Effective Behaviour Modelling for Computer Generated Forces

Joost van Oijen, Armon Toubman, Gerald Poppinga
Royal Netherlands Aerospace Centre
Amsterdam, the Netherlands
{Joost.van.Oijen, Armon.Toubman, Gerald.Poppinga}@nlr.nl

ABSTRACT

In military Modelling and Simulation (M&S), there is an increasing need for Computer Generated Forces (CGFs) with advanced autonomous behaviours for use in training, Concept Development & Experimentation (CD&E) and decision support. However, the development of behaviour models is often resource-intensive and reuse across applications, simulation systems or scenarios is difficult. This is clearly illustrated by the plethora of platform-specific (proprietary) modelling tools on the market and is considered a burden throughout interviews conducted with staff operating various simulators of the Dutch military.

In this paper we address current practices and propose directions for *effective behaviour modelling* for CGFs. Hereby we take a holistic view on the behaviour modelling process, including its stakeholders such as subject-matter experts (SMEs), designers, developers and end-users (e.g. operators/instructors). In this view, we regard effectiveness as optimising *efficiency* in the workflow process, and increasing *reusability* of CGF behaviours models. We highlight directions for effective behaviour modelling, taking the perspective from different stakeholders. These relate to expert knowledge management; data-driven modelling techniques; resource management; service-oriented deployment; and model transparency and explainability.

Based on the above directions, we present a design concept for a general-purpose executable behaviour construct, modelled specifically for reusability. It addresses stakeholder needs and requirements in a synergetic manner through pragmatic design practices. This concept is evaluated in a use case for air-to-air combat behaviour. We demonstrate the reusability of tactical CGF behaviours across two simulation systems, for CD&E purposes and fighter pilot training. The behaviours employ both hand-crafted and machine learning techniques and are (partly) generated from (reusable) behaviour descriptions created by former F-16 pilots acting as SMEs.

ABOUT THE AUTHORS

Joost van Oijen, PhD is a Research Scientist at the Royal Netherlands Aerospace Centre (NLR). With a background in Computer Science and Artificial Intelligence, he has over ten years of experience in AI for modelling & simulation, both in the industry and academia. At his current position, Joost leads several R&D projects focused on human behaviour modelling for training and decision-support. Having a strong software engineering background, he is actively involved in the development of multi-agent systems and behaviour modelling tools for military simulation systems.

Armon Toubman holds a Master of Science in Artificial Intelligence. He works as an R&D engineer at the Netherlands Aerospace Centre NLR. In this capacity, he designs and tests new modelling and simulation concepts. Furthermore, as a simulation operator, he has first-hand experience with the application of behaviour models. In an international context, Armon has multiple publications in simulation conference proceedings. He is currently completing his PhD study on the use of machine learning for the generation of air combat behavior models, and is expected to receive his doctorate in the summer of 2019.

Gerald Poppinga is a Computer Scientist that has been working in Aerospace for more than twenty years now. He has been a member of the Information Systems Technology (IST) panel of the NATO Science and Technology Organization (STO) for several years and contributed to the Research Task Group IST-121 on Machine Learning Techniques for Autonomous Computer Generated Entities. Current projects include the application of Artificial Intelligence to Modelling and Simulation with a focus on the re-use of Computer Generated Forces, route optimization for aircraft and robot control.

Effective Behaviour Modelling for Computer Generated Forces

Joost van Oijen, Armon Toubman, Gerald Poppinga
Royal Netherlands Aerospace Centre
Amsterdam, the Netherlands
{Joost.van.Oijen, Armon.Toubman, Gerald.Poppinga}@nlr.nl

INTRODUCTION

In military Modelling and Simulation (M&S), there is a growing interest for Computer Generated Forces (CGFs) with advanced autonomous behaviours for use in training (c.f. training simulators), Concept Development & Experimentation (CD&E) in Battle Labs and decision-support (e.g. course-of-action analysis). To keep up with this demand, a paradigm shift is required in CGF modelling where traditional scenario-oriented modelling approaches (where behaviour is predominantly driven by scenario events and triggers) are replaced with agent-oriented modelling approaches (where CGFs are treated as fully autonomous entities with their own local view of the environment and capabilities to express doctrinal behaviour in a dynamic environment).

However, the development of autonomous CGF behaviour models is often resource-intensive and reuse across applications, simulation systems, or scenarios is difficult. This is clearly illustrated by the plethora of platform-specific (proprietary) modelling tools on the market and is considered a burden throughout interviews conducted with staff operating various simulators of the Dutch military. If a new simulation system is acquired, existing models typically have to be built completely from scratch. Additionally, models are often designed for specific applications or scenarios and transfer to other domains is not possible (e.g. to transfer behaviour of an enemy CGF designed for training purposes in one simulator to another simulator for war-gaming purposes).

Effective reuse of behaviour models remains a huge challenge. Similar to simulation models in general, reusability is enabled by forms of standardization (e.g. for 3D objects, geospatial data, scenario definitions). When considering CGF behaviour models, standards for internal representations are essentially non-existent and seem improbable. History has shown that mainstream techniques used in practice change and evolve over the years (e.g. rule-bases, state machines or behaviour trees). This evolution is expected to continue as AI technologies such as machine learning are maturing and start showing some practical applications for behaviour modelling (Roessingh et al., 2017). At a more abstract level, there have been recent efforts investigating standardization of a reference architecture for human behaviour representation, although its results are preliminary and further studies are required (Lewis, Alexander, Huiskamp, & Blais, 2019). When regarding a model's interfaces in terms of interoperability, standardization is more feasible and some examples exist. E.g. C2SIM is a standard for communication of C2 messages which could be used for inter-CGF communication (Brook, 2015); and LLBML has been proposed as a standard for controlling a CGF's physical model (Alstad et al., 2013). Still, such standards are often geared towards very specific domains.

In this paper we address reusability from a different angle. Rather than focusing on standardization efforts of the computational behaviour models, we take a more holistic approach by considering the workflow of the behaviour modelling process itself, including the involved stakeholders. In this scope we address alternative practices for effective behaviour modelling (increasing efficiency and reusability), originating from the needs of the stakeholders, which include subject-matter experts (SMEs), designers, developers, and end-users. Concretely we address (1) efficiency in knowledge elicitation and knowledge management from a designer's perspective, (2) the expected impact of maturing AI technologies on the modelling process from a developer's perspective (e.g. data-mining and machine learning), and (3) enabling factors for reusability from an end-user's perspective, with regard to resource management, service-oriented integration and model transparency.

Based on the above focus areas we present the concept of a general-purpose behaviour construct which offers an integrated and synergetic design approach for effective behaviour modelling, focusing on military tactical behaviours for CGFs. This concept is evaluated in a use case for air-to-air combat behaviour where we demonstrate the development efficiency and reusability of tactical behaviours across two simulation systems: for CD&E purposes and fighter pilot training.

DIRECTIONS FOR EFFECTIVE BEHAVIOUR MODELLING

Agent-oriented design approaches have become more common practice for modelling tactical CGF behaviours. These behaviours can include mission-oriented behaviours, doctrinal rules, operating procedures, communication protocols or tactical maneuver execution. The implementation of such behaviours requires expert knowledge, design expertise and engineering skills, typically involving multiple persons. The associated process can be represented as the behaviour modelling workflow as illustrated in Figure 1.

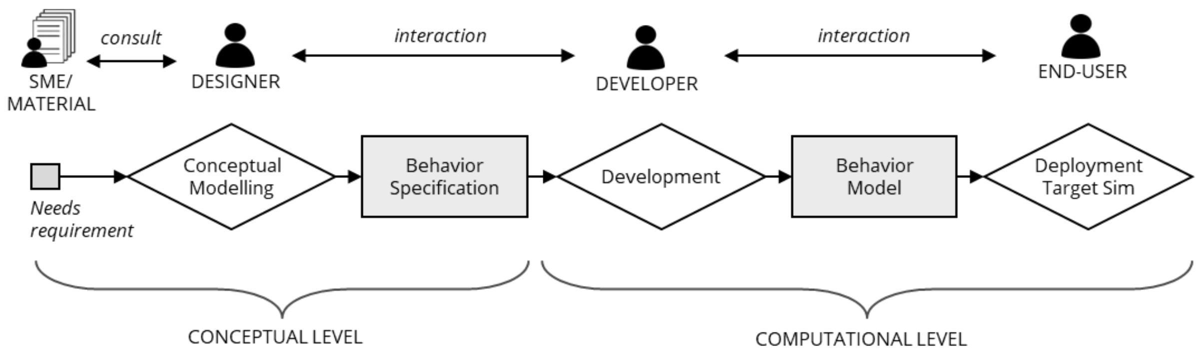

Figure 1. The Behaviour Modelling Workflow

The workflow starts with a needs requirement which is established by the user of a simulation system. For instance, an instructor requires enemy aircraft behaviour in a training simulator to practice personnel skills for tactical intercepts; or an operator of a Battle Lab requires behaviours for both blue and red forces for simulation-based CD&E (e.g. evaluating tactics or weapon system performance). At the start of the process, a designer translates the needs to more concrete behaviour specifications for individual CGFs, capturing required capabilities. The designer can consult subject-matter experts (SMEs) or external reference material to obtain required expert knowledge (e.g. tactical decision-making). In the next phase, a developer uses the specification to engineer a behaviour model that can be deployed in a target simulator. Finally, an end-user (c.f. a simulator operator) integrates the model into an operational scenario used for training or analysis purposes.

The roles of the designer, developer and end-user (c.f. the stakeholders) do not operate fully independently. Interactions are common between the end-user and the developer due to operational issues of the model (e.g. the performance of the behaviour does not meet the end-user's need or expectations). The designer and developer require interactions when a specification gives insufficient insight for a developer on how to implement the model. They collaborate on filling in the missing details, which in turn may require additional consults with SMEs.

The presented workflow takes different forms when comparing to industry practices. For instance, a single person can take on multiple roles (e.g. an end-user which collaborates directly with the SME to design and implement the model). Or behaviour specifications are either non-existent or range from more informal descriptions (free text or illustrations) to more formal knowledge management through structured templates or behaviour diagrams. Still, the workflow can be generally recognized in a more or less structured form.

Within the scope of the behaviour modelling process, we define *effective behaviour modelling* as:

(1) Optimizing *efficiency* in the workflow process (both in terms of stakeholder interactions and speed of model development), and
(2) Increasing *reusability* of CGF behaviours (either as specifications or computational models).

When addressing effectiveness, one must include the perspective from all stakeholder roles. Below we present three directions that contribute to effective behaviour modelling, related to the needs and requirements of the different stakeholders.

1. Knowledge Management & Elicitation
Taking a designer's perspective, we stress the need for more formal knowledge management for behaviour specifications to store expert knowledge. Formalized behaviour specifications can be seen as a *design contract* between on the one hand operational needs and expert knowledge, and on the other hand the computational result in the form of an executable model. It benefits effective behaviour modelling because:

- It facilitates and promotes a more structured knowledge elicitation process between a subject-matter expert (SME) and a designer to secure detailed professional knowledge (operational, tactical or procedural) about a CGF's behaviour. This limits valuable SME interaction time, which could otherwise be caused by feedback from developers (insufficient information for implementation) or end-users (experiencing undesired behaviours).
- It offers possibilities for behaviour reuse at a technology- and implementation-independent level. Currently, reuse of computational models between different simulation systems is practically non-existent as each system uses different underlying technologies for implementing CGFs. This is unlikely to change because of the challenges in standardization. A technology-independent specification can preserve explicit behaviour knowledge which would otherwise become lost in translation towards an implementation.

Some examples of related work addressing the above issue exist. For instance in (Gonzalez, Castro, & Gerber, 2005), a support tool is proposed to automate the acquisition of tactical knowledge from SMEs using context-based reasoning (CxBR) (Stensrud, Barrett, & Gonzalez, 2004). As CxBR is also used as a behaviour modelling paradigm for CGFs, the encoded (implementation-independent) expert knowledge directly facilitates model development. A comparable approach is used in (Potts, Griffith, Sharp, & Allison, 2010), where SMEs can define modular behaviours to compose executable behaviour models.

2. Data-driven Modelling
From a developer's perspective, one should be flexible in terms of adaptation to different implementation technologies. AI technologies such as machine learning and data mining are becoming increasingly adept at coping with more complex data and problems. They have the potential to generate (parts of a) behaviour model and replace or reduce hand-crafted modelling efforts (hereby increasing efficiency of model development). We highlight two approaches that are active areas of research.

The first approach involves using machine learning for generating behaviour models. For instance, supervised learning can be used to imitate behaviours from human example data. Data could be collected from human role-players demonstrating behaviours in a simulator (Luotsinen & Løvlid, 2015), or from real-life data that has been recorded during operation (Karli, Efe, & Sever, 2017). Reinforcement learning can be used to automatically learn behaviours while a CGF is interacting with a simulation environment, hereby learning through experience (Toghiani-Rizi, Kamrani, Luotsinen, & Gisslén, 2017). The use of machine learning for generating CGF behaviours has also been investigated in the NATO task group IST-121 (Roessingh et al., 2017; Toubman, Roessingh, van Oijen, et al., 2016).

The second approach involves the use of military reference material for extracting human knowledge. Much information about the operation or tactics of military actors is secured in documents: e.g. consider doctrine documents, tactical operating procedures (TOPs), combat instruction sets (CISs), rules of engagements (RoE) or mission planning information. We expect that advances in technologies such as natural language processing will enable harvesting this information and extract relevant knowledge that could be used for generating behaviour specifications of the models themselves, e.g. see (Balint, Allbeck, & Hieb, 2015; Smith & Dunn, 2016).

3. Model Discovery, Deployment & Transparency
The mere existence of a reusable behaviour model does not automatically lead to reusability in practice. From an end-user's perspective (c.f. simulation engineers & operators), two requirements are essential:

First, an end-user should have the technological support for discovering suitable models and deploying them into the target simulation system. In the M&S domain, these are aspects that apply to all reusable M&S resources. They concern (1) effective resource management and (2) effective integration through technological compatibility. Both are active and ongoing fields of research within the NATO M&S community. Regarding resource management, stakeholders highlight the importance of meta-data for managing resources for purposes of discovery and providing access to the actual models (NATO STO, 2018). Regarding model integration, an active area is the concept of modelling and simulation as-a-service (MSaaS), which advocates treating M&S models in a more service-oriented manner (Berg et al., 2018). Both topics are directly relevant to CGF behaviour models as well, in terms of reusability.

Second, an end-user requires a certain level of transparency from a behaviour model in the form of explicit knowledge. This includes ‘offline’ knowledge concerning a model’s purpose, scope and capabilities (what does it offer), and ‘online’ knowledge that allows inspection and tracing of the model’s run-time performance (what decisions did it make and why). As CGF behaviour models become more complex, the need for end-users to have (at least) some rudimentary insight into the model’s decision-making grows. Explainability of behaviour models is an active area of research, both for rule-based approaches and connectionist approaches (Miller, 2018). Sufficient transparency increases the end-users’ trust and confidence in the quality and performance of a model, thereby increasing the chance of reuse.

TOWARDS A REUSABLE BEHAVIOUR CONSTRUCT

In this section we present a concept for a general-purpose behaviour construct. This construct is modelled specifically to promote efficiency and reusability throughout the behaviour modelling process. It addresses the three directions for effective behaviour modelling that were identified in the previous section in a synergetic manner. Throughout this paper we refer to this construct as a Behaviour Building Block (BBB).

The notion of a behaviour here is quite broad, though its focus is geared towards CGF behaviours at the *tactical* level. A BBB can represent a cognitive activity at any level of abstraction. On the one hand, it can represent a complete behaviour model for a CGF (e.g. fighter pilot behaviour). On the other hand, it can represent sub-activities such as the execution of a mission (e.g. a defensive counter air), a coordinated behaviour with a team-member (e.g. formation), communication of situational awareness elements (e.g. locations of hostile treats), a tactical maneuver (e.g. an evasive tactic), or an atomic action (e.g. a change in heading).

Figure 2 shows the concept model of a BBB with involved stakeholders, highlighting properties that facilitate efficiency and reusability.

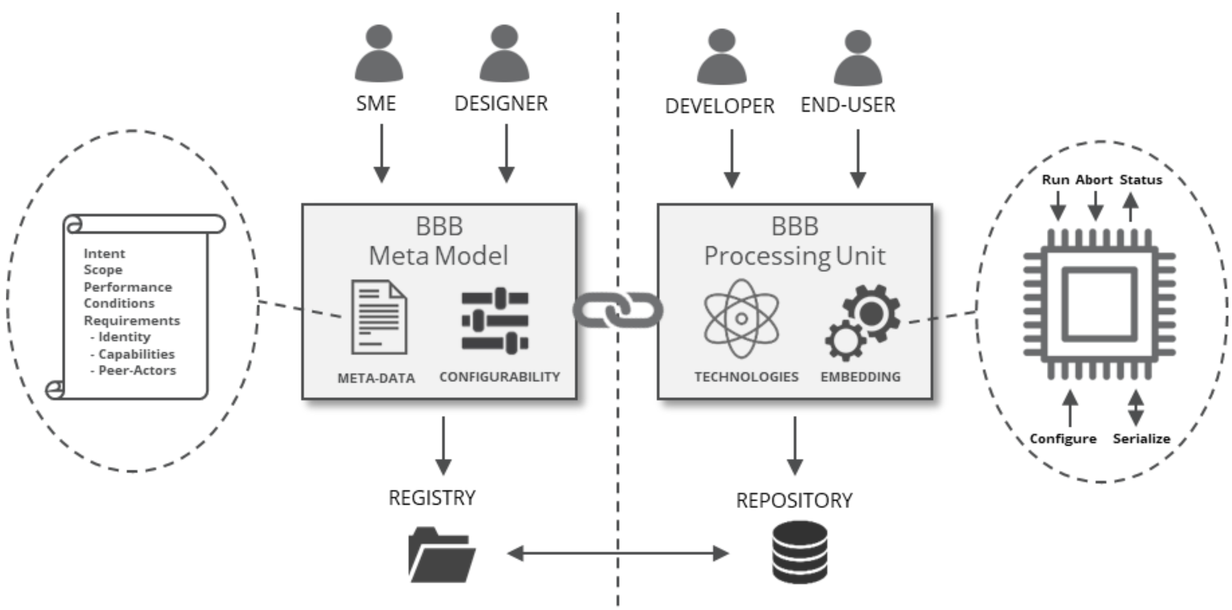

Figure 2. Behaviour Building Block Concept

The BBB concept consists of a Meta Model (left-side) and a Processing Unit (right side). They relate to the behaviour modelling artifacts defined in Figure 1: The Meta Model represents the (technology-independent) behaviour specification, acting as a formal design contract between stakeholders on the meaning and purpose of a behaviour construct. The Processing Unit represents a developer’s implementation of a Meta Model that can be deployed by end-users in a target simulator. Multiple implementations can exist for the same Meta Model (e.g. for the purpose of difference in procedure details between different organizations). Management of these M&S resources can be achieved through storage in registries and repositories to enable efficient discovery and integration, following recent trends in M&S resource management (NATO STO, 2018).

In the remainder of this section we describe the Meta Model and Processing Unit in more detail, focusing on properties that increase efficiency and reusability in the behaviour modelling workflow.

BBB Meta Model

The purpose of the Meta Model is to streamline the behaviour modelling workflow between designers, developers and end users through technological support. It is most strongly related to the knowledge management and elicitation process. Efficiency and reusability is achieved through the following properties:

Property: Meta-data
Meta-data provides explicit knowledge about a behaviour, including expert knowledge obtained through knowledge elicitation. It captures (semi-)formal information that serves all stakeholders:

- *Behaviour Intent:* A description of the main purpose of the behaviour.
- *Operational Scope:* The design boundaries and constraints for this behaviour in terms of situations or contingencies it can cope with. E.g. mission behaviour design for air-to-air combat that does not take into account any presence of air-to-ground weapon systems.
- *Performance Characteristics:* Details on how to realize the behaviour, such as expert knowledge on the tactical reasoning or details on the performance of a specific tactical maneuver.
- *Behaviour Conditions:* Preconditions for the behaviour to be applicable, success conditions stating successful realization of the behaviour, or failure conditions stating when the behaviour cannot be accomplished.
- *Identity Requirements:* The type of CGF this behaviour can be applied to. Depending on the generality of the behaviour, it may be applicable to only a specific type of military aircraft, all aircrafts, or all platforms from air, land and maritime domains.
- *Capability Requirements*: The excepted platform capabilities of the CGF (e.g. the behaviour assumes active missile capabilities and the presence of a radar warning receiver)
- *Peer-Actors:* Dependencies with potential peer-actors required to achieve the behaviour. E.g. other members of a team required for possible coordination or communication part of the behaviour.

For a designer, the above information serves as an explicit agreement between the designer and an SME on a behaviour definition. For a developer, it should capture sufficiently detailed information to allow independent implementation. For an end-user, it should offer sufficient information (*transparency*) for the end-user to judge whether the behaviour is a suitable candidate for integration into a CGF behaviour model. Lack of transparency is regularly an obstacle for users to integrate predefined behaviour models, being uncertain about its performance and how it affects the simulation.

Meta-data can provide the above information in different forms such as free text, graphical illustrations, formal data fields or references to external (military) documents. Effective use of meta-data increases efficiency in the modelling workflow: securing information as explicit knowledge limits the need for inter-communication between SMEs, designers, developers and end-users (caused by a lack of information or uncertainties about semantics).

Property: Configurability
Configurability can be achieved through parameterization of a behaviour. User-defined parameters allow the behaviour to become applicable to a wider variety of situations for a CGF. For instance, an *intercept* behaviour could be parameterized by a *target*, a *velocity* and *angle of approach*, or by parameters on a higher abstraction level such as a specific *Tactical Operational Procedure* to be applied. A *communication* behaviour can contain parameters for a *receiver* and a *message content*. It is a designer's challenge to define suitable parameters. A developer has the responsibility to create a successful implementation taking into account the possible parameter values. An end-user configures the behaviour during integration by providing parameter values to make it applicable to the specific situation.

BBB Processing Unit

The BBB Processing Unit is explicitly linked to a Meta Model. Its purpose is to (1) offer developers a mechanism to implement the Meta Model and (2) offer end-users the ability to embed the implementation into action-selection structures. The processing unit can be seen as a wrapper for the BBB's internal implementation that can be executed in a standardized manner through a general-purpose execution interface (see the illustration on the right side in Figure 2). It contributes to efficiency and reusability through the following properties.

Property: Paradigm-agnostic implementation
A BBB does not enforce specific behaviour modelling or action-selection paradigms. Developers have the freedom to choose a suitable technology for implementing a behaviour, which can range from hand-crafted approaches (e.g. scripts, state machines or planners) to machine learning approaches (e.g. supervised learning based on available behaviour data or reinforcement learning). From an end-user point of view, the BBB hides the internal implementation details. Explicit knowledge about the behaviour's purpose and performance can be obtained through the associated Meta Model.

Property: Effective integration
A BBB can be embedded in a variety of different behaviour modelling paradigms as a logical unit of behaviour. It could represent a *script* in a scripting engine; a *state* in a (hierarchical) finite state machine (FSM); a *node* in a behaviour tree (BT); a *goal* in a belief-desire-intention (BDI) system; or a *context* in a context-based reasoning (CxBR) engine. Although many different paradigms exist for orchestrating decision-making, they often employ similar approaches for executing units of behaviour: they can be initiated or aborted by an external cause (e.g. a transition firing in an FSM, a context-switch in a BT or the selection of a higher priority goal in a BDI system), and internally they can succeed (finish execution or meet some goal condition) or fail (an unrecoverable problem occurred that prevented successful execution). A BBB exploits this shared characteristic through its standard execution interface for controlling the unit of behaviour. This facilitates reuse of BBBs through effective embedding in modelling paradigms in a more service-oriented manner.

Hierarchical Decomposition

As mentioned earlier, BBBs can represent behaviours at different levels of abstraction. This implies some form of (recurrent) hierarchical decomposition of BBBs into sub-BBBs (e.g. high-level activities that are decomposed into sets of sub-activities, until reaching the level of atomic actions). Hierarchical decomposition of BBBs facilitates all stakeholders:

- For a designer in collaboration with an SME, it facilitates conceptual modelling by following a *divide-and-conquer strategy* to break down the problem of understanding complex behaviours. This is also a natural approach for humans to explain behaviours and is a feature that is employed in most behaviour modelling paradigms.
- From a technological and developer's point-of-view, it allows *hybrid technology solutions* for implementing BBBs or composing CGF behaviour models consecutively. Through the properties of the BBB Processing Unit, hybrid models come for free. For instance, a BBB that represents a high-level goal may use a behaviour tree (BT) implementation to represent goals as a set of tasks (sub-goals). These could be implemented by sub-BBBs using a mixture of other symbolic approaches (e.g. state machine or a script) or sub-symbolic approaches (e.g. a neural network).
- For an end-user, it promotes *transparency* as it increases explainability and traceability. The use of hierarchical BBBs gives insight into the reasoning process at different levels of abstractions (through their linked Meta Model specifications). At run-time they can be traced and logged for post-analysis.

An open issue that remains is what methodology to follow for an effective decomposition of BBBs to satisfy all stakeholders. Decomposition strategies fall outside the scope of this paper, though related design patterns have been proposed for human behaviour modelling (Taylor & Wray, 2004) and knowledge acquisition strategies for military tactics (Gonzalez et al., 2005; Stensrud et al., 2004).

In conclusion, in this section we have taken the directions for effective behaviour modelling from the previous section and materialized these into a concrete concept for a reusable behaviour construct. It illustrates an integrated approach for modelling behaviours where the emphasis is put on *design contracts* (the Meta Models), rather than focusing on specific technological solutions. The designer's specification is directly accessible to end-users at the computational level, under the assumption that developers offer implementations in compliance with the contract. In the next section we illustrate how the proposed concept can be applied in practice.

CASE STUDY

In this case study we apply the concept of a reusable behaviour construct (BBB) from the previous section by demonstrating its feasibility in effective behaviour modelling in a use case for air-to-air combat behaviour. We illustrate the reusability of tactical CGF behaviours across two simulation systems. The first system is a multi-ship tactical training research simulator for fighter pilots named Fighter-4-Ship (F4S), located at a Netherlands Airforce Base. The simulator uses the commercial-of-the-shelf (COTS) product *Presagis STAGE* for generating threats. The second system is used for CD&E purposes and analyzes tactics and missile performance in air-to-air encounters. It is part of an in-house Battle Lab and uses *Matlab Simulink* for running flight- and missile models.

System Implementation

The system implementation used in this case study is illustrated in Figure 3. Its components are described below.

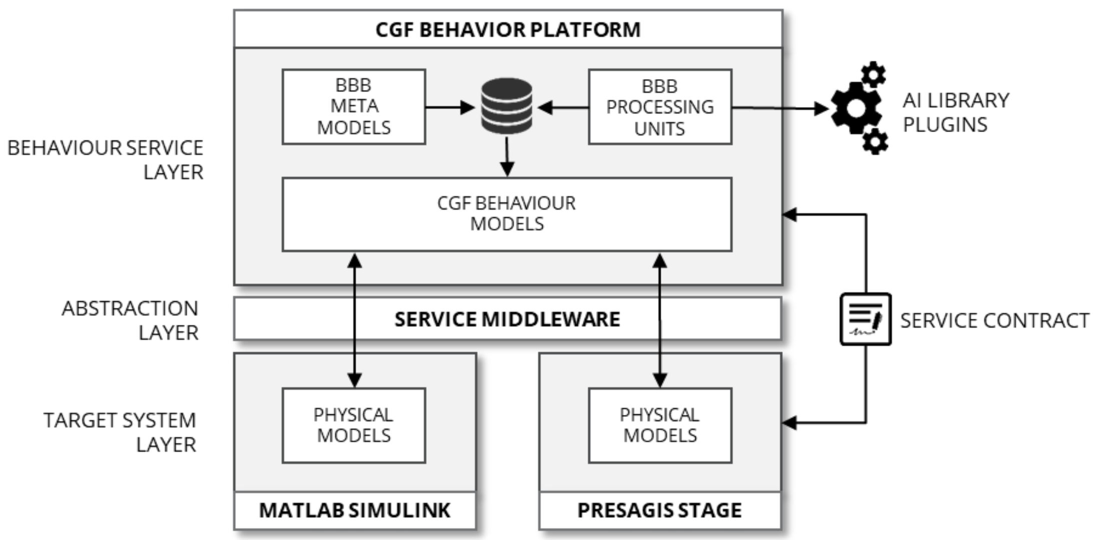

Figure 3: System Implementation

The BBB concept is implemented within an existing in-house developed CGF behaviour platform (operating at the Behaviour Service Layer). This platform can be used to create CGF behaviour models and provide these in a service-oriented manner to a target simulation system.

In the platform, BBB Meta Models can be specified through a graphical user-interface and stored in a BBB Library. Consequently, one can create implementations for Meta Models (the BBB Processing Units) which are stored in the same library. Implementations can be created by means of plugins for different AI technologies. Currently plugins exist for hierarchical finite state machines (HFSMs) and Dynamic Scripting, which is a rule-based machine learning algorithm (Toubman, Roessingh, Spronck, Plaat, & van den Herik, 2016). Additional preliminary plugins exist for context-based reasoning (CxBR) and a Utility-based system.

A service middleware at the Abstraction Layer manages the interface between a CGF's behaviour model and its physical model (e.g. associating a behaviour model for a pilot with a specific fighter platform in the target system). Interoperability is achieved through *service contracts* which define the inter-model interface: observations (sensor data) and actions (control commands) (van Oijen, Vanhée, & Dignum, 2012). Couplings between the behaviour platform and the two target simulation systems that are used in this case study are pre-existent. They share the same service contract (interface definitions for perception and control of sensor systems, navigation systems, weapon systems and communication systems). In the remainder of this section we evaluate the system using a concrete behaviour use case.

Behaviour Modelling Use Case

As an experiment we demonstrate the behaviour modelling process in the scope of modelling fighter pilot behaviour in a Defensive Counter Air (DCA) mission. The goal is to efficiently define and develop reusable BBBs that can be used to compose CGF behaviour models, applicable in scenarios for two different simulation systems.

In the first phase of the process, the aim is for the designer to understand the behavioural activities of fighter pilots in a DCA mission. Together with an SME (a former F-16 pilot), the activities are decomposed into logical units of sub-behaviours such as contexts, tasks and actions. These are defined as BBB Meta Models. In the scope of the DCA mission, activities of both blue and red forces are addressed. Figure 4 shows a high-level sketch of the identified and implemented BBBs. The sketch and operational information is simplified for illustration purposes.

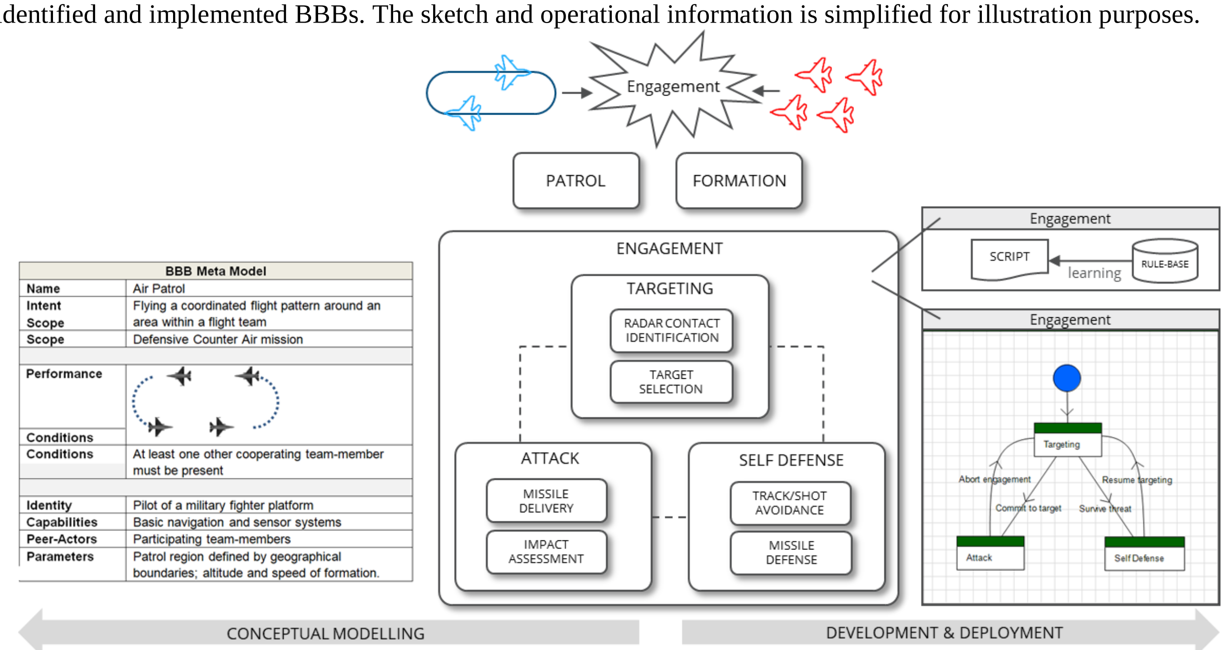

Figure 4: BBBs for a fighter pilot

In the center of the figure, the BBB decomposition is illustrated. At the highest level three BBBs are identified: a *patrol* behaviour for the blue fighters, a *formation* flight for the red fighters and an *engagement* for both blue and red which resembles the air-to-air combat. The engagement is decomposed into three contexts (targeting, attack and self-defense) which internally contain more low-level activities and tactics. At any time a pilot can switch between contexts, depending on the situation. An example of a BBB's specification is shown on the left side of Figure 4.

In the next phase, the developer uses a mixture of AI techniques to implements the BBBs. We highlight the *Engagement* which was implemented twice using different techniques: one *controlled* model and one *explorative* model. In the controlled model, the engagement and all sub-BBBs were implemented as hierarchical finite state machines (HFSMs). Being hand-crafted allows full control over the tactical decision-making within the model. The explorative model was implemented using Dynamic Scripting, a rule-based machine learning algorithm that learns optimal scripts from a database of decision-making rules. In this case the algorithm learns an optimal strategy for the full engagement (sub-BBBs are not present).

At the final phase, the end-user creates CGF behaviour models by combining the BBB implementations for concrete scenarios. In the CD&E system, behaviour models are prepared for both blue and red whereas in the training system, only models for red are used. Whether or not to select the controlled or explorative model for the *Engagement* depends on the purpose of the scenario. The controlled model can be used for analysis or training based on known blue or red tactics. Note that in this study only a single generic model was used, though in practice different models can be used to differentiate between own tactics and known enemy tactics. The explorative model can be used for red to test or train against unknown enemy tactics, or for blue to explore new tactics. Even though the model was generated using machine learning, learned scripts are still human-readable and can be inspected to trace decision-making (a property of Dynamic Scripting).

In this case study we illustrated how the development of CGF behaviour models benefit from reusable behaviour building blocks. From the point-of-view of the simulation systems in which the CGFs operate, two levels of abstractions can be seen. First, the CGFs' behaviour models are abstracted from their physical counterpart in the simulation environment through the behaviour service platform, allowing the same models to be used in different systems. Second, individual sub-behaviours are abstracted from the CGF behaviour model. This allows models to be composed from a library of sub-behaviours, geared towards a specific application domain. The end-user can compose a CGF behaviour model based on the descriptive Meta Models for the BBBs, offering a degree of transparency without being aware of the implementation techniques or details. Although a quantitative evaluation for efficiency and reusability has not been performed in this study, it can reasonably be argued that effectiveness is increased.

CONCLUSION AND DISCUSSION

In this paper we presented directions for effective behaviour modelling from a holistic point of view, taking into account the workflow process and involved stakeholders for developing and employing CGF behaviour models. Effective behaviour modelling in this view relates to any *efficiency within the process* (both in communication between stakeholders and model development) and *reusability of data* (behaviour specifications or models).

We have identified a set of properties for effective behaviour modelling and combined them into a single concept (the BBB concept) where they operate together in a synergetic manner, serving all stakeholders. To summarize they include:

- *Meta-model specifications* for effective knowledge management and elicitation by designers and SMEs.
- *Behaviour composition & configurability* to increase reuse through behaviour building blocks.
- *Hybrid technologies* that allow freedom of technology including symbolic and sub-symbolic approaches.
- *M&S resource management* for storage, discovery and accessibility of models.
- *Service-oriented integration* for effective integration into simulation systems by end-users.
- *Transparency & traceability* of models to increase explainability and end-users' trust in models.

The above properties each are R&D fields on their own. It is not our goal to propose fundamentally new contributions to individual fields, but rather to raise awareness that they are equally important in terms of effective behaviour modelling. In our case study these properties have been demonstrated in practice in a limited example.

There are two additional properties for effectiveness that are open challenges. The first one relates to *Model interoperability*. The demand for service-oriented CGF behaviours increases when current proprietary simulation systems offer insufficient or impractical means to create these behaviours internally. Current efforts for providing M&S as a service can facilitate and has been considered for terrain and weather services, weapon models or communication services (Siegfried & van den Berg, 2015). However for a CGF behaviour model to be offered in a truly service-oriented manner requires interoperability and compatibility with a CGF's physical model in the target system. E.g., a pilot behaviour model requires knowledge about the fighter platform it will control (what systems are on-board, what sensor data can be expected, what are its actuator capabilities). In our system implementation, this knowledge is managed by a *service contract*, allowing the same behaviour model to be associated with CGFs in different target systems that comply with the contract. In our experience, defining such a service contract even for a single CGF is not trivial and standardization efforts in this area will become increasingly important.

The second property relates to *Behaviour representation*: Effective standardization of a behaviour representation (language) seems improbable, considering the fragmented and evolutionary landscape of modelling approaches. Additionally, we have argued that developments in machine learning will lead to opportunities for generating (parts of) behaviour models. In order to cope with the transition from more hand-crafted to data-driven implementation approaches, it is important to stay flexible in behaviour model design and support hybrid solutions.

Besides the above facets for effective reuse of CGF behaviour models, there are organisational issues that can prevent reusability: e.g. (1) *Governance*: How to incentivize designers to specify and manage reusable models that go beyond the scope of their immediate needs?; and (2) *Security*: How to deal with classification restrictions for employing expert knowledge into models? Especially for realistic behaviour models dealing with tactical decision-making, one is likely to deal with sensitive and confidential information (from data sources or SMEs).

In conclusion, in this paper we have focused on effective CGF behaviour modelling based on interests from different stakeholders. We have presented pragmatic design practices which address the underlying challenges: How to effectively manage and exploit expert knowledge to design behaviour models? How to prepare for potentially disruptive AI technologies (i.e. transition to data-driven approaches)? And how to keep complex behaviour models transparent and understandable in order to increase end users' trust? These challenges become increasingly more important as the demand for more advanced CGF behaviour models increases.

REFERENCES

Alstad, A., Mevassvik, O. M., Nielsen, M. N., Løvlid, R., Henderson, H., Jansen, R. E. J., & de Reus, N. M. (2013). Low-level battle management language. *Proceedings of the 2013 Spring Simulation Interoperability Workshop, No. 13S-SIW-032*. Simulation Interoperability Standards Organisation.

Balint, J., Allbeck, J. M., & Hieb, M. R. (2015). Automated simulation creation from military operations documents. *Interservice/Industry Training, Simulation and Education Conference*, 15227. I/ITSEC Orlando, Florida.

Berg, T., Lloyd, J., Huiskamp, W., Grom, A., Siegfried, R., & Phillips, R. (2018). Modelling and simulation as a service: Rapid deployment of interoperable and credible simulation environments Final Report of NATO MSG-136. *2018 Winter Simulation Innovation Workshop, SIW 2018, 2018 Winter Simulation Innovation Workshop, SIW 2018, 21 January 2018 through 26 January 2018*. SISO-Simulation Interoperability Standards Organization.

Brook, A. (2015). *C2SIM: C2 and Simulation Systems that have been Implemented* (No. STO-EN-MSG-141). NATO STO.

Gonzalez, A. J., Castro, J., & Gerber, W. (2005). Automating the acquisition of tactical knowledge for military missions. *The Journal of Defense Modeling and Simulation*, *2*(3), 145–160.

Karli, M., Efe, M. Ö., & Sever, H. (2017). Air combat learning from F-16 flight information. *2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, 1–6. IEEE.

Lewis, M., Alexander, T., Huiskamp, W., & Blais, C. L. (2019). *A Reference Architecture for Human Behaviour Representation* (No. MSG-127). NATO STO.

Luotsinen, L. J., & Løvlid, R. A. (2015). Data-driven behavior modeling for computer generated forces. *NATO Modelling and Simulation Group Symp. M&S Support to Operational Tasks Including War Gaming, Logistics, Cyber Defence (MSG-133)*, 1–13.

Miller, T. (2018). Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*.

NATO STO. (2018). *M&S Resource Discovery and Access* (No. TR-MSG-100). NATO STO.

Potts, J., Griffith, T., Sharp, J., & Allison, D. (2010). Subject matter expert-driven behavior modeling within simulation. *Proceedings of the 2010 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC)*.

Roessingh, J. J., Toubman, A., Oijen, J. van, Poppinga, G., Løvlid, R. A., Hou, M., & Luotsinen, L. J. (2017). Machine learning techniques for autonomous agents in military simulations - Multum in parvo. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 3445–3450. Banff, Canada.

Siegfried, R., & van den Berg, T. (2015). M&S as a Service: Paradigm for Future Simulation Environments. *Proceedings of the 2015 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC)*.

Smith, R. E., & Dunn, A. G. (2016). *Tactical Behavior Mining of a Soldier-Based Gaming Environment (Briefing Charts)*. Army Tank Automotive Research Development and Engineering Center.

Stensrud, B. S., Barrett, G. C., & Gonzalez, A. J. (2004). Context-Based Reasoning: A Revised Specification. *FLAIRS Conference*, 603–610.

Taylor, G., & Wray, R. E. (2004). Behavior design patterns: Engineering human behavior models. *Proceedings of the Behavior Representation in Modeling and Simulation Conference*.

Toghiani-Rizi, B., Kamrani, F., Luotsinen, L. J., & Gisslén, L. (2017). Evaluating deep reinforcement learning for computer generated forces in ground combat simulation. *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 3433–3438. IEEE.

Toubman, A., Roessingh, J. J., Spronck, P., Plaat, A., & van den Herik, J. (2016). Rapid adaptation of air combat behaviour. *Proceedings of the Twenty-Second European Conference on Artificial Intelligence*, 1791–1796. IOS Press.

Toubman, A., Roessingh, J. J., van Oijen, J., Løvlid, R. A., Hou, M., Meyer, C., … Turčaník, M. (2016). Modeling behavior of computer generated forces with machine learning techniques, the nato task group approach. *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 001906–001911. IEEE.

van Oijen, J., Vanhée, L., & Dignum, F. (2012). CIGA: A Middleware for Intelligent Agents in Virtual Environments. In M. Beer, C. Brom, F. Dignum, & V.-W. Soo (Eds.), *Agents for Educational Games and Simulations* (pp. 22–37). Berlin, Heidelberg: Springer Berlin Heidelberg.