Viewpoint from a Defence Expert

The development of future war fighting concepts and capabilities involves the exploration of complex and multi-dimensional military domains. The effective use of traditional modelling and simulation approaches in this area is often constrained by the time and resources required to prepare, run and process high resolution combat models. Consequently, studies are often based on a limited set of runs that restricts examination of the wider problem space.

The use of agent-based simulation models has been adopted as a means of circumventing these shortcomings, enabling the modelling of complex military systems and thus enhanced scope for studies and exploratory experimentation. Typically, agent-based simulations are easy to use, flexible and have fast runtimes. The characteristics of agent-based simulations provide a tool that is well suited for the initial exploration of advanced war fighting concepts. This ability to explore the function landscape quickly is a major benefit of agent-based simulations and can be exploited by efficient experimental design that exhibits good space filling properties, allowing subsequent statistical investigation. Such tools can supply the analyst with a vehicle to explore the problem landscape, determine the opportunities and limitations of a new concept, gain insights into critical issues and thus identify areas requiring more detailed investigations. Importantly, military activities can be investigated at the effects level, without recourse to modelling of actual hardware and systems. Thus more abstract issues can be explored and this provides an independent force development process that could lead to disruptive rather than incremental changes in development trajectories.

Employing agent-based simulations as a exploratory tool provides a twofold benefit. They can provide the means to minimise resource costs, in terms of information, manpower and time, and increase flexibility. They also allow more creative investigation of new concepts without the constraints imposed by the properties of existing military systems. Critically, current agent based distillation studies have been found useful in that the game play and outputs are thoroughly compatible with the perspectives of military practitioners, and thus the technique is gaining ready acceptance.

John Hall

Land Operations Division—Canberra

Defence Science and Technology Organisation (DSTO), Australia

8. WISDOM-II: A Network Centric Model for Warfare

Ang Yang, Neville Curtis, Hussein A. Abbass, Ruhul Sarker
and Michael Barlow

Abstract

With recognition of warfare as a complex adaptive system, a number of Agent-Based Distillation Systems (ABDs) for warfare have been developed and adopted to study the dynamics of warfare and gain insight into military operations. These systems have facilitated the analysis and understanding of combat. However, these systems are unable to meet the new needs of defence, arising from current practice of warfare and the emergence of the theory of Network Centric Warfare (NCW). In this chapter, we propose a network centric model which provides a new approach to understand and analyse the dynamics of both platform centric and network centric warfare.

Introduction

Simulation is an important part of the Operations Researcher's tool box in assisting decision makers. Traditionally, Defence uses human-based warfare simulation to assess risks, optimise missions, and make tactical, operational and strategic decisions on change. However, simulation can be expensive and may not explore enough aspects of the problem or provide statistical validity.

Recent research (Ilachinski 1997; Lauren 2000) shows that warfare is characterised by nonlinear behaviours and that combat is a Complex Adaptive System (CAS). This has attracted attention from researchers and military analysts and a number of ABDs have been developed to understand and gain insight into military operations. They help the military analysts explore *What if?* questions that allow investigation of concepts and development of capability options. However, with recent practice of warfare and the emergence of the theory of NCW (Alberts 1999; Wilson 2004), analysts have found that it is hard to use these systems to understand and verify the new theory and concepts in warfare (Yang 2004). The military analysts often ask questions such as: how can we reach certain goals; why certain strange behaviours have emerged; and why the same setup has produced different results. Because almost all existing ABDs were built on platform centric warfare and existing agent architectures, it is very difficult for them to answer these questions. In this chapter, we propose version II of the Warfare Intelligent System for Dynamic Optimization of Missions (WISDOM)

(Yang et al. 2004, 2005b), which is built on a novel agent architecture, called the Network Centric Multi-Agent Architecture (NCMAA) (Yang et al. 2005a). With such agent architecture, WISDOM version II (WISDOM-II) allows analysts to study the dynamics of warfare easily, especially for NCW.

In this chapter we identify the key limitations of existing ABDs in the following section, then the NCMAA architecture and WISDOM-II are described. After that, scenario analysis is conducted before conclusions are finally drawn.

Limitations of existing agent-based distillation systems

Based on the understanding of warfare as a complex adaptive system (Ilachinski 1997, 2000, 2004; Lauren 2000; Scherrer 2004), a number of ABDs have been developed. These include the Irreducible Semi-Autonomous Adaptive Combat (ISAAC) (Ilachinski 1997, 2000) and the Enhanced ISAAC Neural Simulation Toolkit (EINSTEIN) (Ilachinski 2000, 2004) from the US Marine Corps Combat Development Command, the Map Aware Non-uniform Automata (MANA) (Lauren 2000; Lauren and Stephen 2002; Galligan et al. 2003; Galligan 2004) from New Zealand's Defence Technology Agency, BactoWars (White 2004) from the Defence Science and Technology Organisation (DSTO), Australia, the Conceptual Research Oriented Combat Agent Distillation Implemented in the Littoral Environment (CROCADILE) (Barlow and Easton 2002) and the Warfare Intelligent System for Dynamic Optimization of Missions (WISDOM) (version I) developed at the University of New South Wales at the Australian Defence Force Academy (UNSW@ADFA) (Yang et al. 2004).

ISAAC and EINSTEIN created a new era for warfare analysis. They are the first 2 systems that modelled warfare as a CAS. Almost all latter ABDs were inspired by them. MANA introduced the concept of way-points, internal situational awareness (SA) map and event-driven personality changes. The latest version of MANA (released at the end of 2004) concentrated on the model of communication, including the reliability, accuracy, capacity and latency of each communication channel. BactoWars focused on problem representation and tried to provide a simple framework that allows analysts to model the real world problems more adaptively and flexibly. CROCADILE used a 3D continuous environment with a higher fidelity than ISAAC, EINSTEIN and MANA. Version I of WISDOM improved the movement algorithm used by the above systems and adopted evolutionary computation techniques to search for optimal strategies for pre-defined scenarios (Yang et al. 2005b).

These systems have facilitated the analysis and understanding of combat, for example, using MANA to explore factors for success in conflict (Boswell et al. 2003). They offer an opportunity to analyse the behaviours that we would intuitively expect on the battlefield. Through the use of these systems, people are able to gain understanding of the overall shape of a battle and what factors are

playing key roles in determining the outcome of a battle. However, in our opinion, current ABDs are facing several shortcomings:

- Hard to validate and verify. System behaviours emerge from simple low level rules in any CAS. In current ABDs, agents are programmed without an underlying theoretically sound software architecture. Therefore, it is very difficult to validate and verify them.
- No reasoning during the simulation. Due to the high degree of nonlinear interaction between agents, it is impossible to reason at the agent level, which makes it hard to understand the results of the whole simulation.
- Can be a computationally expensive exercise in some systems. This is either because of a bad design, unnecessary fidelity, or fancy tools without proper modelling.
- No connection between tactic and strategy. Existing ABDs are developed either on the reactive agent architecture, which focuses on tactics, or on the BDI (Belief-Desire-Intention) architecture, which focuses on strategies (Wooldridge and Jennings 1995; Nwana 1996; Sycara 1998; Wooldridge 1999). There is almost no interaction between tactics and strategies being modelled by existing ABDs.
- Hard to capture the underlying structural interaction between agents. Although existing ABDs embed the structural interaction between agents, there is no explicit model for such interactions. It is hard for the user to capture these interactions during the simulation, which is a crucial point of a CAS.
- Difficulty in application to complexity. Current ABDs are based on conventional military tactics and tend not to be approached from an overarching systems view. Concepts such as NCW, with its inherent complexity and interdependency, present challenges to identifying correct inputs at the entity level. Thus, techniques addressing higher level manipulations must be employed.

To address the limitations above, WISDOM-II, which is re-designed and re-implemented based on the NCMAA architecture, is proposed.

Network Centric Multi-Agent Architecture

The limitations of existing ABDs and current agent architectures motivate the appearance of the new agent architecture of NCMAA, which is purely based on network theory (Wasserman and Faust 1994; Dorogovtsev and Mendes 2002). The system is designed on the concept of networks, where each operational entity in the system is either a network or a part of a network. The engine of the simulation is also designed around the concept of networks. Each type of relationship between the agents forms a network. We distinguish between a static network, where the topology does not change during the course of the simulation, and a dynamic network, where the topology changes. An example of the former is a

network of families, while an example of the latter is a communication network. It is important to emphasise that the definition of static or dynamic may vary from one application to another.

The properties of each network may change because the agents take actions that may trigger changes in agents' states, environmental states, or simply simulation clock advances. These triggers affect the dynamic relationships and the cycle continues. Figure 8.1 depicts a coarse-grained view of the system.

Figure 8.1. A coarse-grained view of NCMAA

To design a system based on NCMAA, the developer first needs to define an influence diagram of concepts, and then develop a finite state machine to control the simulation engine. The influence diagram is a directed graph of concepts defining the interdependency of concepts in the concept space. It provides the basis for establishing a meta-level reasoning system. The finite state machine is a collection of states, each representing the state of a network in the system. The finite state machine represents the sequence of executing each network in the system and the control of the system clock.

NCMAA adopts a two-layer architecture (Figure 8.2). The top layer, which we call the influence network, based on the influence diagram, defines the relationship types and how one type of relationship influences other types. Each of these relationship types is reflected in the bottom layer by a set of agents who interact using that relationship. For example, agents that can see each other would be connected in the lower layer, and agents that can communicate with each other would also be connected in the lower layer. The influence of vision on communication would form a connection in the top layer.

Figure 8.2. wo-layer architecture in NCMAA

WISDOM Version II

WISDOM (Yang et al. 2004) was first proposed in 2004 and is similar to other existing ABDs. Although it improved the movement algorithm and used the relational database to store information during simulations, it faced the same problems as other ABDs. In order to overcome these problems, WISDOM-II has been developed based on NCMAA. It not only uses the spirit of CAS in explaining its dynamics, but also centres its design on fundamental concepts in CAS. In WISDOM-II, concepts have been abstracted at a level that balances efficiency and depth of understanding. It is implemented based on the concept of keeping it simple, informative, efficient but realistic. Generally speaking, there are 5 components in WISDOM-II. The first 3 components are used to model the internal behaviours of warfare, and the last 2 components are analysis tools:

- *the C3 component*—including both Command and Control (C2), and communication. C2 is a key concept in military operations. It is the process where the commander directs and coordinates the assigned forces to achieve certain missions according to their own knowledge and information collected. Allard (1996) defined C2 as:

'The exercise of authority and direction by a properly designated commander over assigned forces in the accomplishment of the mission.

Command and control functions are performed through an arrangement of personnel, equipment, communications, facilities, and procedures which are employed by a commander in planning, directing, coordinating, and controlling forces and operations in the accomplishment of the mission.'

Dissemination of information in a command and control system requires secure and robust communication network. The C3 component in WISDOM- II has 2 computer models, one for C2 and the other for the communication system;

- *the sensor component*—retrieving information from environment;
- *the engagement component*—including firing and movement activities;
- *the visualisation component*—presenting various information with graphs; and
- *the reasoning component*—interpreting the results in natural language during the simulation process.

Architecture

There are 5 concept networks—relationships between agents (Figure 8.3)—that form the influence network in WISDOM-II and make up the top layer of the NCMAA. These 5 concept networks are:

- *The C2 network* defines the command and control hierarchy within one force. Each force has its own commander and control structure. Figure 8.4 depicts the C2 hierarchy in WISDOM-II. Each force may have several teams, each of which may include several groups. Each group may have a number of agents with different characteristics. We first introduce heterogeneous agents at the group level in WISDOM-II. Since the commands can only be sent from agents at the higher level to agents at the lower level, the C2 network is a directed graph.
- *The vision network*—if agent A can see agent B, then there is a link from Agent A to Agent B. The vision network is also a directed graph.
- *The communication network*—communication only occurs within the same force in WISDOM-II. Therefore, there are 2 instances of the communication network within the system. These communication networks could carry 2 types of information: situation information and commands. In a traditional force, the situation information typically flows from an agent to other agents in the communication channel, from the group leader to the team leader, and from the team leader to the general commander. In a networked force, the situation information flows directly from the agent to its general commander. In either a traditional force or a networked force, a Common Operating Picture (COP) is developed at the headquarter based on the collected information. Based on the COP the general commander makes decisions for each group in

the battlefield and sends commands to the team leader, and then the team leader sends commands to the group leaders. However, in a networked force, all agents in the battlefield can access the COP through the communication channel. Therefore, each agent has a global view of the battlefield, while in a traditional force each agent only has its own local view of the battlefield. Since we use network to model communication, it is easy for WISDOM-II to support various types of communications: Point to Point directly (P2Pdirect), Point to Point indirectly (P2Pindirect), and Broadcast (BC). Because the information flows from source to sink, the communication network is obviously a directed graph.

- *The information fusion network*—defines current knowledge about friends and enemies through vision and communication. We fuse the information collected by vision and communication and then develop this network. Since both vision and communication are direction dependent, the information fusion network is a directed graph too.
- *The engagement network*—defines the agents being fired at, based on the firing agent's current knowledge about its enemies and friends. This network is also a directed graph.

Figure 8.3. Influence Network for NCMAA in Warfare

Figure 8.4. Command and Control Hierarchy in WISDOM-II

Each of these networks influences each other. The C2 network specifies how the command is flowing, therefore it influences the communication network. Both vision and communication network define the information fusion network, based on which agents decide which agent they may fire at. Agents may die through firing, so the engagement network will influence the C2 network in the next time step. Each of the above concept networks may have one or more instances defined for the blue and red agents with their interactions. These instances make up the lower layer of the NCMAA. Since different forces have different C2 structures, and communication can only occur within the same force, there are 2 instances of the C2 network and 2 instances of each communication network, while there is only one instance for each other network in WISDOM-II.

Characteristics of agents

4 types of agents are supported in WISDOM-II: combatant agent, group leader, team leader and general commander. Agents are defined by their characteristics and personalities. Both team leader and general commander are virtual agents that exist in the force headquarters. They only have one capability: communication. Each combatant agent and group leader has 8 characteristics: health, skill, probability to obey command, invisibility, vision, communication, movement and engagement. The initial levels of health, skill, visibility and vision are set by the user. They may differ based on different agent types. The health defines the level of energy for an agent. With user defined wounded threshold and immobile threshold, each agent may have 4 health statuses: healthy, wounded, immobile, and dead.

The skill level of each agent is used to set the hit probability, which is the probability of the enemy being hit. High skill level means high hit probability. The probability to obey command is the probability to follow the command sent by its leader. When an agent receives a command from its commander, it may obey the command or take actions based on its own knowledge.

Invisibility defines how hard it is to be detected by other agents if it is within their vision range. The value of invisibility should be between $[0, 1]$, with 1 meaning invisible to all. Each combatant agent has its own sensor that is defined by sensor range and detection. Detection defines what kind of agents can be detected by using this sensor. If the detection of agent A is equal to or larger than the invisibility of agent B, and agent B is within agent A's vision range, then agent A may detect agent B. The value of detection should be between $[0, 1]$, with 1 representing able to detect all.

Combatant agents can communicate with other agents linked directly to them through the communication network. This communication occurs through a communication channel, which is modelled by noise level, reliability, latency and communication range. The agent may only communicate with the agents within the range of that communication channel. We also adopted a probabilistic model to implement the noise level and reliability of a communication channel. Each communication channel has 2 probabilities corresponding to noise level and reliability. At each time step the message can only be transferred from one agent to another agent. The message will permanently be lost if it is older than a number of time steps predefined by the user.

How agents move is determined by their speed and personality. WISDOM-II supports 4 kinds of speeds: still, low speed, medium speed and high speed. Agents with a low speed must wait for 2 time steps until they can move, and agents with a medium speed must wait for 1 time step until they can move, while agents with a high speed can move every time step. The movement algorithm is described in the section about tactic decision making.

Engagement in WISDOM-II is determined by what kind of weapon the agent uses. The weapon is defined by the weapon power, fire range and damage radius. Based on the damage radius, 2 types of weapon are supported in WISDOM: point weapon, the damage radius of which is zero, and explosive weapon, the damage radius of which is larger than zero. WISDOM-II also supports direct and indirect fire. Indirect fire can fly over obstacles.

Personalities of agents

Existing ABDs use personality to define the tendency of an agent to move close to or far from certain types of agents, while WISDOM-II uses it to define the influence of other agents on this agent. The influence of other agents may attract

the agent to move closer to them or repulse the agent far from them. The personality in WISDOM-II is a vector quantity specified by its magnitude and its direction. The magnitude is the strength of the influence which is between 0 (no influence) and 1 (strongest influence). The direction defines at which direction the influence occurs. There are eight directions, defined by a value between 0 and 1 (Figure 8.5).

Figure 8.5. Influence directions

The influence of agent A on agent B may be different from that of agent B on agent A. This difference can be reflected in either the value of the influence, or the direction of the influence, or both. Similar to existing ABDs, the personality is used to determine the new location the agent should move to. For an agent, its movement is influenced by all the agents which it can detect or communicate with. The algorithm to aggregate and resolve movement is described later. The personalities for each type of movement influence are:

1. Influence vector of healthy friend agent via sensor.
2. Influence vector of wounded friend agent via sensor.
3. Influence vector of immobile friend agent via sensor.
4. Influence vector of healthy hostile agent via sensor.
5. Influence vector of wounded hostile agent via sensor.
6. Influence vector of immobile hostile agent via sensor.
7. Influence vector of healthy friend agent via communication.
8. Influence vector of wounded friend agent via communication.

9. Influence vector of immobile friend agent via communication.
10. Influence vector of healthy hostile agent via communication.
11. Influence vector of wounded hostile agent via communication.
12. Influence vector of immobile hostile agent via communication.
13. Influence vector of group leader.

The group leader has the same number of personalities as a combatant agent. However, the group leader has the influence vector of the target flag assigned by the higher level commander instead of the influence vector of the group leader. Since they are virtual agents and do not move, the team leader and general commander do not have personalities.

Decision-making mechanism

There are 2 decision making mechanisms in WISDOM-II: tactical and strategic. The former is used by each agent to determine where it should move to and is based on the agent's current knowledge and personalities, while the latter is used by each general commander to determine the way point for each group and is based on the group's mission type and the firepower of each force in the battlefield.

Tactical decision-making mechanism

The movement of each agent is determined by its knowledge and personality. Only a healthy or wounded agent in the battlefield can move to a new location. In each time step, the agent can only move to its neighbour cell at the direction of the overall influence of all influencing agents. A movement function, as in Equation 1, is constructed on the influence vectors, and an agent moves in the direction of the resultant influence vector. The sum of the influence vector in the equation is a vector summation. Calculations are done synchronously with the moves. This process is repeated for each time step in the simulation. This movement algorithm is totally different from that implemented in any other ABDs.

$$RF_k = \frac{\sum_i^n \vec{F}_i^v + \sum_j^m \vec{F}_j^c + \vec{F}^t}{D} \quad (1)$$

where:

RF_k denotes the resultant influence of an evaluated agent k ;

n denotes the total number of agents perceived by agent k through vision in the information fusion network;

$\vec{F}_i^v$ denotes the influence vector from agent i , who is perceived by agent k through vision in the information fusion network;

m denotes the total number of agents perceived by agent k through communication in the information fusion network;

$\vec{F}_j^c$ denotes the influence vector from agent j , who is perceived by agent k through communication in the information fusion network;

$\vec{F}^t$ denotes the influence vector from agent k 's target. If the evaluated agent is a combatant agent, then the target is its group leader. If the evaluated agent is the group leader, then the target is the group waypoint.

D denotes the distance between the agent and the influencing agent.

In WISDOM-II, each cell can only accommodate one live agent. If more than one agent would like to move to the same cell, a collision occurs, then the collision resolution mechanism is used to remove it. The collision resolution mechanism is defined by a set of rules:

- the agent that occupied this cell in the previous time step has the highest priority to stay in this cell;
- the group leader has the second highest priority to move to this cell;
- the wounded agent has the third highest priority to move to this cell, the wounded agent normally goes back to its hospital for recovery; and
- if multiple agents with the same priority wish to occupy the cell, one is uniformly randomly chosen to move and the others stay in their original cells.

Strategic decision-making mechanism

A strategic decision is made for each group by the general commander of each force based on the COP, which is the global view of the battlefield for that force, and the mission type. Each team is assigned one mission in the input XML configuration file by the user. Then all the groups in that team have the same mission as the team mission. Three types of missions are supported in WISDOM-II: defend, occupy and surveillance. Defend means that the group is trying to protect a certain area; occupy means that the group is trying to occupy a certain area; and surveillance means that the group is trying to collect all possible information about the battlefield.

Three types of decisions can be made for each group: advance, defend and withdraw. Advance means that the group needs to go forward; defend means that the group needs to protect the current location; and withdraw means that the group needs to escape from the current location. After making decisions,

the high level commanders send the decision/command/intention to the lower level agents. To simplify the implementation of the command, the command sent to the group leader is the location of the waypoint.

The general commander abstracts the whole environment into $n \times n$ (n is predefined by the user) hyper cells and calculates the total firepower of the hostile and own force for each hyper cell. The total firepower is a function of individual agent weapon power, fire range, damage radius and health level (see Equation 2).

$$P = \sum_i^n (f_i^P \times f_i^R \times r_i \times h_i) \quad (2)$$

where:

P denotes the total fire power of one force in this hyper cell;

n denotes the total number of agents of one force in this hyper cell;

f_i^P denotes the weapon power of agent i in this hyper cell;

f_i^R denotes the fire range of agent i in this hyper cell;

r_i denotes the damage radius of agent i in this hyper cell;

h_i denotes the health of agent i in this hyper cell.

For each group with the mission of occupy, the general commander calculates the force power ratio defined in Equation 3 for each of the surrounding hyper cells.

$$R_p = \frac{P_o}{P_h} \quad (3)$$

where R_p is the fire power ratio, P_o is the total fire power of own force in that hyper cell, and P_h is the total fire power of hostile force in that hyper cell. The waypoint for each group is generated based on Equation 4.

$$WP = \begin{cases} C_i, & \text{if } \exists R_p^i \leq \theta_1, C_i \in C^b; \\ C_0 & \text{else if } \exists R_p^i > \theta_1, R_p^i \leq \theta_2, C_i \in C^b; \\ C_j & \text{else } \forall C_k \in C^s, R_p^j = \min(R_p^k) \end{cases} \quad (4)$$

where:

θ_1 is the advance threshold while θ_2 is the defend threshold. Both of them are defined by the user in the input configuration XML file;

C^b is the set of hyper cells which is between the group and its goal. It is defined in Table 8.1. The maximum number of the hyper cells in this set is 3;

C^s is the set of hyper cells which are surrounding the hyper cell of that group.
The maximum number of the hyper cells in this set is eight;

C_0 is the hyper cell which is occupied by that group;

C_i is a hyper cell in C^b ;

C_j is a hyper cell in C^s ;

R_p^i, R_p^j, R_p^k is the force power ratio in the hyper cell i, j or k respectively.

Table 8.1. The set of hyper cells in between

<i>Goal relative position to group position</i>	<i>The set of hyper cells in between (C^b)</i>
Top-left	Left, top-left and top hyper cell
Top	Top-left, top and top-right hyper cell
Top-right	Top, top-right and right hyper cell
Right	Top-right, right and bottom-right hyper cell
Bottom-right	Right, bottom-right and bottom hyper cell
Bottom	Bottom-right, bottom and bottom-left hyper cell
Bottom-left	Bottom, bottom-left and left hyper cell
Left	Bottom-left, left and top-left hyper cell

If the firepower ratio of one hyper cell in C^b is less than or equal to the advance threshold $\mathcal{A}_1$, the centre of this hyper cell will be assigned to the group as the waypoint. If the firepower ratios of all hyper cells in C^b are between the advance threshold $\mathcal{A}_1$ and the defend threshold $\mathcal{A}_2$, the group will stay in the original hyper cell. Otherwise the commander selects the centre of the hyper cell which has the minimal firepower ratio among all surrounding hyper cells as the waypoint for the group.

If more than one hyper cell meets the condition, the general commander will randomly select one. When the command is sent to the group, the group leader may misunderstand it. We model such misunderstanding by the probability and the variance of misunderstanding. The probability of misunderstanding stands for the chance that a group leader may misunderstand the command. The variance defines the degree of misunderstanding. In other words, it defines the degree of a received waypoint deviating from its correct location. After receiving the command, the group leader may or may not follow it, based on the probability of following the command mentioned above.

For a group with the mission of surveillance, the waypoint is the surrounding hyper cell with highest hostile force power. In order to maximise the information collected, no more than one group will be assigned to the same hyper cell. Since the area the team needs to defend is predefined by the user in the input configuration XML file, no command will be sent to the group with the mission of

defend. The frequency of the general commander sending the command is also defined by the user in the input configuration XML file.

Recovery

One of the most important aspects of military operations is logistics, and the medical treatment system is one of its key components. The model of the artificial hospital is first introduced in WISDOM-II. Each team may have a hospital in the team base, which is defined by the number of doctors and the recovery rate. If the team has a hospital, the wounded agent will move back to the hospital for treatment. Each doctor can treat only one wounded soldier at each time step and the health of that treated soldier will be increased by the recovery rate at each time step. If all doctors are already treating, the wounded soldier will be put in the queue to wait for treatment. When the agent is fully recovered, it will move back to the battlefield near its group leader. If its group leader is in the hospital, or if it is the group leader, it will be positioned in a cell around the team base.

The waiting queue and the number of doctors in the hospital are used to model the limited resources available and the recovery capability of a force. The capability to make a wounded soldier fully recovered is a key issue in warfare. With this kind of recovery model, analysts may search for the minimal resources needed to maintain a minimal level of recovery capability for a force.

Visualisation

One drawback of current ABDs is that they only provide limited information to analysts during simulations. WISDOM-II fills this gap and visualises many types of information. The visualised information in WISDOM-II can be in two categories: information about entities in the battlefield, and information about the interaction between entities. The entity information includes the agent status (position and health), the number of dead agents, the number of agents in hospital, the number of wounded agents, the total number of agents in each force, and the total damage for each force. With such information the user may quickly identify the damage, casualty and firepower for each force. The interaction information between agents includes: C2 structure, information flow chain, vision, knowledge, and engagement. By spatially visualising the interaction information, WISDOM-II provides a graphic view for each type of interaction between agents. It offers a window for the user to know what kind of interaction occurs between agents, and to identify the role of each interaction during the simulations. For example, visualising the C2 and communication network for each force may allow the user to see how information is transferring, how the C2 structure is evolving, and if it is efficient for a certain communication structure. Combining with visualisation of the entity information, the user may develop the relationship

between the damage or casualty of a force with the structure of C2 or communication network.

Spatially visualising the information also makes it possible to validate and verify the system itself. If something happens, it will immediately be reflected on the visualisation. The developer or the user can then identify and capture it quickly.

Finally, combining with the real-time reasoning component described in the following subsection, the analysts may gain deeper understanding of what happens in each time step during the simulation and how the simulation is progressing.

Reasoning

The architecture of NCMAA makes it possible and easy to conduct real-time reasoning during the simulation. BDI agents do not scale well because all reasoning is at the individual level. The nonlinear interaction between agents makes it almost impossible to reason at the individual level. To overcome this, WISDOM-II conducts reasoning at the network (group) level. We define the reasoning component through 2 stages: establishing causal relationships, and establishing degree of influence. The first stage is taken care of through the influence diagram, which defines the causal relationship. The second stage is undertaken through statistical inference.

We define 5 interactions (concept networks) between agents in WISDOM-II and the influence network identifying the influence between them (Figure 8.3). In order to see which network (interaction) among these 5 concept networks is the key aspect leading to the damage of the enemy and to see why certain behaviours emerge during the simulation, 2 kinds of reasoning are conducted in each time step: time series analysis, and correlation analysis.

Time series analysis allows the analysts to capture the dynamics over time during the simulation. It includes:

- damage of each force over time;
- average degree of each network over time;
- average path length of each network over time;
- clustering coefficient of each network over time;
- spatial entropy at the coarse, medium and fine level over time; and
- combat entropy over time.

When any network collapses, the user may immediately capture it according to these network measures.

Correlation analysis may allow the analysts to understand which network is playing the key role in damaging the enemy. We don't randomly select two parameters and conduct the correlation analysis between them. Correlation

analysis is based on the influence network (Figure 8.3). For example, in the influence network, the information fusion network directly influences the firing network, while the information fusion network is based on the communication network and the vision network. In order to see that the damage of the enemy is mainly caused by the activity in the vision network or in the communication network during a certain time frame, we may conduct correlation analysis of some network measures of the vision and communication network with the damage of the enemy.

The window and data-streaming techniques are used in the correlation analysis. All the data of the parameters, which are used to conduct the correlation analysis, are streamed and filtered. Only the valid data within the window size, which is defined by the user in the input configuration XML file, are captured for the correlation analysis. For example, if the window size is set to 5 and we would like to see if the communication plays a key role in damaging the enemy, we might calculate the correlation coefficient between the average degree of communication network and the damage of the enemy. Only the data of the average degree of communication network and the damage in the last 5 time steps are captured and used for the calculation. Following are 11 examples of correlation coefficients calculated between the *damage of the enemy* and the:

- average path length of communication network;
- average degree of communication network;
- ratio of the number of links from vision to the total number of links in information fusion network;
- ratio of the number of links from communication to the total number of links in the information fusion network;
- correctness of the knowledge about their enemy at the force level;
- correctness of the knowledge about their enemy at the agent level;
- correctness of the knowledge about their friend at the agent level;
- correctness of the knowledge about their friend at the force level;
- spatial entropy at the coarse level;
- spatial entropy at the medium level; or
- spatial entropy at the fine level.

In WISDOM-II, two kinds of knowledge correctness are measured: correctness of knowledge about the friend (C_f) and the enemy (C_e), which can be calculated as:

$$C_f = \frac{N_f^a}{N_f} \quad (5)$$

$$C_e = \frac{N_e^a}{N_e} \quad (6)$$

where N_f^a denotes the number of friends perceived by a single agent or a force; N_f denotes the total number of friends; N_e^a denotes the number of enemies perceived by a single agent or a force; N_e denotes the total number of enemies.

Based on the real-time reasoning, WISDOM-II provides an English-like interface to interpret the dynamics during the simulation to the user. Since the command sent by the general commander for each group is very important too, we integrate the command log to this English-like interface.

Terrain feature

WISDOM-II supports impassable objects. Agents cannot see or travel through impassable blocks. With the indirect weapon, the agent can shoot its enemies through them. The impassable blocks can be defined in the input configuration XML file by specifying each block's coordinates and the corresponding length and width.

Interactive simulation

Any simulation system is designed and implemented based on the domain knowledge of its developers. The limitation of the domain knowledge of the system developers may lead to misunderstanding of the simulations. This requires the simulation system to be able to import the domain knowledge from the domain experts during the simulation. Most existing ABDs generally don't allow interaction between the system and the user during the simulation.

WISDOM-II supports interactive simulation. In any time step, the users may send a command to any group based on their knowledge of the military domain. Therefore, the simulation may automatically run based on a predefined scenario or be guided by the user.

Scenario analysis

A simple scenario has been built to exemplify the use of our model. The red force is a traditional force with a large number of soldiers and traditional weapons while the blue force is a networked force with a small number of soldiers and advanced weapons. There are two surveillance agents in the blue force, they do not have weapons but are invisible to the red force. The scenario settings for each force are shown in Table 8.2. Figure 8.6 depicts the initial position of each force. The blue force is initialised at the top left corner of the environment while the red force is initialised at the bottom right corner.

Table 8.2. Scenario settings

	<i>Blue Force</i>	<i>Red Force</i>
<i>Number of Agents</i>	11	50
<i>Vision</i>	9 short	50 medium
	2 long	
<i>Communication</i>	Networked	Medium range
	2 no weapon	
<i>Weapon</i>	8 P2P	50 P2P
	1 explosive	

Figure 8.7 presents the damage of each force over time. From this figure, it is easy to see that the engagement started around time step 15. Although the red force has over 4 times number of agents than the blue force, the losses of the red force are much higher than that of the blue force. This is consistent with the importance of communication in combat. Two blue surveillance agents collect information and send it back through communication. Based on the COP developed in the headquarter of the blue force, the blue agent with the long range, explosive weapon may then shoot enemies. However, the red agents only have local information and they do not know where their enemy is, so they always get fired upon.

Figure 8.6. Initial position

Figure 8.7. Damage of each force over time

Figure 8.8 presents the average degree and average path length of the blue and red communication network over time. It supports our intuitive view above. The average path length of the blue communication network is always longer than that of the red communication network. The average degree of the blue communication network is larger than that of the red communication from about the time step of 35. This implies that information can be transmitted to most of the blue agents in the battlefield. We also note that the average degree and path length of the blue communication network are almost unchanged over time, while those of the red communication network are continuously going down. This may imply that the red communication network has broken and that they cannot exchange information properly. They know much less than the blue force and cannot win the game.

Figure 8.8. Average degree and average path length of the blue and red communication network over time

Figure 8.9 depicts the correlation coefficients between the damage of the red force and the knowledge of the blue force. It is obvious that the red damage is strongly related to the knowledge about both friend and enemy at the agent level, while at the force level it is more related to the knowledge about enemy than friend. At the agent level, agents may get more information about their enemies through their friends, so the knowledge about friend is of similar importance to that about enemy. However, at the force level, the only thing agents need to know is where their enemies are. Therefore, here the knowledge about enemy is more important than that about friend.

Our scenario demonstrates that a lesser number of networked blue agents can overtake a large number of red agents. Obviously, one cannot generalise from a single run when using a stochastic simulation. However, the objective of this chapter is to introduce a new model for warfare and it is not our objective to provide a detailed analysis of warfare simulations.

Figure 8.9. Correlation coefficient between red damage and the knowledge correctness at agent level (left) and force level (right) (window size is 5)

Conclusion

WISDOM-II is different from other existing ABDs in the sense of architecture, functionality and capability. Since it is developed on NCMAA architecture, it can overcome the limitations discussed above to some degree. With the help of network measures, WISDOM-II may easily validate the underlying structure. If a certain network collapses, the user can immediately detect it through network measures and visualisations. WISDOM-II conducts the real-time reasoning at the network (group) level. Such reasoning captures the domain specific interaction between networks (interactions), and a natural language interface provides a real-time interpretation of the simulation process in natural language for the analyst. This may overcome the limitations of the reasoning at the individual agent level. Since everything in WISDOM-II is a network or part of a network, the theory of NCW can be easily modelled, analysed and verified by using WISDOM-II.

The use of a network as the representation unit in WISDOM-II also facilitates efficient parallelism based on network structure and grounded modelling. In WISDOM-II, a rule based algorithm is used to make strategic decisions that guide a semi-reactive agent to make tactical decisions. Therefore, the interaction between tactics and strategies is easily captured. Concepts such as information misunderstanding, communication, and level of information fusion can now be studied in a unified framework. Overall, WISDOM-II is a promising ABD system that creates a new approach for analysts to understand the dynamics of and gain insight into warfare.

Acknowledgement

This work is supported by the University of New South Wales grant PS04411 and the Australian Research Council (ARC) Centre of Complex Systems grant number CEO0348249.

References

- Albert, R. and A.L. Barabási (2002) Statistical mechanics of complex networks. *Reviews of Modern Physics* 74: 47–97.
- Alberts, D.S., J.J. Garstka and F.P. Stein (1999) Stein Network centric warfare: developing and leveraging information superiority (2nd ed.). CCRP Publication Series.
- Alberts, D.S. and J.J. Garstka (2001) Network centric warfare: Department of Defense report to Congress. Technical report, Department of Defense, United States of America.
- Allard, K. (1996) *Command, Control, and the Common Defense*. (rev. edn.) Washington, D.C.: National Defense University, Institute for National Strategic Studies.
- Barlow, M. and A. Easton (2002) Crocodile: an open, extensible agent-based distillation engine. *Information and Security* 8: 17–51.
- Boswell, S., N. Curtis, P. Dortmans and N. Tri (2003) A parametric study of factors leading to success in conflict: potential warfighting concepts. In *Proceedings of the Land Warfare Conference*, pp. 369–81. Adelaide, Australia.
- Dorogovtsev, S.N. and J.F.F. Mendes (2002) Evolution of networks. *Advances in Physics* 51: 1079–187.
- Galligan, D.P., M.A. Anderson and M.K. Lauren (2003) *MANA: Map Aware Non-uniform Automata Version 3.0 Users Manual*. Devonport, Auckland: Defence Technology Agency. (Draft version).
- Galligan, D.P. (2004) Modelling shared situational awareness using the mana model. *Journal of Battlefield Technology* 7: 35–40.
- Ilichinski, A. (1997) Irreducible semi-autonomous adaptive combat (isaac): an artificial life approach to land combat. Research Memorandum CRM 97–61. Alexandria: Center for Naval Analyses.
- Ilichinski, A. (2000) Irreducible semi-autonomous adaptive combat (isaac): An artificial life approach to land combat. *Military Operations Research* 5: 29–46.
- Ilichinski, A. (2004) *Artificial War: Multiagent-Based Simulation of Combat*. Singapore: World Scientific Publishing Company.

- Lauren, M.K. (2000) Modelling combat using fractals and the statistics of scaling systems. *Military Operations Research* 5: 47–58.
- Lauren, M. and R. Stephen (2002) Mana: Map-aware non-uniform automata: A New Zealand approach to scenario modelling. *Journal of Battlefield Technology* 5: 27–31.
- Newman, M.E.J. (2003) The structure and function of complex networks. *SIAM Review* 45: 167–256.
- Nwana, H.S. (1996) Software agents: an overview. *Knowledge Engineering Review* 11: 205–44.
- Rao, A.S. and M.P. Georgeff (1995) Bdi agents: from theory to practice. In *Proceedings of the 1st International Conference on Multi-Agent Systems*, pp. 312–319, San Francisco, USA.
- Sycara, K.P. (1998) Multiagent systems. *AI Magazine* 19: 79–92.
- Scherrer, J.H. (2003) *Risks and Vulnerabilities of Network-Centric Forces: Insights from the Science of Complexity*. Newport, R.I.: Naval War College.
- Wasserman, S. and K. Faust (1994) *Social Network Analysis: Methods and Applications*. Cambridge, UK: Cambridge University Press.
- White, G. (2004) The mathematical agent a complex adaptive system representation in BactoWars. In First workshop on complex adaptive systems for defence, Adelaide, Australia.
- Wilson, C. (2004) Network centric warfare: background and oversight issues for Congress. Report for Congress, CRS.
- Wooldridge, M.J. and N.R. Jennings (1995) Intelligent agents: theory and practice. *Knowledge Engineering Review* 10: 115–52.
- Wooldridge, M.J. (1999) Intelligent agents. In G. Weiss (ed.), *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, pp. 3–44. Cambridge, MA:MIT Press.
- Yang, A. (2004) Understanding network centric warfare. *ASOR Bulletin* 23: 2–6.
- Yang, A., H.A. Abbass and R. Sarker (2004) Landscape dynamics in multi-agent simulation combat systems. In *Proceedings of 17th Joint Australian Conference on Artificial Intelligence*, LNCS. Cairns, Australia: Springer-Verlag.
- Yang, A., H.A. Abbass and R. Sarker (2005a) Wisdom-II: A network centric model for warfare. In *Ninth International Conference on Knowledge-Based Intelligent Information and Engineering Systems*, KES, LNCS. Melbourne, Australia.

Yang, A., H.A. Abbass and R. Sarker (2005b) Characterizing warfare in red teaming. *IEEE Transactions on Systems, Man, Cybernetics, Part B*, (forthcoming).