Calculated Moves: Generating Air Combat Behaviour

Toubman, A.

6 A validation procedure for generated air combat behaviour models

In this chapter, we investigate research question 4. This research question reads: *How should we validate machine-generated air combat behaviour models for use in training simulations?*

Validation is an important step in the development of behaviour models, since it provides a structured way to determine whether the models are useful with regards to their intended purpose. However, there is no *one-size-fits-all* solution to the validation of behaviour models. Many different validation methods are available, each with their own strengths and weaknesses. It is up to the developer of the behaviour models to consider which validation methods are best applied.

We begin this chapter by briefly reviewing the available literature on (1) the validation of behaviour models and (2) the validation methods (Section 6.1). Next, we introduce new terminology (Section 6.2) tuned to the behaviour models designed for groups of four CGFs. These models and their validation are the subject of this chapter. Therefore, we design a validation process in a step-by-step manner (Section 6.3). Subsequently, we describe two specific elements of the validation process in detail. These elements are (1) the novel Assessment Tool for Air Combat CGFs (ATACC) which is presented in Section 6.4, and (2) the statistical analysis that is performed on the results of the ATACC, which is described in Section 6.5. Then, we present the steps for implementing the validation process (Section 6.6). Finally, we conclude the chapter by answering research question 4 (Section 6.7).

This chapter is based on the following publication.

- A. Toubman (2019). Validating Air Combat Behaviour Models for Adaptive Training of Teams. In: *Adaptive Instructional Systems*. Ed. by R. A. Sottilare and J. Schwarz. Springer International Publishing, pp. 557–571. DOI: 10.1007/978-3-030-22341-0_44

6.1 Validating behaviour models

Since the advent of the use of simulation in military training (see, e.g., Sargent, 1939) there has been a rising interest in the *validation*[1] of simulation models (cf. Sargent, 2011; Kim, Jeong, Oh and Jang, 2015). Many definitions of validation have been stated throughout the literature (cf. Petty, 2010; Birta and Arbez, 2013; Bruzzone and Massei, 2017). When military simulations are discussed in particular, references are made to the definition of validation that is used by the US Department of Defense (2009). We use this definition from now onwards. For convenience, we restate the definition.

Definition 6.1 (Validation)**.** Validation is "[t]he process of determining the degree to which a model or simulation and its associated data are an accurate representation of the real world from the perspective of the intended uses of the model" (US Department of Defense, 2009).

The definition names four important concepts. The concepts are (1) a process, (2) a degree of accuracy, (3) a model (or simulation), and (4) the intended use of the model. We can readily fill in concepts (3) and (4). Regarding concept (3), the models that we wish to validate are newly generated behaviour models. Furthermore, regarding concept (4), the intended use of these models is to produce behaviour for opponent CGFs in air combat training simulations. However, this leaves open two questions for us to investigate: (1) what does the process precisely entail?; and (2) how should we determine the accuracy of the models? We discuss the two questions in Subsection 6.1.1 and Subsection 6.1.2, respectively. Subsection 6.1.3 concludes the section and provides an outlook on the remainder of the chapter.

6.1.1 What does the validation process precisely entail?

First, we investigate the question of *what the process precisely entails*. There is no *one-size-fits-all* solution for validation processes, since all different models have (1) different intended uses, and (2) different *associated works* available for use in the validation. Here, we use the notion of associated work to refer to a range of results of works performed, e.g., (1) baseline models, (2) expected output data, (3) conceptual diagrams of the modelled phenomenon, or (4) expert knowledge. This being so, we still observe that the various validation methods to be applied are well described in the literature. Petty (2010) names four types of validation methods for behaviour models: (1) informal methods, (2) static methods, (3) dynamic methods, and (4) formal methods. Below, we briefly describe these four validation methods, and provide examples of each. The descriptions and the examples are based on (Balci, 1994; Petty, 2010; Sargent, 2011).

[1] Validation is often paired with the related concept of *verification*. Whereas validation tries to answer the question *did we build the right model?*, the question that verification tries to answer is *did we build the model right?* We informally verified the generated models in Chapters 3, 4, and 5 by measuring their performance in automated simulations. The validation procedure that we design in this chapter is intended for determining whether the generated models are suitable for human-in-the-loop simulations.

Type 1: Informal methods. Informal methods are (mostly) qualitative methods that rely on subjective evaluations by subject matter experts of (1) the model or (2) associated works. Examples of informal methods are (a) inspection, (b) face validation, and (c) the Turing test.

Type 2: Static methods. Static methods evaluate (1) the structure of the model and (2) the flow of data within the model, both without executing the model. Examples of static methods are (a) data analysis, and (b) cause-effect graphing.

Type 3: Dynamic methods. Dynamic methods execute the model and evaluate the output that is produced by the model. Examples of dynamic methods are (a) sensitivity analysis, (b) predictive validation, (c) comparison testing, (d) regression analysis, and (e) hypothesis testing.

Type 4: Formal methods. Formal methods are methods that are based on mathematical proofs of correctness. According to both Balci (1994) and Petty (2010), formal methods provide (1) the most reliable conclusions of all validation methods, but at the same time are (2) the most difficult methods to apply to complex models. Examples of formal methods are (a) inductive assertions, and (b) predicate calculus.

An important factor in the choice of validation method(s) to use is the availability of associated works (Petty, 2010; Sargent, 2011). For example, dynamic methods can only be applied if (1) it is possible to execute the model with input that is relevant with regard to the intended use of the model, (2) data can be collected on the execution of the model, and (3) it is known how the collected data should be interpreted (e.g., compared to another available set of data). In other words, the choice of validation methods is always limited by practical considerations.

6.1.2 How should we determine the accuracy of the models?

The second question we would like to investigate reads: *how should we determine the accuracy of the models?* For instance, for a physics-based model, the accuracy of the model can be defined in terms of the number of faults that is allowed when the data that the model produces is compared to data that is measured in the real world. However, for behaviour models the question is particularly difficult to answer, since the notion of fault is difficult to grasp (see, e.g., Hahn, 2013; Hahn, 2017). Goerger, McGinnis and Darken (2005) identify five causes to the difficulty of validating behaviour models in general. Four[2] of these causes relate to the problem of defining the accuracy of a behaviour model. These four causes are: (1) the cognitive processes that are modelled may be nonlinear, which makes the processes as well as their models hard to reason about, (2) it is impossible to investigate all possible interactions that may arise in simulations

[2]The fifth cause is the lack of a standard validation process, which we discussed in Subsection 6.1.1.

because of the large number of interdependent variables in the models, (3) the metrics for measuring accuracy are inadequate, (4) there is no "robust"[3] set of input data for the models.

An important consequence of the difficulty of validating behaviour models is that the outcome of a validation should not be interpreted as either "the model is valid" or "the model is not valid", as it is practically impossible to "completely validate" a model (Birta and Arbez, 2013). Therefore, Birta and Arbez (2013) note that "degrees of success must be recognized and accepted." For them, it is important that the chosen validation methods are able to adequately reflect on the extent of the validity of the models.

6.1.3 Section conclusion and outlook

In summary, it is impossible to have a straightforward, general validation of behaviour models. Therefore, in the remainder of this chapter, we design a validation procedure that is tailored to (a) the generated behaviour models that we wish to validate (see Chapters 3 to 5), and also (b) the application (viz. training simulations) for which the behaviour models are intended (see Chapter 1). In the design, we consider (1) the associated works that are available, (2) the expert knowledge that may be applied, and (3) the measurement of degrees of accuracy of the models.

Looking forward, our validation procedure will consist of many interlocking parts (see Section 6.3). It is our opinion that the description of each part in the procedure must be accompanied by a comprehensive rationale behind each part. The reason should be *trust* in the validation process. Ultimately, validation is a matter of trust, i.e., establishing the trust that behaviour models are suitable for their intended application. Therefore, if the rationale behind one part of the validation process cannot be trusted, the wrong conclusions could be drawn from the results of the process. We acknowledge that the rationales provided in this chapter make the chapter quite lengthy and somewhat abstract. Still, we believe that these rationales are essential for appreciating the actual validation of newly generated behaviour models (i.e., the implementation of the validation process), which we perform in Chapter 7.

6.2 Terminology

In the previous chapters, we have mostly considered *two-ships* of CGFs. However, the human-in-the-loop simulations that we will discuss in this chapter (as well as in the next chapter) are designed to accommodate four human participants. In the simulations, the human participants are opposed by a team of four CGFs. Therefore, we now introduce the term *four-ship* to refer to such a team.

The larger team size requires us to rethink the manner by which we will discuss the behaviour models that produce the behaviour for the CGFs in a four-ship. So far, our experience has been that the behaviour models for the CGFs in a four-ship are treated as a single model. In particular,

[3] We interpret Goerger et al.'s (2005) use of "robust" here as "exhaustive".

when these behaviour models are designed by professionals, the behaviour models are carefully tuned to each other. So, they usually provide the illusion of a cohesive team at work. For this reason (being a cohesive team), we henceforth consider the four behaviour models that together control the behaviour of a four-ship to be an indivisible unit. For convenience, we introduce the term *4-model* to refer to the behaviour models of a four-ship. We define this term below.

Definition 6.2 (4-model)**.** A 4-model is a combination of four behaviour models, which together are used to control the behaviour of a four-ship of air combat CGFs.

Using the term 4-model, we are now able to make a distinction between (1) 4-models that have been written by the professionals, and (2) 4-models that have been generated by means of machine learning. We introduce the terms *4P-model* (where the P stands for *professional*) and *4M-model* (where the M stands for *machine learning*) to refer to these two kinds of 4-model, respectively. We define these terms below.

Definition 6.3 (4P-model)**.** A 4P-model is a 4-model that is written by professionals.

Definition 6.4 (4M-model)**.** A 4M-model is a 4-model that is generated by means of machine learning.

6.3 Designing a validation process

In this section, we design a validation process for the validation of air combat CGF behaviour models. We do so along five design steps.[4] These design steps are: (1) outlining the process, (2) adding a baseline, (3) obtaining behaviour traces in human-in-the-loop simulations, (4) assessing the behaviour traces, and (5) equivalence testing. Below, we describe each of the five design steps and the rationale behind them.

Design step 1: Outlining the process. As the first design step, we draw the outline of the validation process. Figure 6.1 shows the outline. The validation process is placed in the middle of the figure. To the left of the process are the 4M-models that we wish to validate. Therefore, the 4M-models are the input to the validation process. The output of the validation process is the extent of the validity of the 4M-models (right).

Design step 2: Adding a baseline. The subjects of the validation (i.e., the 4M-models) are by themselves not sufficient input for the validation process. As Petty (2010) stated succinctly, validation "[is a] process[] that compare[s] things." Therefore, we require either (1) a

[4] The five design steps that we present in this section are an idealised abstraction of the design of our validation process. This abstraction is presented for the reader's convenience. In reality, the design was a demanding fuzzy optimisation task that required careful balancing of (1) the objective that we were trying to reach, and (2) the resources (both digital and human) that were available to us.

baseline model, (2) a set of expected output data, or (3) implicit expert knowledge as a reference to compare against the 4M-models.

For complex air combat behaviour models, it is almost infeasible to compile a set of expected output data, since the output depends on a wide range of possible interactions with other entities.[5] However, what we *do* have available are behaviour models that have been written previously by professionals (i.e., 4P-models). These 4P-models constitute a *sample* of all behaviour models that have been written by the professionals. The sample is in some sense comparable (see below) to the sample of 4M-models that have been generated by machine learning. Furthermore, we argue that since the 4P-models have been developed by means of the behaviour modelling process (see Section 2.1), the 4P-models have been validated to some extent. As a second design step, we therefore add 4P-models as the second input to the validation process (see Figure 6.2, highlighted).

Design step 3: Obtaining behaviour traces in human-in-the-loop simulations. Currently, a comparison of the 4M-models to the 4P-models in a meaningful way is a hard problem because of the aforementioned dependency on a wide range of input. We are unable to accurately predict if the models will produce comparable behaviour purely by inspecting the models. Therefore, as the third design step, we provide the models with the necessary input. We do so by submitting the models to human-in-the-loop simulations.

In the simulations, human pilots provide realistic input to the models, meaning that the behaviour of the pilots makes sense in the context of the training simulations for which the models are intended. Furthermore, by letting human pilots engage CGFs in simulations, we are able to obtain a sample of *behaviour traces*, i.e., recordings of the behaviour that the CGFs display. Figure 6.3 shows the composition of this design step. The human-in-the-loop simulations and the pilots are highlighted. The behaviour traces (not shown) serve as input to the remainder of the validation process.

Design step 4: Assessment of the behaviour traces. As the fourth design step, we aim to summarise the behaviour that is encoded in the behaviour traces into values that are (1) meaningful and (2) comparable between the 4M-models and the 4P-models. We do so by a structured form of face validation, which is one of the informal validation methods.

However, there is little to no information available on measures for CGF behaviour that are relevant to training simulations.[6] Therefore, in this design step, we make use of the

[5] The solution to this objection is using scenario models. However, well-balanced, adequate scenario models are beyond the scope of our research. Still, we use a similar idea by introducing the use of behaviour models which are written by professionals.

[6] An idea that was put forward at an early iteration of the design was to measure the improvement in skills of the *human* pilots after training in simulations with CGFs with 4M-models, in contrast to 4P-models. However, this idea brought along new problems, such as (1) selecting the right task to train in simulations, (2) choosing the right measures for the performance of the humans, and (3) using an appropriate training schedule.

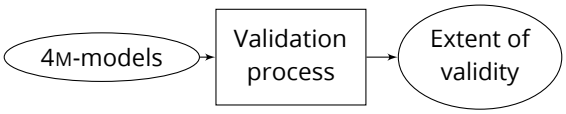

Figure 6.1 Design step 1. The outline of the validation process.

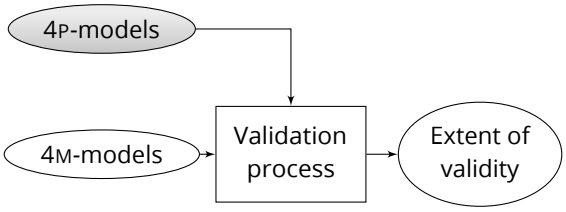

Figure 6.2 Design step 2. The 4P-models are added as a baseline.

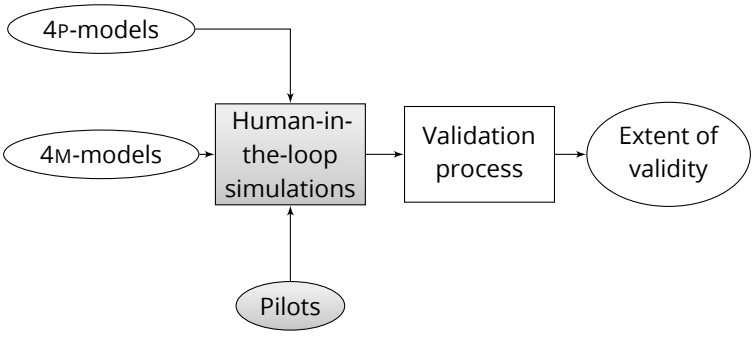

Figure 6.3 Design step 3. The 4M-models and the 4P-models are executed in human-in-the-loop simulations with the participation of human pilots.

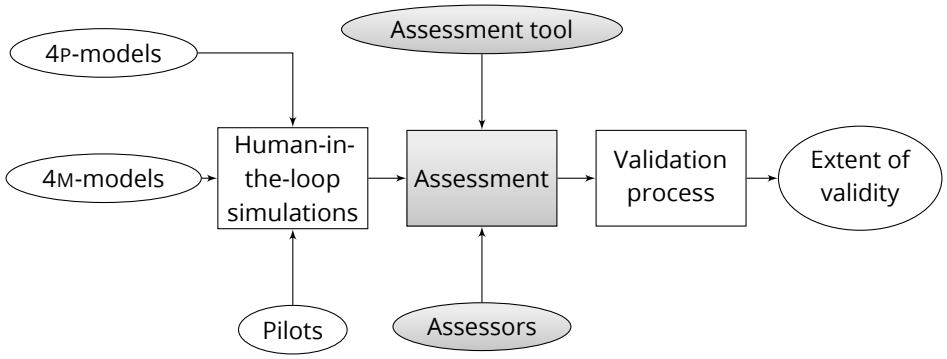

Figure 6.4 Design step 4. The results of the human-in-the-loop simulations are subjected to assessments by assessors that make use of an assessment tool.

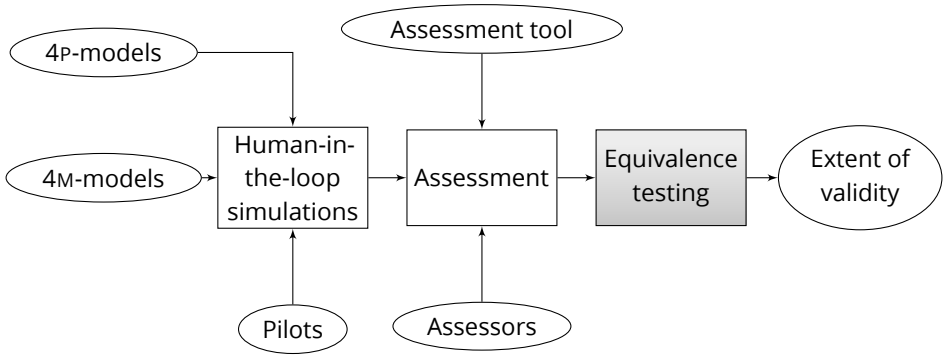

Figure 6.5 Design step 5. The results of the assessments are analysed by means of equivalence testing.

implicit knowledge of expert evaluators. We leverage this knowledge in two manners. First, we elicit knowledge on measures for behaviour of air combat CGFs, and then structure this knowledge into an assessment tool (see Section 6.4). This tool enables a structured assessment of CGF behaviour. Second, expert evaluators review the behaviour traces that we have collected, and then assess the behaviour that the CGFs display. The assessments are performed by means of the newly developed assessment tool.

The use of expert evaluators as assessors relates back to the question of *how should we determine the accuracy of the models*. Since we are unable (as of yet) to codify the measures for the accuracy of CGFs behaviour in a manner that is (1) complete and (2) objective, the main source of these measures is the implicit knowledge of expert evaluators. Sadagic (2010) used a similar validation method in a similar context (i.e., the behaviour of urban warfare CGFs).

Figure 6.4 shows the addition of (1) the assessment (centre, highlighted), including (2) the assessors (bottom, highlighted) and (3) the assessment tool (top, highlighted) to the validation procedure.

Design step 5: Equivalence testing. At this point in the validation process, we have two sets of data: (1) the assessments of the 4P-models, and (2) the assessments of the 4M-models. We wish to compare these two sets of data in a meaningful way. Since we used the 4P-models as the baseline, we assume that the assessments of the 4P-models contain information about the desirable properties of air combat CGF behaviour. Based on this assumption, we define the following measure of validity of the 4M-models.

Definition 6.5 (Measure of validity of the 4M-models)**.** The 4M-models are valid to the extent that (1) the assessments of the 4M-models and (2) the assessments of the 4P-models can be measured to be equivalent.

Obviously, a simple comparison (viz. determining if the difference between the assessments equals zero) of the assessments is too strict. The results of our assessments include noise from multiple sources (e.g., the pilots in the human-in-the-loop simulations, and bias of the assessors). Furthermore, standard statistical significance tests do not suffice, since these tests check for differences rather than for equivalence. We found a solution in a form of comparison testing that is called *equivalence testing*. We further describe the equivalence testing in Section 6.5.

Figure 6.5 shows the result of this design step. We replace the remainder of the validation process by equivalence testing (highlighted). The output of the equivalence testing is the extent of the validity of the 4M-models.

In summary, we have designed a validation procedure by means of which behaviour models for CGFs may be validated. In the procedure, we use two validation methods: (1) face validity in a structured form by means of an assessment tool, and (2) comparison testing between the assessment results of the 4P-models and the 4M-models. However, two gaps remain in the procedure. The first gap is the assessment tool by which the assessors can assess the behaviour that the models produce in a structured manner. We develop this tool in Section 6.4. The second gap is the comparison testing that is performed on the results of the assessments. We describe the comparison testing in Section 6.5. Afterwards, in Section 6.6, we provide a step-by-step procedure for implementing the validation procedure.

6.4 The Assessment Tool for Air Combat CGFs

In this section, we present the Assessment Tool for Air Combat CGFs (ATACC). Below, we first describe the development of the ATACC. Next, we look at its implementation.

We consulted four former instructor pilots for the development of a novel assessment tool. During multiple brainstorming sessions, we identified (1) an appropriate format for the tool, and (2) the specific behaviour that we wished to measure with the tool.

The assessment of behaviour is a major topic of research in the fields of (1) behavioural sciences and (2) human resource management (cf. DeNisi and Murphy, 2017). For this reason, we performed a literature review in order to find formats which could be used as a basis for our assessment tool. The review guided us towards the tool known as the behaviourally anchored rating scale (BARS) (Debnath, Lee and Tandon, 2015).

A BARS (plural: BARSS) is a scale that is intended to measure specific *performance dimensions* (Snell, Morris and Bohlander, 2015, p. 321). In order to aid the assessors who use the BARS in identifying the behaviours, the levels of the scale are marked with anchors. These anchors consist of *critical incidents*, e.g., objectively observable behaviours that are (un)desirable in the performance dimensions. We refer to the work by Phillips, Shafer, Ross, Cox and Shadrick (2006) for an example of BARSS for tactical behaviour in the military domain.

Together with the instructor fighter pilots, we identified three performance dimensions that should be taken into consideration in the assessment of the behaviour of air combat CGFS. These performance dimensions are (1) the *challenge* provided by the CGFS, (2) the *situational awareness* that the CGFS display, and (3) the *realism* of the behaviour of the CGFS. Below, we briefly describe the three performance dimensions.

Performance dimension 1: Challenge. The tool should measure whether (1) the CGFS behave in such a way that the human participants in the simulations need to think about and adjust their actions, and (2) whether the CGFS provide some form of *training value* to the simulations.

Performance dimension 2: Situational awareness. The tool should measure whether (1) the CGFS appear to sense and react to changes in their environment, and (2) whether multiple CGFS belonging to the same team appear to acknowledge each other's presence.

Performance dimension 3: Realism. The tool should measure (1) whether the CGFS behave as can be expected from their real-world counterparts, and (2) whether the CGFS use the capabilities of their platform (including, e.g., sensors and weapons) in a realistic manner.

After the identification, we attempted to formulate examples of behaviour that relate to each of the performance dimensions. This was done in an iterative manner, so that examples that were proposed could be critically analysed by each of the instructor fighter pilots. We formulated eight examples of behaviour in total. Below, we list these eight examples of behaviour. In each of the examples, *red air* refers to the CGFS, whereas *blue air* refers to the human participants in the human-in-the-loop simulations. Four of the examples relate to performance dimension 1, *Challenge*.

Example of behaviour 1. Red air forced blue air to change their tactical plan.

Example of behaviour 2. Red air forced blue air to change their shot doctrine[7].

Example of behaviour 3. Red air was within factor range[8].

Example of behaviour 4. Blue air was able to fire without threat from red air.[9]

Subsequently, two examples relate to performance dimension 2, *Situational awareness*.

Example of behaviour 5. Red air acted on blue air's geometry.

Example of behaviour 6. Red air acted on blue air's weapon engagement zone[10].

The remaining two examples relate to performance dimension 3, *Realism*.

Example of behaviour 7. Red air flew with kinematic realism.

Example of behaviour 8. Red air's behaviour was intelligent.

Next, we attempted to define *critical incidents* based on the eight examples of behaviour. In other words, we tried to formulate desirable and undesirable instances of the examples of behaviour, that could be observed in an objective manner. The critical instances could then be placed as anchors on their respective performance dimensions in order to form the BARSS. However, despite our best efforts, we were unable to define satisfactory critical incidents that (1) objectively described situations that could be observed, and (2) once observed in a simulation, would indicate the performance of the CGFs in a performance dimension for the *entire* simulation. Consequently, we were unable to use the BARS format for our assessment tool.

Rather than abandoning the examples of behaviour that were formulated, we decided to substitute the BARS format by a related format. This format is the behaviour observation scale (BOS). In contrast to a BARS, a BOS defines examples of behaviour and attempt to measure the frequency of the occurrence of the examples (Snell et al., 2015, p. 321). Following the new way, rather than requiring predefined anchors to guide the assessors, it is the assessor who determines if a given behaviour is displayed, and if so, how often. Here, an appeal is made to the implicit expert knowledge that the assessor possesses on critical incidents that we are as of yet unable to explicitly define.

We created a new BOS for the assessment of air combat CGFs. In this BOS, we used the eight examples of behaviour that were defined earlier in this section. We attached a five-point Likert scale to each example of behaviour, to indicate that example's occurrence in a simulation: (1) never, (2) rarely, (3) sometimes, (4) often, or (5) always.

[7] Jargon: pre-briefed instructions for the use of air-to-air weapons.

[8] Jargon: the range within which opponents have to be taken into account in the selection of tactical actions.

[9] We formulated this behaviour from the viewpoint of blue air, since we were unable to satisfactorily state the behaviour from the viewpoint of red air.

[10] Jargon: the airspace in front of a fighter jet in which a fired missile can be effective.

In addition to the eight examples of behaviour, we added a ninth example. This example states on a high level the behaviour that we desire from the CGFs that are being assessed. The purpose of the ninth example is to capture the general opinion on the suitability of the behaviour of CGFs . Therefore, this example functions as a sort of *control item* on the BOS. Below, we state the ninth example of behaviour.

Example of behaviour 9. Red air's behaviour tested blue air's tactical air combat skills.

The ninth example of behaviour is also rated using a five-point Likert scale, but with different options than the first eight examples: (1) strongly disagree, (2) disagree, (3) undecided, (4) agree, or (5) strongly agree.

6.5 Equivalence testing

We incorporate Schuirmann's (1987) two one-sided t-tests (TOST) method in the validation process to determine the equivalence of (1) the responses given on the ATACC for 4P-models, and (2) the responses given on the ATACC for 4M-models. The TOST method involves the application of two one-sided t-tests. They should calculate to what extent two measured means do not differ from each other, given a margin of error that is called the *indifference zone*. We briefly introduce the TOST method below (Subsection 6.5.1). Next, we explain how we use the TOST to measure the *extent* of the validity of the models that are the subject of the validation (Subsection 6.5.2).

6.5.1 Equivalence testing with TOST

The TOST method tests for equivalence of the means of two populations (cf. Meyners, 2012; Anderson-Cook and Borror, 2016; Lakens, 2017). This means that the method (1) starts with the assumption that two populations are different, and then (2) collects evidence to show that the populations are the same. Note that this is the opposite of traditional tests that compare two populations (e.g., Student's t-test), which (1) start with the assumption that two populations are similar or even the same, and then (2) collect evidence to show that the populations are different.

In TOST, the assumption that two populations are different (viz. the *null hypothesis* or H_0) is stated as follows.

$$H_0 : \quad \mu_A - \mu_B \leq \delta_L \quad \text{or} \quad \mu_A - \mu_B \geq \delta_U \tag{6.1}$$

Here, the difference of the means of two populations A and B are compared. Two populations are considered different if the difference of their means lies outside of the indifference zone $[\delta_L, \delta_U]$. For the remainder of this chapter, we assume that the indifference zone is symmetrical, i.e., $\delta = \delta_U = -\delta_L$. However, we are interested in examining the alternative hypothesis (or

H_1) that the means are *not* different, i.e., the difference between the means lies inside of the indifference zone. Following from H_0, we formulate H_1 as follows.

$$H_1: \quad \delta_L < \mu_A - \mu_B < \delta_U \tag{6.2}$$

If the TOST finds evidence that the difference of the means lies within the indifference zone under the assumption that it does not, we reject H_0 and do not reject H_1, meaning that we conclude that the populations are the same (up to a very small difference). Finding this evidence is done by splitting H_0 into two hypotheses which can be tested using standard one-sided t-tests. The p-value of the TOST then becomes the maximum of the two p-values that are obtained from the two one-sided t-tests.

The outcome of the TOST greatly depends on the value chosen for δ. Until recently, δ could not be calculated directly. It was either (1) prescribed by regulatory agencies (e.g., in the field of pharmacology) or (2) determined by subject matter experts based on reference studies or expectations about the data (e.g., in psychology) (cf. Meyners, 2012; Anderson-Cook and Borror, 2016; Lakens, 2017). For our validation, it is difficult to determine a suitable δ, since we have neither a regulatory agency, nor a reference study available. However, in 2016, an objective calculation of δ was introduced by Juzek (2016). The calculation of this delta δ (henceforth: Juzek's δ) is as follows.

$$\delta = 4.58 \frac{s_p}{N_p} \tag{6.3}$$

Here, s_p is the pooled standard deviation in the two samples under comparison, and N_p is the pooled number of data points in the samples. Juzek found the coefficient (4.58) by simulating a large number of TOST applications. The coefficient was approximated in such a way that Juzek's δ gives the TOST the appropriate statistical power ($1-\alpha = 95\%$, $1-\beta = 80\%$).

6.5.2 Measuring an extent of validity

As mentioned in Section 6.1, the validation process should not produce an absolute outcome. Rather, the process should reflect *degrees of success*, i.e., the extent to which models can be said to be valid. Although we have selected the TOST method for our equivalency tests, we have not yet defined how the results of the TOST should be interpreted to arrive at a judgement on the validation of the 4M-models.

The TOST provides us with a test of equivalence of the assessments for each example of behaviour. In other words, nine tests of equivalence are performed in total to compare the ATACC assessments of the 4M-models to the ATACC assessments of the 4P-models. Therefore, we propose that we measure the extent of the validity of the 4M-models along the number of equivalences that are found by the TOST.

6.6 Implementing the validation process

In this section, we briefly state a step-by-step procedure that can be followed to implement the validation process that was described in this chapter. The procedure consists of five steps. We describe these steps below.

Step 1. Defining the baseline. We collect a set of 4P-models to serve as the baseline. The size of this set is a trade-off between (1) the 4P-models that are available to use, and (2) the number of 4P-models that can be practically used in human-in-the-loop simulations, so that after the generation of 4M-models (see Step 2) sufficient behaviour traces per 4P-model can be (a) collected (see Step 3) and (b) assessed (see Step 4).

Step 2. Generating models by means of machine learning. We generate a set of 4M-models by means of dynamic scripting. These 4M-models are the subject of the validation. Here, the same trade-off on the size of the set of 4P-models (see Step 1) holds for the size of the set of 4M-models.

Step 3. Human-in-the-loop simulations. The 4P-models and the 4M-models are used to control the behaviour of a four-ship of CGFs in human-in-the-loop simulations. In the simulations, the CGFs are opposed by a four-ship that is controlled by human participants. The behaviour that both the CGFs and the human participants show is recorded as behaviour traces that can be reviewed at a later time.

Step 4. Assessments. Subject matter experts assess the behaviour traces that were obtained from the human-in-the-loop simulations. The assessments are performed by means of the ATACC.

Step 5. Equivalence testing. We perform equivalence tests to compare (1) the behaviour produced by the 4P-models to (2) the behaviour produced by the 4M-models.[11] The results of the equivalence tests indicate to what extent the 4M-models are valid for use in training simulations.

6.7 Answering research question 4

In this chapter, we addressed research question 4. This research question reads: *How should we validate machine-generated air combat behaviour models for use in training simulations?* To answer this question, we investigated the validation methods that are available in the literature (Section 6.1). Next, we defined new terminology (Section 6.2) that allows us to refer concisely

[11] In the future, the validation procedure presented in this chapter may be adapted to compare the behaviour of 4M-models that have been generated using different machine learning techniques, such as deep learning as it has been applied in the ALPHAGO program (see, e.g., Silver et al., 2016, 2017b).

to the combined behaviour models of a four-ship of CGFs. With the use of the new terminology, we designed a validation process for the validation of behaviour models for air combat CGFs (Section 6.3).

The validation process has two important features. The first feature is the use of a novel assessment tool for the assessment of the behaviour that CGFs display in human-in-the-loop simulations (Section 6.4). The second feature is the use of equivalence testing, a form of hypothesis testing that determines whether two sets of data may be considered equivalent (Section 6.5). In the validation process, equivalence testing is used to determine whether the behaviour that is produced by generated models is assessed as equivalent to the behaviour that is produced by models that are written by professionals. Finally, we summarised the implementation of the validation process, including (1) the use of the ATACC and (2) the equivalence testing by means of TOST, into a step-by-step procedure (Section 6.6). This procedure forms the answer to research question 4.