ROYAL MILITARY COLLEGE OF CANADA

MODELLING OF SMALL-UNIT TACTICAL DECISION-MAKING IN A CONSTRUCTIVE SIMULATION

MODÉLISATION DE DÉCISIONS TACTIQUES POUR ESCOUADES DANS UNE SIMULATION CONSTRUCTIVE

A Thesis Submitted to the Division of Graduate Studies
of the Royal Military College of Canada
by

Dean J. Morrissey, CD1, MASc
Lieutenant Commander (ret'd)

In Partial Fulfillment of the Requirements for the Degree
Doctor of Philosophy

January 2017

Acknowledgments

To my supervisor, and good friend, Dr. Alain Beaulieu, thank you for your unwavering support for my thesis which was, to say the least, outside the norm in ECE. I also want to thank you, and my co-supervisor, Dr. Sidney Givigi, for pushing me to always strive for the highest possible standards.

To my wife, Louise, words cannot begin to express my gratitude for your support and encouragement. Without you, this work would not be the success that it is. Method L is still a source of inspiration and creativity for me.

To my daughter, Andrée-Anne, and my son, Benjamin, thank you for your support, and for showing me that inspiration can come from the most innocuous of places.

Abstract

Modelling of Small-Unit Tactical Decision-Making in a Constructive Simulation

The Canadian Forces, like most military organizations around the world, has increasingly turned to the use of Synthetic Environments, including serious games, to offset training and mission rehearsal costs in the face of shrinking budgets. As the shift to simulation-based training and experimentation continues, the complexity of the simulation scenarios grows and the realism of synthetic entities becomes a limiting factor for effective experimentation and training. While serious games provide unprecedented levels of graphical realism, the AI which controls the game entities has not kept pace. As a result, while serious games provide a framework within which to run training simulations, the lack of Human-Level AI means that in order to achieve training and replay value, these exercises are inherently manpower intensive. While Boyd, Ensley and Klein are major contributors to the field of human decision-making, their models are theoretical and are not easily implemented in a software architecture. We have developed a practical application to decision-making by merging those various theoretical models in order to demonstrate the competitive nature of warfare in the field, and thereby create more credible AI actors. To accomplish this, we have developed a software architecture using Soft Computing techniques to effectively model the time-competitive nature of Boyd's Observe, Orient, Decide and Act (OODA) loop. This has allowed us to create a more realistic model of the dynamics associated with the tactical decision-making process of small-unit leaders in a synthetic environment.

Résumé

Modélisation de décisions tactiques pour escouades dans une simulation constructive

Les Forces canadiennes, comme la plupart des organisations militaires du monde entier, se tournent de plus en plus vers l'utilisation d'environnements synthétiques, y compris de jeux sérieux, pour compenser les coûts de formation et de répétitions de mission face aux contractions budgétaires. Alors que la transition vers la formation et l'expérimentation basées sur la simulation se poursuit, la complexité des scénarios de simulation s'accroît et le réalisme des entités synthétiques devient un facteur limitant pour une expérimentation et une formation efficaces. Alors que les jeux sérieux fournissent des niveaux sans précédent de réalisme graphique, l'IA qui contrôle les entités de jeu n'a pas suivi le rythme. Par conséquent, alors que les jeux sérieux fournissent un cadre dans lequel des simulations de formation sont exécutées, le manque d'IA de niveau humain implique que, afin d'atteindre la formation et la valeur de reprise de jeu, ces exercices exigent intrinsèquement de la main-d'œuvre intensive. Bien que Boyd, Ensley et Klein sont les principaux contributeurs au domaine de la prise de décision humaine, leurs modèles sont théoriques et ne sont pas facilement implémentés dans une architecture logicielle. Nous avons créé une application pratique à la prise de décision en fusionnant ces différents modèles théoriques afin de démontrer la nature concurrentielle de la guerre dans un théâtre d'opérations, créant ainsi des acteurs IA plus crédible. Pour ce faire, nous avons développé une architecture logicielle utilisant des techniques de calcul souple afin de modéliser efficacement la nature compétitive temporelle de la boucle Observe, Oriente, Décide et Agir (OODA) de Boyd. Cela nous a permis de créer un modèle plus réaliste de la dynamique associée à la prise de décision tactique, dans un environnement synthétique.

Contents

List of Figures

List of Tables

List of Abbreviations

AAPG	America’s Army: Proving Grounds
ACM	Anti-Coalition Militia
AI	Artificial Intelligence
ANA	Afghan National Army
ANN	Artificial Neural Network
BDI	Belief–Desire–Intention
BG	Battle Group
BMU	Best Matching Unit
CGF	Computer Generated Forces
CDMM	Competitive Decision-Making Model
COA	Course of Action
FCM	Fuzzy Cognitive Maps
FIS	Fuzzy Inference System
FL	Fuzzy Logic
FPS	First Person Shooter
GPMG	General Purpose Machine Gun
JSON	Java Script Object Notation
L1SAE	Level 1 Situation Awareness Element
MF	Membership Function
NDM	Naturalistic Decision Making
OODA	Observe-Orient-Decide-Act
PMF	Performance Moderator Function
PR	Probabilistic Reasoning
RPC	Remote Procedure Call
RPD	Recognition-Primed Decision model
ROF	Rate of Fire
SA	Situation Awareness
SAF	Semi-Automated Forces
SAW	Squad Automatic Weapon
SC	Soft Computing
SE	Synthetic Environment
SME	Subject Matter Expert
SOF	Special Operations Forces
SOM	Self-Organizing Map

TSK	Takagi-Sugeno-Kang
UDK	Unreal Development Kit
VQ	Vector Quantization

1 Introduction

1.1 Background

1.1.1 Modelling and Simulation

The Canadian Forces, like most military organizations around the world, has increasingly turned to the use of Synthetic Environments (SE) to offset training and mission rehearsal costs in the face of shrinking budgets. Bassarab [1] states that, “During the last ten to fifteen years, the Land Force has witnessed a tremendous shift towards the use of specially developed simulation equipment to assist in training our personnel.” Similarly, Roman and Cyr [2] note that, “SEs will provide the Army with a powerful and resource efficient medium for the exploration of doctrinal alternatives, realistic training, support to operations, and experimentation into the applicability of leading-edge technologies.” Roman and Brown [3] note that the impressive levels of resolution offered by the latest in video game technology permit the emulation of real-world environments to such a degree that “... serious games have a role to play in military tactical training”—a finding echoed by Bruzzone et al [4]. The paper also provides several examples of trainees who had serious games included with regular training methods outperforming those who did not.

Simulation provides a training and experimentation capability that is either time-consuming and expensive to duplicate in the real world, or cannot be duplicated (i.e., testing systems that are not yet in service). As the shift to simulation-based training and experimentation continues, the complexity of the simulation scenarios grows and the realism of synthetic entities becomes a limiting factor for effective experimentation and training [5] [6]. The current workaround for the problem of lack of realism is to employ a team of human operators to manage synthetic entities [7], which can range from individual soldiers to formations of soldiers and/or vehicles. The human operator provides the knowledge and skills to ensure that synthetic entities perform in a realistic manner so the training can be effective or the experimental results valid. However, the availability of such a team—both in terms of numbers and individuals with the right skill-sets—then becomes a constraint on the availability of training and experimentation, and a limitation on the complexity (i.e., number of synthetic entities) that can be employed in a scenario. In a recent exercise [8] there was a roughly one for one mapping between human operators and synthetic entities, which is typical. As a result, the lack of autonomous realistic synthetic entities forms an artificial limit on the provision of effective training and experimentation capabilities using simulation. There are three broad categories of simulation [9]. These are:

- Live – Real people operating real systems, (e.g. a pilot flying a CF-188);
- Virtual – Real people operating simulated systems. Virtual simulations inject a Human-in-the-Loop into a central role to exercising motor control skills (e.g. a pilot flying a simulated CF-188); and
- Constructive – Simulated people operating simulated systems. Humans provide input to such simulations, but are not involved in determining the outcomes. It is the area of constructive simulation that we are targeting for our work.

Categorizing a simulation as live, virtual, or constructive can be difficult, as there is often no clear division between these categories (e.g. most First-Person-Shooters (FPS) are a combination of Virtual and Constructive simulations) [9].

Computer Generated Forces (CGF), sometimes referred to as Semi-Automated Forces (SAF) are synthetic entities within a battlefield simulation that are controlled by a software system [10]. CGF provide opposing forces, neutral entities, as well as supplementary friendly forces, and are key components for Virtual and Constructive simulations.

Abdellaoui et al. [11] note that current CGFs are, "...predictable, non-adaptable, with a distinguishable behaviour, and are easily defeated by human crews, which substantially reduces the replay value of training scenarios." As an example, if the first time a scenario is played, the player is ambushed by an enemy tank that is hidden behind a building, and the enemy tank always appears behind the same building in every run of the scenario, the player adapts, but from that point forward he is essentially learning the wrong lessons (negative learning). The current workaround to overcome this shortcoming is to use human operators to control critical entities [7]. As a result, Subject Matter Expert (SME) availability becomes a training bottleneck and an additional expense. Abdellaoui et al. conclude that to obtain realism while minimizing human intervention "...would require an AI capability that can overcome the current critical CGF shortcomings" [11].

Around the late 1990s, early 2000s, researchers began to seek what George and Cardullo, in their paper "Application of neuro-fuzzy systems to behavioral representation in Computer Generated Forces" [12], referred to as "humanlike expertise in the military domain."

While George and Cardullo were specifically interested in military simulations, the interest of researchers was not simply limited to this domain. In 2001, Laird and van Lent used the term "Human-Level Intelligence" in their paper "Human-Level AI's Killer Application: Interactive Computer Games" [13]. They state that: "Although one of the fundamental goals of AI is to understand and develop intelligent systems that have all the capabilities of humans, there is little active research directly pursuing this goal. We propose that AI for interactive computer games is an emerging application area in which this goal of human-level AI can successfully be pursued" [13].

They go on to state: "The thesis of this article is that interactive computer games are the killer application for human-level AI. They are the application that will need human-level AI. Moreover, they can provide the environments for research on the right kinds of problem that lead to the type of incremental and integrative research needed to achieve human-level AI" [13].

In their conclusion, Laird and van Lent write: "One attractive aspect of working in computer games is that there is no need to attempt a "Manhattan Project" approach with a monolithic project that attempts to create human-level intelligence all at once. Computer games provide an environment for continual, steady advancement and a series of increasingly difficult challenges" [13].

1.1.2 Cognitive Modelling Approaches

While there are a number of cognitive modelling approaches—ACT-R [14], SOAR [15], CoJACK [16], and PMFserv [17] to name a few—we will focus on one in particular by Jones et al. [18] as it was developed with Endsley, and it purports to fully support Endsley's SA model, whereas other approaches, according to Jones et al., do not. While CoJACK claims that it implements both Boyd's Observe, Orient, Decide and Act (OODA) loop and Endsley's model of SA, we can find no evidence that it does either in anything but a superficial manner.

In support of Endsley et al.'s research [19] for the U.S. Army in determining SA requirements for the infantry platoon leader, follow-on research conducted by Jones et al. proposes the use of Fuzzy Cognitive Maps (FCM)—which they refer to as SA-FCM in their implementation—as a way of improving the representation of SA and developing a model that replicates human cognition as it relates to SA [20]. In their paper, they state that while traditional modeling approaches are capable of representing Level 1 SA, an FCM can represent and help the human decision-maker understand the "so what" (Level 2 SA) and "now what" (Level 3 SA) of how the data impacts current and future situations, which are needed for effective decision-making. The report, however, does not provide an explanation as to why traditional modelling approaches are limited to Level 1 SA and, in fact, earlier in the same document, they state that, "... there is a tendency when using them [other SA modelling techniques] to model SA (as defined by Endsley) with an emphasis on Level 1 SA (perception) over the other higher levels." In an earlier paper, Jones et al. [18] note that other modelling approaches, "... generally do not include the comprehension (Level 2 SA) and projection (Level 3 SA) levels of situation awareness" thus their SA-FCM constitutes an advancement to cognitive modelling.

Aside from the fact that their implementation, which makes use of a simple, non-dynamic, planning-oriented scenario for validation, could easily fall prey to fuzzy logic's greatest weakness—combinatorial rule explosion—we believe that it only has, at best, utility for offline mission planning and is, therefore, unsuitable for the real-time approach that we present in Section 4.

Therefore, our research did not reveal any software architecture developments that could be used to create credible synthetic forces that can react to the environment, provide situational awareness or incorporate the uncertainty induced by the fog of war. One of the aims of our work was to research the state of the art in time-competitive decision-making models in order to simulate credible autonomous agents. From this research we have chosen the methods and models to develop a scalable architecture that can support these agents.

1.2 Problem Statement

Current CGF are predictable and non-adaptable, typically because their behaviour is scripted. The net result of this is that training scenarios lack replay value. The workaround to this problem is to employ human operators to control critical entities. The availability of these SMEs, however, then becomes a bottleneck and an extra expense for training.

1.3 Objective

As a first step toward solving the problem stated in section 1.1, our objective is to create an AI model that will reduce the considerable manpower requirements for running military simulations, to make CGF that respond appropriately to military tactics.

We have used state of the art models in situation awareness and decision making to create a more realistic decision model for AI decision making. By setting our software architecture on this more realistic model and by using soft computing to elaborate its behaviour, we can generate and validate credible synthetic forces.

1.4 Thesis Contributions

Our research shows that Boyd, Ensley and Klein are major contributors to the field of human decision-making. They have developed theoretical models which, however, have not yet been implemented in any kind of software architecture. Our thesis' first contribution is that we have developed a practical application to decision-making by merging those various theoretical models in order to demonstrate the time-competitive nature of warfare on the modern battlefield.

Based on his observations on air combat between MiG-15s and F-86s in the Korean War, former United States Air Force Colonel John Boyd developed a time-competitive decision-making model that is now commonly referred to as the OODA (Observe, Orient, Decide, and Act) loop. According to Boyd's theory, the key to victory is to be able to create situations wherein one can make appropriate decisions more quickly than one's opponent [21] [22] [23].

While useful in terms of illustrating the time-competitive nature of the decision-making cycle, Boyd's model alone is not sufficiently detailed to allow for the creation of a software architecture that supports this cycle.

Endsley's [24] model, for its part, describes the stages of Situation Awareness (Perception, Comprehension and Projection) which maps to Boyd's concepts of Observe and Orient. In a similar manner, Klein's [25] Recognition Primed Decision-Making model develops a naturalistic theory on how humans make decisions. The models of Endsley and Klein, however, are stand-alone models; therefore, our second contribution is that through our analysis, we have selected the strengths of each model and integrated them within the framework of the OODA loop, thereby providing a much better representation of the decision-making cycle. We must note, however, that the addition of Endsley's and Klein's models still did not allow us to fully represent the intent of Boyd's work.

What sets Boyd's work apart from other models is the human dimension—how humans react to change, and the stress that it brings—and how the human brain processes information under these conditions. Based on Boyd's model, each leader's OODA loop will continue until one side possesses a decisive advantage and emerges victorious. What is not accounted for in this description, however, is how we model this "mental lag". Boyd's model accounts implicitly for the differences between the abilities of the opposing commanders, but not how they must be modelled in order to depict an engagement between an expert and a novice decision-maker, or between two experts who possess varying degrees of expertise. While both Endsley and Klein discuss the importance of various environmental and individual factors in situation awareness and decision-making, they do not articulate how these factors would be accounted for in their respective models. Therefore, our third contribution was to introduce the cognitive Transition Model of Bridges [26] which allows us to model the effects of change on individual decision makers.

Another element which we must consider that complicates the development of a software model is the difficulties associated with decision-making in a synthetic environment, which is where our software model must work. Synthetic environments pose a unique challenge in that they provide a discrete representation of environmental change, which is not how humans perceive an essentially continuous world [27]. Similarly, synthetic actors do not 'see' in the same manner as humans [27], and this must also be accounted for in a model for a human decision-making architecture.

The fourth and final contribution of this thesis was to develop a software model using Soft Computing (SC) techniques [28]—specifically a fuzzy-neuro system—in order to effectively model the time-competitive nature of Boyd's OODA loop. We have demonstrated that by representing the interaction between opposing, or competitive, OODA loops, we have developed a more realistic model of the dynamics associated with the tactical decision-making process of small-unit leaders in a Constructive Simulation.

1.5 Conclusion

While modern video games provide unprecedented levels of graphical realism, the AI which controls the game entities has not kept pace. As a result, while these games provide a framework within which to run training simulations, the lack of what Laird and van Lent describe as Human-Level AI[13] means that in order to achieve training and replay value, these exercises are inherently manpower intensive.

The remainder of this dissertation is divided as follows: Chapter 2 will provide an in-depth look at the Boyd Cycle, followed by a detailed examination of Endsley's model of Situation Awareness and Klein's decision-making model. Chapter 3 will look at the specific SC methods that we have used in implementing the architecture. In Chapter 4, we will provide an overview of our software architecture. Chapter 5 will lay out the primary components and distributed nature of our software model. Chapter 6 will discuss the behavioural models that form the core of the CDMM. Chapter 7 will review the scenarios that we used to validate our model, and Section 8 will provide conclusions and proposals for future work.

2 Decision-Making

2.1 The Boyd Cycle

In the mid-1970s, former United States Air Force Colonel John Boyd began studying and writing about conflict and warfare. Based on his experience during the Korean War, he proposed a decision-making model to try to explain the success of the F-86 fighter pilots over their MiG-15 opponents. Despite the fact that the MiG-15 was considered an overall superior plane, with a higher ceiling, tighter turn radius and higher maximum speed, the kill ratio was 10:1 in favour of the F-86. While some of this advantage could be explained by superior US pilot training, it could not, Boyd believed, explain the entire difference since the North Korean pilots often achieved numerical superiority during air-to-air combat [22].

Boyd believed that other factors were at play, ones that offered the F-86 a decisive advantage in combat. The advantages that the F-86 had over the MiG-15, he believed, were its canopy design, which afforded the F-86 pilot better visibility and, because it used fully hydraulic controls, compared to the MiG-15's hydraulic-assist controls, its ability to transition more quickly between manoeuvres. These advantages permitted the American pilot to observe more and, as a result, orient himself more quickly to the changing situation and maneuver the aircraft in response. He hypothesized that the F-86 gained a time advantage with each new action and with each change in action the MiG-15's reaction was increasingly inappropriate which eventually resulted in the F-86 obtaining a good firing position [23].

Boyd proposed that conflict can be seen as time-competitive Observation-Orientation-Decision-Action (OODA) cycles. Each party to a conflict begins by observing. He observes himself, his physical surroundings and his enemy. On the basis of his observations, he orients, meaning he makes a mental image of his situation. He then makes a decision and acts on it. Because he assumes that his action has changed the situation, he observes again, and starts the process anew. This version of Boyd's decision-making model (Figure 2-1) is referred to as either the Boyd Cycle or the OODA loop, and it forms a connection between information and action [22].

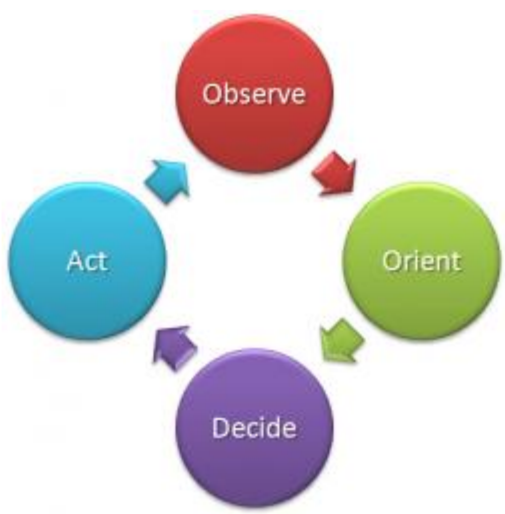

Figure 2-1: Simplified or "Rapid" OODA Loop [23]

In the John Boyd Roundtable [23], Osinga refers to this version of the Boyd Cycle as the rapid OODA loop and argues that, while it is the popular interpretation of Boyd's work, this view is too narrow an interpretation of the general OODA loop (Figure 2-2).

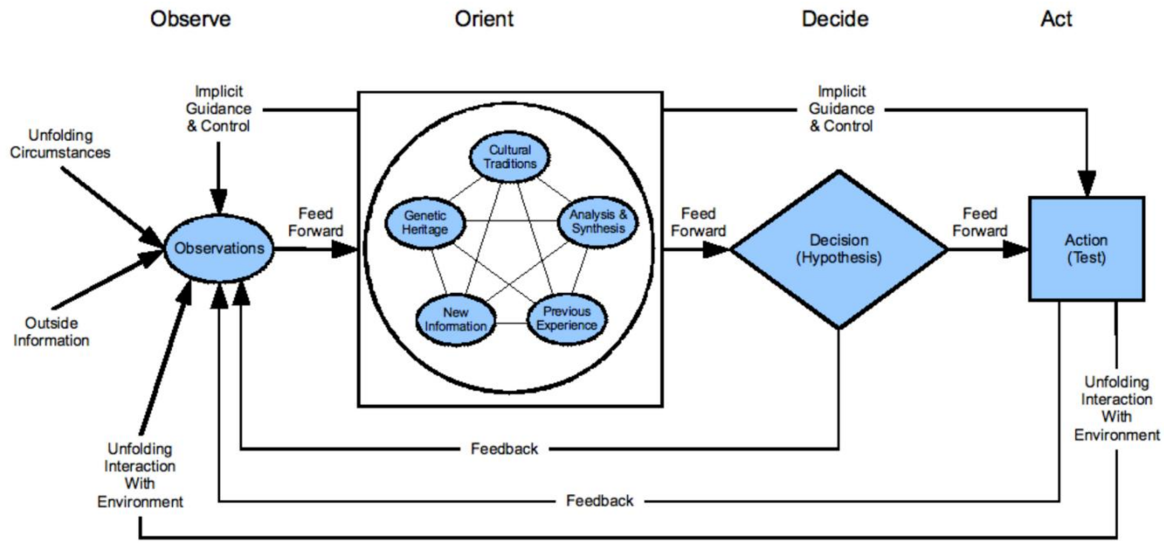

Figure 2-2: The General OODA Loop [22]

The Boyd Cycle has gained widespread acceptance, and has been expanded outside of tactical fighter combat to include business applications [23] and is even used in robot soccer [29]. That said, as Osinga notes, many interpretations of Boyd's work are, at best, incomplete or oversimplified and, at worst, wrong [22]. The Orient step presented in [29], for example, is often seen as simply a physical orientation, rather than the cognitive orientation proposed by Boyd. This is also frequently seen in books [30] and websites that purport to use the OODA loop in support of training for law enforcement agencies. Many who support the concept of the OODA loop focus exclusively on the speed of the decision-making cycle and speak of "getting inside" the opponent's OODA loop in the same way that a plane with a tighter turning radius could "turn inside" that of its opponent. While this is not incorrect, it is incomplete.

Lind's explanation summarized the popular understanding of the OODA loop: "In any conflict, the actors who can consistently and effectively cycle through the OODA loop faster, who can maintain a higher tempo of operations, gains an ever-increasing advantage with each cycle affording tactical initiative. The slower actor falls further and further behind in his actions and becomes increasingly unable to cope with the deteriorating situation. With each cycle the slower actor's actions become less relevant to the true situation, and become increasingly ineffective. This is the OODA loop in its simplest form" [31].

Osinga [22] notes that the UK military doctrine description of the doctrinally preferred method of war fighting, the maneuvrist approach, is pure Boyd (and fully in line with the US Marines doctrine) and provides a deeper understanding:

> The maneuvrist approach to operations is one in which shattering the enemy's overall cohesion and will to fight, rather than his materiel is paramount [...] significant features are momentum and tempo, which in combination lead to shock and surprise. Emphasis is on the defeat and disruption of the enemy—by taking the initiative, and applying constant and

unacceptable pressure at the times and places the enemy least expects—rather than attempting to seize and hold ground for its own sake. It calls for an attitude of mind in which doing the unexpected and seeking originality is combined with ruthless determination to succeed. A key characteristic of the maneuvrist approach is to attack the enemy commander's decision process by attempting to get inside his decision making cycle. This involves presenting him with the need to make decisions at a faster rate than he can cope with, so that he takes increasingly inappropriate action or none at all, thereby paralyzing his capability to react. Clearly any degradation of the overall command system which can be achieved by physical or other means will hasten the onset of paralysis.

As discussed in Section 1.4, we have developed a multi-stage architecture to model the time-competitive nature of Boyd's OODA loop. In the next two sections we will show how we decomposed the OODA loop further in order to facilitate the development of our decision-making architecture.

2.2 Observe and Orient

According to Boyd, decision-making occurs in a recurring cycle of Observe-Orient-Decide-Act. An entity (whether an individual or an organization) that can process this cycle quickly, observing and reacting to unfolding events more rapidly than an opponent, can thereby "get inside" the opponent's decision cycle and gain the advantage [21].

Boyd developed the concept of the OODA loop to explain how to direct one's energies to defeat an adversary and survive. Boyd emphasized that "the loop" is actually a set of interacting loops that are to be kept in continuous operation during combat. He also indicated that the phase of the battle has an important bearing on the ideal allocation of one's energies [21] [22].

Figure 2-2 shows that all decisions are based on observations of the evolving situation, tempered with implicit filtering of the problem being addressed. These observations are the raw information on which decisions and actions are based. The observed information must be processed to orient it for further making a decision. In notes from his talk "Organic Design for Command and Control", Boyd said: The second 'O', orientation—as the repository of our genetic heritage, cultural tradition, and previous experiences—is the most important part of the OODA loop since it shapes the way we observe, the way we decide, the way we act [21].

The first two steps of the OODA loop, Observe and Orient bear a strong similarity to Endsley's definition of Situation Awareness (SA). Although numerous definitions of SA have been proposed, Endsley's formal definition [24], "the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future," is the most widely used as it is applicable across multiple domains [32]. More informally, it is often stated as simply "knowing what's going on" [32]. Endsley's three levels of SA—Perception, Comprehension and Projection—are depicted in Figure 2-3.

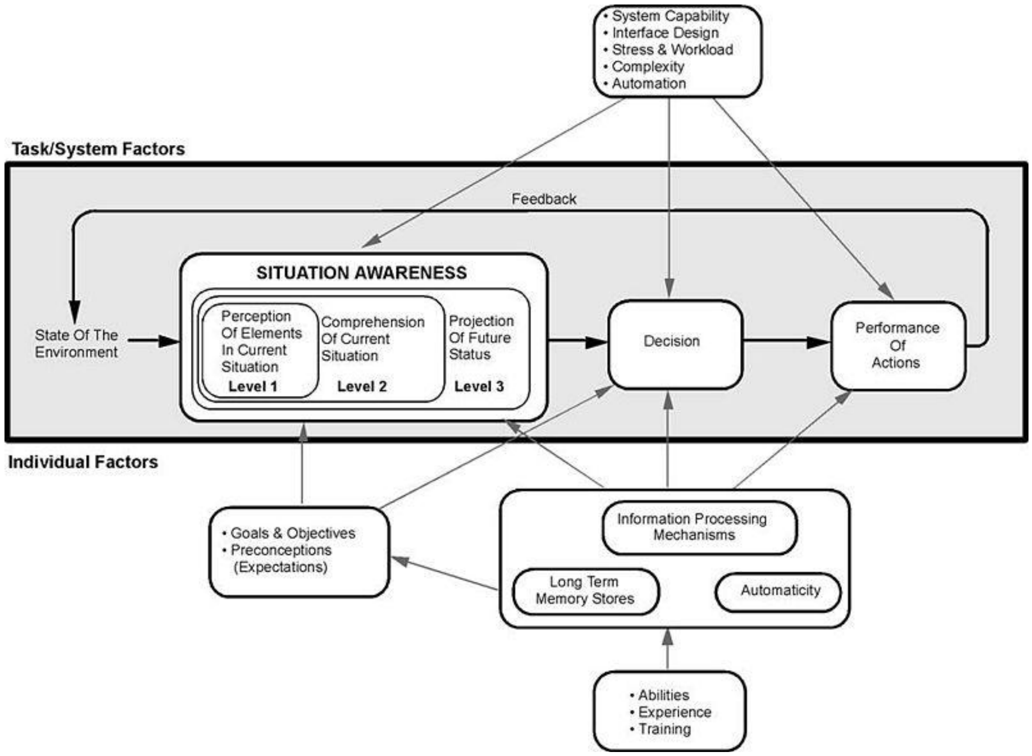

Figure 2-3: Endsley's model of situation awareness [24]

Perception, or Level 1 SA, involves the processes of monitoring, cue detection, and simple recognition, which lead to an awareness of multiple situational elements (objects, events, people, systems, environmental factors) and their current states (locations, conditions, modes, actions) [24]. It is important to make the distinction here between perception and knowledge or fact. Bratman [33], who developed the Belief-Desire-Intention (BDI) model, deliberately chose the term belief, rather than knowledge, in recognition of the fact that what an agent believes may not necessarily be true, and that it may change in the future.

Comprehension is the next step in SA formation and involves a synthesis of disjointed Level 1 SA elements through the processes of pattern recognition, interpretation, and evaluation. Level 2 SA requires integrating this information to understand how it will impact upon the individual's goals and objectives. This includes developing a comprehensive picture of the world, or of that portion of the world of concern to the individual [24].

Projection is the third and highest level of SA, which involves the ability to project the future actions of the elements in the environment. Level 3 SA is achieved through knowledge of the status and dynamics of the elements and comprehension of the situation (Levels 1 and 2), and then extrapolating this information forward in time to determine how it will affect future states of the operational environment [24].

Endsley's model of SA illustrates several variables that can influence the development and maintenance of SA, including individual, task, and environmental factors. For example, individuals vary in their ability to acquire SA; thus, simply providing the same system and training will not ensure similar SA across different individuals. Endsley’s model shows how SA “provides the primary basis for subsequent decision making and performance in the operation of complex, dynamic systems" [34]. Although alone it cannot guarantee successful decision making, SA does support the necessary input processes (e.g. cue recognition, situation assessment, prediction) upon which good decisions are based [35].

SA also involves both a temporal and a spatial component. Time is an important concept in SA, as SA is a dynamic construct, changing at a tempo dictated by the actions of individuals, task characteristics, and the surrounding environment. As new inputs enter the system, the individual incorporates them into this mental representation, making changes as necessary in plans and actions in order to achieve the desired goals. SA also involves spatial knowledge about the activities and events occurring in a specific location of interest to the individual. Thus, the concept of SA includes perception, comprehension, and projection of situational information, as well as temporal and spatial components [34].

Roman and Chapman [36] put Endsley's work into a military context in the following manner: To perceive the situation at the first level, the commander must have accurate static and dynamic data on all forces and environmental elements that can potentially affect operations. This would be termed Blue (friendly), Red (enemy), White (neutral) or Brown (environmental) situation awareness in current doctrine. When these disparate elements of Level 1 situation awareness are assembled and patterns emerge, the commander forms a holistic picture of the environment and comprehends the significance of the various objects and events, thus achieving Level 2 situation awareness. This Level 2 awareness is enhanced into Level 3 awareness through the combination of the holistic picture with the dynamic (movement or activity) information related to the object.

Strater et al. [37] note that infantry platoon leaders operate in a complex environment requiring that they attend to multiple information sources, prioritize among competing and sometimes conflicting goals, and make rapid decisions, all under highly stressful conditions where the loss of life, either their own or others', is a constant threat. To complicate the matter, platoon leaders are often relatively inexperienced officers, with minimal service time, training and experience to draw on. In this environment, superior SA provides tremendous advantages to those with the ability to acquire it and the experience to use it. Endsley et al. [19] reinforce the importance of the concepts of ability and experience when they state that intangible human skills, such as tactical competence, problem-solving abilities, and the capacity to make decisions under the pressures of time and high risk are mitigating factors and are also heavily affected by the state of SA. We will explore how these concepts fit into our decision-making model in Section 4.

Endsley et al. [19] provide a more complete overview of the challenges of SA in infantry platoon operations, where she concludes that achieving high levels of SA in the highly complex and dynamic environment of combat is not easy. Many stressors act to degrade the platoon leader's SA, or to prevent him from gaining a high level of SA to begin with. Time pressure and the rapid tempo of operations can significantly challenge platoon leaders who often must struggle to maintain an up-to-date awareness of a rapidly changing reality. The conditions for gathering and assimilating information may rapidly deteriorate during combat operations.

Endsley goes on to note that fatigue brought on by heavy physical exertion, lack of sleep and night-time operations also degrade the platoon leader's ability to detect and process information vital to good SA. Poor environmental conditions, including noise, fog, weather, and smoke can directly obscure critical information. Stress and anxiety associated with warfare and the inherent uncertainty and confusion can all act to reduce SA. Periods of significant task underload or task overload can also lead to SA problems [19].

The factors that shape SA also can be greatly influenced by the enemy, who can alter the tempo of the battle and dramatically affect the conditions under which a battle is fought. Thus, infantry operations frequently must be conducted under the challenges of a number of factors, some naturally occurring, some task or enemy induced, that can all act to seriously degrade SA [19].

While not stated explicitly, we can again see elements of Boyd's competitive OODA loop emerge in Endsley's model, and in her description of rapid tempo, quickly changing circumstances, and of how enemy actions can degrade our own SA.

Returning to the competitive nature of Boyd's model, we can see that it is impossible to fully understand the situation that one is facing. Perceptions may, or may not, be true and, unless it is expressly stated, intent must be inferred. Given that opponents are attempting to disguise their actions from one another in order to create false perceptions which, in turn, could lead to incorrect conclusions (i.e. deny to them SA), both sides must operate in a climate of uncertainty.

Lind states: "Leaders have always faced a complex environment of imperfect knowledge, uncertainty and ambiguity in battle" [31]. Boyd [38] states that uncertainty is a fundamental and irresolvable characteristic of our lives, no matter how good our observations and theories for explanation are.

Once a leader has a mental picture of what he believes is going on—and this distinction is important, given the degree of uncertainty in his environment—he must make a decision and act on it, thereby closing the decision-making loop.

2.3 Decide

We can see from Figure 2-2 that, in addition to the feedback loop from Act back to Observe (which closes the OODA loop) there is an intermediate loop from Decide back to Observe. This is due to the fact that cognitive decision-making takes time, and this is exacerbated by the many battlefield conditions described by Endsley [19]. Therefore, with each iteration through the loop, the decision-maker must determine if he has enough information to make a decision. If the answer is no, then he returns to the Observe step and continues.

While Endsley's model, depicted in Figure 2-3, contains both "Decisions" and "Performance of Actions" steps, they are not discussed in her work. Therefore, for information on how individuals make decisions, we must look beyond Endsely's model.

In a whitepaper on Realism in Cognition and Emotion, members of the AOS Group [16] noted that many factors contribute to the decision-making process, including:

- Situation Awareness – the appreciation of those aspects of the current situation that are relevant to the question at hand.
- Predictive Capability – the ability of the agent to foresee the consequences of actions and the likely actions/reactions of other entities that are part of the scenario.
- Response Repertoire – the known action sequences for dealing with the current situation (skill set).
- Personal Preference – preferred methods of dealing with the current situation, often based on experience of previous successes and failures.
- Cognitive Effectiveness – the current state of the underlying cognitive architecture, affecting capabilities such as ability to recall facts, hold intermediate results in working memory, and stay focused on the problem.
- Affective State – the emotional factors that can influence a decision; for example, a high-level of fear can predispose a person to make an irrational decision.

Prior to the emergence of the Naturalistic Decision Making (NDM) model in 1989, research in the field of human decision-making focused on optimal ways of making decisions, defined as choices among alternatives, in well-structured settings that could be carefully controlled [25]. Kahneman, Slovic and Tversky [39] demonstrated that people did not adhere to the principles of optimal performance. Instead, respondents relied on heuristics as opposed to algorithmic strategies even when these strategies generated systematic deviations from optimal judgments as defined by the laws of probability, the axioms of expected utility theory, and Bayesian statistics [39]. In other words, people do not generate alternative options and compare them on the same set of evaluation dimensions. They do not generate probability and utility estimates for different courses of action and elaborate these into decision trees. In fact, even when they do compare options, they rarely employ systematic evaluation techniques [25].

NDM researchers sought to discover how people were able to make tough decisions under difficult conditions such as limited time, uncertainty, high stakes, vague goals, and unstable conditions. Klein postulates in his Recognition-Primed Decision (RPD) model that people use their experience in the form of a repertoire of patterns, and that the patterns highlight the most relevant cues, provide expectancies, identify plausible goals and suggest typical types of reactions in that type of situation [40].

However, there is more to the RPD model than pattern matching. How can a person evaluate an option without comparing it with others? Klein found that the fireground commanders that he studied evaluated a course of action by using mental simulation to imagine how it would play out within the context of the current situation. If it would work, then the commanders could initiate the action. If it almost worked, they could try to adapt it or else consider other actions that were somewhat less typical, continuing until they found an option that felt comfortable. This process exemplifies Herbert Simon's (1957) notion of "satisficing" – looking for the first workable option rather than trying to find the best possible option. Because fires grow exponentially, the faster the commanders could react, the easier their job. Therefore, the RPD model is a blend of intuition and analysis. The pattern matching is the intuitive part, and the mental simulation is the conscious, deliberate, and analytical part.

This blend corresponds to the System 1 (fast and unconscious) / System 2 (slow and deliberate) account of cognition put forward by Kahneman [41] and Epstein [42]. Evertsz et al. [27] refer to these concepts as pre-cognitive and cognitive, and Grossman [43] refers to them as the midbrain and forebrain. Grossman takes the position that the expression, "to be scared out of your mind" is actually a literal expression and represents a mental state that he refers to as Condition Black, where the midbrain takes over. Harland [44] refers to these as Emotional and Cognitive centres, and states that the brain's emotional centres actually receive signals before they reach the cognitive centres of the brain. The emotional centres will process input more quickly and typically trigger emotional responses, such as fear. He notes, however, that following this response by the midbrain, the stimuli are still transported to the cognitive centres where they will be processed more logically.

Klein found that a purely intuitive strategy, relying only on pattern matching, would be too risky because sometimes the pattern matching would generate flawed options. A completely deliberative and analytical strategy, however, would be too slow, as the fires would be out of control by the time the commander finished deliberating [25]. Klein's RPD model is shown in Figure 2-4.

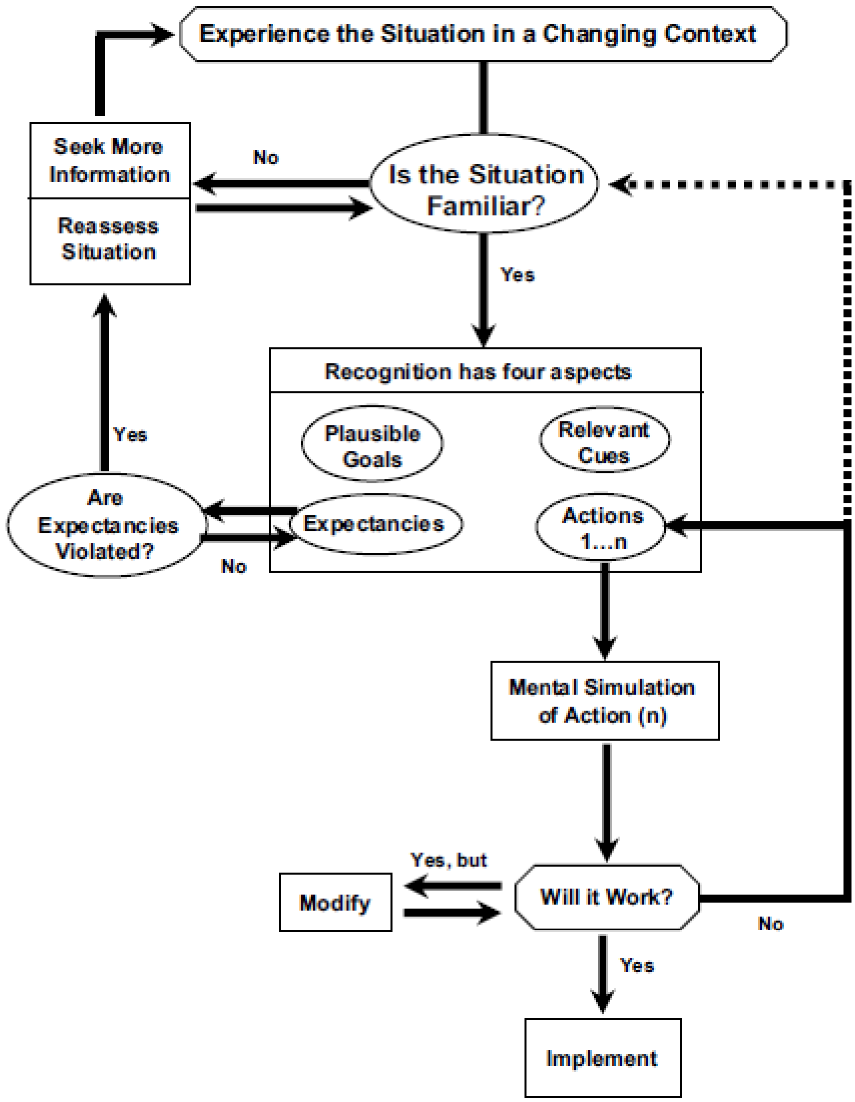

Figure 2-4: Model of recognition-primed decision making [25]

Here, we can see a very close parallel between Klein's NDM model, RPD, and the time-competitive tactical environment of a small-unit leader. In this way, Klein's decision-making model appears to be a natural fit with Endsley's SA model. Where Boyd's model describes the "what", these models describe the "how".

2.4 Act

While the "decision" step in the Boyd loop is the core of leadership, the execution of the chosen "action" is the core of a well-trained force. One of the hallmarks of military training is the emphasis on what could simply be called immediate-action (or battle) drills.

Infantry battle drills describe how platoons and sections (squads in US terminology) apply fire and maneuver to commonly encountered situations. They require leaders to make decisions rapidly and to issue brief oral orders quickly. In this manner, battle drills resemble software design patterns in that they represent general solutions to commonly occurring problems, with the details of the solution being adapted to the specific problem.

The Canadian Army's Tactical Aide Memoire states: "the aim of drills is to permit a Battle Group (BG) and a sub-unit under pressure of time to react quickly in a coordinated manner by executing one or more standard rehearsed manoeuvres. Use of drills reduces greatly the need for detailed orders, and it is the removal of this portion of the decision-action cycle that saves time and frees the commander's mind to think ahead. In short, drills help increase the tempo of operations."

The US Army field manual, FM 25-101, defines a battle drill as "a collective action rapidly executed without applying a deliberate decision-making process."

As the overarching goal of this research is to model the time-competitive decision-making that Boyd referred to, the supporting architecture of this thesis must support the complete decision-making cycle. The primary focus of this research, however, has been centred on decision making. Battle drills, therefore, provide a convenient, pre-made, set of "decisions" that can be made (selected), as well as the steps required to implement the action, thus completing the OODA loop.

2.5 Conclusion

In this chapter we provided an overview of the time-competitive decision-making model proposed by Boyd that forms the basis of our model and architecture. We have decomposed Boyd's model in order to more intuitively map the steps to a software architecture, and then examined each of the decomposed steps (Observe-Orient, Decide, and Act) in more detail. We then examined Endsley's work on Situation Awareness to better understand the complexities involved in SA, followed by Klein's Recognition-Primed Decision model.

While the models of Endsley and Klein are considerably more detailed than Boyd's high-level OODA loop, they are still theoretical. More specifically, they are descriptive rather than prescriptive in nature and thus are still too high-level to be implemented directly in a software model.

Our approach was to take these high-level concepts and break them down further into discrete steps that can be modelled in a software architecture. In the next chapter, we discuss our use of a Hybrid Intelligent System and demonstrate how this approach is well suited to decomposing the models of Endsley and Klein, and how it maps well to the architecture that we will present in Chapter 4.

3 Hybrid Intelligent Systems

In this chapter we discuss AI technologies and develop our argument for our choice of tools, methods and design in support of our thesis. Given the number of AI technologies available, it can be challenging to select which approach(es) will work best for a given problem. As Marvin Minsky [45] writes:

> Many students like to ask, "Is it better to represent knowledge with Neural Nets, Logical Deduction, Semantic Networks, Frames, Scripts, Rule-Based Systems or Natural Language?" My teaching method is to try to get them to ask a different kind of question. "First decide what kinds of reasoning might be best for each different kind of problem – and then find out which combination of representations might work well in each case."
> [...] My opinion is that we can make versatile AI machines only by using several different kinds of representations in the same system! This is because no single method works well for all problems; each is good for certain tasks, but not for others. Also, different kinds of problems need different kinds of reasoning.

This need for a combination of different intelligent technologies, whose individual strengths offset each other's weakness(es), has led to the emergence of hybrid intelligent systems [12] [46]. As we noted in Chapter 1, synthetic environments are comprised of models of physical entities, environmental factors, in addition to human models, and that the human models are not as sophisticated as the other two. This is, in part, due to the fact that human behavior represents highly complex nonlinear and adaptable systems. While conventional approaches using state machines and expert systems have been applied to CGF, the results in some cases have been synthetic force portrayals that are not totally autonomous, are unrealistic and do not support the full requirements that militaries have for analysis and training [12].

3.1 Soft Computing

A new mathematical approach, known as Soft Computing (SC), shows promise in dealing with the inherent complexity of modeling human behavior. SC is a discipline that is comprised of a combination of several distinct mathematical techniques: Fuzzy Logic (FL), Artificial Neural Networks (ANN) and Probabilistic Reasoning (PR), which includes genetic algorithms, chaos theory, belief nets and learning theory.

SC differs from conventional computing in that, unlike hard computing, it is tolerant of imprecision, uncertainty and partial truth. In effect, the role model for soft computing is the human mind [28]. SC is not, however, an uncoordinated combination of FL, ANN and PR; it is a partnership in which each method contributes, usually at different organizational levels, thereby creating a hybrid system. From this perspective, the contributions of FL, NN and PR are complementary rather than competitive [12] [28].

In traditional, or "hard" computing the primary factors are precision, certainty and rigor. Zadeh [28] writes, "... the point of departure in soft computing is the thesis that precision and certainty carry a cost and that computation, reasoning, and decision making should exploit—wherever possible—the tolerance for imprecision and uncertainty." He goes on to say, "... in raising the banner of 'Exploit the tolerance for imprecision and uncertainty,' soft computing uses the human mind as a role model and, at the same time, aims at a formalization of the cognitive processes humans employ so effectively in the performance of daily tasks." In other words, soft computing is capable of operating with uncertain, imprecise and incomplete information in a manner that reflects human thinking. A case in point would the human tendency to represent data with words, rather than numbers.

Zadeh [47] [48] notes that, while words are inherently less precise than numbers, the offset to this is that the cost of precision is high. Words can be used when there is a tolerance for imprecision. Likewise, soft computing exploits the tolerance for uncertainty and imprecision to achieve greater tractability and robustness, and lower the cost of solutions.

Intelligent systems such as CGF must possess humanlike expertise in the military domain. Like a human or a group of humans in a military organization, they must be able to adapt to change in a highly dynamic synthetic environment. This must be done within the constraints of doctrine, tactics, experience and performance of military systems. It therefore seems reasonable that it would be advantageous to use several mathematical techniques together to form a hybrid system that leverages off the advantages of various modeling techniques.

We noted in Chapter 2 that the environment in which a small-unit leader must make decisions in is replete with the factors that soft computing aims to address. Our research will show that soft computing provides both a good solution for the problem space that we are dealing with and a very flexible approach to further decomposing the models of Endsley and Klein into workable software solutions.

3.2 Neuro-Fuzzy Systems

Our approach was the use of a neuro-fuzzy system, which applies a combination of an Artificial Neural Network (ANN) and a Fuzzy Inference System (FIS) [12] [49]. This hybrid technique uses the power of artificial neural networks to classify patterns in data and adapt that classification with highly dynamic environments. ANN's have been employed in several applications ranging from target recognition to financial forecasting. They are particularly powerful in clustering the solution space, thereby identifying important features, an attribute that we will develop further in Chapter 4. Fuzzy logic is based on the idea that sets are not crisp but are fuzzy, and these can be modeled in linguistic human terms such as large, small and medium [50]. In fuzzy systems, rules can be formulated that use these linguistic expressions and apply them to the human behavioral problem. The combination of ANN and fuzzy sets offers a powerful method to model human behavior, as their roles are complementary, rather than competitive [12] [28].

Neuro-fuzzy systems can be broken down into two broad categories: heterogeneous and homogeneous [51]. We chose to use a heterogeneous system, where the FIS and ANN work as complementary components. The FIS accepts perceptions from the simulation engine and generates a high-level comprehension of the situation. The FIS then feeds the results to the Project step in order to project our current comprehension into the near future, and then generate the leader's SA of the current situation.

3.2.1 Fuzzy Logic

Fuzzy logic is a superset of conventional (Boolean) logic that has been extended to handle the concept of partial truth values. As its name suggests, it is the logic underlying modes of reasoning which are approximate rather than exact. FL is based on the concept of fuzzy sets, and provides a means for representing uncertainty [47] [50]. While probability theory is the primary tool for analyzing uncertainty, it assumes that uncertainty is a random process. However, uncertainty is not always random, and fuzzy set theory can be used to model the kind of uncertainty associated with imprecision, and lack of information [50]. This was an important point in our selection of FL in our model, as the small-unit leader will rarely have complete and/or timely information available in order to make decisions.

Fuzzy concepts are represented using fuzzy variables, fuzzy sets and fuzzy values. A fuzzy variable is used to describe a general fuzzy concept. It consists of a name (for example, air temperature), its units (such as Degrees C), a range (for example, from 0 to 100) which is often referred to as the universe of discourse, and a set of fuzzy terms that can be used to describe specific fuzzy concepts for this variable. The fuzzy terms are defined using a term name such as *cold* or *hot*, together with a fuzzy set that identifies the degree of membership of the term over the range of the fuzzy variable. The fuzzy variable terms along with a set of system or user supplied fuzzy modifiers (like *very* or *slightly*), sometimes referred to as hedges, and the operators *and*, *or* and *not* provide the basis for a simple grammar that allows one to write fuzzy linguistic expressions to describe fuzzy concepts in a natural language format, rather than mathematically. The logic of a FIS is encoded in fuzzy rules [52].

More specifically, fuzzy sets can be interpreted as membership functions μx that associate with each element x of the universe of discourse U, a number $\mu_x(x)$ in the interval [0,1] depending on the exclusion or the inclusion of the element in the fuzzy set. Therefore, a fuzzy set is constructed as a set of ordered pair of element x and its degree of membership $\mu_x(x)$. The degree of membership specifies how strongly an element belongs to a set, and the shape of the Membership Functions (MF) are determined by the system designer, based on the kind and quantity of uncertainty present [53].

In general, a FIS consists of three basic processes: fuzzification; fuzzy inference, using a fuzzy rule base and an inference engine; and defuzzification. Fuzzification maps the crisp inputs into fuzzy sets, which are subsequently used as inputs to the inference process. Fuzzification typically involves converting the crisp value to either a singleton or a fuzzy set. In the case of a singleton, the crisp point $x \in U$ is mapped into a fuzzy set X, where $\mu_x(x_i) = 1$ for $x_i = x$ and $\mu_x(x_i) = 0$ for $x_i \neq x$. In the case of a non-singleton fuzzifier, the point $x \in U$ is mapped into a fuzzy set X, where μ_x achieves maximum value at $x_i = x$ and decreases while moving away from $x_i = x$. Non-singleton fuzzification is useful in cases where the input data to the FIS is imprecise, due to factors such as measurement noise [53].

A fuzzy system is represented by a set of linguistic statements based on expert knowledge which is usually in the form of “if/then” rules. The collection of these rules forms a rule base which provides a linguistic approach to modeling systems in a human-like manner. Fuzzy reasoning or a fuzzy algorithm is developed to implement fuzzy implication relations. Four commonly used types of fuzzy reasoning are [53] [54] [55]:

1) Mamdani – This approach is considered the most intuitive to use and thus has widespread acceptance due to its ease in capturing and describing expert knowledge. It is not as computationally efficient as TSK or Tsukamoto, however, due the defuzzification process;

2) TSK (Takagi-Sugeno-Kang, sometimes referred to simply as Sugeno) – TSK replaces the consequent portion of the model with an equation, thereby producing a crisp output which does not require defuzzification. Rather, the output is the weighted average of the consequents. This makes TSK models well suited to complex, high-dimensional problems;
3) Tsukamoto – In this model, the consequent of each fuzzy rule is represented by a fuzzy set in a manner similar to Mamdani. There is a requirement, however, that the consequent fuzzy sets have monotonic MFs. The output of each rule is then defined as a crisp value corresponding to the rule's firing strength, and the overall output is taken as the weighted average of each rule's output; and
4) Larsen – The Larsen model is similar to the Mamdani approach, but uses the max-product method where the consequent of a rule is scaled with the degree of fulfillment of that rule. Aggregation of the output fuzzy sets is the same as for Mamdani.

FL's Achilles heel is the risk of combinatorial rule explosion [56] [57] [58]. A typical system, with n input variables, and S states per variable, will require S^n rules. Therefore, a system with five input variables, each with five states, would require over three thousand rules, which is clearly impractical to implement. Combs and Andrews [56] introduced a technique, referred to as the Combs method, which replaces multi-antecedent rules with an interconnection of single antecedent rules, thereby eliminating the rule explosion that is associated with multi-antecedent rules. The Combs method is based on the propositional logic equivalence shown in Equation 1:

$$[(p \wedge q) \Rightarrow r] \Leftrightarrow [(p \Rightarrow r) \vee (q \Rightarrow r)] \quad (1)$$

This approach, however, has its detractors [57] [58] who raise questions regarding the method's validity. Both note that while this equivalence is easily proven in Boolean logic, it does not strictly hold to be true under the generalized modus ponens used by FL. For Combs' method to be valid in a FL system, the predicates p and q must be independent. Combs acknowledges in his reply to Mendel and Liang [57] that one cannot, based on his theorem alone, convert an intersection-based set of rules directly to a union-based set of rules. Rather, the system needs to be designed from the ground up using, to the greatest extent possible, a union-based rule set. Combs also notes in his reply that, in the case of dependent predicates, the use of multi-antecedent rules is unavoidable. Mendel and Liang [57] go on to state that while many variations of intersection-based FL systems have been shown to be universal approximators, it still has not been proven that the same is true for union-based systems.

Since we require our model to work in real time, we chose to use Combs' approach to reduce the number of rules required in our FIS, thereby greatly decreasing the computational load. For similar reasons, we chose to use the TSK method of fuzzy reasoning for its computational efficiency.

3.2.2 Artificial Neural Networks

Despite their widespread use in academic AI, the use of ANNs in modern game AI is quite rare. The black box nature of ANNs makes them difficult to debug, and due to the still relatively small amount of CPU time that a game's AI gets, the approach is not well suited to online learning. Charles and McGlinchey [49] believe, however, that progress has been limited largely due to a lack of understanding among game developers of the range of methods available. They note that there are many neural algorithms that have a comparatively low computational cost, such as some Hebbian learning methods, topology-preserving maps, radial basis networks and Vector Quantisation (VQ), that would make ANNs suitable for use in computer games.

Learning mechanisms for ANNs may be either offline or online. For offline learning, the ANN is trained during the development process only. With online learning, however, the ANN continues to learn even when the end product is being used. The implementation of online learning is much more difficult because it is a real-time process and many of the commonly used algorithms for learning are not considered suitable [49]. An additional problem is the possibility that the ANN learns the wrong lessons and behaves incorrectly from that point forward. As this occurs in the final product, the developer has no control over this happening [46].

There are three categories of learning for ANNs: supervised, unsupervised and reinforcement learning. With supervised learning—sometimes referred to as learning with a teacher—the network is provided with both input data and the correct answer. The input data is typically propagated forward through the network until activation reaches the output neurons. The answer that the network has calculated is compared with the desired result. If the answers agree, there is no need to make changes to the network. If, however, the answer which the network is giving is different from the desired answer, then the weights are adjusted to ensure that the network is more likely to give the correct answer in the future if it is presented with the same (or similar) input data. This type of learning allows an ANN, once trained, to generalize when presented with new or incomplete data [46] [49] [59].

With unsupervised learning there is no external teacher and learning is generally based only on information that is local to each neuron. This is also often referred to as self-organization, in the sense that the network self-organises in response to the data presented and detects the emergent collective properties within the data. As a result, unsupervised learning is often used in an exploratory manner. Unlike supervised learning, the correct answers are not known before training begins [46] [49] [59].

The third form of neural learning is known as reinforcement learning. This learning relates to maximizing a numerical reward signal through a trial-and-error approach. In order to learn, the network is not told which actions to take but instead must discover which actions yield the most reward by trying them. If an action has been successful, the weights of the ANN are altered to reinforce that behaviour. Otherwise that action is discouraged in the modification of the weights. Reinforcement learning is different from supervised learning in that with supervised methods, learning is from examples provided by some knowledgeable external supervisor. The advantage of reinforcement learning is that with interactive sorts of problems it is quite often unrealistic to expect to be able to provide examples of desired behaviour that are both correct and representative for all scenarios which an agent may encounter [46] [49] [59].

In the next sub-section, we will introduce a form of unsupervised learning, known as Kohonen Self-Organizing Maps, and discuss our rational for its choice as the Projection portion of our model.

3.2.3 Self-Organizing Maps [60]

Our choice for a neural network—one that has a reasonably low computation cost—is Kohonen's topology-preserving map. Kohonen formulated the principle of topographic map formation [60], which states that the spatial location of an output neuron in the topographic map corresponds to a particular feature of the input pattern. Kohonen's approach, which he calls a Self-Organizing Map (SOM), uses competitive unsupervised learning, where neurons compete among themselves to be activated. While in other ANN architectures more than one output neuron can be active at the same time, there is only one "winner-takes-all" neuron in a SOM [46].

Each node in a SOM contains a weight vector of the same dimension as the input vectors. When a new feature vector is presented to the network, every node is examined to determine which node's weight vector most closely corresponds to the input vector. Although there are a number of ways to compare the distance that each node is from the input vector, Euclidean distance is frequently used for its low computation cost. The weight vector of the winning node—often referred to as the Best Matching Unit (BMU)—is then adjusted as shown in Equation 2.

$$W(t+1) = W(t) + L(t)(V(t) - W(t)) \tag{2}$$

In Equation 2, t represents the time-step and L is a small variable called the learning rate. Therefore, the new weight of the node is equal to its current weight, plus a percentage of the difference between the current weight and the input vector. The learning rate decays over time, typically in an exponential fashion, so that the network will converge.

Once the weight of the BMU is adjusted, a secondary search is conducted of all nodes to find which ones are within a pre-defined distance of the BMU, called the BMU's neighbourhood. Nodes within the neighbourhood have their weight vectors adjusted in the same manner as was done for the BMU, but the impact of the weight delta is further reduced based on the distance of the node from the BMU. Equation 3 illustrates the overall equation for calculating the new weight of the node.

$$W(t+1) = W(t) + \Theta(t)L(t)(V(t) - W(t)) \tag{3}$$

The symbol theta in Equation 3 represents the amount of influence a node's distance has on its learning, and has a value of 1 for the BMU. In the same manner as the learning rate, the size of the neighbourhood and the learning distance both decay over time.

The SOM encompasses a number of characteristics which bear similarities to the way in which the human brain is thought to work. It accomplishes this by grouping neurons, via the learning process, which specialize in the identification of certain types of patterns. A SOM also provides a way of representing multidimensional input data in much lower dimensions in the output space, thereby providing a form of data compression. As will be discussed in the next section, the ability to pattern-match, combined with the ability to compress the many SA-related factors down to the primary decision-making criteria, was of great benefit in our model.

3.3 Conclusion

In this chapter, we have motivated the use of a hybrid intelligent system for our architecture by demonstrating how, by combining different AI techniques, we were better able to reproduce how humans make decisions. While a single method could produce some results, they would be partial at best, and not sufficient to meet the requirements of our thesis.

As noted in Sections 2.2 and 2.3, the variability in the types of uncertainty in the environment of small-unit leaders, combined with variations in experience, cognitive (SA) and decision-making abilities, makes for truly unscriptable opponents. This reinforces the requirement for the type of robust and tractable architecture that SC provides.

We then presented a heterogeneous fuzzy-neuro system as the approach that is best suited to the types of problems faced by small-unit leaders in combat. In the next section we will present our architecture, and elaborate on how it addresses the specific needs of Boyd's time-competitive decision-making loop.

4 Approach and Architectural Design and Elaboration

4.1 Introduction

The insight provided by Boyd, Endsley and Klein shows us that any architecture that attempts to implement a competitive decision-making model must account for both cognition and intuition.

As noted in Section 2, the quality of information in NDM environments is usually ambiguous, and past experience and the ability to interpret information are vitally important. Thus the adequacy of classical decision theory to explain decision-making in an NDM environment is unsuitable for our problem.

4.2 Model Characteristics

As we discussed in Section 2, an architecture that is based on Boyd's OODA model can be decomposed into sub-models, based on the work of Endsley and Klein. This new model, which is the subject of this thesis and is henceforth referred to as the Competitive Decision-Making Model (CDMM), incorporates Endsley's model for Situation Awareness, as it provides a good fit to expand Boyd's Observe and Orient steps. Klein's RPD model is an equally good fit for the Decide step, and for the Act component, we confined ourselves to the use of a set of pre-made battle drills. However, as we also discussed in Section 2, the models proposed by Boyd, Endsley and Klein are largely theoretical models. The following discussion, therefore, describes how we elaborated these theoretical models into our own CDMM, which we then used as the foundation for our own software architecture.

4.2.1 Expanded Boyd Model

By integrating Endsley's and Klein's models into Boyd's, the steps of our expanded decision-making model become:

- Sensory cues are sent from the simulation engine to the Perception state in the CDMM.
- Perceive – Level 1 SA Elements (SAE) are generated by the Perception state in response to the cues received from the simulation engine.
- Comprehend – A Fuzzy-Neuro System combines the various SAEs that represent an understanding of the meaning of interrelated cues and forwards the output to the Projection state.
- Project – The past and current outputs of Comprehension are used to project our current understanding into the near future in order to predict how these elements will impact the current situation. These projections are then forwarded to the Decision state.
- Decide – Determine if the current situation is recognizable and, if it is, what course of action best corresponds to it, and then mentally simulate the Course of Action (COA) to ensure that it is, in fact, suitable.
- Act – Select the appropriate set of pre-defined actions.

This expanded Boyd model is shown in Figure 4-1

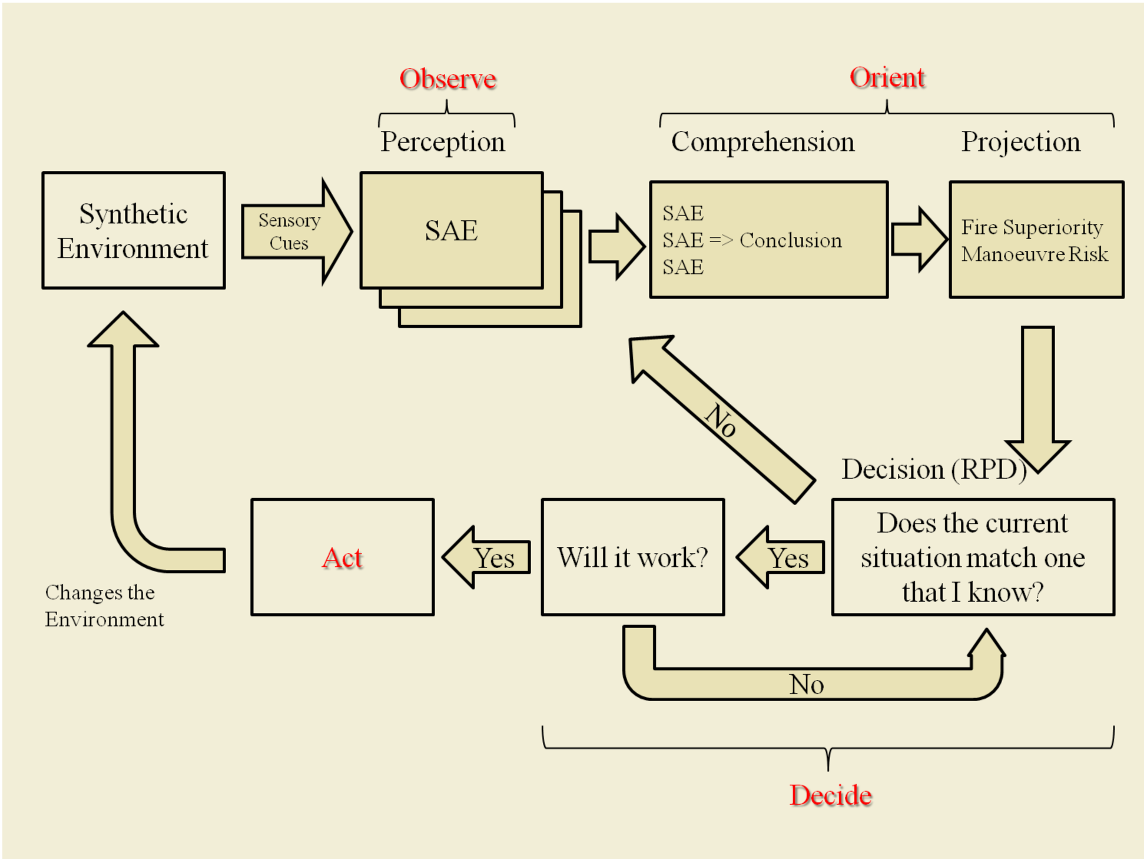

Figure 4-1: Expanded Boyd Model

Lipshitz [61], in his study on NDM-related research noted that all nine of the theories that he reviewed included an element of situation assessment, and that expert decision makers are able to perform situation assessment more quickly and accurately than novices. It is believed that this superiority in situation assessment skills accounts for much of the ability of experts to make rapid decisions and contributes to their decision-making accuracy. As Klein [25] notes, cue recognition/significance gives expert decision-makers an advantage because they can recognize cues more quickly and completely than novices, and they recognize patterns of cues better than novices. They can also detect important features of a stimulus more readily than novices so that they can detect the underlying structure of a problem. Here we can see that the Comprehension and Projection steps in Endsley's model for situation awareness allow us finer grained control in representing the relevance and potential impact of an actor's observations/perceptions. This, in turn, allows for better separation of individuals in their decision-making ability.

4.2.2 The Human Element

Thus far we have been able to map Endsley's and Klein's work to Boyd's model, thereby providing finer granularity in the steps that must be taken in the decision-making process. These two models alone, however, do not allow us to fully represent the intent of Boyd's work. What sets Boyd's work apart from other models is the human dimension—how humans react to change, and the stress that it brings—and how the human brain processes information under these conditions.

Therefore, based purely on the steps of our expanded Boyd model introduced in Section 4.2.1, we can see that our expanded model does not account for the System 1 (fast and unconscious) and System 2 (slow and deliberate) levels of cognition described in Chapter 2.

The concept of pre-cognitive processing complicates our model, as we must also account for behaviour which has the effect of 'short-circuiting' our decision-making process. Militaries around the world repeatedly train for such "act without having to think" responses in order to have their soldiers react instinctively in the face of danger. In fact, battle drills are exactly that. While, according to Harland, the functioning of these separate processing centres in the brain remains largely speculative, it is clear that, regardless of how the mechanic actually works, fear clouds judgement, and at some point, individuals will cease to behave rationally [44].

4.2.3 Accounting for Change

The last factor that needs to be accounted for in our model is the effect(s) caused by change; specifically the effect in the mind of the opposing decision-maker. In Sections 2.1and 2.2 we discussed the importance of acting quickly in the decision-making cycle in order to change the environment that the opposing decision-maker is using to formulate his own decisions. This, in turn, makes his current assessment increasingly less valid. Based on Boyd's work, this circle (loop) will continue until one side possesses a decisive advantage and emerges victorious. What is not accounted for in this description is how we model this "mental lag". Neither are the differences between the abilities of the opposing commanders discussed, but they must be modelled if we wish to depict an engagement between an expert and a novice decision-maker, or between two experts who possess varying degrees of expertise.

We have established that we need to model the effects of change, but not how. Bridges [26], in his Transition Model makes an interesting distinction between 'change' and 'transition'. According to Bridges, change is situational and happens without people transitioning, while transition is psychological and is a three-phase process (Ending, Neutral and Beginning) where people gradually accept the details of the new situation. It is equally insightful to examine the emotions associated with these three phases:

- Ending, triggered by events or stimuli, leads to shock, denial, disorientation and self-doubt;
- The Neutral phase is the unknown area, and typical emotions are Confusion, Lack of Focus, and Fear/Anxiety; and
- The Beginning is acceptance of the new situation, and in Bridges model often leads to feelings of renewed energy and commitment. That said, depending on the new situation, it could just as easily lead to uncertainty and fear—the emotions that Boyd's model attempts to generate—if the recipient of the new situation is not the one that initiated the change.

Bridges goes on to state that change can be "Wanted" or "Unwanted", and "Expected" or "Unexpected", to various degrees. Bridges' model, therefore, provides us with another model element that can be incorporated into our architecture to allow us to model the most critical element of the OODA loop, the speed of the loop, by modelling how change affects the decision-maker's mental state.

4.2.4 Tactical Decision-Making

The U.S. Field Manual 3-21.8 (Infantry Platoon and Squad) [62] describes tactical decision-making as a process of the leader collecting information, employing a decision-making process, and giving an order to subordinates.

The specific “Actions on Contact” are as follows:

- Deploy and Report;
- Evaluate and Develop the Situation;
- Choose a COA; and
- Execute the COA

It should come as no surprise that these steps mirror the OODA loop so closely. As we have mentioned previously, there is nothing particularly novel in the concept of Observe, Orient, Decide and Act in decision-making. Rather, it is the competitive nature of Boyd’s loop that is novel, as the focus is not just on decision-making, but on the speed of decision-making. This requires acquiring situation awareness faster than one’s adversary and acting decisively, thereby creating the effects described by Bridges in his transition model.

The “Choose a COA” step states that, in general, the following options are open to the leader:

- Assault/Attack
- Support by Fire for another unit
- Break Contact
- Defend
- Bypass enemy position

It goes on to state that the listed COAs are relative to the effectiveness of fire and strength of the enemy position, and are listed in order of preference. Each of the above, except for the Bypass option, has one or more battle drills associated with it. Therefore, once the COA is chosen (Decide), the appropriate battle drill can be invoked (Act). As the “Bypass Enemy Position” option does not involve combat, it was not considered in this work.

4.2.5 Simulation Engine

In order to visualize and test our model, we require a synthetic environment in which our decision-makings must operate. While the intent was to make the CDMM as engine-agnostic as possible, there are some things that a simulation engine is optimized to do, such as ray tracing and obstacle avoidance, and we have availed ourselves of these optimizations whenever possible.

Decision-making in a synthetic environment presents a number of problems. As noted by Evertsz et al. [27], synthetic environments provide a discrete representation of environmental change, which is not how humans perceive an essentially continuous world. Similarly, synthetic actors do not ‘see’ in the same manner as humans, and this must also be accounted for in a model that attempts to model human decision-making.

4.3 Basic Model Architecture

Based on the preceding discussion on the required model characteristics, we can formulate a high-level CDMM architecture as shown in Figure 4-2.

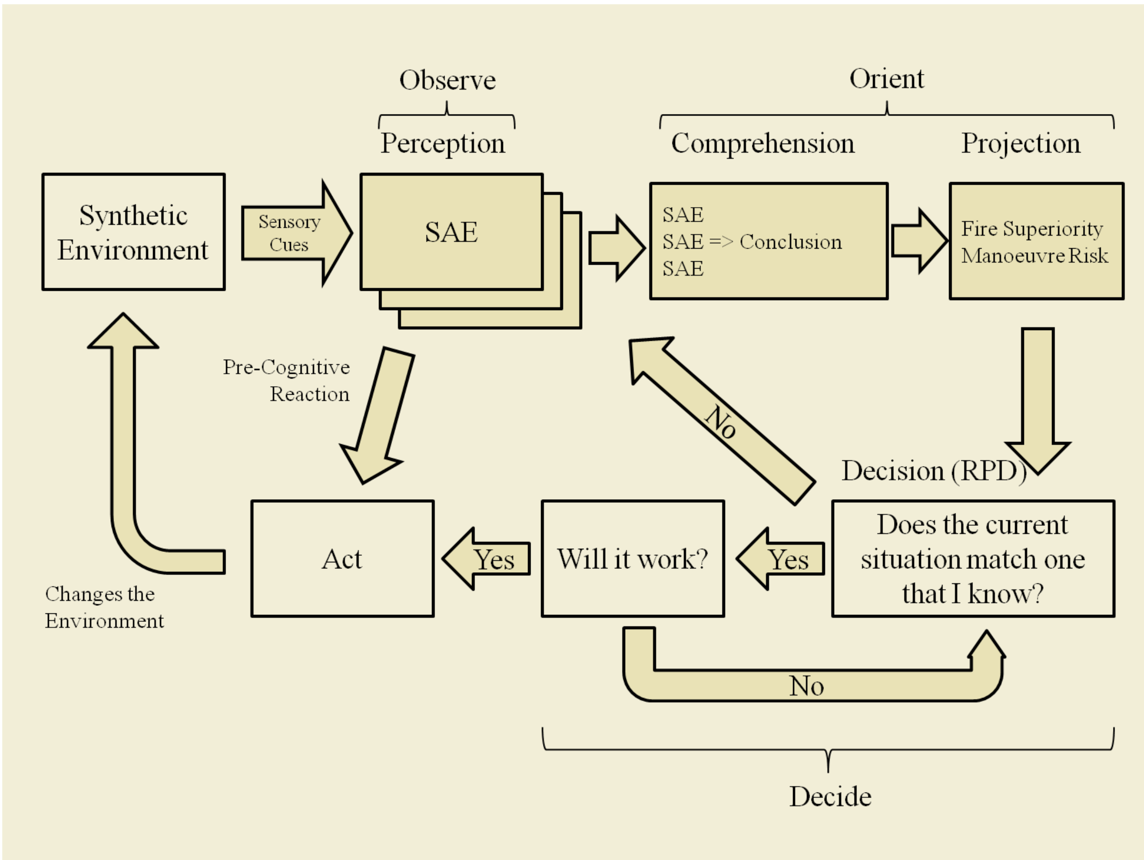

Figure 4-2: High-Level Competitive Decision-Making Model

In Figure 4-2, we see a high-level implementation of both the pre-cognitive and cognitive processes. Table 4-1 provides an example of a pre-cognitive event, where the SAE, upon activating, immediately triggers an Act response. Note that this example does not take into account existing stress, and the rule check has been, for clarity, kept at a high level. However, the firing of this event will generate new stress. The effects of the SAE, regardless of whether or not a pre-cognitive event has been triggered, will carry forward to the Comprehension step (e.g. the effects of being "Under Fire"). The effects of enemy fire will be covered in more detail in our discussion of our combat model, in Section 6.2.

Table 4-1: Cues and L1 SA Elements

Cue	Level 1 SA Element	Act Response

Muzzle flash(s)
Weapon firing sounds
	Under Fire
	=> Trigger Trained response (if available) or
	Emotional response
		if (!InContact && !UnderCover)
		=> ReactToContact
		else
		=> ReturnFire

4.4 Implementation

We have already noted that the environment that a small-unit leader must operate in is challenging and that there will be varying degrees of uncertainty associated with the information that is available. In fact, it is this uncertainty, and the accumulation of stress, that affect the decision-making speed and accuracy of a leader. Our implementation, therefore, must account for these factors. In the next chapter, we will provide a detailed description of the architecture that we have developed to model this challenging environment.

4.5 Conclusion

In this section we demonstrated how Boyd's model can be decomposed and integrated into those of Endsley and Klein in order to define the steps of a more comprehensive decision-making model. To these steps we added what we felt was a critical component of Boyd's model, but is not accounted for in either of the models of Endsley or Klein—the human element—by introducing pre-cognitive events and the effects of change on human decision-makers. We then evolved these steps into the basic outline of our software architecture.

In the next chapter, we will examine the major components our model's architecture and the communications protocol used to bind them.

5 CDMM Architecture

5.1 Overview

The model that we have created is a distributed application composed of two main components. The first is the simulation engine, which will simulate the battlefield conditions where we wish to test our decision-making model. The second is our CDMM where we model the cognitive decision-making behaviour. The two processes communicate via TCP sockets using JSON (JavaScript Object Notation) messages in a remote procedure call (RPC) protocol.

JSON is a lightweight data-interchange format that is easy for humans to read and write, and easy for machines to parse and generate [63]. It is used to transmit data objects consisting of attribute-value pairs.

5.2 Unreal Engine 3

In 2009, Epic Games released the Unreal Development Kit (UDK), free for non-commercial use. The UDK is comprised of the complete Unreal Engine 3, minus the C++ source code. All of the tools available in the commercial version of Unreal Engine 3 are available in the UDK, including all of the UnrealScript source code. UnrealScript is the Unreal Engine's scripting language, and is quite powerful. It is statically/strongly-typed, object-oriented language very similar to Java. The language was designed from the beginning to provide features which, in the view of Epic Games, facilitated the development of games. Therefore, it supports states at the language level, and is heavily event/animation driven.

While the C++ source code is not available in the UDK, its functionality is exposed via the UnrealScript classes. UnrealScript is quite slow compared to C++, and care must be taken so as to not bog down the system with computationally long functions. The time-sensitive operations exposed via UnrealScript are backed by native C++ code. End users, however, cannot create this wrapper functionality. At best, users can create C++ dynamically linked libraries (DLL) and call functions in them from UnrealScript. In our application, the cognitive processing is offloaded from the CPU-intensive simulation engine to the CDMM component. By running the CDMM in a distributed environment, we are free to make our models for cognitive behaviour as complex as required without affecting the frame rate of the simulation.

If communications speed becomes a priority in follow-on use of the CDMM, then the two processes can simply be run on the same computer. This is because, while TCP communications over a network are substantially slower than native function calls, when the two processes are run on the same computer (i.e. the server is using a loopback address like 127.0.0.1), the outgoing packets need only travel down the OSI stack to the Transport Layer, where they are then sent to the receiving process, resulting in extremely fast communications.

Given that the cognitive processing timing requirements are substantially slower than the simulation engine's requirement to maintain a frame rate of at least 30 frames per second to create the illusion of continuous movement, the network TCP speed loss is inconsequential, particularly if the two processes are run on the same computer.

5.3 UE3 Organization

The class diagram shown in Figure 5-1 portrays the basic types of leaders and groups of combatants that one could reasonably expect to find in a small-unit engagement.

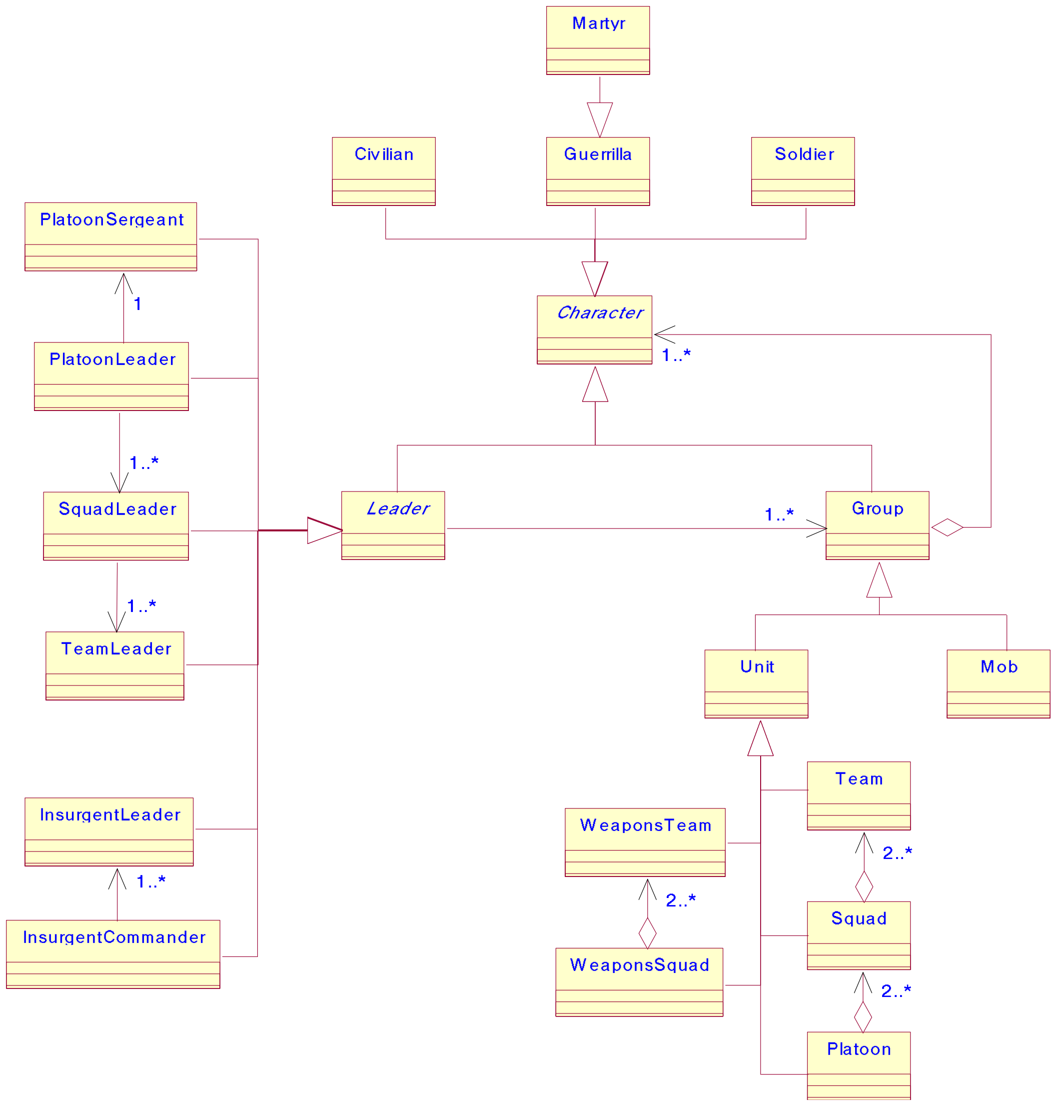

Figure 5-1: Combat Groups Class Diagram

The diagram depicts how groups of combatants are organized, and the various levels of leaders that lead these groups. The classes derived from *Character*, which is an abstract base class, are represented as avatars in the simulation.

The corresponding UE3 *Character* classes are shown in Figure 5-2. You will note that in the following sections that all of our classes are prefixed with “Dm” (Decision Making) to differentiate them from those classes belonging to UE3.

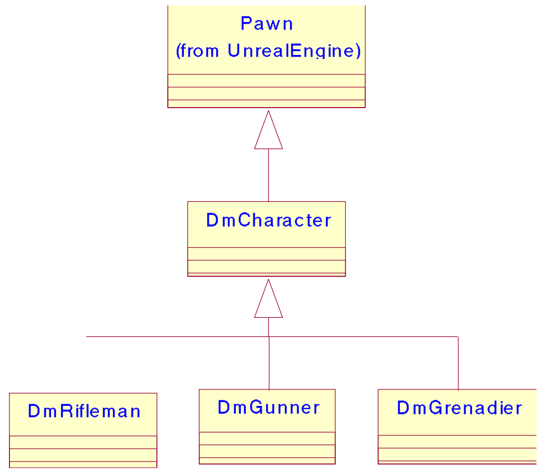

Figure 5-2: Character Classes

We can see from Figure 5-2 that our *DmCharacter* class is derived from UE3's Pawn class, which is a generalized class for avatars that is controlled by some form of artificial intelligence. The Pawn class, itself an abstract base class, contains functionality for dealing with 3D models (Mesh), animation, collision, damage, physics, weapons and sound.

UE3 provides the *Controller* base class specifically to control pawns. In this way, the pawn is analogous to the physical form of the avatar, and the controller represents its brain. The *Controller* class has two derived classes, PlayerController and AIController, where PlayerController handles input from a player and translates those commands into action in the controlled pawn. The AIController is the base class for AI controlled pawns, and it is from this class that our decision-making controllers are derived. These classes are shown in Figure 5-3.

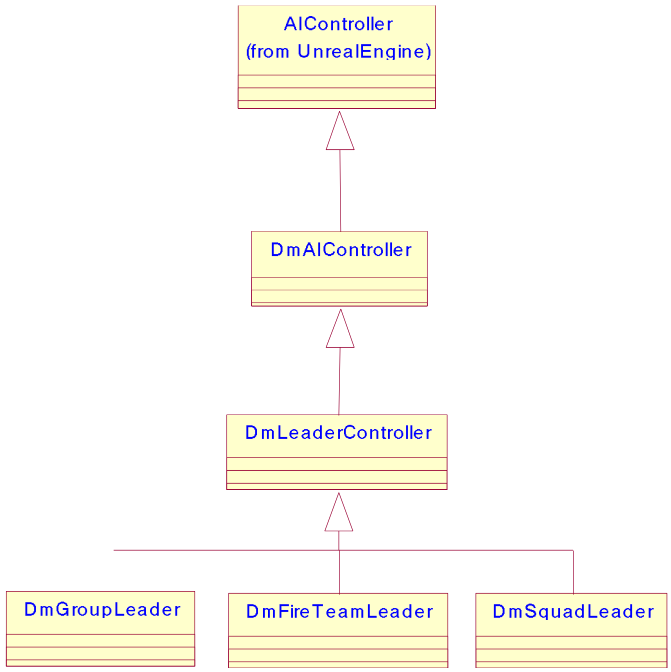

Figure 5-3: Controller Classes

From Figure 5-3 we can see that our base controller class for all avatars is DmAIController, and from it the leader controllers are derived. DmAIController contains all of the physical state-based functionality that handles DmCharacter actions (e.g. moving, crouching, shooting). DmLeaderController, and its derived classes have a counterpart leader class in the CDMM, and it acts as a bridge class between the DmAIControllers and the cognitive states in the CDMM. As such, it contains the logic necessary to establish a TCP socket connection to its counterpart in the CDMM, and to pass information, including start-up administrative information and in-simulation observations. Administrative information includes the leader and group attributes, and scenario-related a priori information, which is set by the scenario designer in the UDK editor,

In Figure 5-4 we see the base class DmGroup and its derived classes. Within our model, the Group classes are responsible for spawning (creating in simulation) the appropriate DmCharacters and providing references to these individual characters back to the group's controlling leader. Groups have no counterpart in the CDMM.

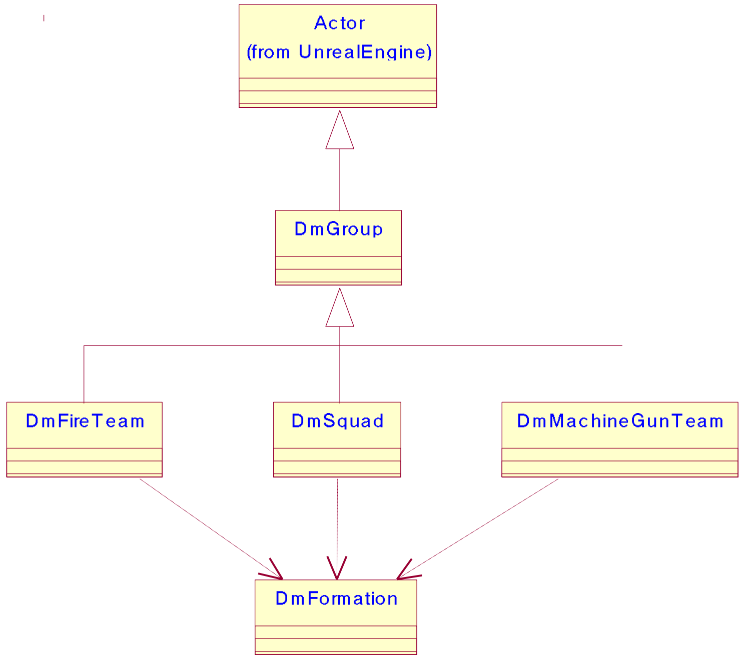

Figure 5-4: Groups Class Diagram

5.4 CDMM

The cognitive side of the simulation, the CDMM, is written in the C# programming language and is divided into modules (a single class, or a group of classes) that correspond to our expanded Boyd model, discussed in section 4.2.1. In this manner, the existing cognitive models can be replaced with more detailed ones if one wished to test a particular behaviour.

5.5 CDMM Organization

Figure 5-5 provides us with a basic overview of the main classes on both sides of the simulation, and how they interconnect.

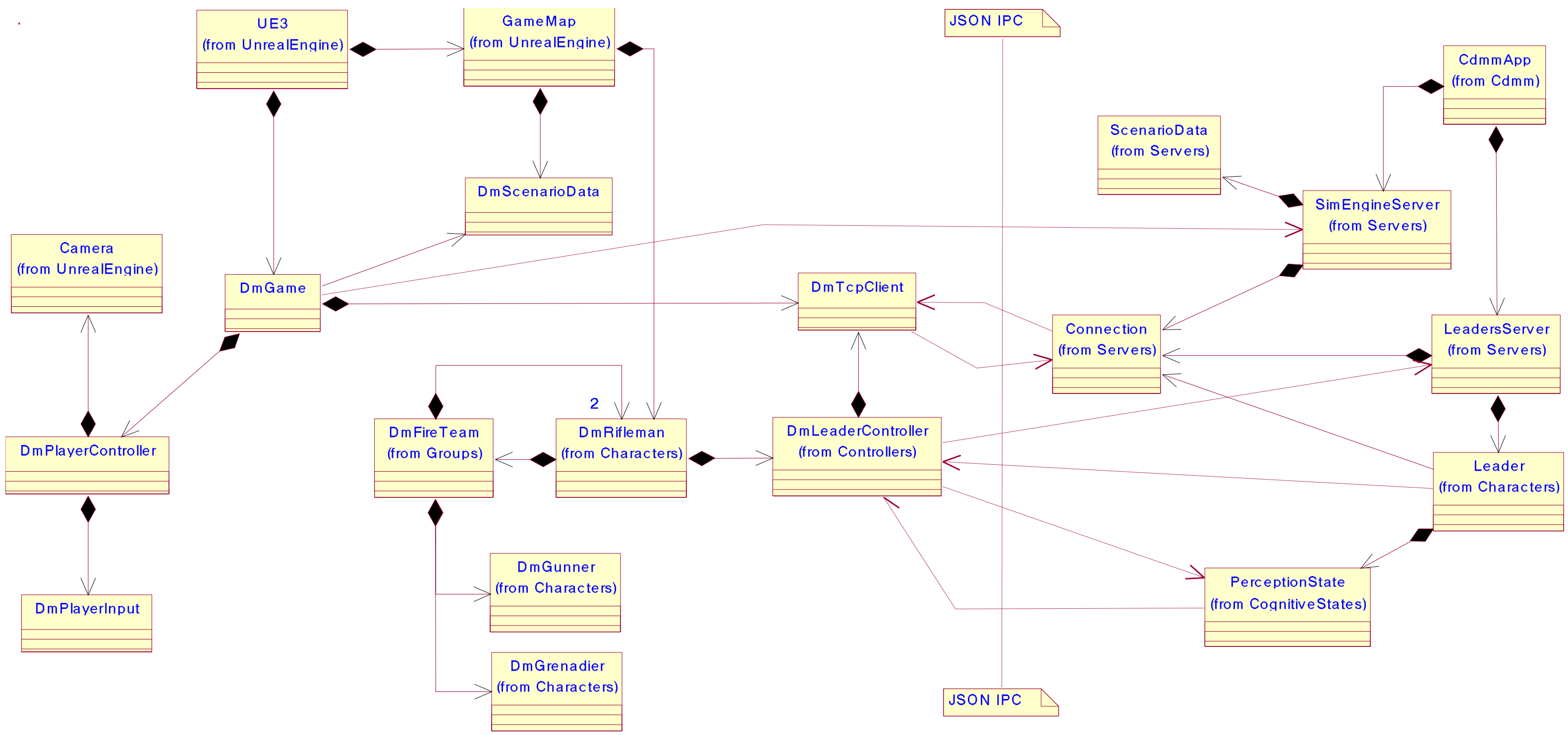

Figure 5-5: General Class Overview

Starting on the UE3 side, we see that the UE3 engine (abstracted here as a single class) creates the DmGame class that provides rules for how to initialize the simulation, including creating the DmPlayerController which, in turn, creates a game camera and a DmPlayerInput class. These last three classes permit us to view our simulation and to move the camera to whichever vantage point we desire.

The UE3 engine also creates the game map, which then creates all of the entities in the map at game start. In Figure 5-5 we can see that this entails a ScenarioData object, as well as a DmRifleman. The DmRifleman object creates the appropriate AIController. The DmFireTeam creates each of the soldiers that make up the team and relays this information to the unit's leader, the initial DmRifleman.

Once created, the DmLeaderController instantiates a DmTcpClient and connects to the LeadersServer in the CDMM. The LeadersServer instantiates the appropriate leader and passes to it a reference to the TCP socket connection. Once instantiated, the Leader will query its DmLeaderController counterpart for information on leader and group attributes, all of which were set by the scenario designer in the UDK editor. After this exchange of administrative information, the CDMM Leader creates its Perception state and passes responsibility to it for listening for messages from its counterpart DmLeaderController in the simulation engine.

In a similar manner, once DmGame is instantiated, it creates a DmTcpClient and connects to the SimEngineServer and passes game-relevant information (e.g. map scale, time of day, weather conditions etc.).

5.6 Communications Protocol

As mentioned previously, TCP sockets are used for communications between the simulation engine and the CDMM, and the communication protocol chosen was JSON. To implement a remote procedure call (RPC) mechanism, message maps were created in both the simulation engine and the CDMM. These mapped the JSON string for the method to be called with a delegate for the actual method. Every method that was subject to RPC received the JSON object as a parameter. In this way, if parameters needed to be passed between the two processes, a secondary JSON object could be added that contained the parameters.

5.7 Conclusion

In this chapter we discussed how our decision to create a distributed application, where the simulation engine and the CDMM operate as separate processes, allows us to offload the cognitive processing from the CPU-intensive simulation engine to the CDMM component. By running the CDMM in a distributed environment we are emulating the Façade design pattern in that each of the components communicates with an interface, without any understanding of the underlying code. In addition, by maintaining a separation of concerns, we are free to make our models for cognitive behaviour as complex as required without affecting the frame rate of the simulation.

We have provided high-level descriptions and class diagrams of Unreal Engine 3, our simulation engine, and the CDMM, and the communication protocol used to connect the two components. In the next chapter, we will decompose the CDMM into the individual behavioural components that make up the decision-making cycle.

6 Behaviour

In this chapter, we will discuss the specifics of the Leader class, and the ancillary classes that make up the decision-making model. In Figure 6-1 we can see all of the decision-making classes, each of which is annotated with the area of responsibility (e.g. SAEs, Reservoirs, and Cognitive States). Note that Leader is subclassed with ForceCommander, of which, there is only one per side in our model.

The ForceCommander represents the highest ranking commander on each side that is in the simulation (i.e. the ForceCommander's superior is not represented in the simulation). From Figure 6-1 we can see that, while the majority of the ancillary classes are represented in the Leader, the Comprehension, Projection and Decision states are only represented in the ForceCommander. This is due to the fact that, in our model, we have decided to limit the cognitive processes to only one level of leader and they reside with the ForceCommander in order to limit the model's initial complexity.

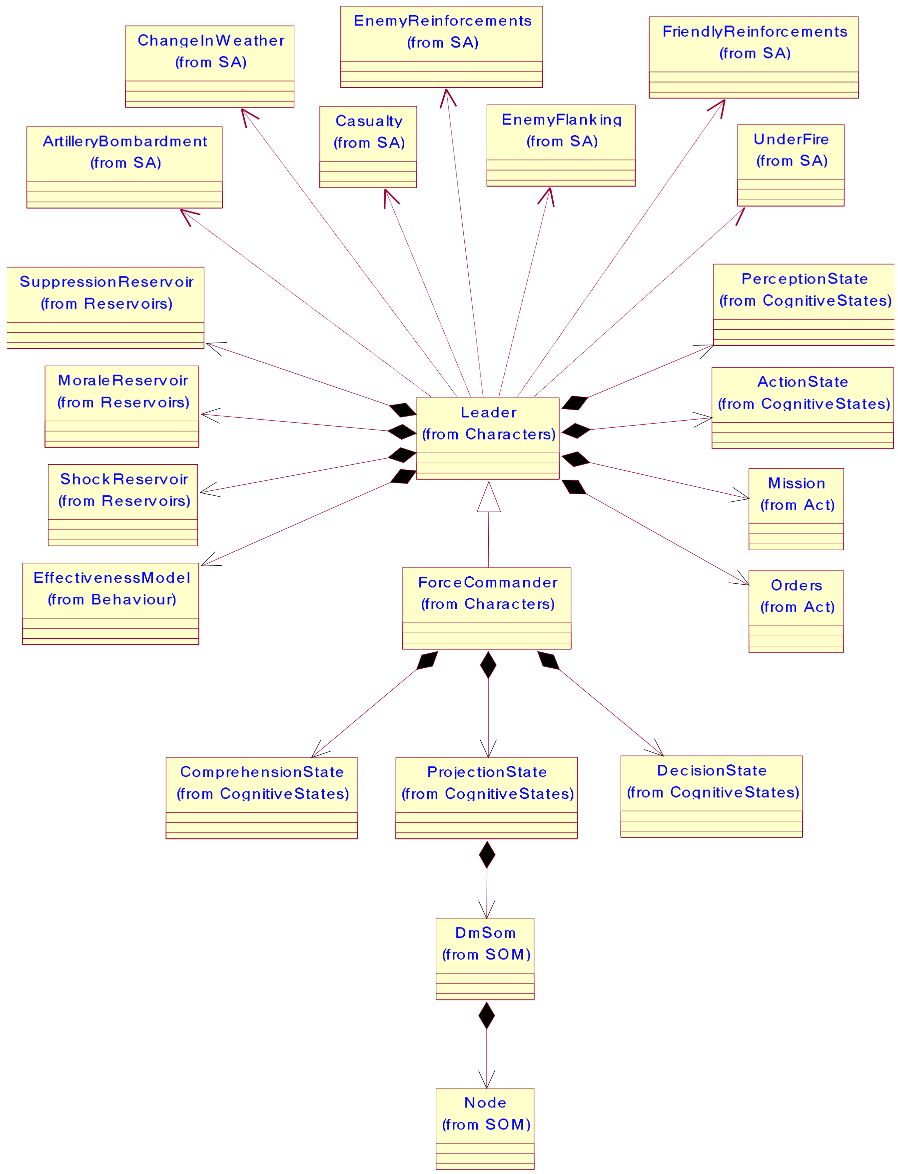

Figure 6-1: CDMM Leader Class

6.1 Troop Quality and Morale

While not universal, references to troop quality [30] [64] frequently fall into one of four abstract categories: Untrained (or Green); Trained (or Regular); Veteran ; or Elite. Similarly, references to unit morale tend to be either: Low; Average; Good; or High.

Given its subjective nature, there is really no way to quantify these categories. While there are marksmanship standards for various armies, these are for qualification on a firing range and not in actual combat. It was assumed that accuracy rates in combat would be substantially lower, so we selected an arbitrary range of skill levels, with a lower bound of 50% and an upper bound of 75%. We also wanted the difference between each level to be more pronounced at the lower levels (i.e. the difference between Level 1 and Level 2 would be greater than the difference between Levels 2 and 3). Given that it is extremely difficult to quantify troop quality, we believe that the subjective range we have chosen represents a reasonable cross-section of the range of troop qualities one might see on the battlefield. In our work, there are no super-soldiers. Therefore, we found that Equation 4 gave a reasonable approximation of the effect that we desired:

$$Skill(or\ Morale) = 50 + 18\ln(Level) \tag{4}$$

Using Equation 4 provides us with approximate skill/morale values of: 0.50, 0.625, 0.7 and 0.75. We felt that this range gave us reasonable starting values that could be increased or decreased by certain stimuli within the simulation (e.g. initial fatigue, SAEs, and/or enemy action).

Table 6-1 provides a representative breakdown of common troop types as defined by our 4-level system. The values range from poorly trained and motivated troops like the Afghan National Army (ANA) in its early days, to regular military units and various levels of Anti-Coalition Milita (ACM).

Table 6-1: Troop Quality and Morale Ratings

	Skill				**Morale**			
	Untrained (1) [0.50]	**Trained (2) [0.625]**	**Veteran (3) [0.70]**	**Elite (4) [0.75]**	**Low (1) [0.50]**	**Average (2) [0.625]**	**Good (3) [0.70]**	**High (4) [0.75]**
ANA	X				X			
Non-Cbt	X					X		
Regulars		X				X		
Veterans			X				X	
SOF			X					X
Tier 1				X				X
Local ACM	X					X		
Hardcore ACM	X						X	
Foreign Fighters		X						X

6.2 Combat Model

A major consideration when attempting to simulate the effects of small unit engagements is the combat model that will be used. Looking at U.S. Field Manual 3-21.8, Infantry Platoon and Squad [62], we see that, from the leader's perspective, in order to conduct an attack, he must answer the question: "Can the unit in contact suppress the enemy?" The following question: "Can the follow-on unit(s) manoeuvre?" is, unless cover is exceptionally good and the enemy's position exceptionally bad, directly impacted by the degree of enemy suppression.

From U.S. Field Manual 23-14, Marksmanship Training, Section 5-4, Training for Combat Conditions, we note the following challenges of infantry combat:

- Most engagements will be within 300 yards;
- Enemy personnel are seldom visible except when assaulting;
- Most combat fire must be directed at an area where the enemy has been detected or where he is suspected of being but cannot be seen. Area targets consist of objects or outlines of men irregularly spaced along covered and concealed areas (ground folds, hedges, borders of woods);
- Most combat targets can be detected by smoke, flash, dust, noise, or movement, but they are visible only for a moment;
- Most combat targets have a low contrast outline and are obscured; and
- Time-stressed fire in combat can be divided into three types:
    1) A single, fleeting target that must be engaged quickly,
    2) Distributed targets that must be engaged within the time they remain available, and
    3) A surprise target that must be engaged at once with instinctive, accurate fire.

We can see from these extracts from the U.S. Field Manuals that combat is imprecise and that suppression is the primary goal based on the simple expedient that suppressive fire against a fleeting enemy is typically the most that can be hoped for. Suppression, therefore, against an area target must be the primary goal of our combat model.

Kushnick and Duffy [65] conducted an extensive study on Vietnam veterans in order to quantify the effects of small arms fire on suppression. Their paper: "The Identification of Objective Relationships Between Small Arms Fire Characteristics And Effectiveness Of Suppressive Fire" notes that "An analysis of interviews with nearly 200 combat veterans led to the conclusion that combat soldiers perceive a personal danger radius outside of which a passing round is heard but is not perceived as dangerous or as producing suppression. These interviews indicate that the length of the danger radius varies with the individual soldier, the weapon employed against him, the volume of fire, and the general situation at the time he receives incoming fire." They further note that: "...it is concluded that perceived dangerousness predominantly increases in a linear fashion with increases in perceived loudness of projectiles." and that "...perceived dangerousness increases linearly, with increases in volume of fire."

This result is interpreted as demonstrating that the perceived dangerousness of incoming fire decreases in a linear fashion with increases in the lateral miss distances of passing projectiles, and increases linearly with increasing volume of enemy fire.

It was the opinion of both the subjects and the DSL analysts who conducted the study that the basic stimulus that allowed the subjects to perceive and note the dangerousness of the events in the field experiment was produced by the projectile signatures and not by the characteristics of the muzzle blasts of the weapons themselves. Therefore, they attempted to relate projectile characteristics to the perceived dangerousness of each "weapon."

The obvious overt characteristic producing the perception of danger is the loudness of the signature of passing projectiles. They noted, however, that the sensation of loudness is a complex function, relating to both the physical parameters of the stimulus and the physiological apparatus of the ear. It was, therefore, considered too complex a function to derive on the basis of the data obtained in this study. However, they go on to note that the loudness phenomenon is in part based on kinetic energy. As a result, it was determined that the kinetic energy (K.E. = 0.5 MV^2) of each projectile would suffice as a first approximation to predicting perceived dangerousness of a projectile.

Calculation of kinetic energy for any given projectile is a multistep process:

$$ARC = BC \text{ x } MC \text{ x } MV^{0.045} \tag{5}$$

$$BV = MV \text{ x } (1 - 3 \text{ x } RCR \text{ x } range / ARC)^{(1 / RCR)} \tag{6}$$

$$BKE = BW \text{ x } BV^2 / 450380 \tag{7}$$

where:

ARC = Adjusted Retardation Coefficient
BC = Ballistic Constant
MC = Mayewski Constant
MV = Muzzle Velocity
BV = Bullet Velocity (at a given range)
RCR = Retardation Coefficient Rate
BKE = Bullet Kinetic Energy
BW = Bullet Weight (in grains)

Table 6-2 provides a breakdown, by weapon, of bullet kinetic energy for ranges of 0, 150 and 300 yards. From this table, we can see that even at 300 yards, bullet kinetic energy is still quite a large number. We need, therefore, a method to scale this number to a reasonable level.

Table 6-2: Weapon Ballistic Data

Weapon	Range (Yards)	Velocity (ft/sec)	Energy (ft-lb)	Scaled Energy
M4	0	2900	1158	0.290
	150	2450	826	0.207
	300	2052	580	0.145
M249	0	3200	1410	0.353
	150	2723	1021	0.255
	300	2300	728	0.182
M240	0	2750	2468	0.617
	150	2404	1887	0.472
	300	2091	1427	0.357
AK47	0	2350	1520	0.380
	150	1939	1035	0.259
	300	1582	689	0.172
PKM	0	2707	2408	0.602
	150	2365	1838	0.460
	300	2056	1388	0.347

If we look at values from the military simulation, America's Army: Proving Grounds (AAPG), we see that the developers consider the amount of suppression per shot for the M4 assault rifle to be 0.10, while for the M249 automatic rifle, the value is 0.35 for a 3-round burst. The sustained rate of fire for the M4 is 15 rounds per minute, and for the M249 the value is 50 rounds per minute, which is 3.3 times larger (i.e. close to the 0.35 value for the M249 given in AAPG). Given that the M249 has a slightly higher muzzle velocity than the M4, this ratio is reasonable.

Looking at the value of kinetic energy for the M249 in Table 6-2 (1410 ft-lb), we can see that a 3-round burst would deliver a cumulative kinetic value of 4230 ft-lb. Dividing this value by the kinetic energy of an M4 round (1158 ft-lb) gives us a value of 3.65. Given the closeness of the two ratios (3.5 in AAPG, 3.65 when considering kinetic energy values only), we felt that it was reasonable to consider bullet kinetic energy as being synonymous with the bullet's suppression value. We further considered the scale of the AAPG values to be a better fit for PMF reservoirs that will have ranges of 0–1 or, at most, 0–2. Therefore, we have divided the value from Equation 7 again by a factor of 4000. This modified Equation 7 gives us the values found in the "Scaled Energy" column of Table 6-2.

The next question that we were faced with was how many rounds, and at what miss-distance, does it take to suppress a soldier. Murray [66] notes that a common rule of thumb states that a soldier will become suppressed if a bullet passes within one metre of him every second, and stay suppressed if one bullet passes within one metre every three seconds. He goes on to point out that this rule of thumb does not fit with what has been seen in real small-arms firefights. Second World War field studies, he notes, suggest that one round passing within three metres every six seconds would appreciably degrade return fire from a whole fire-team, and two rounds every three seconds would prevent any return fire at all.

When one considers that Kushnick and Duffy [65] found that the effect of miss distance also varies by troop quality (up to 15 metres for new recruits), we are faced with a problem that could be the subject of its own study. We have already determined that a unit's fire is typically at an area, as individual targets are rarely seen. We found that multiplying the suppression value (scaled energy at a given range) by the unit's current effectiveness, and then further modifying this value by factors such as the target unit's troop quality and type of cover, we obtained a workable approximation of combat suppression. We will discuss the effects of our combat model on suppression further when we discuss the Suppression Moderator Function in Section 6.4.1.1.

Casualties are possible in our model, based on the range and effectiveness of the firing unit. In the environment that we are modelling, however, casualties are rare and can be turned off completely in order to demonstrate fire and manoeuvre scenarios.

We found in testing that our combat model provides realistic-looking results. As with other features of our model, the individual factors can be manipulated if one desires a different effect. Marcus Luttrell, in his book "Service – A Navy SEAL at War" [67], mentions during his description of an Al Qaeda ambush of his unit in the Iraqi city of Ramadi, "At least there was no belt-fed stuff, thank goodness." The weapon he was referring to was the Russian PKM General Purpose Machine Gun (GPMG), and we can see from Table 6-2 that the bullet energy of the two GPMGs (the Russian PKM and the U.S. M240) is more than double that of an M4, the U.S. Army's standard assault rifle, and slightly less than double the bullet energy of the U.S. M249 Automatic Rifle, which is also belt-fed, but fires the same 5.56 mm round as the M4. According to Kushnick and Duffy [65], however, the suppressive value of a bullet is only half of the equation. The other half is the volume of fire, which is based on the Weapon's Rate of Fire (ROF). Note that the sustained and rapid rates of fire are in rounds per minute, and these values are not the same as the weapon's cyclic rate of fire. Sustained rate of fire can be maintained for long periods of time, while Rapid rate of fire is prone to weapon overheating and must be used judiciously.

Table 6-3: Weapon Suppressive Effects

Weapon	Sustained ROF (Rnds/Min)	Sustained ROF Energy (Energy/Min)	Rapid ROF (Rnds/Min)	Rapid ROF Energy (Energy/Min)
M4	15	4.35	40	11.5
M249	100	35.3	200	70.6
M240	100	61.7	200	123.4
AK47	20	7.6	40	15.2
PKM	125	75.25	250	150.5

We can see from the results in Table 6-3 that for Russian-built weapons (the ubiquitous AK-47 and the PKM GPMG), the PKM is ten times more effective at suppressing enemy troops than the AK-47. For the U.S.-built weapons, M249 Automatic Rifle is approximately nine times more effective than the M4, and the M240 GPMG is close to 15 times more effective. The high rate of fire of the belt-fed weapons, combined with their increased weight to absorb recoil, makes these weapons very dangerous. We have already noted that enemy combatants, if seen, are fleeting. The time, therefore, to shoot at an enemy soldier is short, and muzzle rise, caused by recoil, quickly throws off initial aim. Having a weapon that allows a soldier to fire the highest number of bullets possible in the shortest time increases the likelihood of a hit before recoil overcomes the inertia of the weapon and pushes its aiming point upwards.

The values in Table 6-3 are theoretical maxima in our model. The values used in the simulation are multiplied by unit skill level and current effectiveness. This last point will be discussed further in Section 6.4.1.1.

In testing, the addition of the PKM to a competent unit completely changed the results of an encounter, due to its high suppressive and casualty-causing effects. To some degree, this validates the findings of Kushnick and Duffy [65] regarding the strong correlation between bullet energy, rate of fire and suppressive effect.

The final point regarding the combat model concerns fatigue. While the majority of the a priori information passed from the simulation engine to the CDMM is handled in the Comprehension state (e.g. soldier load, ground and weather conditions), fatigue is incorporated directly into the unit's characteristics immediately on start-up. A unit's initial morale and suppression reduction rate are both negatively impacted by fatigue, which is a value set by the scenario designer.

6.3 Sensory Cues

As discussed in Section 4.2.5, synthetic environments provide a discrete representation of environmental change, which is not how humans perceive an essentially continuous world. Similarly, synthetic actors do not 'see' in the same manner as humans. Sensory cues are transmitted from a DmLeaderController to its counterpart Leader either as they occur (e.g. coming under fire) or at a set periodicity. Through experimentation, we found that a value of 4 Hz was fast enough to capture environmental changes, but not so fast that it overloaded the Perceive state with excessive amounts of redundant data. Some decision-making elements, such as weather and ground condition are applicable to all avatars in the scenario, so this information is transmitted at scenario start from DmGame to the SimEngineServer in the CDMM. Other information, such as troop quality, morale, soldier equipment load and soldier fatigue, are requested by each Leader upon instantiation for itself and for all subordinates from its corresponding DmLeaderController. Once each Leader possesses this information, it transfers responsibility for the reception and routing of incoming cues to the Leader's Perception State.

6.4 Perception State

Within the CDMM, each Leader's Perception state possesses an event map that contains all of the possible JSON messages that can be received from the simulation engine. The event map parses the incoming messages according to the Leader's current plan and provides cue recognition in order to create/trigger SAEs, which represent situations or events in the environment. Some events like "Under Fire" will immediately trigger an SAE, whereas the message "In Position" is context dependant. If the unit is patrolling, the message, sent from the lead unit, simply means that the patrol has reached its destination. If the force is "In Contact" then the message would mean that the sending unit has reached its target location, either through a direct order, or via a precognitive reaction. In this way, the simulation is event driven. The individual SAEs will be discussed in more detail in Section 6.4.2.

6.4.1 Performance Moderator Functions

There is a considerable body of research in the field of human physiology and the effects of stress, and a number of software tools that attempt to model their effects on human behaviour and decision-making. Among these tools, CoJACK [16], PMFserv [68], and ACT-R Phi [14] present themselves as applicable for use in military simulation. These tools use cognitive moderators, referred to as Performance Moderator Functions (PMF) in CoJACK and PMFserv, to model the accumulation or depletion of various physiological and stress factors. Silverman et al. [68] cite the work of Janis and Mann [69], as the inspiration for their work, describing it as, "... what is probably the most widely cited methodology of decision strategies for coping under stress, time pressure and risk." Despite the fact that Janis and Mann's work has extensive empirical backing, and that their work has been validated by other researchers, we find that there are no actual implementations of their model. The authors of CoJACK acknowledge that their work has, in turn, been influenced by that of Silverman et al. Both of these tools are closed-source, and their published results to date show that their work is still very much in the early stages, and that the tools appear to be used primarily for research. Cassenti [17], in his report on the utility of PMFserv for military applications found that, while the tool might serve to provide insights into Army problems, it would be of limited use.

We have chosen to leverage the modelling concept of PMFs in order to moderate cognitive and combat efficiency. While our model provides a framework for exploring the concepts of physiology and cognitive psychology, our primary goal is measuring the real-time effects of these concepts on the speed of the decision-making loop. As a result of the modular fashion of our framework, more detailed cognitive models could be added in future work.

In his book, "Into the Fire", Dakota Meyer [64] notes, "Fear slows down your logic circuits, gives you tunnel vision, and triples your heart rate". Grossman [43] is even more specific, stating that heart rate increase in response to fear is correlated with a deterioration of motor skills and senses like vision and hearing. Eventually, cognitive abilities degrade to a point that he calls condition black, based on work by Cooper's "Principles of Personal Defense", where the individual is completely overwhelmed and is incapable of rational thought.

While the concept of stress provides an abstract concept for moderating the physical and cognitive abilities of and individual, it was deemed to be too high-level for the purpose of our model. Therefore, throughout the following discussion on PMFs, the reader will note that stress is represented across a number of PMFs, including Suppression, Morale and Shock.

We chose to use the effects of stress-related factors to moderate both cognitive and combat efficiency, albeit at different rates. The effects of stress can be accumulated through a number of both combat factors (under fire, friendly casualties) and environmental factors (fatigue, ground conditions, equipment load).

Figure 6-2 is a screenshot of a scenario in progress in the CDMM display. To the left (Blue) and right (Red) of the centre "Projection" display, we can see graphical depictions of four PMFs: Effectiveness; Suppression; Morale; and Shock. Each grouping of PMFs (Blue and Red) represents the overall state of each side, as seen through the eyes of that side's Force Commander.

The Effectiveness PMF has a range of 0–1 and all forces start a scenario with an effectiveness of one. As we will see in Section 6.4.1.2, the rate of decay of this PMF is a function of morale (discussed in Section6.4.1.3), which is variable at scenario start. Of note in Figure 6-2 is the difference in colour between Blue's Effectiveness PMF (yellow), and Red's (green). The colour of the effectiveness bar changes as a visual cue when effectiveness enters certain regions. Effectiveness values between 0.5–1.0 are green, and represent troops engaged in a firefight. Values between 0.2–0.5 are yellow, and represent troops that are sufficiently suppressed so as to have a major impact on their performance. Values less than 0.2 are red, and represent troops who are pinned down and are no longer combat-effective. Each of the PMFs will be explained in more detail in the following sub-sections.

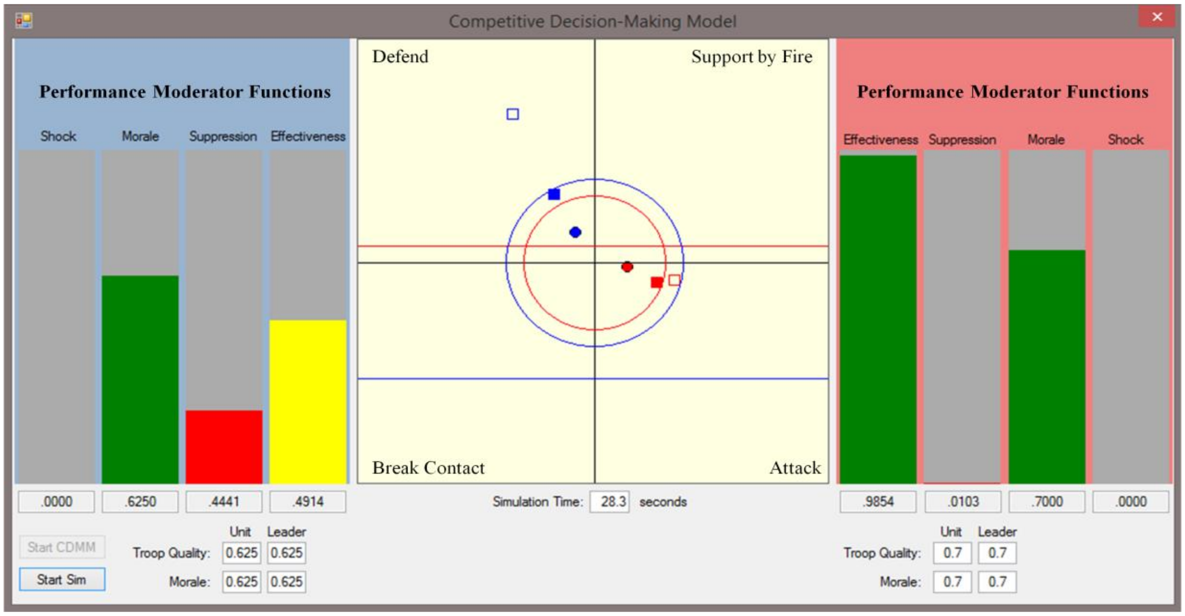

Figure 6-2: Performance Moderator Functions

6.4.1.1 Suppression Moderator Function

Our previous discussions have shown that suppression of enemy troops is a key consideration in infantry combat, particularly since it is difficult to spot, let alone kill an enemy soldier. NATO's publication on terminology, AAP-6 [70], defines suppressive fire as "Fire used to protect troops when they are within range of enemy small arms." Suppressive fire achieves its effect by threatening casualties to individuals who expose themselves to it. Willingness to expose themselves varies depending on the morale and leadership of the target troops, concepts that we will elaborate on further in later sections. The term suppressive fire is frequently used interchangeably with the term "Covering Fire" and is also referred to as "winning the firefight." Smoke used to blind enemy observation is a form of non-lethal suppression, and is also factored into our simulation.

AAP-6 [70] stipulates that suppressive fire requires sufficient intensity over the target area, with intensity being the suppressive effect per unit of target area per unit of suppression time. The Fleet Marine Force Manual MCWP 3-11.2 – Marine Rifle Squad [71] describes suppressive fire as a tactic to reduce casualties to friendly forces and enable them to conduct their immediate mission. A suppressed target will be unable to engage vulnerable forces that are moving without cover. This enables forces to advance to new positions or to close with the enemy.

The primary intended effect of suppressive fire is psychological. Rather than directly trying to kill enemy soldiers, it makes the enemy soldiers feel unable to safely perform any actions other than seeking cover. When a unit is completely suppressed, it is considered "Pinned Down." A unit in this state has lost its ability to move, lost all or most of its ability to return fire, and lost much of its ability to gather real-time intelligence on the enemy position, as it is concerned with staying under cover. Often, a suppressed unit will lose organizational efficiency and morale if kept in that state for an extended period of time.

Enemy suppression, therefore, is key not only in protecting friendly troops from enemy fire, but also for denying the enemy the real-time intelligence that he needs to drive his OODA loop forward. In essence, it slows the speed of the enemy's OODA loop. For these reasons, the effects of suppression are an important component in our model.

Whereas most of our PMF reservoirs range in value from zero to one, our suppression reservoir ranges from zero to two. Suppression values from zero to one reflect a unit that is suffering from suppression, but is not pinned down (i.e. it still retains combat effectiveness). Once the suppression value exceeds one, the unit has lost most of its combat effectiveness. Soldiers in this state will be more inclined to blind-fire at their opponents, or simply peek in order to get an idea of what they are up to. As suppression approaches two, the unit will have lost all ability to move, return fire or gather intelligence on their enemy. The effects of suppression on unit effectiveness will be discussed further in Section 6.4.1.2.

As discussed earlier, units also have the ability to recover from suppression. Suppression recovery rate is the most subjective part of our model. After some initial testing, we selected a value of 30 seconds for a Level 1 unit to fully recover from suppression, given that it was no longer taking fire. For each unit level above Level 1 we subtracted 4 seconds from that time, so a Level 4 unit would fully recover in 18 seconds. Given the weapon values for suppression that we were working with, we found with continued testing that these values gave reasonable results, particularly given the effects we were trying to demonstrate.

The final point regarding our suppression moderator function is in regards to the point made earlier that "...suppressive fire requires sufficient intensity over the target area..." Our combat model already provides us with fire intensity, so we divide this value by the number of soldiers in the target group. In this way, a group of four soldiers would find it more difficult to suppress an enemy group of six soldiers than a group of four (assuming that the target troop quality remains the same).

6.4.1.2 Effectiveness Moderator Function and the Effectiveness Model

Effectiveness is the primary moderator in our model and is a high-level representation of both physical and cognitive effectiveness. We have seen how the primary effect of suppression is psychological, but we have also discussed its physical ramifications. In this regard, the cognitive and the physical are intertwined.

Numerous after-action accounts [66] [30] [72] refer to the initial effects of an encounter with an enemy force as being very dynamic, where the desire to avoid harm is in direct opposition to a soldier's desire to not let his teammates down. However, once a unit has essentially lost the firefight and is in the process of becoming pinned down, there is an inertia that sets in where the desire to avoid harm becomes the overriding factor. Once a unit reaches this state, suppression recovery becomes more difficult.

The other factor in this equation is unit morale, where morale represents a unit's willingness to expose itself to fire. Low morale troops become suppressed very quickly, as their self-preservation instincts take over almost immediately. Troops with high morale, on the other hand, are much more difficult to suppress, as they are driven to succeed, even to the point of being foolhardy.

It was felt that the best way to integrate all of these elements into a model for unit effectiveness was through the use of an exponential decay function, shown in Equation 8. In our model, Morale has a minimum value of 0.2, which avoids a "divide by zero" error.

$$Effectiveness = e^{(-Suppression\,/\,Morale)}\ x\ Skill\ Level \qquad (8)$$

Figure 6-3 provides us with a graphical depiction of Equation 8 for Level 2 troops. We can see from Figure 6-3 that the starting unit effectiveness has the same value as the unit's troop quality. As unit suppression approaches one, unit effectiveness has been reduced to approximately 12% and the curve begins to flatten as the unit spends most of its time at this point under cover and is making very little attempt to return fire.

As our Effectiveness PMF represents an abstraction for both physical and cognitive effectiveness, we can use the effectiveness value to determine the probability (willingness) of each soldier popping up to shoot. As group effectiveness decreases, fewer and fewer soldiers will pop up, and will either blind-fire, peek, or simply remain under cover. The value on the x-axis is the current value of the Suppression reservoir, which has a range of 0–2. Effectiveness is at its maximum when Suppression is 0.

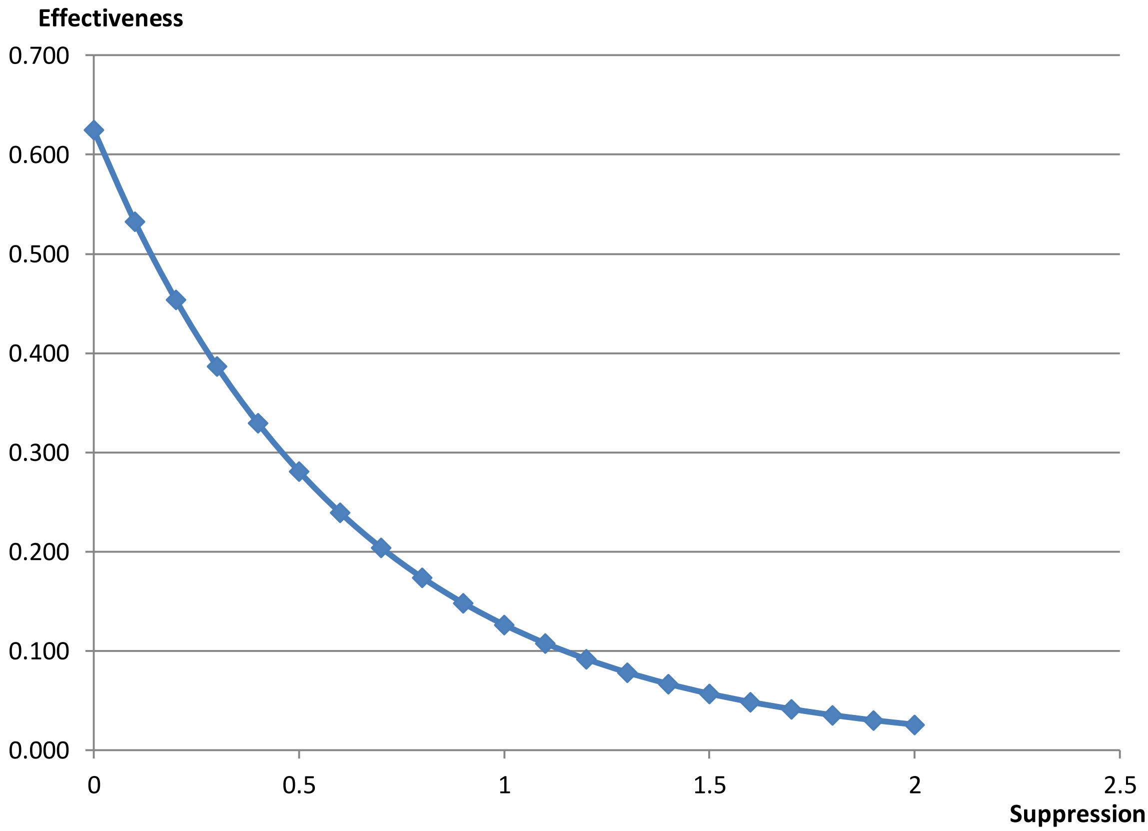

Figure 6-3: Effectiveness vs. Suppression for Level 2 Troops

6.4.1.3 Morale Moderator Function

Unit morale is affected in a number of ways in our model. Being pinned down erodes morale, so in our model, once a unit enters that state (suppression value between 1 and 2), morale begins to decrease at a rate that is proportional to how full the suppression reservoir is. The decay rate of unit morale is never more than 1/100 of its value, but even at that rate, prolonged suppression can seriously degrade a unit's combat effectiveness. This is because morale lost due to suppression is not regenerated when the unit is no longer suppressed. Combat takes its toll and, eventually, soldiers will become ineffective to the point of no longer being able to undertake high-risk courses of action, such as assault.

Morale erosion is modelled as follows:

```
if (suppressionLevel >= 1.0)
{
    suppressedAmount = 3.0 - suppressionLevel;
    double moraleReduction = Math.Exp(-suppressedAmount / leader.InitialMorale)
                                                      * leader.InitialMorale
                                                      / 1000.0;
    leader.MoraleReservoir.Remove(moraleReduction);
}
```

Another manner in which morale is affected is through the effects discussed in Bridges' [26] Transition Model. From it, we can see that events that are either Wanted or Unwanted will have a direct impact on morale, and these effects are driven by the various SAEs, which will be discussed in Section 6.4.2. Wanted events will increase morale, while Unwanted events will decrease morale.

The final way in which morale can be affected in our model is through interaction with a leader. Once a leader joins a group (team leaders start with their groups), his leadership affects the joined group in several ways. Immediately on joining, the joined unit inherits the leader's morale and suppression recovery rate. Murray [66] notes that it is almost universal that individuals perform better when the boss is watching, and military units are no exception. The unit inheriting the leader's values is representative of the leader's ability to motivate the joined unit to perform better. Even leaders of the same troop quality as the unit they are joining provide a small bonus to these values.

It should be noted that, while in most armies, leaders are selected for their ability to lead. In some armies, leaders are appointed for political reasons and may actually have a negative impact on the troops that they are leading (they have lower morale than the group they are joining) [73]. This effect is also reflected in our model. It should be noted that the leadership effect is removed when the leader leaves the unit.

6.4.1.4 Shock Moderator Function

In the same way that morale can be affected by the Wanted and Unwanted elements of Bridges [26] model, shock is the result of the Expected and Unexpected elements generated by individual SAEs. While it may seem strange to associate an Expected event with shock, an event that is Expected but Unwanted can still cause shock, although the degree to which shock affects the unit is dependent on how unexpected the event is. A unit on patrol that believes that the likelihood of contact is "Highly Likely", will experience some shock when first fired upon, but that value will be nowhere near the value associated with a unit that believes that the likelihood of contact is "Highly Unlikely" (i.e. they are completely surprised).

The shock values associated with the various SAEs is inversely proportional to the expectedness of the event and the skill level of the affected troops. On each iteration of the simulation loop, a unit whose shock reservoir is not empty may take no action (the unit is temporarily frozen). Units dissipate shock at the same rate that they dissipate suppression (section 6.4.1.1). The effects of shock on other PMFs are discussed in the next section on SAEs.

6.4.2 Situational Awareness Elements

Figure 6-1 shows all of the SAE's associated with the Leader class. Of the SAEs listed, our testing has only incorporated UnderFire, EnemyFlanking, ArtilleryBombardment and Casualty, so we will limit our discussion to these four. Based on the following descriptions, however, it should not be difficult for the reader to generalize how the other SAEs would affect a scenario.

All SAEs are derived from the base class SAE and, as such, are required to check upon instantiation whether or not a precognitive reaction is possible, and calculate their degree of Wanted/Unwanted and Expected/Unexpected.

When an SAE fires, it stores the outputs of its calculations in properties of the Leader class in a manner that is analogous to the way that values are centrally stored in a blackboard architecture. In this way, the properties,—which, in addition to SAEs, includes all values which affect a leader's state, such as fatigue, and modified troop quality and morale—can be thought of as short-term memory storage that other cognitive processes can tap into.

6.4.2.1 Under Fire

By nature of what we are simulating, UnderFire is the most prominent of the SAEs, and affects both the shooting group as well as the one receiving fire.

To some degree, all units engaged in a firefight suffer from tunnel vision [66] [30] [67]; the degree being determined by how highly trained the troops are. This tunnel vision is particularly dangerous for a leader if he involves himself in the firefight, as he loses his perspective of the battlefield. In our model, leaders can be either "Acting" or "Assessing the Situation", but not both at the same time.

When a unit first perceives that it is under fire, it reacts according to its level of training and the degree of shock associated with the event. The amount of shock is inversely proportional to the expectedness of the event and the skill level of the affected troops. The unit may take no action until the level of shock in the shock reservoir has decayed to zero.

Once the initial shock has worn off, the unit may react to contact, which normally means either heading for the closest cover, or dropping prone, and returning fire.

The UnderFire SAE increases the level of suppression in the unit's suppression reservoir based on the intensity of the incoming fire. As we have seen earlier, this increase in suppression will directly impact the unit's effectiveness. Unlike the other SAEs, the UnderFire SAE does not perform a Wanted/Unwanted check as this is already factored into the unit's decreased effectiveness.

6.4.2.2 EnemyFlanking

The more a group is out-flanked, the less secure they feel. The realization that the enemy is not only getting closer, but also improving its firing position, creates stress, which is manifested as suppression (i.e. the unit becomes less effective), and causes a temporary paralysis (shock). As the unit cannot perform any action, including shedding stress (suppression) while it is suffering from shock, an aggressive enemy can exploit this by pushing hard for the enemy's flank while the base of fire unit keeps the enemy's heads down. The degree of suppression generated is a function of the cosine of the angle between the flanking unit and the target unit's facing (values vary from 0 to 1).

The U.S. Field Manual 3-21.8 (Infantry Rifle Platoon and Squad) [62] defines a dilemma as a situation in which the enemy is presented with two or more equally bad alternatives, whereas a problem is a situation in which the enemy is presented with only one bad COA. When a leader can push his manoeuvre element towards the enemy's flank while his base of fire unit engages them, he creates a dilemma for his opponent—which unit should he engage. His first reaction is not knowing what to do as he attempts to decide between equally bad options. The second reaction would be to engage one of the two units, leaving himself vulnerable to fire from the other. The enemy's reaction is a function of his own troop quality and current effectiveness. As we will see in Chapter 7, enemy response will vary between "do nothing" and "break contact".

6.4.2.3 Artillery Bombardment

The ArtilleryBombardment SAE covers a range of artillery strikes, ranging from light mortars up to heavy artillery. The Wanted/Unwanted calculation of an artillery bombardment causes suppression, based on the calibre of the firing artillery and the rate of fire. The Wanted/Unwanted method does not directly affect morale. Instead, prolonged maximum suppression reduces morale, which is not regenerated. We discuss how suppression is calculated in section 6.6.4

An artillery bombardment also creates shock, as even if it is expected, it is definitely not wanted. This shock will inhibit the unit from shedding stress, which mounts with each artillery round that falls. Units subject to an artillery bombardment frequently become pinned down and, as this is on the relatively flat portion of the Effectiveness exponential decay curve, effectiveness recovery after a bombardment is generally slow. As mentioned in Sections 6.4.1.1 and 6.4.1.4, suppression and shock recovery rates are based on the unit's troop quality, an effect we will see more clearly in Chapter 7.

6.4.2.4 Casualty

Casualties directly affect a unit's morale. The morale reduction is the ratio of the casualty to the total number of soldiers in the unit (e.g. 1 in 3 would cause a reduction of 0.333%). Note that this is not a percentage of the current morale level; it is a reduction of the starting value. This percentage is then modified by the unit's skill level so that highly motivated troops are affected less by casualties than units with low morale.

6.5 Comprehension and Projection

We have already noted in Section 4.2.1 that Lipshitz [61], in his study on NDM-related research observed that all nine of the theories that he reviewed included an element of situation assessment, and that expert decision makers are able to perform situation assessment more quickly and accurately than novices. It is believed that this superiority in situation assessment skills accounts for much of the ability of experts to make rapid decisions and contributes to their decision-making accuracy. As Klein [25] notes, cue recognition/significance gives expert decision-makers an advantage because they can recognize cues more quickly and completely than novices, and they recognize patterns of cues better than novices. They can also detect important features of a stimulus more readily than novices so that they can detect the underlying structure of a problem. Here we can see that the Comprehension and Projection steps in Endsley's model for situation awareness allow us finer grained control in representing the relevance and potential impact of an actor's observations/perceptions. This, in turn, will allow for better separation of individuals in their decision-making ability.

6.5.1 Comprehension

In the Comprehension step, SAEs and a priori information, comprised of both environmental and physical factors, are combined via fuzzy rules, into a high-level appreciation of the situation. In essence, the outputs of the fuzzy rules provide us with the ground truth of the current situation.

The FIS is comprised of 23 rules, based on five input and two output fuzzy variables. The five input fuzzy variables were chosen as a representational cross-section of the factors that a leader has to contend with, and are not meant to be exhaustive.

Figure 6-4 shows the fuzzylite editor [74] used to create and test the fuzzy variables and rules. The author participated in the development of this tool in the testing and validation steps.

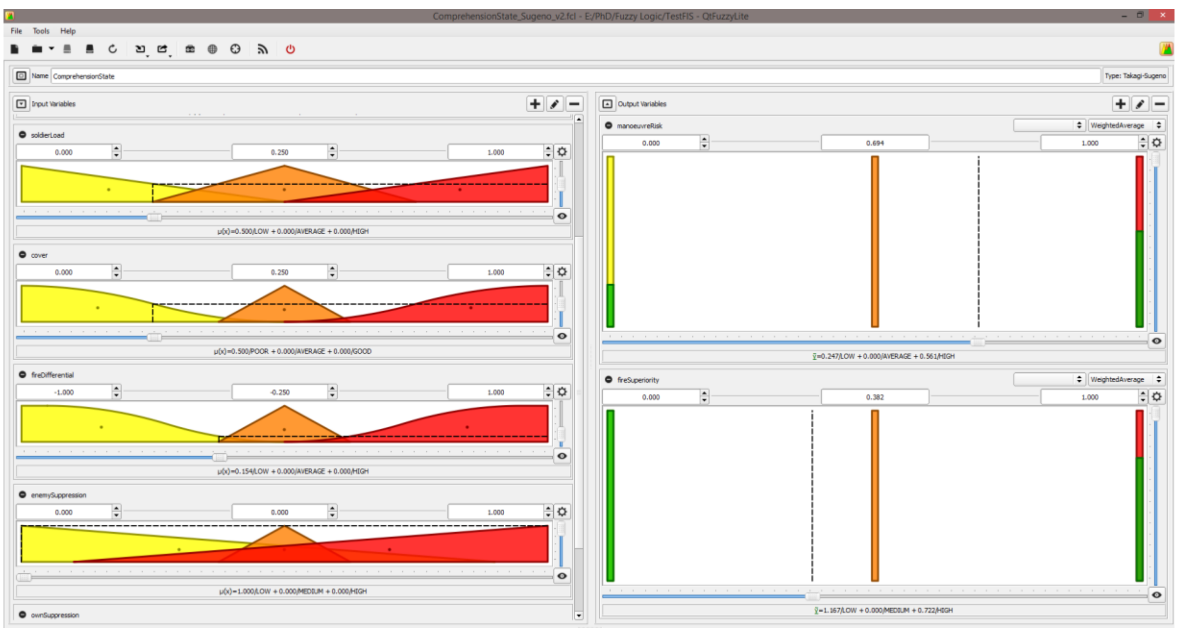

Figure 6-4: FuzzyLite FIS Editor

As discussed in Section 3.2.1, we chose to use the TSK method of fuzzy reasoning for its computational efficiency. We can see from Figure 6-4 that the output fuzzy variables use constants, which is consistent with a zero-order Sugeno model [53]. The representation of fuzzy sets in the input fuzzy variables is a mix of the familiar triangles and ramps, and "Z" and "S" curves. The latter were chosen for use in the fuzzy variables `Cover`, `GroundCondition`, and `GroundElevation` as we sought to define fuzzy variables that were relatively insensitive to change around their neutral point. This was done as a result of early testing that showed that in the early phases of a firefight, where neither side held an advantage, the high-level comprehension jumped around in a manner that we felt was inconsistent with the actual situation.

Given that the TSK method of defuzzification is by weighted average, we noted that as we added more input fuzzy variables, we began to dilute the strength of some of the fuzzy variables that we felt should be more prominent, such as cover and enemy fire. We addressed this problem by assigning decreased rule weights to the other fuzzy variables in order to retain the balance that we sought.

Some of the input variables, such as `SoldierLoad`, `GroundCondition` and `GroundElevation` are simply fuzzified and fed directly into a fuzzy rule. Others, such as `Cover`, are more involved. In the Comprehension state, the leader's assessment of the quality of cover is based on the distance his troops have to move from one piece of cover to reach the next one. The U.S. Field Manual 3-21.8 (Infantry Rifle Platoon and Squad) [62] stipulates that when rushing forward on the attack, soldiers are to remain exposed for no more than five seconds at a time. The degree of risk, therefore, when moving from one piece of cover to the next is directly proportional to the time that soldiers are exposed in the open (5 seconds being the maximum time a leader is willing to expose his soldiers to enemy fire, which equates to a manoeuvre risk value of 1).

It should be noted that the leader's assessment of cover at this point in the SA process provides only a rough sense of the quality of the cover available. Regions of cover, however, must pass this initial test to be considered further in the Decision state. In other words, if a cover region is deemed too risky to cross, based on this initial assessment, then it will not be considered further. In the CDMM, this assessment is accurate, and there is no chance that the AI will mistakenly choose the wrong cover region.

Utmost in the mind of any small unit leader is the concept of "winning the firefight" [62]. In other words, can his side suppress the enemy sufficiently that his troops can manoeuvre with minimal risk from enemy fire. Rather than feeding each side's fire effectiveness into a fuzzy rule, we calculate the effectiveness delta and pass that value to a fuzzy rule. While we could have simply fed the effectiveness values directly into fuzzy rules, as we do with friendly and enemy suppression values, and obtained the same results, we found the effectiveness differential to be a useful comparator further down stream in the Projection and Decision states. This will be discussed further when we reach those sections.

Figure 6-4 shows that our two output variables are `ManoeuvreRisk` and `FireSuperiority`, which are in keeping with the description in FM 3-21.8 [62], where the leader's decision-making can be distilled down to two questions: 1) Can I suppress the enemy?, and if the answer is yes; 2) Can I manoeuvre? If the answer to both of these questions is yes, then the leader can attack. If the answer to the first is no, then the leader must determine if he has a subordinate unit that he can add to suppressing the enemy. If he does, then his best COA now becomes Support by Fire for another leader's unit. In this way, we can see that the responses to "Do I have fire superiority?" and "What is my manoeuvre risk" ultimately shape his comprehension of the current situation.

Figure 6-5 shows the CDMM display with the four possible COAs, each in its own quadrant. `FireSuperiority` is shown along the x-axis, while `ManoeuvreRisk` is on the y-axis. The lines which divide the central display into the four quadrants represent a value of 0.5 along each axis, where `FireSuperiority` and `ManoeuvreRisk` are equal for both sides. In the scenario shown, we get an indication of Blue's high-level comprehension of the situation, which is that he does not have Fire Superiority, his Manoeuvre risk is high, and his side is slowly being forced onto the defensive.

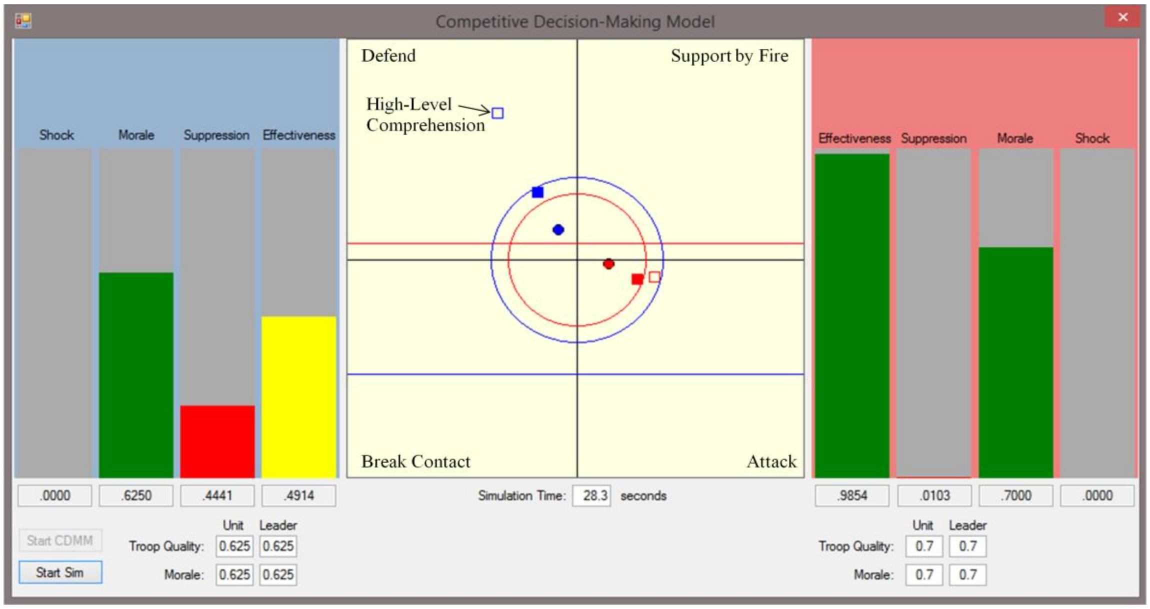

Figure 6-5: Comprehension

The following are the fuzzy rules used in the CDMM:

```
if groundCondition is POOR then manoeuvreRisk is HIGH;
if groundCondition is AVERAGE then manoeuvreRisk is AVERAGE;
if groundCondition is GOOD then manoeuvreRisk is LOW;
if groundElevation is UNFAVOURABLE then manoeuvreRisk is HIGH;
if groundElevation is AVERAGE then manoeuvreRisk is AVERAGE;
if groundElevation is FAVOURABLE then manoeuvreRisk is LOW;
if soldierLoad is LOW then manoeuvreRisk is LOW;
if soldierLoad is AVERAGE then manoeuvreRisk is AVERAGE;
if soldierLoad is HIGH then manoeuvreRisk is HIGH;
if cover is POOR then manoeuvreRisk is HIGH;
if cover is AVERAGE then manoeuvreRisk is AVERAGE;
if cover is GOOD then manoeuvreRisk is LOW;
if fireDifferential is LOW then manoeuvreRisk is HIGH;
if fireDifferential is very LOW then manoeuvreRisk is very HIGH;
if fireDifferential is AVERAGE then manoeuvreRisk is AVERAGE;
if fireDifferential is HIGH then manoeuvreRisk is LOW;
if fireDifferential is very HIGH then manoeuvreRisk is very LOW;
if enemySuppression is HIGH then fireSuperiority is HIGH;
if enemySuppression is MEDIUM then fireSuperiority is MEDIUM;
if enemySuppression is LOW then fireSuperiority is LOW;
if ownSuppression is HIGH then fireSuperiority is LOW;
if ownSuppression is MEDIUM then fireSuperiority is MEDIUM;
if ownSuppression is LOW then fireSuperiority is HIGH;
```

In the following sub-section, we will see how this high-level comprehension drives Projection and, ultimately, the leader's SA.

6.5.2 Projection

The Projection module receives the high-level comprehension from the FIS in the Comprehension module.

The SOM used in the Projection module is atypical in that its functioning had to be altered to fit within our architecture. A typical SOM will normally run for a given number of iterations, or epochs. As the time span of a tactical engagement is open-ended, we could not use this aspect of a SOM.

In our implementation, we refer to the SOM's learning rate as the leader's comprehension rate. This is due to the fact that we are simulating the leader's ability to incrementally gain situation awareness in a complex environment. Comprehension rate, like the neighbourhood radius, is not affected by the number of iterations of the leader's OODA loop, but rather by the leader's effectiveness.

In Section 6.4.1.2, we indicated that effectiveness is a high-level representation of both physical and cognitive effectiveness. Therefore, as a leader's effectiveness decreases, so does his comprehension rate. In effect, a leader that is forced to increasingly duck behind cover to avoid enemy fire is increasingly denied the real-time intelligence that he needs to obtain SA. His comprehension of the current situation is degraded.

Even with the use of the "Z" and "S" curves mentioned in Section 6.5.1, the hollow square on the CDMM display that represents the leader's high-level comprehension still jumped in large increments, particularly early on in a firefight where one side, then the other, has the upper hand. At this very early stage in an engagement, the limited data is insufficient for the AI to determine a trend.

We have already noted in Section 6.5 that cue recognition/significance gives expert decision-makers an advantage because they can recognize cues more quickly and completely than novices, and they recognize patterns of cues better than novices. They can also detect important features of a stimulus more readily than novices so that they can detect the underlying structure of a problem.

When observing one of our scenarios, it quickly becomes apparent to the observer if one side has the advantage; knowledge not available to our synthetic leader. The information is there, but initially appears chaotic. In essence, the true situation is obscured by noise and the leader cannot discern a pattern.

To permit our leader to focus only on the relevant pieces of information, we use a standard moving average by way of a circular array to filter out spurious data points. The length of the array is determined by the leader's projection skill which, in turn, is based on the leader's skill level. Highly skilled leaders can manage this filtering by maintaining data for the past 15 seconds. The poorest quality leader, however, is not as quick to recognize non-relevant data and must maintain data for a full minute. In this way, the skilled leader's ability to determine where the current situation is heading is extremely agile and adapts quickly to changes in the environment. The non-skilled leader, however, maintains a great deal of inertia in his cognitive processing and reacts very slowly to situational changes.

In our simulation, the filtered data represents the leader's near-future projection of the situation and is passed to the SOM to compute the leader's SA. The current value of the leader's projection is illustrated in Figure 6-6 as a solid square, and the leader's current level of SA is represented by the solid circle.

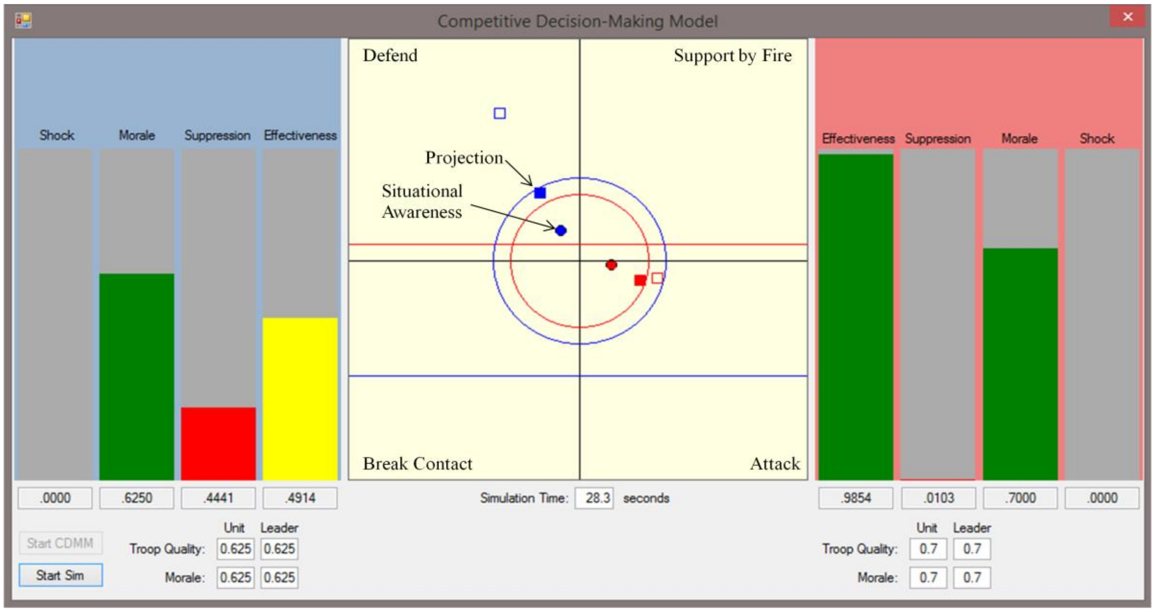

Figure 6-6: Projection

If we take our description of a SOM from Section 3.2.3 and use our own decision-making terminology, we observe that a leader's SA is driven by his interpretation of elements in the environment, not all of which are relevant.

As the BMU in a SOM (the leader's level of SA in our model) moves by a percentage of the distance between the BMU and the latest data point (the leader's projection), it stands to reason that the further apart the two are, the faster the BMU will move. In our model, the more quickly a leader can accurately project where the current situation is heading, the more quickly he will gain sufficient SA to make a decision.

Conversely, the closer the two points are, the longer it will take the leader to make a decision because he lacks sufficient information.

6.5.3 Conclusion

In this section we have seen how the Comprehension and Projection steps in Endsley's model are inextricably linked, as described by our expanded Boyd model in Section 4.2.1, and we have shown that SA is a function of effectiveness. The less effective a leader is—both in terms of inherent skill and enemy action—the slower his OODA loop moves and, thus the longer it takes to obtain sufficient SA to make a decision. We saw how a leader first obtains a high-level comprehension of the situation, and from that is able to discard non-relevant data in order to project where the current situation is heading. From this the leader's current situation awareness is formed.

In the next section, we will describe how a leader determines when he has enough SA to make a decision, and how supporting assets can play a role.

6.6 Decision

The Decision phase is where we determine whether or not the leader has sufficient situation awareness to make a decision (Attack, Support by Fire, Break Contact or Defend). In other words, the decision-maker decides if the current situation matches one that is already known. If yes, the selected COA will be assessed in order to determine if it will work in the current situation, or if another COA must be selected. As the four situations are defined in order of preference, the next COA on the list will be tested next.

6.6.1 Decision Circle

Once the leader's current level of SA is known, and the leader is able to project which COA is the most likely, we need to test to see if the current level of SA is sufficient to make a decision. We accomplish this by first establishing a skill-based decision distance for each leader; that is, how far from the neutral (0.5/0.5) position must the SA circle travel in order to represent that a leader's SA is good enough to make a decision. Figure 6-7 portrays an engagement between an elite leader (Blue) and a conscript one (Red). In the figure, each leader's decision distance is represented by a circle, centred on 0.5/0.5. One can see from the figure the difference in size between Blue's decision circle, and Red's, which shows that the conscript leader requires considerably more information (trips through the OODA loop) than its Blue counterpart.

Once the SA dot has crossed the decision threshold, a determination is made as to which quadrant (COA) best represents the current situation. As we will see in the next sub-section, however, the size of each quadrant is not fixed.

6.6.2 Risk Tolerance

In the same manner that we established a decision distance for each leader, we also determine a base level of risk tolerance that is based on the leader's skill level. In other words, the default manoeuvre risk line at 0.5 on the y-axis is simply a point of reference, and does not necessarily represent a leader's actual tolerance to manoeuvre risk.

Our Effectiveness Model updates a leader's effectiveness in real time (i.e. as soon as an SAE causes a change), but it also monitors the effectiveness differential between the two leaders that we discussed in Section 6.5.1, at a rate of 10 Hz. A leader's actual risk tolerance, therefore, is a function of the effectiveness differential which will either raise or lower the leader's base risk tolerance as shown by the red and blue lines of Figure 6-7.

Referring again to Figure 6-7, we can see that, based on the current effectiveness differential, Blue's risk tolerance is quite high, despite the fact that the enemy is not yet fully suppressed. Red's tolerance to manoeuvre risk is, commensurately, very low.

The shaded blue region in Figure 6-7 shows us the area that delineates the "Attack" COA for the Blue leader, and the shaded red region defines the new "Defend" COA for the Red leader.

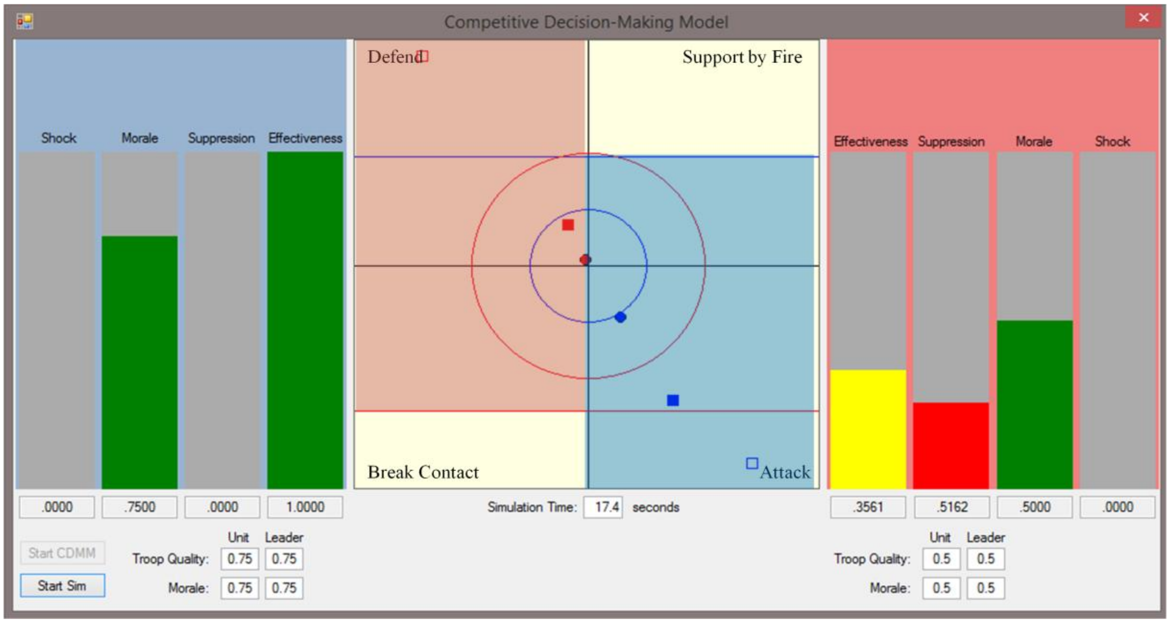

Figure 6-7: Decision with Decision Circles and Risk Tolerance Lines

6.6.3 Mental Simulation

Once it has been determined that a leader has sufficient SA to make a decision which, in this scenario is to Attack for the Blue leader (Red is still a long way from having sufficient SA to make a decision), then as stipulated by Klein, the leader must mentally simulate the COA to see if, in fact, it will work.

In Section 6.5.1, we discussed how the assessment of available cover is simply a quick check. The leader must now look more closely at the route that his assault element will follow in order to position themselves on the enemy's flank.

We discussed our weapon ballistics model in Section 6.2, where we demonstrated that distance from the enemy is a factor in enemy fire effectiveness. As the Blue leader's assault element, with each bound forward, will be under increasingly effective enemy fire, he must now reassess the quality of available cover against the enemy's current effectiveness.

Here, we assess the enemy's current level of lethality at each of the cover points that the assault element must traverse. Enemy lethality varies depending on their current effectiveness and the types of weapons that they are employing. For example, an enemy crew-served weapon will be more dangerous, even at longer ranges, than standard assault rifles, and the leader must take all of this into account before he commits his second fireteam to an assault. The assessment also factors in the team's exposure time, based on the team's running speed and the distance between cover points. In this way, we conduct a more accurate assessment of cover than the one performed in Section 6.5.1.

If the risk is deemed acceptable (i.e., less than his risk tolerance), and the leader has a non-committed unit available to conduct the assault, then the "Attack" COA is implemented. If the risk is deemed to be too high, or the leader has already committed all of his units to suppressing the enemy, then his fallback option is "Support by Fire" (i.e., continue to suppress the enemy and wait for his own superior to commit a unit to the assault).

6.6.4 Additional Support

We saw in Section 6.6.3 that, in addition to determining which COA best fits a given situation, the leader must also possess enough troops to execute it. In military engagements of this nature, leaders frequently have external resources at their disposal. In addition to supporting units, leaders can often call in airstrikes or artillery bombardments, particularly if they are in trouble. The leader is able to conclude this due to the fact that in the CDMM, all leaders are able to assess whether the current firefight is locked in stalemate, or if they are winning or losing, based on the effectiveness differential, which was discussed in Section 6.5.1.

In our model, we have implemented a basic version of the most likely support unit to be available to a squad leader; company-level 60 mm mortars. A company weapons platoon contains a mortar section of three 60 mm mortars, each of which is capable of a sustained rate of fire of 20 rounds per minute, or one round every three seconds. In our implementation, the mortars fire three rounds in a sequential manner, which results in a round landing on target every second for nine seconds. The suppressive energy of each round was found by dividing the calibre of the round (60 mm in our case) by 100, and then adjusting this value to account for the quality of the receiving unit's cover (no cover provides a value of one).

Support of this nature allows a leader to very quickly change the environment that both leaders are working in, and to subject his enemy to rapid, large-scale suppression. If the leader is good enough, he can use this sudden change to the environment to seize the initiative from his opponent and act decisively. We will discuss these effects more in Section 7 when we discuss the scenarios used to validate our findings.

6.6.5 Conclusion

In this section we demonstrated how we simulate the manner in which individual leaders determine if they have enough SA to make a decision, and how the degree of risk that they are willing to tolerate is highly dependent on both the leader's skill level and the current effectiveness differential.

If we look more closely at the scenario depicted in Figure 6-7, we see a Blue Level 4 leader (Elite) pitted against a Red Level 1 leader (Conscript). In a little better than 17 seconds, the Blue leader has determined that the conditions are right for an attack (the blue dot has cleared the decision circle), despite the fact that his enemy is only partially suppressed.

In contrast, the Red leader's current understanding has barely moved. The accuracy and fire discipline of the Blue troops has almost immediately overwhelmed the Red side, thereby denying real-time intelligence on "what is going on" to the Red side as they focus more on self preservation.

Once the Blue attack is launched, the Blue base of fire group will, in accordance with doctrine, switch to rapid rate of fire, further deepening Red side's suppression. When the flanking group begins to manoeuvre, the shock of Red side's predicament will fully paralyze the Red leader, which demonstrates that by seizing the initiative quickly, the Blue leader has ensured that the Red leader will be constantly reacting to the Blue leader's moves instead of focusing on what he should do next.

In the next section, we will close the OODA loop through the action determined by the selected COA.

6.7 Action

As discussed previously in Section 2.4, the primary focus of our research has been on situation awareness. The Action state, therefore, is the least complex of the OODA loop states.

In the UE3 editor, the scenario designer specifies the initial mission for the two opposing leaders. At scenario start, each avatar leader in UE3 passes its initial mission value to its CDMM counterpart. Based on the mission received, orders are specified in the Leader class and passed to the Action class, where one of the predefined plans that match the mission is selected. The plan is a C# dictionary that maps an input signal to a command delegate. In this way, the plan is configured like an event driven state machine, complete with transitions to other states (plans) based on the input received, either from the Decision state or from the Perception state.

By way of example, a unit that receives an “Ambush” mission will be initialized with the “Start” signal that will order the avatar leader and his unit to “Find Cover”. The Action class has an Orders helper class that converts the received order to a JSON-formatted message for onward transmission to the avatar leader.

Once the avatar leader and his unit are in cover, the avatar sends an “In Position” message to the Leader’s Perception state, which forwards the message on to the Action state. Once received by the Action state, the “In Position” message is looked up in the dictionary, and the corresponding order, “Peek” in this instance, is sent to the avatar leader via the Orders class. As mentioned previously, the received signal can also result in a change in state (plan).

A unit that is following “Patrol” orders that comes “Under Fire” (an SAE generated in the Perception state), changes its current plan to a predefined battle drill; “React to Contact” in this instance.

6.8 Conclusion

In this section on Behaviour, we described how our initial CDMM model incorporates the theoretical models of Endsley and Klein into Boyd’s model in order to elaborate the OODA loop into a functioning software architecture. In addition to supporting the decision-making cycle, our CDMM model implements this decision-making process in a time-competitive environment where small-unit leaders are required to operate in the highly complex and dynamic environment of combat. Our synthetic leaders must make decisions under the conditions of time pressure, uncertain information, and competing goals, with self-preservation often being at the forefront of these goals.

The following chapter will discuss how we chose to validate our model, and the scenarios used to do so.

7 Validation

There are two observations which deserve attention at this point: 1) the CDMM is not a fully functional military simulation (however, all those elements and behaviours deemed necessary to demonstrate the time-competitive nature of Boyd's OODA loop have been implemented); and 2) the field of cognitive science is extremely large and diverse, and since the author is not trained in this field, the models used within the CDMM are, by necessity, basic, although every effort has been made to make the behaviour of the synthetic entities as believable as possible. All of the values (and ranges of values) discussed thus far can be replaced, if the user wishes, in order to align the values to better represent what the user is testing for.

The CDMM has been designed in a modular fashion and, as mentioned in Section 5.2, the computational burden of the cognitive models has been offloaded from the process that runs the simulation engine in order to facilitate replacing any or all of the existing cognitive modules with more detailed ones without disrupting the frame rate of the simulation engine.

The scenarios discussed below were chosen so that, while we felt that they represented a good cross-section of the various aspects of the CDMM, their outcomes would be obvious to the observer.

In Figure 7-1 we see the Leader Properties dialog that we created within the UDK to enable the scenario designer to set scenario-specific leader properties. These properties include whether or not the leader is also the force commander, mission type, waypoints for a patrol, initial soldier load, initial fatigue, troop quality and morale, and likelihood of enemy contact.

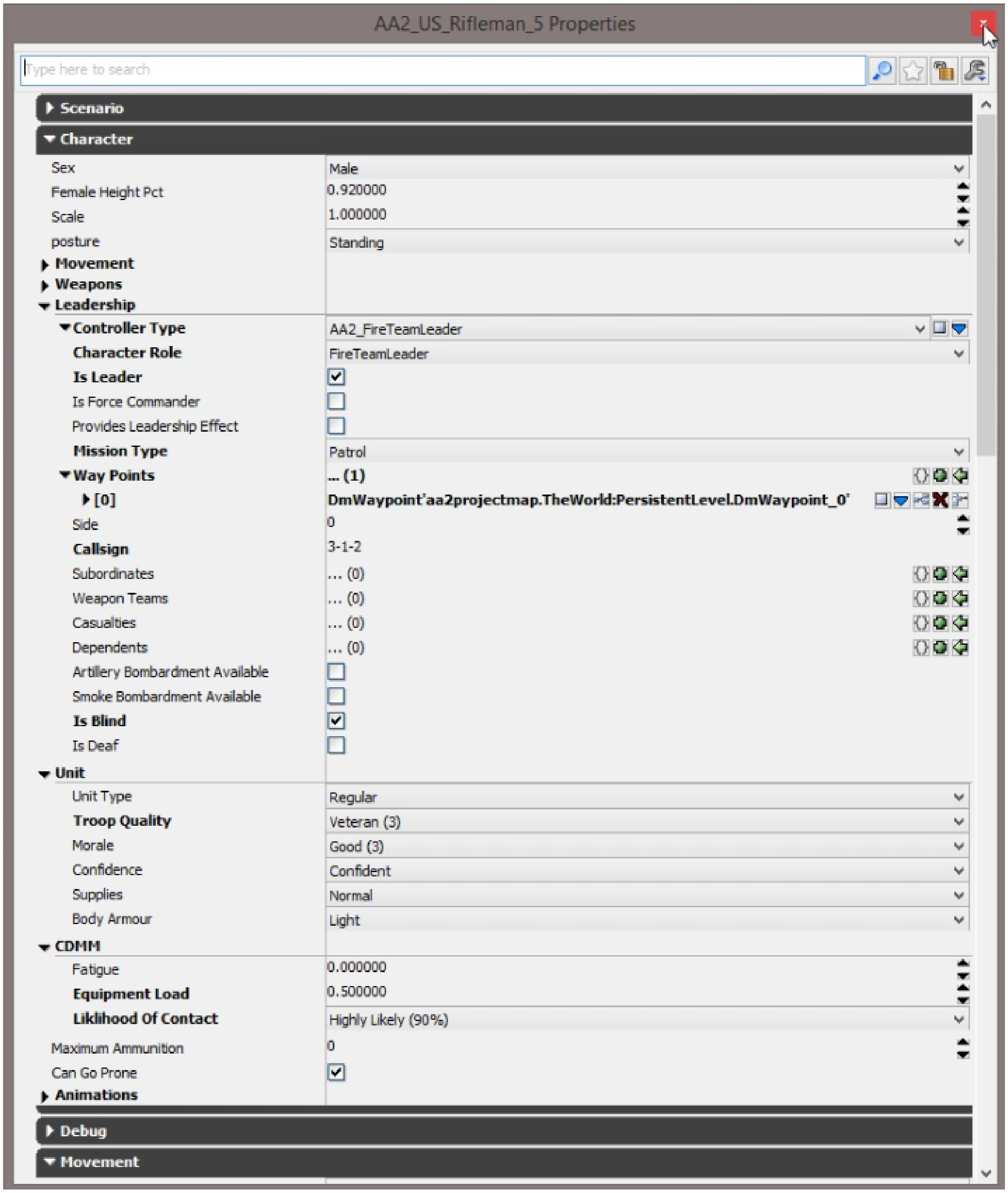

Figure 7-1: Leader Properties Dialog

7.1 Scenario Overview

We used a single test map, shown in Figure 7-2 and Figure 7-3, for all of our test scenarios, each of which begins with a meeting engagement between Blue and Red forces at a range of approximately 150 metres. The same weapons are used for both groups in order to remove weapon variances. As discussed in Section 6.4.2.4, there is an SAE option for casualties, but it is not used in the following scenarios so as to focus solely on leader effectiveness.

Figure 7-2 shows a three dimensional view of our test map, looking from Blue’s end of the map towards Red’s, with Red being at the north end. The map was divided into cover zones (north, south, east and west), with the west and east zones clearly visible in the figure. This was done so that leaders could quantify risk for map areas. From Figure 7-2, the reader can see that there is significantly more cover in the western zone than the eastern one. While our synthetic leaders will assess the manoeuvre risk for both zones in the same manner, it is obvious to the observer that the western zone provides considerably more protection to a manoeuvring unit than does the eastern one.

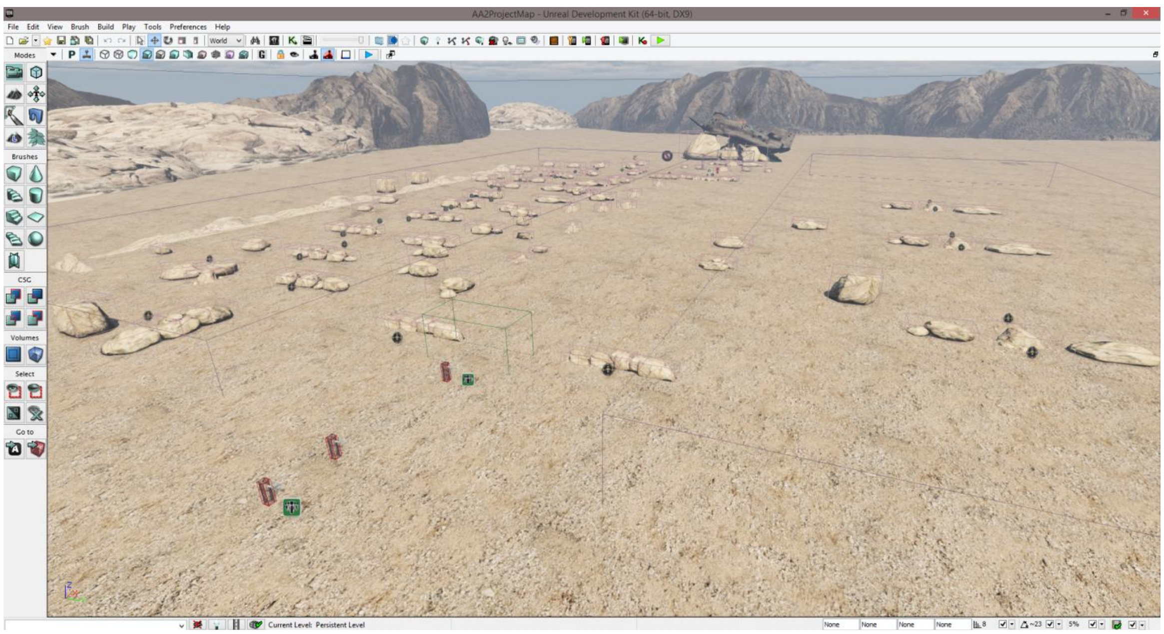

Figure 7-2: 3D View of the Test Map in the Unreal Editor

Figure 7-3 shows the same map, but from a top-down, wireframe perspective. Here we get a better look at the cover zones, including the northern one, which is the route Red would take if they chose to break contact. Interspersed amongst the areas of cover—large rocks in our case—we can see small dark dots. Each dot marks an area of cover on the map for the leaders. As we mentioned in Section 1.4, synthetic actors do not ‘see’ in the same manner as humans. Therefore, we must provide another mechanism for synthetic actors to ‘spot’ cover. In Figure 7-3, the disparity in available cover between the western and eastern cover zones is even more apparent.

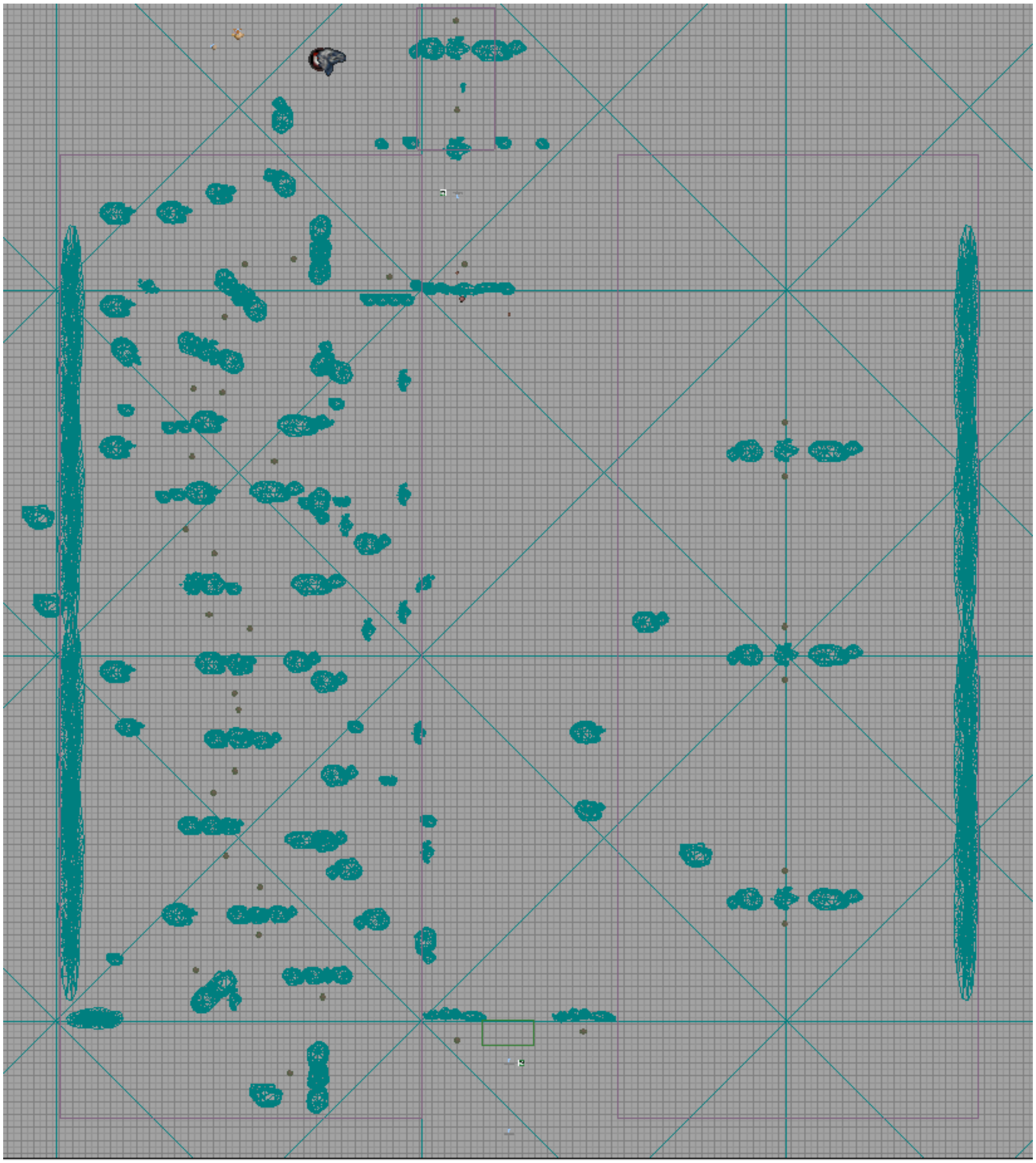

Figure 7-3: Top View to Test Map in Unreal Editor

In the following sub-sections, we will review each of the six scenarios whose outcomes we have used to validate our model. All scenarios are reciprocal in that there is no advantage for a force to be at one end of the map or the other. We have selected the Blue side as the aggressor in all scenarios so that we can use the same starting forces for both sides. In this way, we can see the effects of leadership skill and troop quality, as well as the effects of giving one leader supporting assets.

Both sides in each of the scenarios use U.S.-style formations and weapons. Blue side is a U.S. Army squad of nine men, with a squad leader and two four-man fireteams. The squad leader is equipped with an M4 carbine, and each fireteam contains two M4s (fireteam leader and a rifleman), an M4 equipped with and underslung 40 mm grenade launcher (used only as a third rifleman in our scenarios), and an M249 light machinegun, referred to as a Squad Automatic Weapon, or SAW, in U.S. terminology.

This formation was chosen because the squad leader (i.e. the decision-maker) is separate from the fighting units. Each fireteam is also led by a leader, complete with a CDMM counterpart, in order to establish a chain of command for passing orders.

The Red side is comprised of a leader and a single four-man fireteam, equipped in an identical manner to the Blue side. In this configuration, we have a leader embedded within the fighting unit, which is a configuration more closely related to how, in Commonwealth countries, the squad leader is also a fireteam leader. In our scenarios, the leader is separate from the fireteam which, like the Blue side, is led by a fireteam leader. This provides us with a different chain of command configuration.

The results of each of the scenarios are tabulated in Table 7-1 at the end of this chapter.

7.2 Scenario 1

Scenario 1 depicts two equal, Level 3 forces against each other. In terms of supporting assets, as we mentioned in Section 7.1, Blue has (for all scenarios), two fireteams to Red's one. As such, once contact is made in a scenario, Blue will have an uncommitted unit.

Once contact is made, both sides pause momentarily until the initial shock of meeting the enemy has passed. The likelihood of contact in all of our scenarios is "very high" so the initial shock is short-lived, but still dependant on troop quality. Once the shock has passed, both sides execute the React to Contact battle drill—they move to the nearest cover and return fire on the enemy.

At this point in the scenario, we have a firefight between two equal fireteams which will end in a stalemate, and both leaders will recognize this at approximately the same time. We say approximately because of the ebb and flow of the fight, where one side, and then the other, holds a slight advantage. As a leader's effectiveness drops, his cognitive processing speed slows down—he is spending increasingly more time under cover to avoid enemy fire and, therefore, is not assessing the situation. Therefore, while the length of time it takes a leader to recognize a stalemate is a function of a leader's skill level, the value is based on 100% effectiveness.

Once the Blue leader recognizes that he is in a stalemate, and that he will be unable to attack, he brings his second fireteam forward to help suppress the enemy, and is successful in doing so. He then contacts his immediate superior (not represented in any of the scenarios) and advises him that he can only "Support by Fire" in this engagement and that it is up to his superior to commit additional forces to the battle.

As Red is forced increasingly on the defensive, the option to "Break Contact" is no longer possible due to the intensity of fire from Blue. In CDMM terms, Red's tolerance for manoeuvre risk is now too low to allow them to safely manoeuvre, and their only option is to defend in place.

In Figure 7-4, we see that both sides are in a stalemate, as illustrated by the similarity of each of their PMFs, and their decision dots are still within their respective decision circles. Blue leader, therefore, has made the decision to bring his trail fireteam forward to support by fire.

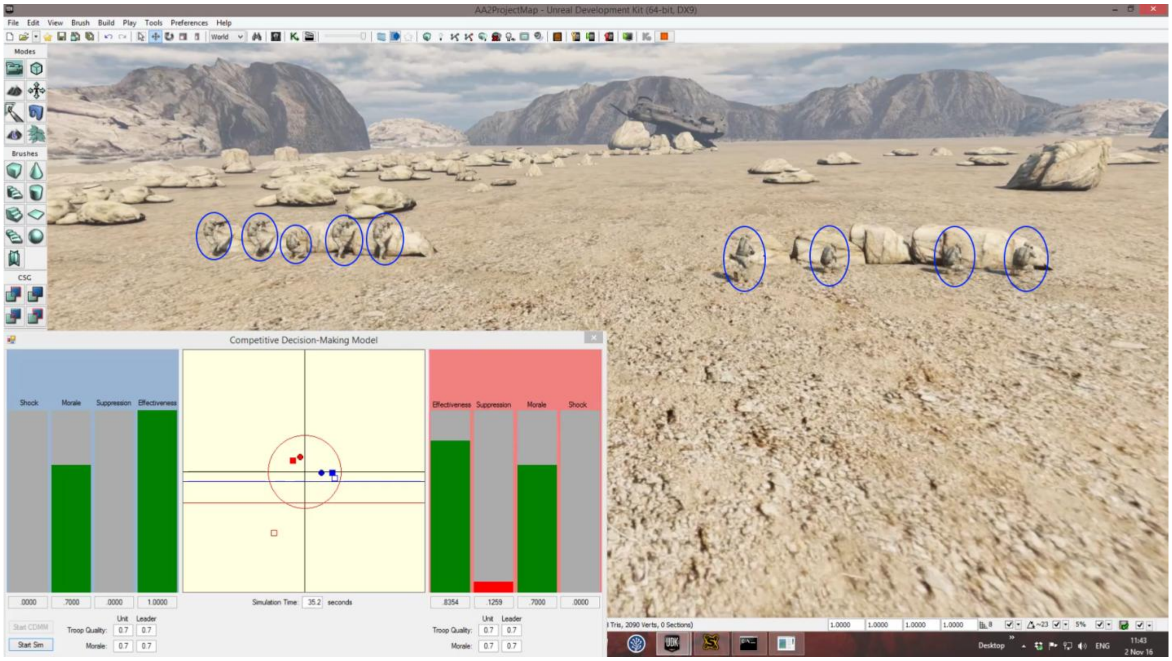

Figure 7-4: Fireteam 2 moving to Support by Fire

7.3 Scenario 2

Scenario 2 is identical to Scenario 1, except that Blue now has a mortar barrage on call as an additional asset. The scenario proceeds in the same manner as Scenario 1, except that once the Blue leader recognizes that the situation is likely to end in a stalemate, he calls in the mortar barrage, rather than deploying his non-committed unit to aid in suppressing the enemy. Now, he is relying on the barrage to suppress the enemy, which it does. Once Red is suppressed, Blue employs his non-committed unit as a manoeuvre unit in a flanking (using the western zone) manoeuvre, and orders his suppressing unit to increase their rate of fire from “sustained” to “rapid”. As we discussed in Section 6.2, sustained rate of fire can be maintained for long periods of time, but Rapid rate of fire is prone to weapon overheating and must be used judiciously. In our model, the AI only uses rapid rate of fire when an assault begins. The effect of weapon overheating is not modelled.

By having the flanking unit aggressively push towards the enemy’s open right flank (Blue’s left), and by increasing the suppressing unit’s rate of fire, Red now has two sources of danger, is suppressed, and is increasingly paralyzed as the flanking unit closes in on his position. By maintaining a high tempo to his attack, Blue ensures that every bound forward by the flanking unit throws Red further and further off balance.

7.4 Scenarios 3 and 4

Scenarios 3 and 4 are both standard attack scenarios. In Scenario 3, we have a Level 3 Blue vs. a Level 2 Red, while in Scenario 4, we have a Level 4 Blue vs. a Level 1 Red. As in Scenario 1, Blue’s only additional asset is the uncommitted fireteam. In both scenarios, Blue will gradually begin to win the firefight (suppress the enemy) and choose the “Attack” COA. The primary difference in the scenarios is time; Blue in Scenario 4 obtains sufficient SA to be confident that he can order an attack in approximately half the time that it takes Blue in Scenario 3. We are demonstrating here that while leader quality is extremely important, the disparity between the two leaders also has a tremendous impact on the outcome of the engagement. Figure 7-5 shows Blue’s trail fireteam manoeuvring to their left to launch a flank attack. We can see from the CDMM inset that Red’s SA is not yet good enough to make a decision.

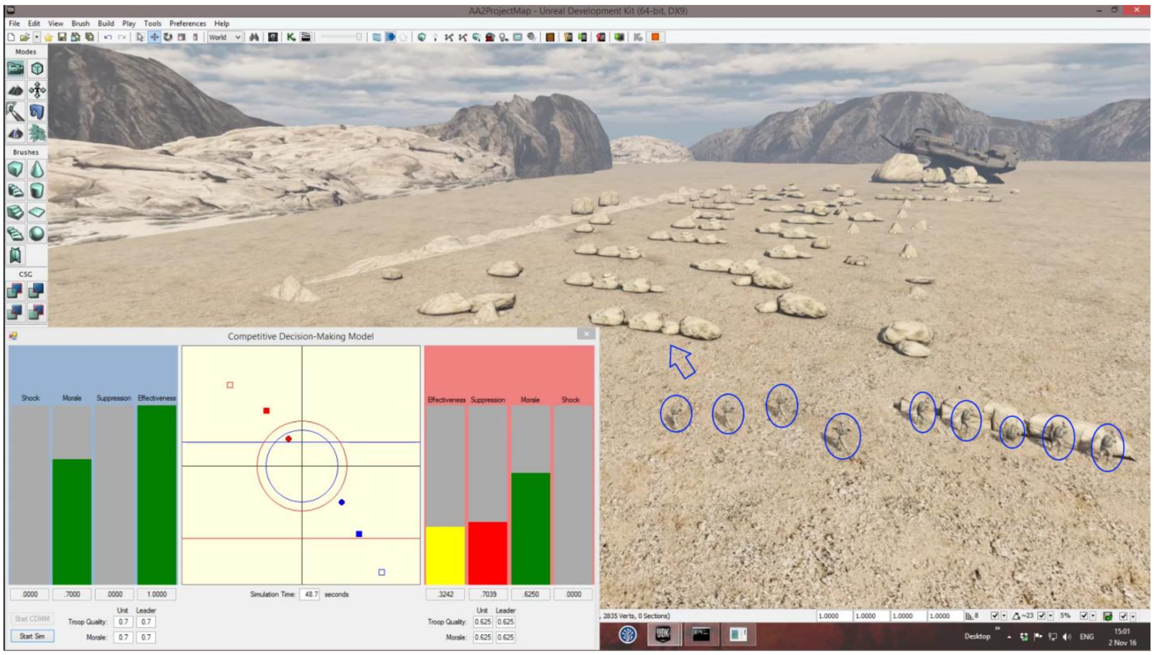

Figure 7-5: Blue launching a flank attack

7.5 Scenario 5

Scenario 5 is a reversal of Scenarios 3 and 4, in that an inferior group (Level 2 Blue) is facing a superior (Level 3 Red) group. Blue, however, has a mortar bombardment on call as a supporting asset, while Red has none. As we would expect from looking at the preceding scenarios, Red slowly begins to win the firefight.

As discussed in Section 6.6.4, in the CDMM, all leaders are able to assess whether the current firefight is locked in stalemate, or if they are winning or losing, based on the effectiveness differential. As a result, once Blue recognizes that he is losing, he calls in the bombardment and, just as in Scenario 2, Blue recognizes that, while Red is suppressed by the bombardment, he can attack.

Once again, the combination of stepping up the suppressing unit’s rate of fire to “Rapid”, and the aggressive flanking move by the manoeuvre unit, puts Red on the defensive and keeps them there.

7.6 Scenario 6

Scenario 6 is configured in much the same manner as Scenario 5, with an inferior force attacking a superior one. This time, however, we have a Level 2 Blue group vs. a Level 4 Red group—a disparity of two levels in the defender's favour.

As in Scenario 5, Red begins to win the firefight and, due to the Red leader's high skill level (elite), recognizes that Blue is sufficiently suppressed for Red to attack (COA of "Attack" is selected). During the mental simulation of the plan in the Decision state, however, Red deems that the attack is too risky because he is outnumbered 2–1 and Blue's second fireteam is not yet committed. He, therefore, falls back to "Support by Fire" as his chosen COA.

Blue, once he recognizes that he is losing, calls in his bombardment, which has the desired effect of reversing the firefight in Blue's favour. Red, however, is sufficiently experienced so as to overcome the effects of the bombardment (suppression inertia) sufficiently to recognize that the situation is lost, and decides to "Break Contact", even before Blue can launch his flanking assault in order to pin Red in place.

Here again, we see how the effects of skill and experience can alter the outcome of an engagement, and the larger the delta between leaders, the greater the effect.

Table 7-1: Results from Validation Scenarios

Number	Situation (Blue vs. Red)	Assets (Blue)	Observed	Explanation
1	Level 3 vs. Level 3 (Equal opponents)	Supporting Unit	Engagement will end in stalemate. Blue calls up supporting fireteam to assist in suppressing the enemy. He then reports to his superior that he cannot Assault, but can **Support by Fire**.	Once Red determines that they are overmatched, they are too late to attempt to break contact, as the manoeuvre risk is too high. Red is now fixed in place.
2	Level 3 vs. Level 3 (Equal opponents)	Bombardment, Supporting Unit	Engagement will end in a stalemate. Blue employs his on-call bombardment, thereby breaking the stalemate in Blue's favour and allowing Blue to **Assault**.	Blue's bombardment tips the balance. Blue capitalizes on the suppression effect of the bombardment to assault, which throws Red further and further off balance.
3	Level 3 vs. Level 2 (Blue slightly superior)	Supporting Unit	It takes Blue approximately 37 seconds to recognize that he can **Assault**.	Blue suppresses Red and launches a flank attack.
4	Level 4 vs. Level 1 (Blue greatly superior)	Supporting Unit	It takes Blue approximately 18 seconds to recognize that he can **Assault**.	Same effect as Scenario 3, but in approximately half the time.
5	Level 2 vs. Level 3 (Blue has slightly inferior troops)	Bombardment, Supporting Unit	Blue is initially losing, as we would expect. The bombardment allows Blue to gain the upper hand and **Assault**.	Once heavily suppressed by the bombardment, Red's only option is to **Defend**.
6	Level 2 vs. Level 4 (Blue inferior) Engagement status code changed	Bombardment, Supporting Unit	Blue is initially losing. Red assesses that it can attack. After completing the mental simulation of the assault, however, it is deemed too risky to attack, and **Support by Fire** is selected instead. Blue's bombardment reverses the situation, but Red is able to break contact before Blue can capitalize on its advantage.	Blue is losing initially and calls in a bombardment which reverses its situation. Red, however, is sufficiently skilled to recover from the bombardment enough that they can **Break Contact**.

7.7 Additional Observations

The following additional observations, while not associated with any one particular scenario, provide the reader with insight into some of the subtler points of the CDMM.

We can see from Table 6-1 that there are, in fact, many more combinations than the nine shown. This is because morale can be specified separately from skill to create a wide variety of units, or unit situations (e.g. lower morale due to being in combat for too long). Additionally, units can have leaders who are at a higher or lower skill level than the troops they are leading. In the CDMM, good leaders cause troops to perform better, and recover from suppression faster, while bad leaders drag down the overall quality of the troops that they are with.

Marcus Luttrell's comment regarding belt-fed weapons [67] that we referred to in Section 6.2 is particularly noticeable in the CDMM, as these weapons cause the most damage, by far. In test scenarios, where U.S. forces were facing insurgents armed with Russian weapons, the PKM light machinegun, which fires a 7.62 mm round vs. the 5.56 mm round from the U.S. M249, in the hands of a skilled operator, could completely dominate the battlefield, particularly when firing at a rapid rate of fire, and it was absolutely devastating when fired from ambush. From Table 6-2 we can see that the PKM can put almost double the energy on target as the M249.

In scenarios similar in structure to scenarios 3 and 4, but with the U.S. forces depicted in the scenarios above vs. an insurgent group with triple the firepower of the U.S. group, superior troop quality and leadership was still often able to offset this firepower disparity, depending on the effectiveness differential.

In the end, we opted for scenarios with forces using identical weapons so that we could focus solely on the effects of troop quality and leadership skills, and their effect on the decision-making process.

7.8 Conclusion

In this chapter, we have validated our thesis, that by merging the stand-alone theoretical models of Boyd, Endsley and Klein, we could create a practical application for decision-making that demonstrates the time-competitive nature of warfare on the modern battlefield.

With the introduction of the cognitive Transition Model of Bridges, we have validated the primary factor that sets Boyd's work apart from other models, which is the human dimension—how humans react to change, and the stress that it brings.

We have tested and validated that our software model, and its elaboration into a software architecture, effectively models the time-competitive nature of Boyd's OODA loop. We have demonstrated that by representing the interaction between opposing, or competitive, OODA loops, we have developed a more realistic model of the dynamics associated with the tactical decision-making process of AI small-unit leaders in a Constructive Simulation and thereby provided more credible synthetic forces.

Holistically, we have validated our findings by demonstrating that, as Boyd predicted, better trained leaders can alter the outcome of a fight through improved morale and experience, leading to faster decision-making. U.S. Field Manual 3-21.8 [62] stresses the importance of maintaining a high tempo in offensive operations, and we have shown that by seizing the initiative and manoeuvring aggressively, this high tempo induces a paralysis in the enemy. We have also demonstrated that more experience/better training can offset a numerical/firepower disadvantage.

In the next, and final, section, we will provide our overall conclusions of our work, and provide suggestions for future work.

8 Conclusions and Future Work

Liddell Hart wrote that: “In war every problem, and every principle, is a duality. Like a coin, it has two faces. This is the inevitable consequence of the fact that war is a two-party affair, so imposing the need that while hitting, one must guard” [75].

This quote underlines the importance of considering that war is a two-sided affair, and that one’s action will have an effect on the other party’s response. Boyd understood this duality, which he captured in his theory of the OODA loop.

In our thesis, we have created a theoretical model, based on the work of Boyd, Endsley and Klein; designed an architecture of distributed processes; and elaborated the software modules using Soft Computing techniques into a functioning architecture that supports a practical implementation of the time-competitive nature of Boyd’s model. Our use of Soft Computing techniques makes our model tolerant of the ambiguity and imprecision found on the modern battlefield, which contributes further to the credibility of our synthetic actors. Our CDMM is a standalone test bed that helps to prove this very concept.

In Section 1 we introduced the current state of AI in military simulations and demonstrated the need for an AI model that would reduce the considerable manpower requirements for running military simulations. In addition, we proposed to make CGF that respond appropriately to military tactics.

In Section 2 we presented Boyd’s time-competitive OODA loop decision-making model which postulates that the key to victory is to be able to create situations wherein one can make appropriate decisions more quickly than one's opponent. We noted that while Boyd’s model is useful as an illustration of the time-competitive nature of the decision-making cycle, it is not sufficiently detailed to allow for the creation of a software architecture that supports this cycle. Therefore, we decomposed the OODA loop into its logical components—Observe and Orient; Decide; and Act—in order to provide additional modelling detail to each of the steps. More specifically, we mapped Endsley’s SA model to Boyd’s concepts of Observe and Orient, and Klein’s RPD model to the Decision step in the OODA loop. Finally, we described how the Act step would be limited to pre-defined military battle drills.

In Section 3 we explored the concept of hybrid intelligent systems and demonstrated how a complementary fuzzy-neuro system could be used to model Endsley’s three levels of SA (Perception, Comprehension and Projection), which feeds forward to the Decision step. The architecture for this system is shown in Figure 4-2.

Section 4 combined our findings from the previous three sections into our expanded Boyd model, and Bridges’ Transition Model to account for the human element in the decision-making process. We further discussed how we intended to use cognitive PMFs to model the effects of effectiveness, suppression, morale and shock on a leader’s ability to make decisions in the highly complex and dynamic environment of combat.

Section 5 provided high-level descriptions and class diagrams of Unreal Engine 3, our simulation engine, and the CDMM. In addition, we discussed the communication protocol used to connect the two components and how our decision to create a distributed application, where the simulation engine and the CDMM operate as separate processes, allowed us to offload the cognitive processing from the CPU-intensive simulation engine to the CDMM component. Finally, we discussed how, by running the CDMM in a distributed environment, we are free to make our models for cognitive behaviour as complex as required without affecting the frame rate of the simulation.

Section 6 described the individual behavioural components of the CDMM that make up the decision-making cycle. Our description demonstrated how our CDMM incorporates the theoretical models of Endsley and Klein into Boyd's model in order to elaborate the OODA loop into a functioning software architecture.

In Section 7 we validated our thesis by demonstrating that better trained leaders can alter the outcome of a fight through improved morale and experience, leading to faster decision-making. We also demonstrated how, by maintaining a high tempo in offensive operations and seizing the initiative, we induce a decision-making paralysis in the enemy. We also demonstrated that more experience/better training can offset a numerical/firepower enemy advantage. All of which validated our thesis that by merging the stand-alone theoretical models of Boyd, Endsley and Klein, we could create a practical application for decision-making that demonstrates the time-competitive nature of warfare on the modern battlefield.

8.1 Model Requirements Review

The decision-making factors identified in Section 2 (repeated here for clarity) are as follows:

- Situation Awareness – the appreciation of those aspects of the current situation that are relevant to the question at hand.
- Predictive Capability – the ability of the agent to foresee the consequences of actions and the likely actions/reactions of other entities that are part of the scenario.
- Response Repertoire – the known action sequences for dealing with the current situation (skill set).
- Personal Preference – preferred methods of dealing with the current situation, often based on experience of previous successes and failures.
- Cognitive Effectiveness – the current state of the underlying cognitive architecture, affecting capabilities such as ability to recall facts, hold intermediate results in working memory, and stay focused on the problem.
- Affective State – the emotional factors that can influence a decision; for example, a high-level of fear can predispose a person to make an irrational decision.

When we map these requirements to the CDMM, we note that:

- The description of Situation Awareness is, essentially, Endsley's model of SA.
- Predictive capability is inherent in the Projection stage of Endsley's SA model.
- Response repertoire is supported by the use of trained responses (battle drills), and is also identified by Klein as an important part of NDM.
- Cognitive Effectiveness and Affective State have both been modelled by the use of PMFs (effectiveness, suppression, morale and shock), where the "fullness" of the reservoir has a proportionate impact on both decision-making speed and ability.

From this comparison of the factors affecting decision making, we can see that these requirements are supported by the CDMM that we have developed. Beyond just supporting the decision-making cycle, however, we have developed an architecture that also supports the time-competitive nature of Boyd's loop, augmented by the models of Endsley and Klein, where decisions are not made in a vacuum and there is a real requirement to be aggressive so as to deny to one's opponent the opportunity to understand what is going on around him; and to paralyse him so that he makes no decision at all. This behaviour is specifically what U.S. Army doctrine encourages [62], and it is precisely this element that is missing in existing decision-making models. Thus, the advantage of our CCDM is that it addresses both the real-time aspect of decision-making and the time-competitive nature of Boyd's OODA loop.

8.2 Future Work

The following sub-sections briefly discuss areas of future work, should development of the CDMM continue.

8.2.1 Upgrade the CDMM to Unreal Engine 4

Unreal Engine 3 is no longer supported by Epic Games, now that Unreal Engine 4 is available, so upgrading the CDMM to UE4 should be seriously considered. UE4, like its predecessor, is also free for non-commercial use and, unlike the UDK, users have access to the C++ source code. Users of UE4 have the choice of programming in either C++ or Epic's new visual scripting language, Blueprints.

This latter point would make it easier for non-programmers to continue to evolve the CDMM. Caution should be exercised here, however, as developing and debugging a distributed application, such as we have, is difficult. Developing and debugging one that also based on timing is considerably harder, yet again.

8.2.2 Upgrade the CDMM to Unity 3D

Extremely popular amongst independent developers, Unity has the advantage that its scripting language is C#, the same language the CDMM is developed in. This would consolidate the programming language requirements, rather than a developer having to learn C++.

8.2.3 Improvements to the CDMM

The following points came up during development of the CDMM, but were deemed to be outside the scope of this thesis. Modifications could include:

- Expanding the scope to include platoon and company-level operations;
- Modifying the suppression model to increase suppression if being fired upon by two or more groups that are separated, as the group being fired would be caught in a cross-fire, and the further apart the firing groups are, the greater the effect should be;
- Taking into account how a casualty is caused. A soldier killed in a firefight would lower morale, but if the same soldier were to be shot by a sniper, this should have a more detrimental effect on unit morale;
- Modifying the cover system so that not all cover has open flanks. The degree to which a flank is open should be a matter of degree, rather than simply binary;
- Factoring soldier load into soldier movement speed, rather than being a fuzzy variable. This would require, however, greater granularity in the definition of soldier attributes (e.g. strength).

References

[1] R. Bassarab and P. Roman, "Meeting the Intellectual Challenge Posed by Synthetic Environment Technology," *The Army Doctrine and Training Bulletin,* vol. 6, no. 1, pp. 20-26, March 2003.

[2] P. Roman and J. Cyr, "Enabling the Army Strategy with Synthetic Environment Technology," *Army Simulation,* vol. 5, no. 3, pp. 23-25, Fall 2002.

[3] P. Roman and D. Brown, "Games – Just How Serious Are They?," in *Interservice/Industry Training, Simulation, and Education Conference (I/ITSEC)*, Orlando, 2008.

[4] A. Bruzzone, M. Massei, A. Solis, S. Poggi, C. Bartolucci and L. Capponi, "Serious games as enablers for training and education on operations on ships and off-shore platforms," in *Proceedings of the 2013 Summer Computer Simulation Conference*, Vista, CA, 2013.

[5] T. Teng, A. Tan, W. Ong and K. Lee, "Adaptive CGF for pilots training in air combat simulation," in *2012 15th International Conference on Information Fusion*, Singapore, 2012.

[6] B. Lu and G. Gong, "A method for computer generated forces behavior modeling based on composability," in *2014 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, Yantai, China, 2014.

[7] S. Lee, C. Yoo, J. C. Park, J. H. Park, H. Lee, W. Lee and C. Chung, "Autonomous Intelligent Simulation Model based on CGF for Military Training and Analysis," in *Proceedings of the 10th WSEAS international conference on Computational Intelligence, Man-Machine Systems and Cybernetics*, 2011.

[8] G. Parkinson, "AI in CGFs comparative analysis," 2009.

[9] E. Page and R. Smith, "Introduction to Military Training Simulation: A Guide for Discrete Event Simulationists," in *Proceedings of the 1998 Winter Simulation Conference*, 1998.

[10] U. Dompke, "Computer Generated Forces – Background, Definition and Basic Technologies," *NATO Research and Technology Organization,* Vols. RTO-EN-017, October 2001.

[11] N. Abdellaoui, A. Taylor and G. Parkinson, "Comparative Analysis of Computer Generated Forces' Artificial Intelligence," in *Symposium Proceedings RTO-MP-MSG-069*, Brussels, 2009.

[12] R. George and F. Cardullo, "Application of Neuro-Fuzzy Systems to Behavioral Representation in Computer Generated Forces," in *Proceedings of the Eighth Conference on Computer Generated Forces and Behavioral Representation*, Orlando FL, 11-13 May 1999.

[13] J. Laird and M. van Lent, "Human-Level AI's Killer Application: Interactive Computer Games," *American Association for Artificial Intelligence AI Magazine,* vol. 22, no. 2, 2001.

[14] C. Dancy, F. Ritter and K. Berry, "Towards Adding a Physiological Substrate to ACT-R," College of Information Sciences and Technology, Pennsylvania State University, 2012.

[15] J. Laird and J. Duchi, "Creating Human-like Synthetic Characters with Multiple Skill Levels: A Case Study using the Soar Quakebot," 2000.

[16] A. Group, "CoJACK - A Moderated BDI Cognitive Architecture for Realistic Virtual Actors," AOS Group, 2008.

[17] D. Cassenti, "Performance Moderated Functions Server's (PMFserv) Military Utility: A Model and Discussion," U.S. Army Research Laboratory, Aberdeen, MD, 2009.

[18] R. Jones, E. Connors, M. Mossey, J. Hyatt, N. Hansen and M. Endsley, "Modeling Situation Awareness for Army Infantry Platoon Leaders Using Fuzzy Cognitive Mapping Techniques," in *Proceedings of the 19th Conference on Behavior Representation in Modeling and Simulation*, Charleston, SC, 2010.

[19] M. Endsley, L. Holder, B. Leibrecht, D. Garland, R. Wampler and M. Matthews, "Modeling and Measuring Situation Awareness in the Infantry Operational Environment," U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia, 2000.

[20] R. Jones, E. Connors, M. Mossey, J. Hyatt, N. Hansen and M. Endsley, "Using fuzzy cognitive mapping techniques to model situation awareness for army infantry platoon leaders," *Computational and Mathematical Organization Theory,* pp. 272-295, 2011.

[21] J. Boyd, "A Discourse on Winning and Losing," 1987.

[22] F. Osinga, Science, Strategy and War: The strategic theory of John Boyd, New York: Routledge, 2007.

[23] D. H. Abbott, S. Deichman, A. Elkus, L. Green, F. Hoffman, F. Osinga, C. Richards and T. Wade, The John Boyd Roundtable, Ann Arbor: Nimble Books LLC, 2008.

[24] M. Endsley, "Toward a theory of situation awareness in dynamic systems," *Human Factors,* vol. 37, pp. 32-64, 1995.

[25] G. Klein, "Naturalistic Decision Making," in *Proceedings of the Human Factors and Ergonomics Society*, 2008.

[26] W. Bridges, Managing Transitions: Making the Most of Change, 3rd ed., Da Capo Lifelong Books, 2009.

[27] R. Evertsz, M. Pedrotti, P. Busetta and H. Acar, "Populating VBS2 with Realistic Virtual Actors," in *Conference on Behavior Representation in Modeling & Simulation (BRIMS)*, 2009.

[28] L. Zadeh, "Fuzzy Logic, Neural Networks, and Soft Computing," *Communication of the ACM,* vol. 37, no. 3, pp. 77-84, 1994.

[29] S. Sari, Kuspriyanto and A. Prihatmanto, "Decision System for Robosoccer Agent based on OODA Loop," in *International Conference on System Engineering and Technology*, Bandung, Indonesia, 2012.

[30] P. R. Howe, Leadership and Training for the Fight, AuthorHouse, 2005.

[31] W. Lind, Maneuver Warfare Handbook, Boulder, CO: Westview Press, 1985.

[32] C. Wickens, "Situation Awareness: Review of Mica Endsley's 1995 Articles on Situation Awareness Theory and Measurement," *Human Factors,* vol. 50, no. 3, pp. 397-403, 2008.

[33] M. E. Bratman, Intention, Plans, and Practical Reason, CSLI Publications, 1987.

[34] M. Endsley, "Measurement of situation awareness in dynamic systems," *Human Factors,* vol. 37, pp. 65-84, 1995.

[35] H. Artman, "Team situation assesment and information distribution," *Ergonomics,* vol. 43, no. 8, pp. 1111-1128, 2000.

[36] P. Roman and B. Chapman, "Using Simulation to Estimate the Performance of the Situational Awareness System," *The Army Doctrine and Training Bulletin,* vol. 5, no. 3, pp. 26-31, Fall 2002.

[37] L. D. Strater, M. R. Endsley, R. J. Pleban and M. D. Matthews, "Measures of Platoon Leader Situation Awareness in Virtual Decision-Making Exercises," US Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia, 2001.

[38] J. Boyd, "Destruction and Creation," 1976.

[39] D. Kahneman, P. Slovic and A. Tversky, "Judgement under uncertainty: Heuristics and biases," Cambridge University Press, Cambridge, MA, 1982.

[40] G. A. Klein, R. Calderwood and A. Clinton-Cirocco, "Rapid decision making on the fireground," *Proceedings of the Human Factors and Ergonomics Society 30th Annual Meeting,* p. 576–580, 1986.

[41] D. Kahneman, "Maps of bounded rationality: Psychology for behavioral economics," *American Economic Review, American Economic Association,* vol. 93, no. 5, p. 1449–1475, 2003.

[42] S. Epstein, "Integration of the cognitive and psychodynamic unconscious," *American Psychologist*, vol. 49, p. 709–724, 1994.

[43] D. Grossman, On Killing, New York: Back Bay Books, 1995.

[44] P. Harland, "Emotion and Cognition in Neuro-Linguistic Psychotherapy," Winter 2002. [Online]. Available: http://www.cleanlanguage.co.uk/emotions-3.html. [Accessed November 2014].

[45] M. Minsky, 1992. [Online]. Available: http://rcaneba.weebly.com/1/post/2010/4/causal-diversity-by-marvin-minsky.html.

[46] M. Negnevitsky, Artificial Intelligence: A Guide to Intelligent Systems, Addison-Wesley, 2005.

[47] L. Zadeh, "Fuzzy Logic = Computing with Words," *IEEE Transactions on Fuzzy Systems,* vol. 4, no. 2, pp. 103-111, 1996.

[48] L. Zadeh, "Computing With Words and Perceptions - A Paradigm Shift," in *IEEE International Conference on Information Reuse & Integration,* 2009.

[49] D. Charles and S. McGlinchey, "The Past Present and Future of Artificial Neural Networks in Digital Games," in *International Conference on Computer Games: Artificial Intelligence, Design and Education*, 2004.

[50] L. Zadeh, "Fuzzy Sets," *Information and Control,* vol. 8, pp. 338-353, 1965.

[51] M. Negnevitsky, "Hybrid Neuro-Fuzzy Systems: Heterogeneous and Homogeneous Structures," *2009 WRI World Congress on Computer Science and Information Engineering,* vol. 2, pp. 533-540, 2009.

[52] R. Orchard, "Fuzzy Reasoning in Jess: The FuzzyJ Toolkit and FuzzyJess," in *Proceedings of the ICEIS 2001, Third International Conference on Enterprise Information Systems*, Setubal, Portugal, 2001.

[53] S. Uddin, Y. Sekercioglu and N. Mani, "Fuzzy logic systems: A classification based on capabilities in handling uncertainty," in *IEEE International Conference on Systems, Man and Cybernetics*, Montreal, PQ, 2007.

[54] S. Schmidt, R. Steele and T. Dillon, "Towards Usage Policies for Fuzzy Inference Meghodologies for Trust and QoS Assessment," in *Computational Intelligence, Theory and Applications*, Dortmund, Germany, 2004.

[55] P. Melin and O. Castillo, Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing, vol. 172, Springer, 2010.

[56] W. Combs and J. Andrews, "The Combs Method For Rapid Inference," *IEEE Transactions on Fuzzy Systems,* vol. 6, no. 1, pp. 1-11, 1998.

[57] J. Mendel, L. Qilan and W. Combs, "Comments on "Combinatorial rule explosion eliminated by a fuzzy rule configuration" [with reply]," *IEEE Transactions on Fuzzy Systems,* vol. 7, no. 3, pp. 369-373, 1999.

[58] S. Dick, A. Kandel and W. Combs, "Comment on "Combinatorial rule explosion eliminated by a fuzzy rule configuration" [with reply]," *IEEE Transactions on Fuzzy Systems,* vol. 7, no. 4, pp. 475-478, 1999.

[59] T. Jones, Artificial Intelligence – A Systems Approach, Jones & Bartlett Publishers, 2008.

[60] T. Kohonen, "The Self-Organizing Map," in *Proceedings of the IEEE,* 1990.

[61] R. Lipshitz, "Converging themes in the study of decision making," *Decision making in action: Models and methods,* p. 103–137, 1993.

[62] U. S. Army, FM 3-21.8: The Infantry Rifle Platoon and Squad, Washington, DC: U.S. Department of the Army, 2007.

[63] D. Crockford, "The JSON Data Interchange Standard," 2013. [Online]. Available: http://www.json.org/. [Accessed 16 November 2016].

[64] D. Meyer, Into the Fire, Random House Trade Paperbacks, 2013.

[65] S. Kushnick and J. Duffy, "The Identification of Objective Relationships Between Small Arms Fire Characteristics and Effectiveness of Suppressive Fire," DARPA, Arlington, VA., 2003.

[66] L. Murray, Brains & Bullets: How Psychology Wins Wars, Biteback Publishing, 2013.

[67] M. Luttrell and J. Hornfischer, Service – A Navy SEAL at War, New York: Little, Brown and Company, 2012.

[68] G. Silverman, M. C. J. Johns and K. O'Brien, "Human Behavior Models for Agents in Simulators and Games: Part I," Ackoff Center for Advancement of Systems Approaches, Dept. of Electrical and Systems Eng, University of Pennsylvania, Philadelphia, 2006.

[69] I. Janis and L. Mann, Decision making: A psychological analysis of conflict, choice and commitment, New York: Free Press, 1977.

[70] N. S. Office, AAP-06, NATO Standardization Office, 2015.

[71] D. Division, MCWP 3-11.2 Marine Rifle Squad, Quantico, VA: Marine Corps Combat Development Command, 2002.

[72] S. Naylor, Not a good day to die: The untold story of operation Anaconda, Penguin Group, 2005.

[73] T. Clancy and G. F. J. (. Franks, Into the Storm: A Study in Command, NewYork: G.P. Putnam's Sons, 1997.

[74] J. Rada-Vilela, "fuzzylite: a fuzzy logic control library," 2014. [Online]. Available: http://www.fuzzylite.com/. [Accessed 28 October 2016].

[75] B. Liddell Hart, Strategy, New American Library, 1974, c1967.

[76] J. Mendel and R. John, "Type-2 Fuzzy Sets Made Simple," *IEEE Transactions on Fuzzy Systems,* vol. 10, no. 2, pp. 117-127, 2002.

[77] B. Kosko, "Fuzzy Cognitive Maps," *International Journal of Man-Machine Studies,* vol. 24, pp. 66-75, 1986.

[78] L. A. F. M. M. Bushey, "The Recognition-Primed Decision Model," *FAMag - Field Artillery Magazine (U.S. Army)*, pp. 10-13, January-February 2006.