Using Intelligent Agents in Military Simulation or "Using Agents Intelligently"

Gil Tidhar　**Clint Heinze**　**Simon Goss**
Graeme Murray　**Dino Appla**　**Ian Lloyd**

Defence, Science, and Technology Organisation (DSTO)
506 Lorimer Street, Fishermen's Bend,
Victoria, Australia
Firstname.Surname@dsto.defence.gov.au

Abstract

Modern defence systems include advanced aircraft, ships, radar, weapons, command and control systems, and most importantly human operators. The main objective of modelling and simulation tools is to allow operational analysts to rapidly specify and evaluate existing and proposed systems and procedures for operating these systems. Such tools are required to model all aspects of defence systems including physical systems and human operators and the reasoning processes that they adopt.

Agent-oriented technology is a natural candidate for developing a model of reasoning processes performed by human operators. It allows the operational analyst to work at a high level, formulating cognitive processes, while keeping the detailed computer programming hidden. This premise has led to the development of the Operator-Agent. The base model was completed in June 1996. The model is fully operational and is an integral part of the tools used by operational analysts from the Australian Department of Defence. It has been successfully used for operational analysis and evaluation of multi-billion dollar acquisitions.

Introduction

> *"Outside, it's pitch black, no moon and a heavy overcast sky has completely obliterated the meagre, night illumination.... It is not the sort of night you would like to be out driving your car, but there you are at 60 meters above the ground travelling at close to 1,000 kph. You're thinking to yourself, `the most intelligent decision I could have made was to stay at home'...."*
>
> ***WGCDR Rick Owen***
> ***Royal Australian Air Force***

Modern defence organisations use advanced systems as part of their military operations. Defence systems are typically both expensive to purchase and operate. Furthermore the circumstances under which such systems are used are not always simple to replicate in training. Modelling and simulation is becoming the main approach used by defence organisations in support of evaluation of existing and proposed defence systems. The key requirements are for tools that allow operational analysts to rapidly specify and evaluate defence systems and develop procedures for best use of these systems.

Human operators are an integral part of any military operation. Such operators include air-combat pilots, mission commanders, fighter controllers, ship captains, and many others. They all reason about the environment, evaluate it, make decisions about controlling defence systems, and interact with other operators.

Modelling human operators engaged in military operations is a very challenging task. This is because humans exhibit intelligent behavior that is at times difficult to understand let alone automate. Building a model of human reasoning processes that can be validated, repeated, and meets real-time performance requirements is even harder.

Development of defence systems' models has followed development of programming languages and software engineering. Current models (e.g., TAC-BRAWLER (Bent 1993) and AASPEM (Boeing 1985)) were implemented using structured or object-oriented approaches. This applies to models of physical systems and reasoning processes. In most current systems models of reasoning processes are intertwined with models of physical systems. The only exception is the TAC-Soar system developed by Tambe et al. (Tambe et al. 1994). It uses Agent-Oriented technology to model air-combat pilots.[1]

Agent-Oriented technology has focused on the development of embedded real-time software systems which also exhibits (1) autonomous behaviour; (2) both reactive-and pro-active behaviour; and (3) the ability to interact with other systems. Furthermore theoretical models of agent-oriented technology have been inspired by philosophical and psychological theories of human behaviour. One agent-oriented approach successfully used in industrial applications is the Belief-Desire-Intention (BDI) approach (Rao and Georgeff 1995). In the past decade BDI systems have matured from experimental systems to commercially developed, fully tested, and

---

[1] The TAC-Soar system is a prototype system and has not been developed for or tested under operational conditions. This primarily manifests itself in system performance.

supported systems.

The BDI approach seems a natural candidate for the development of a new model of the human operator. Seven years ago the Australian Department of Defence engaged in a project to develop Intelligent Computer Generated Forces. This included developing a model of a human operator using the BDI approach. The model is to be used for modelling human operators in large military simulations. It is referred to as the *Operator-Agent*.

Technical benefits of the Operator-Agent include the capacity of an operational analyst to work at the level of tactical representation and the general easing of the software-engineering task in developing and maintaining the knowledge base. An unexpected consequence of the adoption of agent-oriented technologies has been a change at the organisational level. Interactions between analysts and military personnel have been improved and there is now significantly greater cooperation between operational analysts and human-factors experts. These advances can be summarised as a new paradigm in knowledge representation and acquisition. It follows a more natural way of describing the decision making process of operators. This innovation has significantly reduced the time and cost of developing and maintaining a model of a human operator.

In the remainder of this paper we describe the domain of application and the requirements from the system. We also describe the underlying technology used and provide details of the implementation. Furthermore we describe the development processes that led to this implementation and the benefits obtained through the use of the system.

A Typical Simulated Scenario

Let us consider two opposing forces. Red Team is planning a strike mission to destroy a ground target within Blue Team's territory. It assembles a *package* of aircraft adopting different roles. There is a group of sweep aircraft to clear a path ahead of the strike aircraft, there are escort aircraft to accompany the strike aircraft, and there are the strike aircraft themselves. These three groups have a single goal: to attack the ground target that has been designated for them. Each has assigned responsibilities that require communication and interaction with other aircraft.

Blue Team has aircraft operating in an air defence role protecting their airspace and vital areas from the incursion by Red Team. These aircraft may either be launched from an air-base when it becomes apparent that an attack is imminent or, if hostile actions have been occurring for a number of days, the aircraft may be flying patrols over an area where an attack is expected.

The hierarchy of command within the team exists in a flexible dynamic way to allow the team to operate at several levels and to split and reform as the situation dictates (Shaw 1985). Within the operation of a standard mission there are aircraft performing different tasks. An escort aircraft may accompany a strike aircraft as its wingman. A pair of low performance fighter aircraft might accompany a pair of high performance fighter aircraft to give the illusion of four high performance fighters.

Each situation may require the use of a different command and control structure. Thus the sub-teams will, at different times, adopt various command and control roles within the team. The different mission goals adopted by the team may require the sub-teams to adopt functional responsibilities with respect to the conduct of the mission. Thus an aircraft may have both a command and control role (e.g., a leader) and a functional role (e.g., an escort).

Requirements from Reasoning Model

Tambe et al. (Tambe et al. 1994) have provided some insight into building believable agents for simulation environments. Here we provide more detailed requirements. We identify four types of requirements necessary for a model of a human involved in military scenarios: (1) ability to interact with the environment; (2) ability to exhibit rational behaviour when reasoning about the world; (3) ability to exhibit irrational behaviour; and (4) ability to provide a good simulation environment.

A basic aspect of human behaviour is the way humans interact with the environment. These interactions occur through a variety of sensors and actuators. We require that the simulation system include the following features:

1. **Sensing:** The ability to sense the world through multiple sensors, e.g., eyes, ears, etc., and create a single model of the world from multiple sensory input.
2. **Actions and Physical Capabilities:** The ability to act and affect the world, e.g., walk, talk, etc. and conform to physical limitations determined by the human body.

When reasoning about the world humans use a variety of techniques and methods. These include building and maintaining situation awareness, planning, pursuing multiple goals simultaneously, and interleaving pro-active and reactive behaviours. We thus require that the simulation system also include the following features:

3. **Building and Maintaining Situation Awareness:** The ability to analyze the model of the world and identify particular aspects that require a response.
4. **Decision Making and Reasoning:** The ability to perform complex reasoning, e.g., make decisions, plan, perform spatial and temporal reasoning, etc.
5. **Simultaneous Goals:** The ability to hold multiple goals and interleave their achievement.
6. **Proactive and Reactive:** The ability to react to the changing world and to interleave pursuing goals and reacting to the world.

Humans exhibit behaviours that are not always rational or easily explicable. Thus a model of human behaviour should be able to simulate emotions, social awareness, and innovation. We thus require that the simulation system also include the following features:

7. **Emotions:** The ability to represent and manipulate emotions and model the way these emotions affect other processes.
8. **Social Awareness:** The ability to interact with other

humans being modelled and to represent and manipulate social structures.

9. **Innovation:** The ability to adopt innovative and novel responses when faced with unfamiliar scenarios.

The above requirements relate to the fidelity of the simulation. As these models are typically used for the purpose of analysis and evaluation of military scenarios there are additional requirements from such models. These requirements refer to the simulation environment itself and include the following features:

10. **Determinism and Repeatability:** Given a particular scenario, the ability to always exhibit a predetermined behaviour and the ability to repeat the exact simulated behaviour under similar conditions.[2]
11. **High Level Specifications:** The ability to specify and modify the behaviour of the agent using a high-level and relatively appropriately abstract language.
12. **Explanations:** The ability to provide clear and high-level explanations of the way reasoning is performed.
13. **Levels of Knowledge:** The ability to model different types and levels of knowledge (e.g., knowledge about the world and about how to behave in the world).
14. **Real-Time Performance:** The ability to perform activities in a time scale comparable to human activity.

The details of the models developed depend on the required fidelity of the simulation and particular aspects of the scenario that are being investigated. For some investigations it may be sufficient to include a crude model of the behaviour or to ignore some aspects altogether.

The Underlying Technology

To model the reasoning processes performed by human operators we have adopted a Belief-Desire-Intention (BDI) agent model. In particular we have used the dMARS™ system that is an implementation of the BDI approach.

We now provide a description of the underlying BDI theoretical model and the components of a dMARS agent.

The BDI Theoretical Model

The logical foundations of BDI systems are based on the philosophical concepts of intentions, plans, and practical reasoning developed by Bratman (Bratman 1987). The semantic model includes multiple possible worlds that model the uncertainty inherent in the agent's environment and the agent's limited perception of this environment.

The beliefs, goals, and intentions are represented as special modal operators. The language used is a combination of Modal Logic and Computational Tree Logic (CTL).

Beliefs represent the current state of the environment as perceived by the agent. Desires represent the states of the environment the agent would like to be in. The agent's desires are limited only to those desires that are both attainable and consistent (referred to as *goals*). An intention represents the commitment of the agent to achieve a particular goal by progressing along a particular path that leads to that goal.

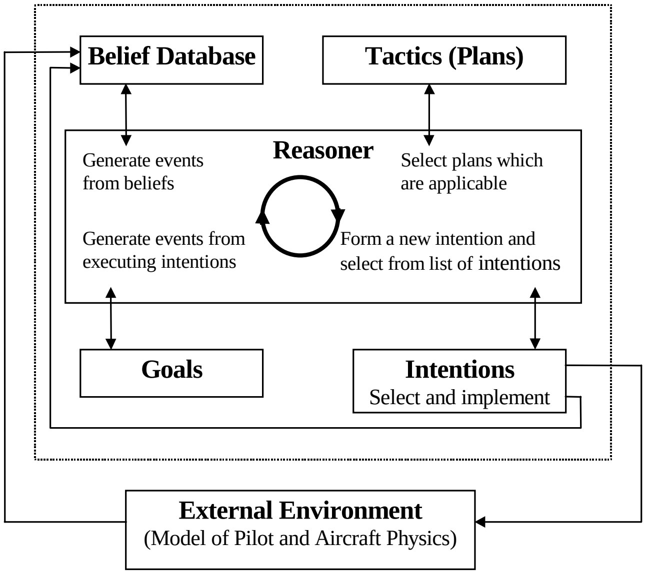

Figure 1: The Operator-Agent Control Loop

This logical model is augmented with plans and an operational semantics to form an abstract architecture of BDI agents (Rao and Georgeff 1995). Underlying the abstract architecture are the concepts of "bounded rationality" and "embedded systems". It is assumed that: (1) the agent has limited computational resources; (2) the agent can actually affect the world by acting; and (3) the environment is changing while the agent is reasoning.

These concepts lead to the development of a model of plans as "recipes" (Rao 1997). Plans are designed to achieve a particular goal under particular circumstances. They are supplied in advance and are not generated by the agent "on the fly". An agent can have multiple plans to achieve the same goal under the same or different circumstances.

Each plan is an abstract combination of sub-goals to be achieved or actions to be executed. Plans can be viewed as representing an abstract notion of a path that leads from the current state to the goal state. Sub-goals represent an intermediate state along the path. Plans are used in combination with a particular goal to form an intention.

Deliberation is done through the selection of goals, plans to be used to form an intention, and intentions to be executed. Decisions are based on the agent's beliefs. The process is known as *means-end reasoning*.

Intentions are executed through the achievement of sub-goals, modification of the agent's beliefs, and execution of actions. Sub-goals are achieved through the formation of sub-intentions. Sub-intentions are formed only when the achievement of the sub-goal is attempted. This is known as the *least-commitment* approach.

---

[2] Under probabilistic behaviour repeatability is measured statistically over repeated simulations.

***dMARS** is a registered trademark of the Australian Artificial Intelligence Institute, Melbourne, Australia.

Components of a dMARS Agent

Each agent is composed of a set of beliefs, goals, plans, and intentions (see Figure 1). Beliefs are represented in first-order logic. For example, a belief that the range from `WARLOCK1` to *BANDIT1* is 40 miles is represented as *(range WARLOCK1 BANDIT1 40)*.

Goals are descriptions of desired tasks or behaviours. Plans are procedural specifications representing knowledge on ways to achieve a goal or react to a situation. Each plan includes an invocation condition, a context condition, and a body (Rao and Georgeff 1995, Rao 1997) (see Figure 2). A dMARS agent can also have meta-level plans. These plans contain information about the manipulation of the beliefs, goals, plans, and intentions of the BDI agent itself.

The invocation condition describes the event that must occur for the plan to be executed. Events may be the acquisition of a new goal (resulting in a goal-directed invocation), changes to the agent's beliefs (resulting in data-directed invocation), or messages from other agents. The context condition describes contextual information relevant for plan execution.

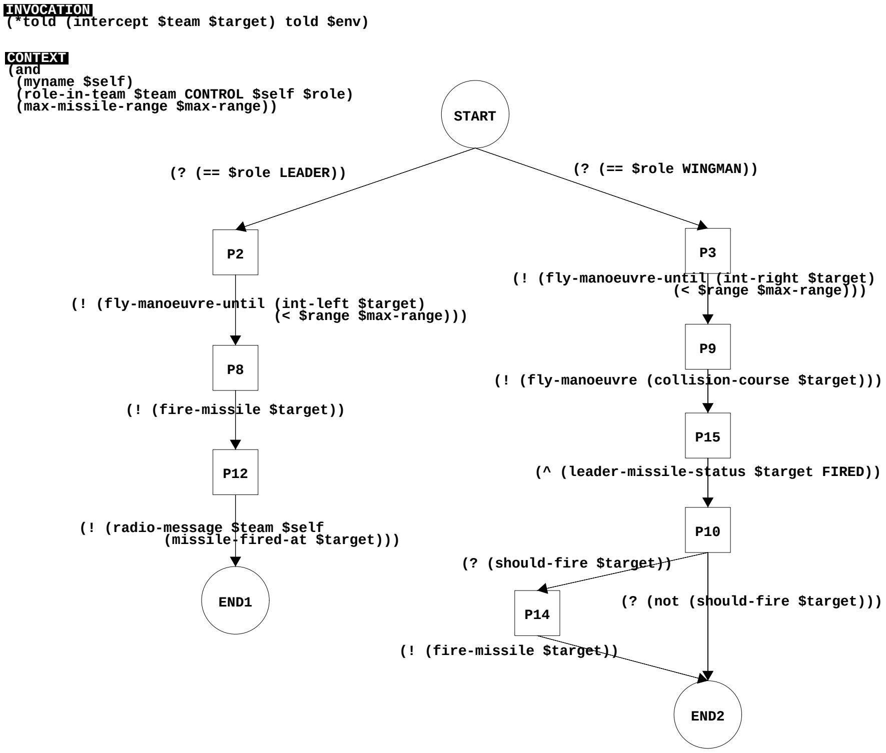

Figure 2: Team Tactics for conducting a pincer intercept

The body of a plan can be viewed as a procedure or a tactic. It is represented as a graph with one start node and one or more end nodes. The arcs in the graph are labeled with sub-goals to be achieved, modifications to the agent's belief database, and actions that should be performed.

In the plan language, an attempt by the team *WARLOCK* to achieve a goal to intercept *BANDIT1* is written as *(! (intercept WARLOCK12 BANDIT1))* and test for a belief that *WARLOCK1* has a control responsibility for the team *WARLOCK12* is written as *(? (role-in-team WARLOCK12 CONTROL WARLOCK1 LEADER)*. Other operators such as asserting (=>) and retracting (~>) a belief and waiting (^) for a belief to be true are also available. The character $ denotes variables in the plan language.

An intention embodies the agent's commitment to achieve a particular goal, respond to a change in its beliefs, or respond to messages from other agents, using a particular plan. It combines an event and an instantiated plan.

The agent's set of intentions contains all those tasks that the agent has chosen for execution, either immediately or at some later time. At any given moment, some of these intentions may be suspended, some may be waiting for certain conditions to hold, and some may be meta-level intentions. Only one intention can be executed at any given time. The choice of that intention depends on the agent's beliefs and the meta-level intentions. Further details on dMARS can be found elsewhere (d'Inverno et al. 1997).

The Operator-Agent

The model of a human operator has been divided into two components: (1) the physical aspects (e.g., sensors, actuators, human body and its limitations, etc.); and (2) the reasoning and decision making performed by the human operator. The physical aspects have been implemented using standard modelling techniques.

The implementation of the BDI agent that models the reasoning of a human operator involved in military operations required the design and development of specialized features. These were done using a variety of BDI features and have resulted in a type of agent referred to as the "Operator-Agent". In the context of the Operator-Agent we refer to plans as *tactics*.

The Design of the Operator-Agent

A model of an operator that operates in highly dynamic environments has to address specific problems such as the strategies for monitoring the environment, the strategies for monitoring and adjustment of the achievement of goals and tactics, and decision making speed. The Operator-Agent includes specialized components for: (1) for dynamic monitoring of the environment; (2) monitoring and adjustment of achievement of goals and tactics; and (3) adjusting the decision making speed.

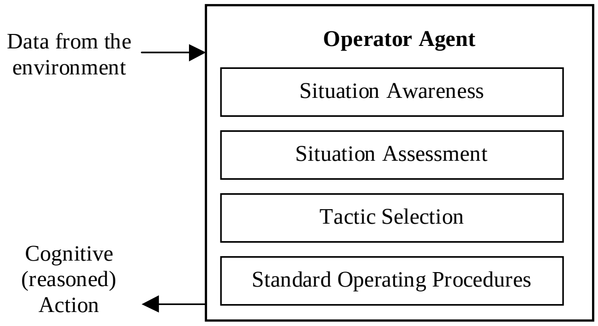

Figure 3: Agent's Architectural Design

The functional decomposition of the Operator-Agent includes the following modules (see Figure 3). **Situation Awareness** module for maintaining the agent's perceived view of the environment. **Situation Assessment** module for examining the perceived situation and producing a subjective evaluation. **Standard Operating Procedures** module containing knowledge about available tactical behaviour. **Tactics Selection** module for responding to the evaluated situation and selecting relevant tactical responses from the available Standard Operating Procedures.

Environment Monitoring

An Operator-Agent receives continuous sensory input. From this sensory input the Operator-Agent identifies an abstract situation and then re-evaluates this situation. This process is referred to as *situation assessment*. The process of situation assessment is relatively computationally intensive and is performed only as required.

Situation awareness allows the agent to determine the conditions for invoking situation assessment. This decision is itself situation specific and depends on the current activities of the agent. The processes of tactics execution and situation assessment can dynamically modify the conditions for invoking situation assessment.

Monitoring and Adjustment of Goals and Tactics

As mentioned before the Operator-Agent must exhibit a combination of reactive and goal-driven behaviour in a dynamic environment. These two features demand that the Operator-Agent continuously monitor and adjust its goals and the tactics it employs in achieving these goals. This is achieved using three specific mechanisms: (1) least commitment approach; (2) elimination of irrelevant goals; and (3) re-evaluation and re-selection of tactics.

The least commitment approach is part of the dMARS model. It allows the agent to commit to the means of achieving the sub-goal at the last possible moment. The agent can select the most appropriate plan for that time instead of committing in advance to a plan that may prove to be inappropriate when it should be executed.

In a dynamic environment it often happens that suspended intentions to achieve a previous goal or respond to a previous situation become redundant. Maintenance and housekeeping of goals and intentions are performed. In the Operator-Agent such maintenance is performed whenever a new goal is added to the Operator-Agent, a new reaction is required, or a tactic has been completed.

When a suspended intention (or tactic) is restarted the agent re-evaluates the tactics used in achieving the goal. This evaluation involves a two step process: (1) determining what would be the best tactics to employ in the new situation; and (2) determining the exact sub-goal in the chosen tactics in which execution should proceed.

Decision Making Speed

In a real military operations there are variety of operators with varying levels of knowledge and experience. An experienced pilot may not only know more tactics but may also react faster under pressure. The agent's tactical knowledge is modelled using a variety of tactical (i.e., plan) libraries. The experience of the operator is modelled through the use of different tactical libraries and a specification of the reaction and decision making. The analyst defining the simulation can determine the time it would take the Operator-Agent to perform certain activities or make a decision. These times could be situation specific and can change dynamically.

Satisfying the Requirements

The simulation environment requirements (Requirements 10-14) state that the model used should include an explicit and well-understood formulation of the modelled behaviour. In the BDI approach specification of agent behaviour is based on the concept of a plan.

A plan is an abstract combination of sub-goals to be achieved and actions to be taken. Such plans can either be generated on the fly using a planner or can be specified in advance in plan libraries. A typical agent-oriented plan language is a high-level language (Requirement 11). This allows the analyst to gain a better understanding of the agent's behaviour (Requirement 12).

Plans are reasoned about and executed using some form of an engine that is capable of performing complex reasoning and follow some decision making procedure, e.g., means-ends analysis (Requirement 4). The nature and complexity of the reasoning is a combination of the engine itself and the plans it manipulates. These could be modified to allow for varying levels of knowledge and abilities (Requirement 13). The decision making speed depends on the complexity of the plan and reasoning. Using abstract plans provided in advance and combining them during execution allows for the real-time response (Requirement 14).

Note that the behaviour of the agent is completely dependent on the knowledge provided in the plans and the algorithm of the engine. Thus, the behaviour of the system is completely deterministic. This together with deterministic simulation of the scenario's dynamics leads to fully repeatable simulation (Requirement 10).

The explicit representation of the goals and intentions of the agent allows the agent to maintain multiple simultaneous goals (Requirement 5). This feature combined with the continuous interleaving of sensing, reasoning, and acting ensures that the agent both reacts to the changing world and interleaves goal-driven and data-driven behaviours (Requirement 6). As to situation awareness (Requirement 3) it seems that as the level of understanding of this mental process increases so does the ability to provide a formal agent-oriented model for it.

A high-level representation of beliefs and knowledge allows the agent to reason about data as well as abstract concepts (Requirement 13). Furthermore the agent can represent in some basic way social concepts such as teams, sub-teams, and roles in a team (Requirement 8). Other social phenomena such as structures for an organisation are still under investigation.

Although agents can exhibit very complex behaviour, this behaviour must be explicitly specified. It follows that they can not actually exhibit behaviour that is not well understood or follow procedures that are not clearly defined. Furthermore, such an approach does not lend itself to performing complex transformations of data (or numbers) or the filtering of such data (or numbers).

Such are the characteristics of some of the required behaviours specified above. In particular it seems that current agent-oriented systems are not very effective in performing sensing (Requirement 1) and incorporating a model of emotions (Requirement 7).

Another required behaviour which current agent-oriented systems are unable to provide is innovative behaviour (Requirement 9). This limitation goes together with the requirement for real-time performance (Requirement 14). With limitations of current technology, and our limited understanding of how humans invent novel responses, real-time simulation of such behaviour is presently impossible.

As mentioned above, the characteristics of the specification language and the reasoning engines that execute and manipulate these specifications in agent-oriented systems make them well suited for simulating human reasoning. By the same token, the characteristics of the dynamics of physical systems and the way actions taken affect the world (Requirement 2) make agent-oriented systems unsuitable for simulating them.

To overcome these limitations, both requirements 1 and 2 are currently modelled using standard modelling techniques. The information collected by the sensors is sent to the Operator-Agent and it in-turn sends high-level acting instructions to the actuators.

Implementing an Operational System

The Development Process

The initial concept of using agent-oriented technology for modelling human operators in military operations was introduced in early 1991 with a concept demonstrator (Rao et al. 1992). The development of an operational Operator-Agent, from initial specification to a fully functional operational system, took close to three years. The system has been in operational use since June 1996.

Wooldrige and Jennings (Wooldrige and Jennings 1998), note that developing agent-oriented systems is a software engineering task that includes additional risks – namely the risks associated with developing embedded real-time distributed systems. We also prescribe to this approach and have adopted state-of-the-art Software Engineering and Software Project Management techniques to mitigate these risks. The system has been developed to IEEE standards and we have adopted an iterative software development process with incremental delivery of functionality.

The behaviour of the Operator-Agent has been independently verified and validated by domain experts. This was achieved through the specification of operational scenarios and the desired behaviour. Furthermore, tracing of plan execution in the Operator-Agent has been used in validating the decision-making processes.

Given the complexity of the development and the risks involved we had to establish a specialised team. In particular we required (1) expert knowledge of the required functionality and the domain knowledge; (2) expert knowledge of the existing technology and simulation systems; (3) expert knowledge of the new technology and artificial intelligence; and (4) expert knowledge of developing advanced software systems.

The key to the successful development was the characteristics of the development team. The team included operational analysts, experts in the existing simulation systems, experts in artificial intelligence, and experts in software engineering and software project management.

The number of active team members and their expertise changed depending on the stage of the project. There were 14 people involved with an average of 8 experts actively engaged throughout the development process.

The operational analysts that use the system would typically have background in operational research, aeronautical engineering, applied mathematics, or physics. The deployment of the system to operational analysts took close to 6 months and commenced close to the completion of development. The deployment included educating operational analysts on the agent-oriented language used in developing the Operator-Agent and its tactical behaviour. Operational analysts are currently performing the maintenance, enhancement, and development of the Operator-Agent as part of their routine development of human operator models.

Occasional reviews of the Operator-Agent are performed by operational analysts and agent-oriented experts. These experts are brought in specifically for this task. To date there have not been any major design changes required.

Implementation Details

The particular implementation of the Operator-Agent involves plans, database relations, database entries, goals, and intentions. The number of plans in the agent's plan library varies from agent to agent depending on the skills and capabilities of that agent. A typical Operator-Agent has over 400 plans in its plan library.

As to the agent's database, again this number varies from agent to agent. A typical agent has close to 300 predicates defined in its database. These predicates represent the types of declarative knowledge the Operator-Agent can reason about. The amount of data stored in the agent's database depends on the size of the unfolding scenario and the information the agent has about it. A typical agent in a scenario with 32 pilots (i.e., 24 vs. 8) has approximately 800 database entries in the database.

The number of concurrent goals held by a typical agent depends on the state of the mission. The agent would typically hold around 10 concurrent goals. These goals relate to the various operations of the aircraft and achievement of the mission.

In addition to these goals the agent also has data driven behaviours in which it reacts to the various inputs from its sensors. In a typical state the agent processes, in parallel, data for as many as 8 contacts, evaluating the threat they pose to the pilot. This is done using between 10 and 14 intentions that are processed in less than 50 milliseconds. Overall, a typical Operator-Agent handles over 25 concurrent intentions on a regular basis. The overall decision making time for the Operator-Agent is modifiable to allow for modelling of pilots with a variety of decision-making capabilities. The Operator-Agent is currently running on Silicon Graphics computers running the IRIX 6.4 operating system.

Integration with Existing Simulation Systems

The approach adopted involves building a powerful operator model capable of interfacing with existing simulation systems. The key is to separate the reasoning models, physical models, and visualisation software.

From a modelling perspective, we replaced the existing reasoning processes in the existing simulation system with interfaces to the Operator-Agent. From a simulation perspective we combined time-stepped simulation of physical systems with event-based (but time dependent) operator reasoning processes. The interfacing software passes messages about the perceived physical world to the reasoning software, and transmits back instructions for continuing or changing the present action.

The existing physical models still contains routines for aircraft and systems control, such as fighter intercept and combat manoeuvres and the logic of highly dynamic one-on-one close combat counter-manoeuvring. Not transferring this to the Operator-Agent reduces the amount of information that must be passed through the interface. The reasoning system performs the role of tactician rather than that of the pilot. Tactical instructions are coded as manoeuvres to be flown or system control instructions.

Benefits and Limitations

Both technical and organisational benefits emerged from the development of the Operator-Agent. At the technical level the main benefit is that the operational analyst can now modify the knowledge base, representing the tactics and decision-making, without being concerned about the remaining physical systems modelling code. This reduced tactical development time from 4-6 weeks to 4-6 days.

In addition this approach is eminently suited to dealing with extensive repertoires of procedural team tactics. Simulating military operations involves modelling multiple aircraft types, multiple roles, multiple weapons systems, multiple sensors, and communication systems. The accuracy of the models directly affects the fidelity of the simulation and the effectiveness of using it as a tool for understanding and analysing military operations. Incorporating intelligent agents enables higher levels of fidelity in studies of larger scenarios.

At the organisational level the main benefit is that operational analyst need only work at the high level, formulating concepts, determining mission goals, and developing pilot tactics. The use of such high level concepts has made the model easier to understand by the military personnel and hence improved the elicitation of operational knowledge and behaviour.

The operational analyst is now concerned with the explicit development of cognitive models of the human operator. This has brought the work of operational analysts and human factors experts closer together. This manifests itself in shared terminology, mutually rewarding interactions, and shared research directions.

In previous work we suggested detailed models and provided detailed analysis of the use of agent-oriented technology for tactics selection (McIlroy and Heinze 1996, Tidhar et al. 1995). These models clearly demonstrate several important technical properties:

- Agent plans are written using a graphical format that is highly recognisable as a logical flow chart. The analyst is able to concentrate on creating the logical processes rather than on the code to represent them.
- The Agent plans are easily read with only minor familiarisation with the system conventions.
- Plans are very readily edited using drag and drop tools. An edited plan can be recompiled and linked without re-compilation of remaining code.
- Plans are executed when their invocation matches the defined conditions (e.g., a radar contact) provided that their context and properties (e.g., priority) are appropriate. The analyst does not have to code the environment that calls a plan; plans invoke other plans.
- The execution of plans (i.e., intentions) is controlled through meta-level plans that resolve conflicts between plans by identification of intentions or goals. A plan can fail without affecting overall execution; alternative plans with the same invocation are used to recover. Otherwise control returns to the meta-level plans. Success or failure of a plan can be noted and used to manipulate the process.
- Intentions can be suspended until conditions are again appropriate (e.g., missile evasion suspends strike mission).
- Plan libraries can represent different types of human operators. The plans can be controlled at run time, enabling control of the representation of participants (e.g., experienced leader, rookie wingman, etc.).
- The system can be stepped or stopped for examination of the reasoning processes. Invoked plans can appear on-

screen with progressive highlighting of execution paths.
These characteristics enable educated users to rapidly develop very complex reasoning processes. The system has the high readability and traceability required for working with domain experts, such as fighter pilots and fighter controllers. This enables Air Force operational personnel to gain confidence in studies of operational effectiveness. A substantial increase in the productivity of operational analysts has been primarily gained due to:

1. The ease of implementing and modifying the behaviour of the model of human operators.
2. The increased level of abstraction in the representation of declarative and procedural knowledge in this model.
3. The improved interaction between operational analysts, human operators, and human factors experts.

The model has been successfully used for operational analysis and evaluation of multi-billion dollar acquisitions.

The main limitation of the above approach is in current state-of-the-art Agent-Oriented technology. There is still no clear and complete Agent-Oriented Analysis and Design methodology. In developing the Pilot-Agent we used a modified Object-Oriented methodology. Although this has proven useful it has been done on a relatively ad-hoc basis. This limitation is particularly significant given the background of the operational analysts. To overcome this limitation emphasis is given to reviews of the design and implementation of the Operator-Agent.

Concluding Remarks

Defence organisations are primarily interested conducting successful and efficient military operations. These operations rely on the performance of the available defence systems and the expertise of the humans operating them.
Modelling and simulation of these operations a priority for defence organisations in evaluating existing and proposed defence systems. Modelling the human operators is critical to conducting such evaluations.
Research into candidate technologies for the modelling of human decision making led to the selection of agent-oriented technology for the development of the Operator-Agent. The Operator-Agent has been integrated into existing simulation systems and has been is in use since June 1996 by DSTO for the conduct of operations research. The model is being successfully used for operational performance analysis, evaluation of multi-billion dollar acquisitions, and development of standard operating procedures with the Royal Australian Air Force. Many of the technical benefits of agent-oriented technology were expected – indeed they were the reason for the adoption of the technology. The organisational benefits were largely unforeseen during the early stages of the project. The proliferation of the technology throughout DSTO and the strength of the commitment to further research are good indicators of the magnitude of these benefits and are at the core of the way DSTO conducts business.

Acknowledgments

We thank Serena Steuart, David McIlroy, Mario Selvestrel, Arvind Chandran, Martin Cross, Anand Rao, Paolo Busetta, and Andrew Lucas for their contribution to the development of the Operator-Agent. We also thank the anonymous reviewers for their valuable comments.

References

1. Bent, N. E., 1993. The TAC BRAWLER Air Combat Analyst Manual (Rev 6.1). Decision-Science Applications Report #906.

2. Boeing, 1985. Advanced Air-to-Air System Performance Model (AASPEM) Users Manual. Boeing Document D180-28938-1.

3. Bratman, M.,1987. Intentions, Plans, and Practical Reason. Harvard University Press, Cambridge, MA.

4. d'Inverno, M., Kinny, D., Luck, M., and Wooldrige, M., 1997. A Formal Specification of dMARS. In *Proceedings of the 4th International Workshop on Theories, Architectures and Languages*.

5. McIlroy, D. and Heinze, C., 1996. Air Combat Tactics Implementation in the Smart Whole AiR Mission Model (SWARMM). In *Proceedings of the SimTecT Conference*, Melbourne, Australia.

6. Rao, A., Morley, D., Selvestrel, M., and Murray, G., 1992. Representation, Selection, and Execution of Team Tactics in Air Combat Modelling. In *Proceedings of the Australian Joint Conference on Artificial Intelligence*.

7. Rao, A. S. and Georgeff, M. P., 1995. BDI Agents: From Theory to Practice. In *Proceedings of the First International Conference on Multi-Agent Systems*.

8. Rao, A. S., 1997. A Unified View of Plans as Recipes. Contemporary Action Theory, ed. Holmstrom-Hintikka, G. and Tuomela, R., Kluwer Academic Publishers, The Netherlands.

9. Shaw, R. L., 1985. Fighter Combat Tactics and Maneuvering. US Naval Institute Press, 6th Edition.

10. Tambe, M. and Jones, R. M. and Laird, J. E. and Rosenbloom, P. S. and Schwamb, K., 1994. Building Believable Agents for Simulation Environments: Extended Abstract. In *Collected Papers of the SOAR/IFOR Project*, Information Sciences Institute, University of Southern California, pages 78-81. Marina del Ray, CA.

11. Tidhar, G., Murray, G., and Steuart S., 1995. Computer-Generated Forces and Agent-Oriented Technology. In *Proceedings of the Australian Joint Conference on Artificial Intelligence Workshop on AI in Defence*, Canberra, Australia.

12. Wooldrige, M. J. and Jennings, N. R., 1998. Pitfalls of Agent-Oriented Development. In *Proceedings of the Second International Conference on Autonomous Agents*, Minneapolis, USA.
