe where we apply the four methods. Table 1.1 gives a summary of the use of the four methods to answer the research questions.

Table 1.1 Research methods used to answer the research questions (RQs).

Method	RQ1	RQ2	RQ3	RQ4	RQ5
Literature review	✓	✓	✓	✓	✓
Automated simulations	✓	✓	✓		✓
Questionnaires					✓
Human-in-the-loop simulations					✓

The literature review is used to answer research questions 1, 2, 3, 4, and 5. Furthermore, we use two simulators for our research: (1) the Lightweight Air Combat Simulator (LWACS) (see Appendix A) and (2) the Netherlands Aerospace Centre NLR's Fighter 4-Ship simulator (see Appendix D). Automated simulations in Lightweight Air Combat Simulator (LWACS) are used to answer research questions 1, 2, and 3. Furthermore, we present the ATACC in Chapter 6. The

ATACC is developed as part of the answer to research question 4, and is then used to answer research question 5. Additionally, both (1) automated simulations and (2) human-in-the-loop simulations in the Fighter 4-Ship are used to answer research question 5.

1.5 Structure of the thesis

This thesis contains eight chapters. Table 1.2 shows which chapters will answer the respective research questions.

Table 1.2 Answering the problem statement (PS) and the research questions (RQs) per chapter.

Chapter	PS	RQ1	RQ2	RQ3	RQ4	RQ5
1	∼	∼	∼	∼	∼	∼
2	∼	∼	∼	∼	∼	∼
3		✓				
4			✓			
5				✓		
6					✓	∼
7						✓
8	✓	✓	✓	✓	✓	✓

∼ contributes to answer, ✓ answers

In Chapter 1 we introduce our problem statement and five research questions. Furthermore, the research methodology by which the research questions are addressed is presented.

In Chapter 2, we provide background information from the literature (see also Section 1.1) on four topics: (1) details of the steps in the behaviour modelling process, (2) the potential benefits and drawbacks of the use of machine learning in training simulations, (3) past approaches to using machine learning for generating air combat behaviour models, and (4) dynamic scripting and its applicability to air combat simulations.

In Chapter 3, we introduce three methods for team coordination: (1) TACIT, (2) CENT, and (3) DECENT. We investigate how beneficial the team coordination methods are by means of an experiment, and then answer research question 1.

In Chapter 4, we zoom in on a specific part of the dynamic scripting process, viz. the reward function. We show how the use of three distinct reward functions influences the behaviour of our CGFs, and then answer research question 2.

In Chapter 5, we investigate to what extent the knowledge that is built by a CGF in some air combat scenario can be transferred successfully to a CGF in a different air combat scenario, and then answer research question 3.

In Chapter 6, we design a validation procedure by which behaviour models that are generated for air combat CGFs may be validated. Furthermore, we present the ATACC, and then answer research question 4.

In Chapter 7, we apply our validation procedure to newly generated behaviour models in the Fighter 4-Ship simulator, and then answer research question 5.

In Chapter 8, we conclude the thesis by providing a summary of the answers to the five research questions. Finally, based on these answers, we formulate the answer to the problem statement. Thereafter we present two recommendations for future work.