

Swarm or Borg? A comparison of Distributed and Collective Control

S. Wheeler¹ and G. White¹

¹*Land Operations Division, Defence Science and Technology Organisation,
PO Box 1500, Edinburgh SA 5111 AUSTRALIA*

Email: {scott.wheeler, geoffrey.white}@dsto.defence.gov.au

Abstract

This paper presents an empirical study of two alternative models for the Command-and-Control of a number of beyond line-of-sight weapons platforms using agent-based modelling based on cellular automata. These two models are used to investigate potential applications for optimisation algorithms embedded within simulations. Using these algorithms, the effect of distributed and collective control of the weapons platforms in these models is explored. Results of multiple runs of the simulator are analysed statistically to provide insight into the relative benefits and limitations of each model.

1. Introduction

This study is predicated on modelling land based warfare as a complex adaptive system. Under this predicate, the impact of two alternative methods for the Command-and-Control of a number of beyond line-of-sight weapons platforms in a networked environment is investigated using simulations built upon agent-based methods. These simulations are developed with a methodology of complex adaptive systems to define models of warfighting based on cellular automata.

Models of warfighting based on cellular automata trade sophistication for speed and lower simulation costs. As a result simulations tend to be less scripted with less user input than high-fidelity high-cost combat simulation software or seminar wargames. In such models, the emergent behaviour of the system as a whole is considered more important than the behaviour of any single constituent part of the system. This emergent behaviour is a characteristic of complex adaptive systems resulting from combined low-level interactions between numerous low-level entities in the system. These entities act according to comparatively simple rules but their behaviours combine in synergy to exhibit complex dynamic behaviour.

One of the most widely known developments of cellular automata used to model warfighting is Irreducible Semi-Autonomous Adaptive Combat (ISAAC), see Ilachinski (1997), developed by the US Center for Naval Analysis. ISSAC proposes to answer the question: “To what extent is land combat a self-organized emergent phenomenon?” The ISAAC white-paper, see Ilachinski (1997), states that ISAAC facilitates the exploration of

evolving patterns of macroscopic behaviour that result from the collective interactions of individual agents, as well as the feedback that these patterns might have on the rules governing the individual agents' behaviour.

Another well known development of cellular automata used to model warfighting is Map Aware Non-uniform Automata (MANA), see Lauren *et al.* (2001), developed by the New Zealand Defence Technology Agency. MANA is a sophisticated model which incorporates a wide range of features not available in other similar products. In comparison to ISAAC and MANA, the BactoWars project, see Millikan *et al.* (1996), is a developmental build of software for the purpose of modelling warfighting. BactoWars was developed by the Australian Defence Science and Technology Organisation (DSTO), Land Operations Division. Without loss of generality, we use BactoWars as a simulation tool for our agent-based models.

Three inputs are used in order to instantiate simulations in BactoWars. First, the nature of the battlespace is defined as a two dimensional grid of squares. This grid is interpreted as a two dimensional topological map by associating each of the grid squares to information about the nature of the terrain in the corresponding map coordinate. The map we use is based on a generic instantiation of an operation held in a littoral environment, see Australian Army (2002). Second, the entities populating the grid are defined on a generic 52-calibre self-propelled 155mm artillery, see Jane's (2002). These entities are associated with concepts such as firepower, red or blue force membership and the status of alive or dead. Entities are initially placed over the grid according to particular scenario requirements. Third, the rules each entity follows are defined. These rules involve the characteristics of movement and life and death. The rules assigned to these entities are designed to reflect the behaviour of the 155mm artillery in the field.

Two alternative Command-and-Control architectures for the entities in our simulations are considered. These two models are based on the *Ant* and *Borg* models as described in Sands *et al.* (2003). These models represent concepts for distributed and collective control respectively in a networked enabled environment. This study uses agent-based modelling to simulate the operational components of these models. Furthermore, this study investigates an application of two different kinds of optimisation algorithms embedded within the behaviours of the agents in the two corresponding models. Using these models, the differences between a user-optimised approach and a systems-optimised approach is discussed.

2. Experimental Method

2.1 Environment and Scenario

Simulations are run in a littoral environment, see Concept for Manoeuvre Operations in the Littoral Environment (MOLE), Australian Army (2002), modelled for agent-based simulations. This environment is divided into a 1000×1000 playing grid. Each of the square regions of the grid represent a 500m^2 area and are labelled with either *minor*, *nominal*, *substantial*, or *impassable* terrain restrictions. The grid size of 500m^2 is chosen for convenience. Hence, ten grid coordinates represents a distance of five kilometers. This size is sufficiently small enough to model lethal area information but is large enough to be tractable when the simulation is run. The four edges of the 1000×1000 grid are marked impassable to prohibit movement outside the playing grid. The map used in this study can be found in Wheeler and White (2004).

Traversal difficulty scores are used during movement of entities, see Table 1. Grid squares with high traversal difficulty scores require more effort to traverse than those with a low score. Impassable grids may not be entered at all. The values listed in Table 1 are representative of actual recorded speeds of 155mm artillery platforms and are scaled for the purpose of simulation. Note that entities in the simulations are positioned at the centre of the grid they occupy and may only move horizontally, vertically and diagonally. The actual movement of entities is explained in greater detail in subsequent sections.

Table 1. *Characteristics assigned to grid coordinates*

Terrain Restriction	Traversal Difficulty Score
Minor	210 units
Nominal	266 units
Substantial	580 units
Impassable	Infinite

The scenario is played out in discrete time $t \in \mathbb{N}$. Each step in time t represents an interval of six minutes in real-time. Hence, ten iterations of the algorithm together correspond to one hour elapsing in real-time. This representation of six minutes is the largest possible interval that is meaningful when considering, for example: movement speeds, rate of fire, and the time it takes to align the gun to a target.

2.2 Entities

Entities are modelled on a generic 155mm 52-calibre self-propelled artillery system, see Jane's (2002). Each entity is assigned a number of characteristics as explained below and summarized in Table 3.

- *Force Membership.* Entities belongs to either the red force (traditionally the enemy force) or the blue force (traditionally the friendly force). Civilian and allied forces are not considered.
- *Movement Counter.* During each time step, entities' movement counters are reduced according to the terrain traversed, see Table 1 above. Entities may not enter a grid that requires more movement units than the entities possess. Movement in an impassable region, such as the edge of the grid, is prohibited. Entities' movement counters are replenished at the start of each new time step.
- *Operational Time of Birth.* Entities are not all placed upon the playing grid at $t = 0$. Instead, entities operational time of birth counters record the time at which those entities are placed on the playing grid. This allows the red and blue forces to receive reinforcements throughout the simulation.
- *Operational Status.* Entities are initially considered to be operational. Entities that are hit by enemy fire are marked as destroyed and removed from the playing grid in the subsequent time step.
- *Position.* The playing grid is referenced by the tuple (x, y) . Both the abscissa and ordinate are defined over the discrete points $1 \dots 1000$. Note that the boundaries $x = 1$, $x = 1000$, $y = 1$, and $y = 1000$ are defined above to be impassable. Hence, entities may not be initialized in these regions.
- *Target Priority List.* A target priority list defines an exhaustive set of acceptable targets for the entities belonging to the blue force. Each acceptable target is ranked, on the natural numbers, according to priority. This ranking increases linearly in proportion to priority. Targets not on the target priority list are not targeted.

- *Target Status.* Target status records which, if any, of the acceptable targets from an entity's target priority list is the current target allocated to that entity. More than one entity may be allocated the same target.
- *Weapons Blast Template.* Shells only strike targets that are allocated by valid target status parameters. Collateral damage is not considered.
- *Weapons Range.* Entities' weapons have a maximum range of 78 grids radius. Firing upon targets beyond this range is prohibited.
- *Weapons Kill Probability.* Shells striking a target destroy that target according to a binomial distribution $p = 0.05$. Hence, if there are n shells striking a target, that target is destroyed with probability $P(\text{destroyed}) = 1 - (1 - p)^n$, where n denotes the number of shells striking the target. This study does not take into account the ability of the weapons platforms to converge their firepower on an enemy by successively correcting their aim. That is, the probability of killing a target with the next shell fired remains constant and is independent of the number of shells previously fired at that target. Hence, the act of firing upon a target is best interpreted as a desire to suppress the enemy by shelling the grid it resides in rather than a necessity to destroy the enemy with precise fire.

Table 2. Example probability of destroying a target

Weapons Rate of Fire	Probability of Kill
1 shell	0.05
2 shells	0.0975
⋮	⋮
10 shells	0.401263
11 shells	0.4312
12 shells	0.45964
⋮	⋮
100 shells	$1 - (0.95)^{100} < 1.0$

- *Weapons Rate of Fire.* Entities may fire more than once each time step. However, a weapons rate of fire of two shells per time step is analogous to modelling assets travelling in pairs and a weapons rate of fire of three shells per time step is analogous to modelling assets travelling in groups of three. We consider assets to travel as individuals with a rate of fire of 12 rounds per time step. Entities fire upon the targets allocated by their respective target status parameters. Hence, entities engage only one enemy target in any single time step irrespective of the potential number of times that they are permitted to fire in that time step. Note that irrespective of the number of shells landing on a target, it is impossible to guarantee that target is destroyed. Instead, if we require that targets be destroyed with at least 45% confidence, for example, we calculate the number of shells required and group that number of platforms together to form the capability brick which defines an entity. In Table 2, it is easy to see that to achieve at least a 45% kill probability, given a single round kill probability of 5%, we require: 12 assets with a rate of fire of 1 round per turn, 6 assets with a rate of fire of 2 rounds per turn, 4 assets with a rate of fire of 3 rounds per turn, 3 assets with a rate of fire of 4 rounds per turn, 2 assets with a rate of fire of 6 round per turn, or 1 asset with a rate of fire of 12 rounds per turn.

Table 3. *Characteristics assigned to artillery entities*

Characteristic	Value and/or Description
Force Membership	either blue or red
Movement Counter	between 0 and 2365 units
Operational Time of Birth	between 1 and 99 time steps
Operational Status	either operational or destroyed
Position	(x, y) -coordinate grid reference
Target Priority List	all valid target status details
Target Status	either no target or identifies current target
Weapons Blast Template	single target only
Weapons Kill Probability	5 percent probability
Weapons Range	78 grids radius
Weapons Rate of Fire	12 shells per time step

Entities act and/or react according to a fixed set of rules. These rules are divided into three categories; movement, target designation and fire, and life/death. These rules are executed in order at each iteration of the simulation. Two distinct behaviour types are developed, one for the red force and one for the blue force. Furthermore, the blue force is given two possible types of rules: behaviour type *Swarm* and behaviour type *Borg*. These behaviour types are mutually exclusive and are broadly based on the *Ant* and *Borg* models respectively from Sands *et al.* (2003).

Table 4. *Interpretation of parameters*

Parameter	Interpretation
Area of 1 grid ²	Area of 500m ²
Simulated time of 1 turn	Real time of 360s
Movement cost of 210 movement points	Speed of 15.6ms ⁻¹
Movement cost of 266 movement points	Speed of 12.3ms ⁻¹
Movement cost of 580 movement points	Speed of 5.7ms ⁻¹
Rate of fire of 12 rounds/turn	Rate of fire of 2 rounds/min
Weapons range of 78 squares	Weapons range of 39km

A summary of the parameters used in this simulation and the real values they represent is provided in Table 4.

2.3 Red Force Behaviour

Red force entities behave according to the rules displayed below. Note that this study is concerned primarily with the measurement of the responsiveness of the blue force to the appearance of a number of red force entities. Hence, we specifically prohibit the red force from engaging the blue force.

- Red force entities move as many grids as possible in a direction chosen randomly with uniform distribution from the set {North, North-East, East, South-East, South, South-West, West, North-West}.

- Red force entities are not allocated targets. Hence, red force entities do not engage blue force entities.

2.4 *Blue Force Behaviour - Both Swarm and Borg*

Blue force entities behave according to the rules displayed below.

- Blue force entities without a valid target allocated by their target status parameter do not move.
- Blue force entities with a valid target allocated by their target status parameter do not move if this target is within a distance defined by their corresponding weapons range parameter.
- Blue force entities with a valid target allocated by their target status parameter move in a straight line towards this target if this target is not within a distance defined by their corresponding weapons range parameter. Such entities move as many grids as possible to bring the target into weapons range in the direction selected above. Once entities have acquired a target, weapons fire is resolved. All entities are considered to fire simultaneously.
- Entities fire upon targets allocated by their corresponding target status parameter, if that target is within weapons range.

2.5 *Blue Force Behaviour - Borg only*

Blue force entities of behaviour type *Borg* are allocated targets according to the rule below.

- Blue force entities of behaviour type *Borg* are allocated targets according to a marriage algorithm, see Ahuja *et al.* (1993) from the field of optimisation. This algorithm is explained below.

A marriage algorithm, more accurately known as the bipartite weighted matching problem or assignment problem, see Ahuja *et al.* (1993, pp. 470–471), matches potential “brides” and “grooms” according to pre-defined characteristics on a sliding scale. Hair color on a scale of 1 to 5, for example. The matching of all the brides and grooms is performed in such a way that the collective group is as “happy” as possible. Hence, it is possible for individuals to be married to a spouse that they are poorly suited to if by doing so one or more other couples increase in suitability by a sufficient amount. In applying this algorithm to targeting we interpret the brides and grooms as friendly entities and enemy entities. The sliding scale is a function of target distance and priority. By maximizing the collective “happiness” of the entities we ensure that the most enemy targets of highest priority are engaged. Hence, the allocation of fire is optimised globally. This corresponds to a system of collective control because the entities work in concert to engage the enemy force. Although each individual in the *Borg* model follows a simple target allocation algorithm, the system as a whole exhibits collective intelligence to optimise the engagement of enemy targets.

We now formally establish how targets are allocated. Let \mathcal{E} denote the set of enemy entities listed on a global target priority list, accessible to all friendly entities, and let \mathcal{F} denote the set of friendly entities. Let $x_{i,j}$ denote the zero–one decision variable which distinguishes whether or not the enemy entity $i \in \mathcal{E}$ is a allocated target of the friendly entity $j \in \mathcal{F}$. That is, $x_{i,j} = 1$, iff the enemy entity i is allocated as a target of the friendly entity j , and $x_{i,j} = 0$ otherwise.

To restrict the number of times an enemy entity may be allocated as a target by friendly entities to one, we require that

$$\sum_{j \in \mathcal{F}} x_{i,j} \leq 1, \quad \forall i \in \mathcal{E}. \quad (1)$$

To restrict the number of enemy targets a friendly entity may be allocated, we require that

$$\sum_{i \in \mathcal{E}} x_{i,j} \leq 1, \quad \forall j \in \mathcal{F}, \quad (2)$$

Let $w_{i,j}$ denote the benefit in the enemy entity $i \in \mathcal{E}$ being targeted by the friendly entity $j \in \mathcal{F}$. This benefit is given by

$$w_{i,j} = \frac{(p_i - b_i + \max_k b_k)}{(1 + d_{i,j})}, \quad (3)$$

where p_i denotes the priority ranking of i from the global target priority list, b_i denotes the operational time of birth measured in simulated time and $d_{i,j}$ denotes the Euclidean two-dimensional distance between i and j measured in grids. The effect that the operational time of birth $b_i, i \in \mathcal{E}$ has in equation (3) is interpreted as a tendency to emphasise enemy entities who have remained operational for a comparatively long time. The term $\max_k b_k$ ensures that equation (3) is always non-negative.

Given the values $x_{i,j}, i \in \mathcal{E}, j \in \mathcal{F}$,

$$\sum_{i \in \mathcal{E}} \sum_{j \in \mathcal{F}} w_{i,j} x_{i,j}, \quad (4)$$

is the collective benefit or fitness function.

An optimal target allocation is obtained as the solution to the integer linear maximization program

$$\begin{aligned} \max \quad & \sum_{i \in \mathcal{E}} \sum_{j \in \mathcal{F}} w_{i,j} x_{i,j}, \\ \text{such that} \quad & \sum_{j \in \mathcal{F}} x_{i,j} \leq 1, \quad \forall i \in \mathcal{E}, \\ & \sum_{i \in \mathcal{E}} x_{i,j} \leq 1, \quad \forall j \in \mathcal{F}, \\ & x_{i,j} \in \{0, 1\}. \end{aligned} \quad (5)$$

The integer linear program (5) can be solved using the Simplex method, developed by Dantzig (1990), because it satisfies Hoffman and Kruskal's (1956) Uni-modularity theorem.

Note that this algorithm is applicable irrespective of whether or not there are more enemy targets than friendly entities, less enemy targets than friendly entities, or no enemy targets at all. Unmatched entities are not allocated a target.

2.6 Blue Force Behaviour - Type Swarm only

Blue force entities of behaviour type *Swarm* are allocated targets according to the rule below.

- Blue force entities j of behaviour type *Swarm* are allocated targets according to

$$\arg \max_i w_{i,j}, \quad (6)$$

breaking ties arbitrarily, where $w_{i,j}$ is defined in equation (3).

Note that both the *Swarm* and *Borg* models, use equation (3) as a benefit or utility function. However, the allocation of fire in the *Borg* model is optimised globally across the entire system where as the allocation of fire, according to equation (6), in the *Swarm* model is optimised for each individual blue force entity independent of any other blue force entity. That is, the *Swarm* model is a system of distributed and decentralised control.

2.7 Instantiation

Twenty entities are instantiated at time $t = 0$. Half of these entities are assigned to the blue force and the remaining entities are assigned to the red force. The (x, y) -coordinate grid references for blue entities are randomly generated from a uniform distribution over the bottom most grid coordinates. That is, $x \in (1, 1000)$, $y \in (1, 500]$. The (x, y) -coordinate grid references for red entities are randomly generated from a uniform distribution over the entire map. When generating entities, red or blue, care is taken not to place them in impassable terrain. The operational times of birth for all blue entities are defined to be $t = 0$. The operational times of birth for all red entities are randomly generated according to a uniform distribution over $[0, 99] \subset \mathbb{N}$. The priority of all red entities is set to 500. We terminate simulations after all red force entities are destroyed.

3. Empirical Results

In this section, we conduct $N = 100$ independent simulations of both the *Swarm* and *Borg* models and measure the response of the system as a whole in each. In these simulations, the statistic we measure is given by the sum, over all red force entities, of the time elapsing between the respective entities operational-time-of-birth and time-of-death, or the time at which the simulation ends in the case that the entity is not destroyed.

Let \mathcal{R} denote the response statistic, let b denote the ordered list that records the operational-times-of-birth for red entities, and let d denote the ordered list that records either the epochs at which red entities die or the number of iterations of the simulation if the entities do not die. Then, the k^{th} response statistic for a simulation of behaviour type $\mathcal{T} \in \{\text{Swarm}, \text{Borg}\}$ is

$$\mathcal{R}_{\mathcal{T}}^k = \sum_i d_i - b_i. \quad (7)$$

Over $N = 100$ independent trials, for some fixed \mathcal{T} , then the sample mean response statistic $\bar{\mathcal{R}}_{\mathcal{T}}$ over these N replications is

$$\bar{\mathcal{R}}_{\mathcal{T}} = \frac{1}{N} \sum_{k=1}^N \mathcal{R}_{\mathcal{T}}^k. \quad (8)$$

The sample variance $s_{\mathcal{T}}^2$ in the response statistic over N independent trials is

$$s_{\mathcal{T}}^2 = \frac{1}{N-1} \sum_{k=1}^N (\mathcal{R}_{\mathcal{T}}^k - \bar{\mathcal{R}}_{\mathcal{T}})^2. \quad (9)$$

We conducted $N = 100$ independent trials of both type *Swarm* and type *Borg* behaviour. For brevity, the raw simulation results are not provided in this paper. Values for $\bar{\mathcal{R}}_{\mathcal{T}}$ and $s_{\mathcal{T}}^2$ are given in Table 5.

Table 5. Sample means $\bar{\mathcal{R}}_{\mathcal{T}}$ and standard deviations $s_{\mathcal{T}}$

	$\bar{\mathcal{R}}_{\mathcal{T}}$	$s_{\mathcal{T}}$
<i>Swarm</i> model	504.2	179.3
<i>Borg</i> model	457.2	165.0

Let μ_{Swarm} and μ_{Borg} represent the true mean response statistics respectively. Then, to statistically test the hypothesis $H_0 : \mu_{\text{Swarm}} = \mu_{\text{Borg}}$ with alternative hypothesis $H_a : \mu_{\text{Borg}} <$

μ_{Swarm} , we use a two sample t -test with $N - 1 = 99$ degrees of freedom. A t -value statistic of 1.93 is calculated. This value corresponds to a probability value p in the range $(0.025, 0.05)$. This means that our null hypothesis H_0 is rejected in favor of the alternative hypothesis H_a at a significance level of $\alpha = 0.05$ but is retained at a significance level of $\alpha = 0.025$. Hence, in rejecting the null hypothesis H_0 at a significance level of $\alpha = 0.05$ we are 95% confident that H_0 is false and that the true mean response statistic for type *Swarm* behaviour exceeds the true true mean response statistic for type *Borg* behaviour. A further set of one hundred simulations is conducted. A t -test over all 200 simulations indicates that the null hypothesis is rejected in favor of H_a at the value of $\alpha = 0.001$.

Our results indicate a significant difference in the operational performance of the two models. That is, the *Borg* model responds to threats in the battlefield faster than the *Swarm* as its response statistic is lower. The implications of this result are that the collective control mechanism for beyond line-of-sight weapons platforms is an efficient use of resources and is also an effective way to fight a battle. In contrast to the *Borg* behaviour, the *Swarm* behaviour is inefficient because the deployment of the weapons platforms is not optimised and the behaviour is ineffective because its response statistic is low.

To explore the robustness of the two models and to demonstrate how easily the behaviours of the blue entities may potentially be exploited by the red force we conducted a second set of experiments. These experiments were designed to test the limitations of the *Swarm* and *Borg* models. These new simulations were conducted identically to those described in Section 2. with one exception. A single decoy entity, belonging to the red force with priority 1000, was initialised at time $t = 100$ at the top left corner of the map. The instant that the blue force destroyed this entity, a second decoy was initialised at the bottom right corner of the map.

Table 6. Sample means $\bar{\mathcal{R}}_T$ and standard deviations s_T

	$\bar{\mathcal{R}}_T$	s_T
<i>Swarm</i> model	681.6	471.2
<i>Borg</i> model	425.6	150.6

In simulations involving a decoy entity, the *Swarm* model displayed substantial degeneration in performance. The majority of blue entities were attracted to the decoy in preference to other targets. In simulations of the *Borg* model, only a single blue entity was deployed to engage the decoy. Hence, in these simulations the differences in the response statistics are negligible. From Table 6, a t -value statistic of 5.18 is calculated. The null hypothesis H_0 is rejected in favor of the alternative hypothesis H_a at a significance level approaching zero.

The evidence supporting the superiority of behaviour type *Borg* over behaviour type *Swarm* in the second set of simulations is overwhelming. These results indicate a catastrophic failure of the *Swarm* model. No method has yet been found to undermine the *Borg* behaviour to the same extent. However, it should be noted that it is impossible to conclusively discount the existence of such a method. Further experiments are necessary to fully understand the benefits of decentralised control, if any. The scenario developed in this study is based upon realistic and practical limitations to the number of weapons platforms and the characteristics of those platforms. These limitations evolve with time. Hence, further studies of this nature are important to consider the impact of future technology on the Command-and-Control of beyond line-of-sight weapons platforms.

4. Conclusion

This document is devoted to a simple investigation of distributed and collective control structures for beyond line-of-sight weapons platforms using an agent-based methodology. A future study is proposed in which the complexity of the model is increased by including a number of additional features such as: ammunition type, collateral damage, formations, logistics and terrain effects. These features will increase the fidelity of the model and the richness of the insights gained through the simulations.

Investigation of the potential use for optimisation algorithms embedded within the behaviours of entities in agent-based models based on cellular automata is a particularly interesting area of research. Simple linear programs such as presented in this study are comparatively easy to modify to suit a wide range of applications. Furthermore, it is not difficult to envisage more advanced techniques from non-linear programming being used. This leads us to ask at what point are agent-based models based on cellular automata interpreted merely as simulations of other well defined mathematical constructs such as linear programs. In this sense, agent-based models lose much of their usefulness because there exist a wide range of alternative techniques superior to or equally well suited to the simulation of mathematically defined systems.

This study presents an argument for the superiority of the *Borg* behaviour over the *Swarm* behaviour. However, it should be noted that the *Borg* model is optimised for the allocation of firepower only and is not guided by the effect that the weapons fire produces on the enemy. Furthermore, this simulation discounts the firepower of the red force because it is primarily concerned with measuring only the response times for the two models. Consider the following contrived example. A pitched battle between five red entities and five blue entities occurs. The five red entities focus their firepower on a single blue entity while the blue force distribute their firepower so that each of the blue force engage a unique target. Let us make the simple assumption that it takes exactly five rounds to destroy an entity. Then, in the initial barrage of combat one blue force entity is destroyed while each of the red force sustains a single rounds worth of damage. In the second barrage of combat the red force again engage and destroy one of the blue force. The red force is now in the position that four of the guns have sustained two rounds worth of damage while one of the guns has sustained a single rounds damage. It is easy to predict the result of this battle. At the end of combat all the blue entities are destroyed for the loss of a single red entity. The remaining red entities have sustained one, two, three and four rounds damage respectively. Hence, the optimised behaviour of the *Borg* model is not the panacea for the Command-and-Control of beyond line-of-sight weapons platforms. Furthermore, it could be argued that the example provided above neglects to take into account any suppression effect on the entire red force from the bombardment blue delivers in the first barrage. This study lays down the foundations for further work in this field which could potentially address these issues.

Acknowledgements

Thanks to Alex Ryan, Daphne Sands and Nikoleta Tomecko, DSTO Ediburgh, for their invaluable assistance with this study.

References

Ahuja, R. K., Magnanti, T. L., and Orlin, J. B. (1993). *Network Flows. Theory, Algorithms, and Applications*. Prentice-Hall, Upper Saddle River, New Jersey.

- Australian Army (2002). *Concept for Manoeuvre Operations in the Littoral Environment (MOLE)*, 2002 Edition, Army's Contribution to Joint Warfighting. Australian Army, Future Land Warfare Branch. Unrestricted.
- Dantzig, G. B. (1990). Origins of the simplex method. In Nash, S. G., editor, *A history of scientific computing*, pages 141–151. ACM Press, Reading, MA.
- Hoffman, A. and Kruskal, J. (1956). Integral boundary points of convex polyhedra. In Kuhn, H. and Tucker, A., editors, *Linear Inequalities and Related Systems*, pages 223–246. Princeton University Press, Princeton, NJ.
- Ilichinski, A. (1997a). *Irreducible Semi-Autonomous Adaptive Combat (ISSAC): An Artificial-Life Approach to Land Warfare*. CNA Research Memorandum 97-61.10. Center for Naval Analysis, Alexandria, VA 22302-1498.
- Ilichinski, A. (1997b). *Irreducible Semi-Autonomous Adaptive Combat (ISSAC): An Artificial-Life Approach to Land Combat*, White paper. Center for Naval Analysis, Alexandria, VA 22302-1498.
- Jane's (2002). *Procurement*, Jane's Sentinel Security Assessment - Oceania, Australia 2002, 2.
- Lauren, M. K., Stephen, R. T., and Anderson, M. A. (2001). *MANA Version 1.0 Users Manual*. DTA Technical Note 01/4, NR 1365, ISSN 1175-6608. Defence Technology Agency, New Zealand.
- Millikan, J., Brennan, M., and White, G. (1996). *BactoWars Project Brief*, Draft version. Defence Science and Technology Organisation, Land Operations Division, Edinburgh, Australia. Unclassified.
- Sands, D. G., Tomecko, N., Wheeler, S., Ryan, A., Kempt, N., and Judd, G. (2003). Alternative force structures for offensive support and their impact on command and control in a networked environment. In Puri, V., Filippidis, D., Quinn, S., and Kelly, J., editors, *Proceedings of the Land Warfare Conference 2003*, pages 649–658. Adelaide Convention Center, Defence Science and Technology Organisation.
- Wheeler, S. and White, G. (2004). *The use of agent based distillations for the modelling of offensive support elements*. DSTO Research Report Series. Defence Science and Technology Organisation, DSTO Edinburgh, Australia. Unclassified.