páscoaAncient and Medieval Battles Conceptualization and Simulation*

Pedro Moraes Vaz

Instituto Superior Técnico e INESC-ID
pedromvaz@gmail.com

Pedro A. Santos

Instituto Superior Técnico
pedro.santos@ist.utl.pt

Rui Prada

Instituto Superior Técnico e INESC-ID
rui.prada@ist.utl.pt

Resumo

Este trabalho apresenta um modelo do desenrolar de uma batalha da antiguidade ou medieval, e de como esta era entendida do ponto de vista do comandante do exército. Com base no modelo foi criada uma plataforma de simulação que permite a jogadores tomarem o papel de um general em batalhas da antiguidade ou medievais. A plataforma permite aos jogadores criarem planos tácticos que descrevem como os seus exércitos vão ser colocados no campo de batalha, como se vão comportar, e depois permite que os jogadores vejam as batalhas nas quais os seus exércitos participaram. Existe uma separação conceptual entre o plano de batalha e as forças à disposição do general.

Palavras-chave: Batalha; Medieval; Antiga; Táctica; Simulação.

Abstract

This work presents a model of ancient and medieval battles from the armies' commander's perspective. This model was used to create a simulation platform that allows players to play the role of a general in ancient and medieval battles. Players, as generals, create tactical plans describing how their armies will be placed in the battlefield, how they will behave, and then they can watch the battles in which their armies participated. There is a conceptual separation between the battle plan and the actual forces at the general disposition.

Keywords: *Battle; Medieval; Ancient; Tactics; Simulation.*

* The authors wish to thank Cor. Rui Moura, who read an earlier version of the manuscript and gave many suggestions for its improvement.

1. Introduction

A battle is a direct conflict between large tactical units that have been committed as part of a major operation (Leonhard, 1991). Western tradition military history is fascinated with Battles (Tarkov, 1989), and the pursuit of the "decisive battle" that will end a protracted conflict. That tradition can be traced back to the Greek city-states, which tried to settle their affairs in a single battle between their citizens. Notwithstanding that fascination, and the thousands of works written describing battles of the ancient and medieval world, the conceptualization of the pre-machinegun battle has, to the authors' knowledge, not been tried. One reason for this is that technological advances in armament (machinegun and automatic small-arms fire, artillery, air power, communications etc) changed considerably the modern battlefield and the current military thinkers turned to the problems raised by the new technologies.

Ancient and Medieval battles study has, nevertheless, kept the interest of historians and general public, as the success of book collections like Osphrey Publishing military history series and many others testify. Also many games (both videogames and board games) have been made trying to reproduce or simulate ancient and medieval battles. A characteristic they all share is that they concentrate on placing forces on the terrain and commanding the specific units at their disposition for specific battle segments. It is possible to divide these games into three categories: games where players can control their soldiers during the battle by moving them around and issuing orders, games where players can choose where to place their soldiers and issue one order each before the battle starts, having few or no choices during that battle, and games where players can only decide what types of soldiers they have in their armies, and how many, while having no control during the battle, because it is processed automatically.

What our research uncovered is that it is possible to create an additional level of abstraction and create a platform where players can create their own tactical plans independently (i.e. without attaching them to a specific army), and then use them in any battle, with whatever army is available. These tactical plans define where to place specific categories of soldiers (e.g. Infantry, Cavalry and Artillery) in the formation, where to place the strongest or weakest, and which orders they would follow during the battle. Apart from the tactical

plans, players have no control over a battle. It is then necessary to base the battle simulation on a Multiagent System (MAS) and rely on autonomous agents to control each contingent (i.e. group of soldiers).

Before creating this platform, dozens of historical battles were analyzed, and simulated using a board game system (Berg et al., 1991-2010). The authors also analyzed seven different games or game systems to understand what kinds of choices players had in order to control their armies in the battlefield, before and during the battle.

The main contributions of the present work are then the conceptualization of an ancient or medieval battle, the study of a possible computer interface to create battle plans, and a MAS-based simulator to implement battle plans and simulate battles.

2. Conceptualization of an ancient or medieval Battle

This conceptual model is based on accounts and analysis of historical battles and commander's decisions (Devries et al., 2006), (Healy, 1994), (Keegan, 1976, 1987), (Warry, 1991). A **Battle** is a direct conflict between two armies, each lead by a different general.

The **General** is the man who commands his army into battle, who places his soldiers in the battlefield and tells them what orders to follow. Generals could be political leaders (Kings, Emperors), or appointed to the office by the political leadership. Independently of their origin, we assume they will make the decisions regarding the battle plan. Some generals would fight together with their men, while others would stay behind, watching the battle unfold and signaling when new orders were to be executed, by using messegers or pre-planned signals like waving a flag or sounding a horn. It is important to notice that changing orders orders during a battle was hard, because messengers were very inefficient, due to the time it took for the information to reach the commander and back to a distant unit, and the possibility of distortion of the information or order. Thus, the placing of the troops and the orders they were to follow would have to be planned before the actual battle took place. Pre-planned signals would be used to tell subordinate leaders when to execute some of those orders, in case something happened during the battle that the general was expecting.

An **Army** is a collection of contingents fighting on the same side, usually lead by a single general. It was common in many historical battles for various allied armies to fight together,

each one lead by a different general, without a common command. For simplification purposes, we will assume the existence of only one army and one general on each side. A **Contingent** is a group of soldiers who perform similar roles: they carry the same weapon and armor, they may walk or ride a horse, and they train and fight together, which gives them a similar battle experience. We divided contingents into five **Categories**:

3. Melee Infantry, composed of soldiers who walk and wield a hand-to-hand weapon like a sword, spear, pike, axe or mace;
4. Ranged Infantry, composed of soldiers who walk and carry a long-range weapon like a shortbow, longbow, crossbow or javelin;
5. Melee Cavalry, horse-riding soldiers wielding a hand-to-hand weapon;
6. Ranged Cavalry, composed of soldiers who ride a horse and carry a long-range weapon;
7. Artillery, composed of heavy machinery like cannons, catapults or trebuchets.

It is important to notice that, while some soldiers might be focused on melee or ranged combat, they might be carrying another weapon. Archers would carry a sword in case the enemy reached their lines, although they were not experts in its use.

The way the general (i.e. the player in the simulation) commands his armies is by creating tactical plans (or **Tactics)** that can be used in any battle and with any army. The general defines the army's **Formation** in each tactical plan, which determines where any available contingents will be placed in the battlefield. A formation is divided into **Sectors** (Center, Left Flank and Right Flank), each having one or more **Sector Lines**. Each sector line is composed of various **Slots** so that, for instance, Melee Infantry can be placed besides Ranged Infantry. Each **Slot** can only be assigned a contingent category, but each sector line can have multiple categories, hence the use of slots to enable that option. Each slot may be assigned a contingent category and a **Rank**. Ranks allow the general (i.e. the player) to determine where to place the strongest or the weakest contingents, inside the formation.

One important point is that formations in the Tactical Plan are in principle independent from any specific army. In this way, abstract tactical plans (e.g.: "Advance in Echelon with a refused right flank where the weaker infantry is", "Form the army in two lines with lighter infantry in the center of the front line, which will slowly give ground, and strong cavalry on

the flanks, to lure and trap an advancing army") can be included in the concept, and applied on several variations of actual army composition.

Each sector line may be assigned one or more **Orders**. Orders are associated to Battle **Events**, and are only executed when those events occur. Each order given to a sector line will have an event associated to it. It is possible to give different orders to the same sector line and each one will be associated to a different event that may occur during a battle. Events are divided into two categories: **Global** events, which trigger when that event happens anywhere in the battlefield, and **Local** events, which trigger when that event happens to a specific sector in the battlefield, friend or foe. Both global and local events are necessary in the model, because sectors need to be able to react to events that concern the whole army, or just a part of it. Events represent the use the pre-planned signals mentioned at the beginning of this section.

Every Event-Order pair is called a **Rule**, and each sector line may have up to five rules with different **Priorities**. Rules with a higher priority will be executed first.

Five global events were considered:

1. Battle : Start, which triggers as soon as the battle starts
2. Battle : Ranged Phase, which triggers as soon as one of the contingents (on any side) starts shooting arrows at an enemy contingent
3. Battle : Melee Phase, which triggers as soon as one of the contingents (on any side) shocks against an enemy contingent, engaging in melee combat after that shock
4. Battle : Enemy Retreat, which happens when the first enemy contingent retreats from the battle
5. Battle : Friendly Retreat, which happens when the first friendly contingent retreats from the battle

Local events represent the events in the battle that are directly observable by the troops in a certain location on the battlefield, like if they anybody in the vicinity is under fire or being attacked, if their flanks are covered, etc. The Conceptual Map is shown in Figure 1.

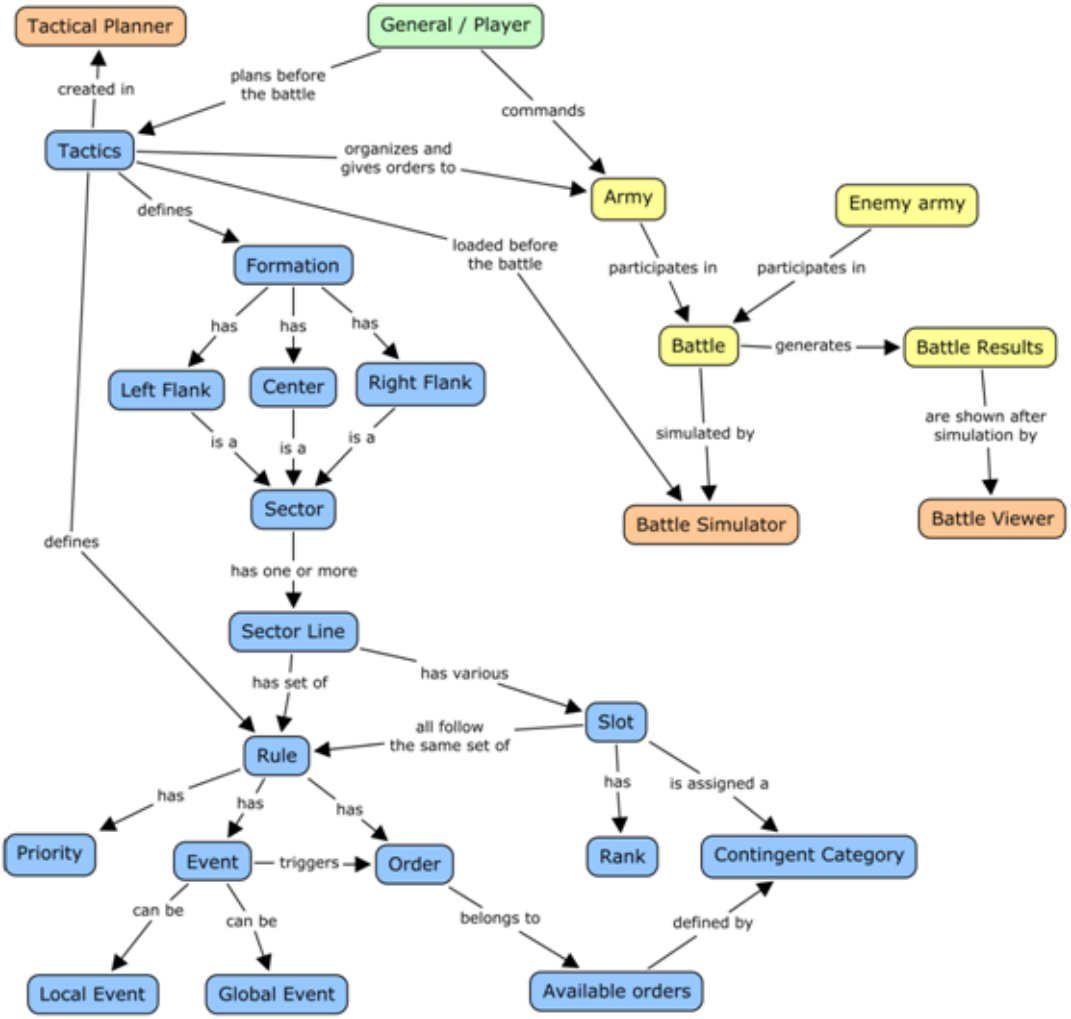


Figure 1 - ***The Conceptual Map***

3. Simulation Platform Architecture

Figure 1 presents the simulation platform's architecture, to give the reader an idea of its components and how they are connected to each other. Figures 3 and 4 show the two Graphical User Interfaces (or GUIs) that we designed to help players use our platform.

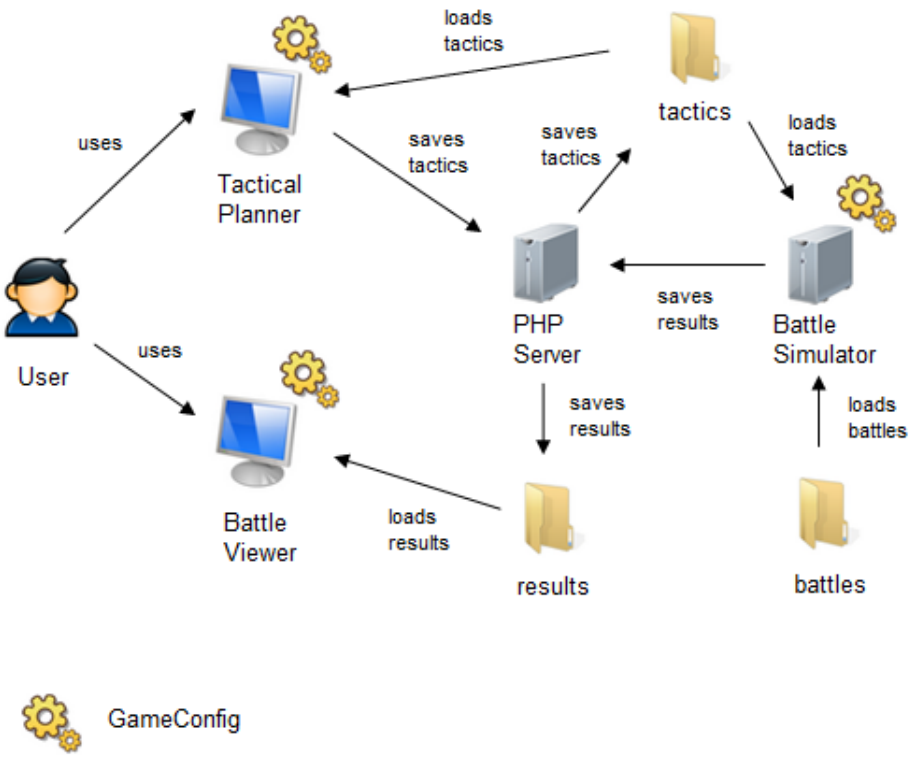


Figure 2 - ***The Platform architecture***

The Tactical Planner is a GUI that allows players to create and save new tactical plans for their armies, as well as load existing ones. We created various prototypes until we concluded that our players could create rich and varied tactical plans with ease. The Battle Viewer is a GUI that allows players to view battles, in which their armies participated, that have already taken place (i.e. that have already been simulated). The Battle Simulator simulates every battle between two armies, and then stores it so players can watch it afterwards. Each army has a specific tactical plan in that battle, which the Simulator must load. The GameConfig loads all the game configurations. This way, we can change the game elements (weapons, contingent categories, contingent types, damage types) in one place and it affects the whole platform.

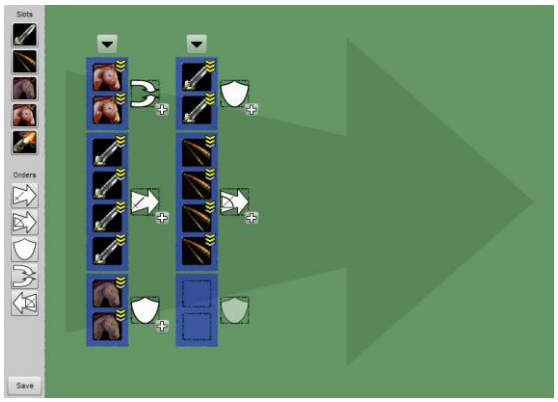


*Figure 3 - **The Platform architecture***

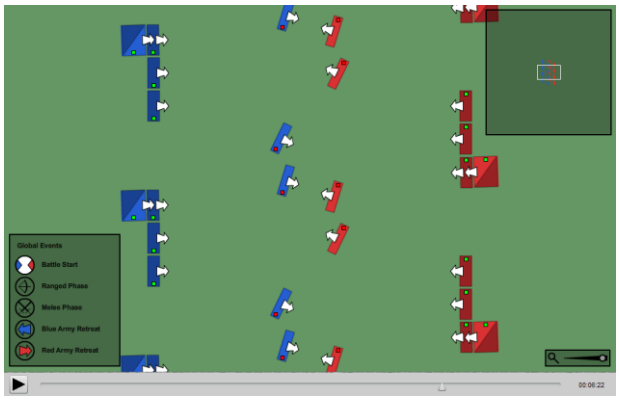


*Figure 4 - **The Platform architecture***

The PHP Server is needed because this whole platform was developed in Adobe Flash, and storing any information created in Flash requires connecting to a server, which then handles the necessary files.

4. Orders and Events

Currently, there are six Orders and five Global Events implemented in the platform.

The orders are as follows:

- **Defend** - agents stay in their position and engage the enemy when in range
- **Melee Attack** - agents move towards the enemy and engage in melee combat
- **Ranged Attack** - agents fire if in range, or move towards the enemy and engage in ranged combat
- **Skirmish** - agents engage the enemy in ranged combat using a loose formation, and keep a safe distance
- **Flank Attack** - agents move around the battlefield and attack the enemy from its side
- **Retreat** - agents run away from the battlefield

The global events are the following:

- **Battle Start** - triggered when the battle starts
- **Ranged Phase** - triggered when the first arrow is fired
- **Melee Phase** - triggered on the first shock between contingents
- **Friendly Retreat** - triggered when a friendly contingent runs away

- **Enemy Retreat** - triggered when an enemy contingent runs away

The interface for orders implemented in the battle planner is shown on Figure 5.

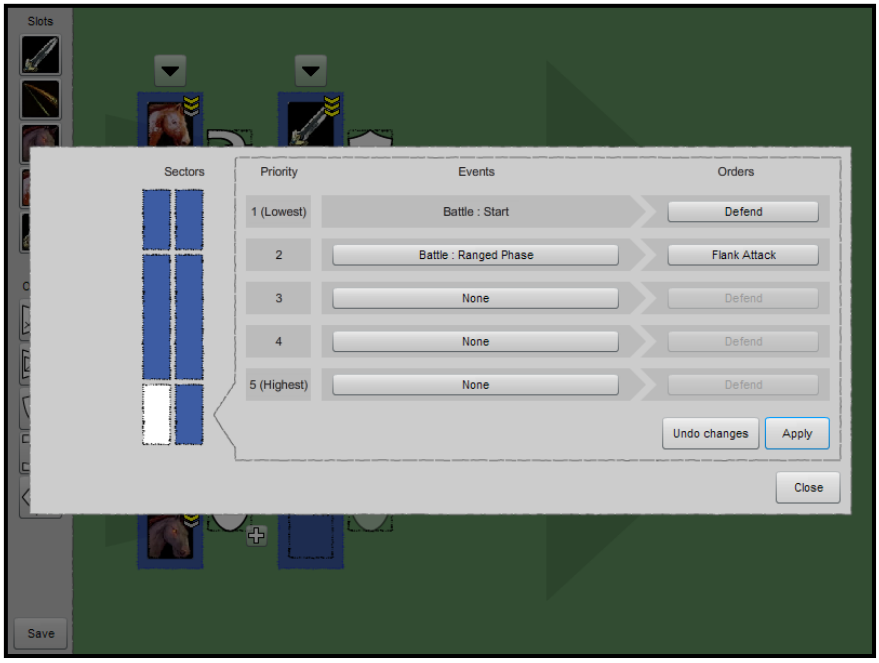


Figure 5 – ***The orders interface***

5. Battle Simulator

In order for a battle to be simulated, the Battle Simulator must load three files: the one containing information about every battle contingent on both sides and two tactical plans (one for each side). After the files are loaded, the battle contingents are placed on the battlefield according to the formation in each tactical plan.

The battlefield is the environment in which the agents (in this case, the contingents) live and interact. Currently, this environment is as simple as a static grassy plain (of approximately 5.4 x 5.4 kilometers) where the contingents are placed and follow their orders.

The reason why we limit the dimensions of the battlefield is due to retreating contingents (contingents that are running away from the battlefield). In real battles, the winning side would sometimes pursue its enemy to exploit the success, either making prisoners or slaughtering the enemy; in some cases, the losing side would manage to escape unharmed.

We want to give the winning side a chance to catch enemy contingents that are retreating, so we made the battlefield big enough for the contingents to take a while to escape. If a retreating contingent does manage to reach the border of the battlefield, it survives the battle but is considered as “retreated”, which means it cannot participate in the current battle anymore.

The simulator is a multi-agent system that runs, in cycles, a set of agents (from both sides), each representing a contingent, and gives them a chance to analyze the environment (i.e. the battlefield) and decide what to do according to the orders assigned to them in the tactical plan. The battle state - consisting of the contingents’ positions, actions, targets and status - is stored after each step of the cycle, for later viewing on the Battle Viewer.

A battle ends when all the contingents from one side are either destroyed or outside the battlefield. When the battle ends, the simulator stops and the battle is stored.

6. Placing the contingents on the battlefield

Since the tactical plans are independent from any army, it was necessary to develop an algorithm to place any specific army on the battlefield, according to the formation defined in its tactical plan for that battle. This algorithm consisted of three steps: distributing all army contingents into the category-assigned slots in the formation, calculating how many blocks were needed for each slot, and aligning the slots on the battlefield.

A block is a 50x50 meter area where only one battle contingent can fit, and the battlefield is a grid of these blocks. When an army contingent is distributed into the category-assigned slots, it must then be split into one or more blocks in the battlefield, depending on its size, because blocks may contain a maximum number of soldiers. Each part of that now-split army contingent is called a Battle Contingent, and is controlled by an autonomous agent during the battle.

Consider an army with 2000 Archers, 3000 Skirmishers and 6000 Swordsmen, and a formation where the slots in both flanks are assigned the Ranged Infantry category. The slots in the left flank have the highest rank, and the ones on the right flank have the lowest rank. Finally, the slots in the center are assigned the Melee Infantry category.

The first step is to group all the army contingents by categories: this means we will group the Archers and the Skirmishers as Ranged Infantry and the Swordsmen as Melee Infantry. The same grouping will be applied to the slots in the formation, so the slots in the flanks will be grouped as Ranged Infantry, and the slots in the center as Melee Infantry.

The second step is to order the army contingents in each group from strongest to weakest and order the slots in each group from highest rank to lowest rank. For this example, we will consider that the Archers are stronger than the Skirmishers. The Ranged Infantry slots in the left flank will come first (because they have the highest rank) and the slots on the right flank will come last (because they have the lowest rank).

The third sub-step is to equally distribute the army contingents from each category into the slots of the same category, in their current order, so that each slot ends with roughly the same number of soldiers assigned. Figure 5 shows the distribution for the Ranged Infantry category.

Slots	Left Flank Slot 1 (Rank 3)	Left Flank Slot 2 (Rank 3)	Right Flank Slot 1 (Rank 2)	Right Flank Slot 2 (Rank 2)
Distribution	1 250 Archers	750 Archers + 500 Skirmishers	1 250 Skirmishers	1 250 Skirmishers
Blocks (Battle Contingents)	2 blocks of 625 Archers	1 block of 750 Archers + 1 block of 500 Skirmishers	2 blocks of 625 Skirmishers	2 blocks of 625 Skirmishers

Figure 6 – ***Distribution of 2000 Archers and 3000 Skirmishers into four Ranged Infantry slots***

Consider that each block on the battlefield can have a maximum of 1000 soldiers. Figure 5 shows how many blocks each slot needs, and how many soldiers stay in each block. Since the slots in the tactical plan are shown side by side, the blocks in the battlefield will be side by side as well, so the formation in the battlefield remains faithful to the tactical plan. In the end, the formation in the battlefield will look like this (from left to right): 2 blocks of 625

archers, 1 block of 750 archers, 1 block of 500 skirmishers, 8 blocks of 750 swordsmen, and 4 blocks of 625 skirmishers.

7. Agents

As stated before, the simulation is based on agents, one agent for each battle contingent. We used state-based agents (Wooldridge, 2002), because it was necessary for them to know their status, which orders they were assigned and what they were doing at any given moment, in order to have a coherent behavior as the battle progressed.

Every battle contingent has a list of attributes that define its role on the battlefield, and how well it will do in battle: the number of soldiers in it, their speed, combat experience, cohesion (or morale), which weapons they use (and their damage), armor and tiredness.

Agents in a game usually act in one of a limited set of ways and they will carry on doing the same thing until some event makes them change. To achieve such behavior we implemented the agents' decision mechanism as State Machines (Millington, 2006) as shown in Figure 6.

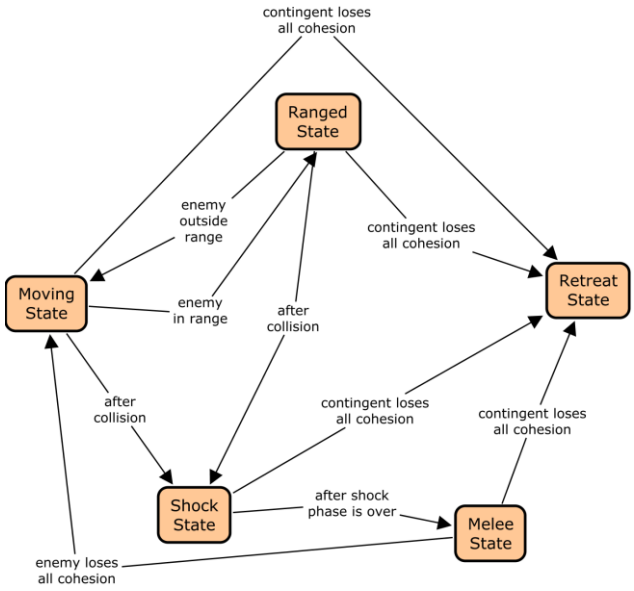


Figure 7 – ***The Agent's State Machine***

When a battle contingent is placed inside a block in the battlefield and assigned an agent to take control over its actions during the battle, that agent is given the set of Rules that the player assigned to the sector line where that battle contingent was placed. These Rules will

define the agent's behavior during the battle. Agents start with their state machine in the Moving State.

The orders that players can give their contingents (in their tactical plans) only affect their movement (i.e. their Moving State). Players do not order their contingents when to engage the enemy in ranged or melee combat, they simply tell them to execute a frontal attack (melee or ranged), a flank attack or a rear attack, for instance, which consist of telling them from which direction to attack the enemy. The "Melee Attack" order will make the contingent move towards its nearest target, the "Flank Attack" order will make the contingent move around the battlefield to attack the enemy from its flank, and so on. The moment when they actually attack the enemy depends on when they get in range for it, which the player has no control of.

Apart from the "Battle Start" event, all the other events will be triggered by the agents in the battlefield, whether they are in their ranged state, their melee state or retreating. When those events occur, agents may be assigned a new order (according to their tactical plan) and their movement may change due to that new order. This means their Moving State will tell them to move to a new point in the battlefield, because their order changed.

If an agent is in any state except the Moving State and a new event occurs, the order associated to that event will have no effect on the agent's current action -- meaning the agent will not change its current state -- until the agent returns to its Moving State because, as was said, orders only affect the agent's Moving State.

The exception is the "Retreat" order, which is of the highest priority, because the player wants to rescue the remaining soldiers in the battle contingent. When a contingent retreats, either by the player's order or because it lost its cohesion, the player loses control over it, meaning the contingent will no longer follow the orders assigned to it in the tactical plan.

Every agent needs to avoid colliding against friendly contingents when moving, because contingents in ancient and medieval battles were in a tight formation (soldiers were close to each other), which left no space for another contingent to move through it. Also, friendly contingents keep a distance from each other, so they can reach their target without being blocked. The only exceptions are when a contingent is skirmishing (moving away from its

target) or retreating from the battlefield, because in these situations the contingent is in a loose formation, meaning its soldiers have some space between them.

When an agent chooses a target to engage, it takes various parameters into consideration: the distance to the target, if the target is obstructed by other agents (i.e. contingents), the angle of rotation needed to face the target, if the target is retreating from the battle, if the target is already attacking the agent, and if it was the agent's last chosen target.

There are two types of damage in combat: physical damage and cohesion (or morale) damage. Any form of physical damage inflicted on a contingent also inflicts cohesion damage. For instance, when two contingents are in melee (i.e. hand-to-hand) combat, soldiers will die in the process, and the remaining soldiers will start losing their will to fight, which affects their cohesion. When a contingent is retreating from the battlefield, it inflicts cohesion damage to nearby friendly contingents, because seeing a contingent flee will cause some panic in the remaining contingents.

Agents can react to the events that occur during a battle. Consider Figure 7, where an agent is assigned three Rules to follow. When the battle starts, the agent will defend its position, due to the first rule. The second and third rules do not apply, because the ranged phase has not started and no one has retreated. When the ranged phase starts, the agent will start a melee attack, because the second rule can now be executed and it has a higher priority than the first. Finally, when a friendly contingent retreats, the agent will retreat as well, because of rule 3, which is of the highest priority.

Priority	Event	Order
1	Battle Start	Defend
2	Ranged Phase	Melee Attack
3	Friendly Retreat	Retreat

*Figure 8 – **Three rules assigned to an agent***

The Battle Simulator produces a log file that captures the position and status of each contingent on the battlefield. This log file can then be fed into the Battle Viewer for analysis and feedback to the players, which can thus observe the result of their plans (see figure 8).

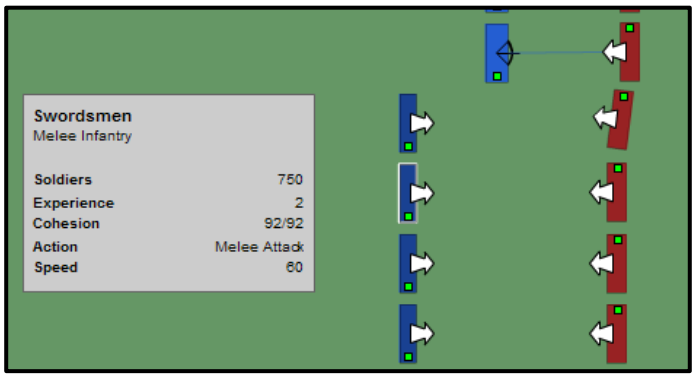


Figure 9 – ***A detail of a Battle seen on the Battle Viewer***

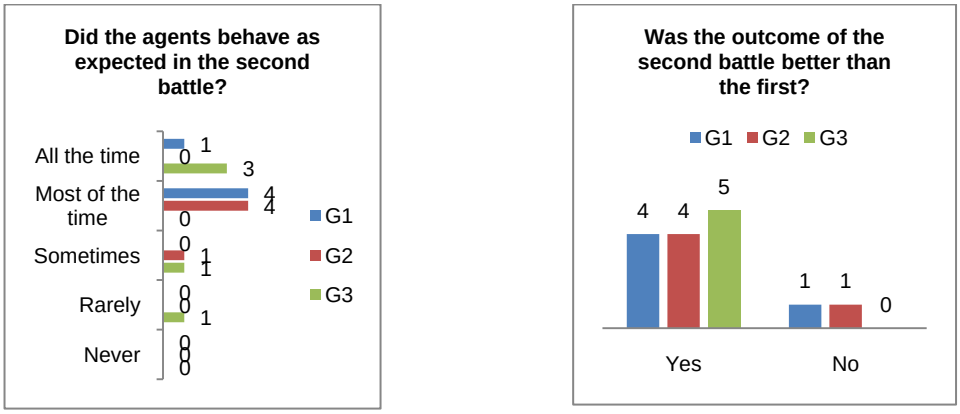
8. Results

After completing the framework, we tested it with fifteen subjects with ages ranging from 25 to 46. All the subjects were observed during the tests and filled a survey in the end, a usual playtesting process explained in (Schell, 2008). The subjects were divided into three groups: the first group was made of five inexperienced players in this genre, who were given no prior information about the framework, the second group was made of five experienced players in this genre, who were also given no prior information about the framework, and the last group was made of five inexperienced players who were told how the GUIs in the framework worked. This way we could compare the results from three different situations.

The test consisted of showing each subject a pre-simulated battle where both sides used the same tactical plan and had similar armies. The subject's army - slightly smaller and weaker - lost the battle and, after watching it, each subject was asked to find a new tactical plan that he thought would defeat the enemy army.

From the Battle Viewer surveys (related to the first battle), we found out that three of the subjects from the first group had difficulties recognizing the various contingents on the battlefield (swordsmen, archers, etc), while the remaining subjects from the three groups did not. Only two subjects from the first group had difficulties understanding actions performed by the contingents during the battle. Thirteen found the agents' behaviors acceptable; one from the third group found them realistic, while and one from the first group found them unrealistic.

From the Tactical Planner surveys, we found out that eleven of the test subjects understood how every order worked in the battlefield (three from the first group and the second group, and five from the third group). Only one subject from the first group used the ranks available in the Tactical Planner, while the second group had three subjects who used it and the third group had another three. The first group had two subjects who used the global events, the second group had three and the third group had five. Still, we noticed that global events were not as straightforward to use as we hoped.



(a) (b)

*Figure 9 – **Results from the survey***

From the Battle Viewer surveys (related to the second battle), we found out that twelve of the subjects felt the agents executed their tactical plan most (or all) of the time (see Figure 9). Only two of the subjects (one from the first group and one from the second group) felt that the second battle did not have a better outcome than the first.

9. Conclusions

From the survey results, we concluded that the platform is easy for experienced players to use, since they are familiar with the concepts, but some of the less experienced subjects had some problems understanding and using the concepts involved. Therefore, our intention is to create tutorials to help new players learn how to use our platform, by explaining the concepts and showing some practical examples.

We planned on having global (army-related) and local (sector-related) events available in the platform, but only the global events were implemented, which significantly limited the

number of events our agents can react to in a battle. We are aiming to add local events in the future.

A feature that is absent from the simulator is the terrain where the battle is fought. Many ancient battles were fought on featureless terrain or where the terrain had no big influence on the plans or outcome (Marathon, Gaugamela, Cannae, Vercellae, Zama, Bouvines, Leignitz, Najera, Legnano, just to name a few). In fact, a commander who did not fear enemy numbers or cavalry would in general prefer a flat and featureless battlefield. In many other battles (Heraclea, Hastings or Pharsalus, for instance), a terrain feature like a river, or a hill was chosen by the defender to break the enemy impetus. Including the effect of terrain in the simulation would add another layer of depth, at the cost of less generic plans.

10. References

BERG, R., Herman, M. (1991-2010). Great Battles of History. GMT.

DEVRIES, K., Dougherty, M., Dickie, I., Jestice, P., & Jorgensen, C. (2006). Battles of the Medieval World. London: Amber Books.

HEALY, M. (1994) Cannae 216 BC - Hannibal smashes Rome's Army. Osphrey Publishing.

KEEGAN, J. (1976). The Face of Battle. London: Penguin Books Ltd.

KEEGAN, J. (1987). The Mask of Command. New York: Penguin Books Ltd.

LEONHARD, R. (1991). The Art of Maneuver. Presidio Press.

MILLINGTON, I. (2006). Artificial Intelligence for Games. San Francisco: Morgan Kaufmann Publishers.

PREECE, J. (2007). Interaction Design: beyond human-computer interaction. John Wiley & Sons.

SCHELL, J. (2008). The Art of Game Design: A Book of Lenses. Burlington: Morgan Kaufmann.

TARKOV, J. (1989). The Hoplite in Battle. MHQ I (Spr 1989): pp. 20-1

WARRY, J. (1991). Alexander 334–323 BC - Conquest of the Persian Empire. Osphrey Publishing.

WOOLDRIDGE, M. (2002). An Introduction to MultiAgent Systems. West Sussex: John Wiley & Sons Ltd.