An Agent-based Computational Model for the Battle of Trafalgar: A Comparison Between Analytical and Simulative Methods of Research

Giuseppe Trautteur, Raniero Virgilio
*Dept. of Physical Sciences, University of Naples "Federico II", Italy**
virgilio@dsf.unina.it

Abstract

A computational agent-based model is proposed to study the historical naval battle of Trafalgar. The model, implemented using the Swarm simulation system, allows a dynamical study of system evolution at a high level of detail. Results agree in a very strict way with historical data. A comparison between the computational model and Lanchester's analytical model is proposed. Lanchester's forecasts and analysis of the battle are in substantial contrast with results obtained by the computational model. Moreover, counterfactual experiments have also been performed in order to investigate the possibility that a different outcome of the battle might have occurred: Nelson's strategy turned out to be not only the winning strategy, but the safest as well. The proposed model appears to be a very flexible tool for a quantitative analysis of a conflict and an interesting conceptual framework for the general study of conflict resolution.

Keywords: CAS, agent-based simulation, SWARM, conflict resolution, combat analysis, Lanchester

1. Introduction

In the last years many studies on complexity have come up in order to find modern techniques to tackle the problems where complex phenomena appear. While new computational approaches to these problems have shown significant results, even from a foundational point of view, they are often considered funny intellectual games with a low scientific rank[8] or, at best, tools that can produce just a visualization -through a virtual reproduction- of the problem, not providing any explanation of it. Warfare analysis is typical of this perspective: conflicts between opposed factions can be studied as complex systems composed of many interacting elements, adapting themselves to a changing environment, and agent-based simulations represent a very powerful application in this context[10]. Yet, today most conflict analyses are still conducted using analytical models [5][7][9] based on Lanchester's set of differential equations [2], even if many authors have proved their inadequacy [3].

1.1. Lanchester's analytical model

Making just few and easy assumptions, F.W.Lanchester found a set of differential equations which he considered a formalization of the empirical military strategy. In the following equations:

$$\frac{dR(t)}{dt} = -aB(t); \frac{dB(t)}{dt} = -bR(t); \qquad \text{[I]}$$

R(t) and B(t) are the sizes of red and blue forces at time t, while coefficients a and b represent the effectiveness of each red and blue unit respectively. It is simple to prove that any solution of system [I] A(t), B(t), has the following property :

$$aA(t)^2 - bB(t)^2 = const \qquad \text{[II]}$$

Lanchester claimed that these equations could describe the evolution of a land or sea conflict in which factions use modern target-rich weapons[6]. The system is soluble exactly, the solution depending on the initial values R(0) and B(0) and the effectiveness coefficients. In particular, from the [II], we obtain that a condition for a stalemate to happen is:

$$B(0) = \sqrt{\frac{r}{b}}R(0) \qquad \text{[III]}$$

also known as Lanchester Square Law. It says that, for example, if you want to stalemate an adversary three times as numerous, you must be nine times as effective. The present relation also shows the necessity, for the outnumbered faction, to divide the enemy lines trying to split the entire battle in several sub-battles, concentrating forces. Lanchester mentioned the Trafalgar naval battle of 21^{st} October 1805 between English and Franco-Spanish fleets, as a kind of validation of his analytical model.

* Work performed with the partial support of INFM-G4

1.2. An agent-based computational model for Trafalgar

We decided to design and implement a new computational model of Trafalgar Battle, using an agent-based simulation. In this case, war is seen as a Complex Adaptive System (CAS) composed of many different elements –called agents- which interact with one another and adapt themselves to the changing environment. More specifically, each ship of the conflict is represented by a goal-based agent: each of them has a list of sequential goals to reach and act in order to satisfy its specific plan. In this way, it is possible to see a global strategy as an emergent behavior and a result of single agent strategies.

We would like to remark that this "bottom-up" approach is completely independent from the analytical model of Lanchester, and radically different from all the computational methods based both on difference equations and on numerical simulations [4]: we do not implement a local Lanchester law in it, neither we use numerical solutions of his equations. We just give a specific behavior to each element of the simulation and analyze the global dynamics of the system. The simulation parameters are editable through a simple LISP file, which is loaded by the program at each run. This allows the simulation of the historical battle, and also to build a set of counterfactual simulations -in which the inserted parameters are different from the historical ones- to examine the possible different outcomes of the battle. This leads us to very interesting results and a quite different interpretation of the conflict compared to the Lanchester's one.

The model is not deterministic: this is due to the several stochastic elements implemented in it. For this reason, many different runs of the same simulation are needed, with the purpose of obtaining statistically significant results –although not a Montecarlo simulation.

2. The Battle of Trafalgar

On October 19th, 1805, Adm. Villeneuve with 18 French and 15 Spanish ships slipped out of Cadiz heading for the Strait of Gibraltar. Two days later, a British fleet of 27 ships captained by Adm. Nelson sighted the allied vessels ten miles off Cape Trafalgar. The combined Franco-Spanish fleet turned north in a single line, while the English approached in two different columns, headed respectively by Nelson and Adm. Collingwood, from the west, splitting the enemy line and exploiting the N-NW direction of the wind. The outcome was a great English victory: 17 allied ships surrendered, 16 escaped, while no English ship escaped or was captured or destroyed. The allies lost 4408 men, the English 449 [1].

3. Model implementation: Swarm

Our model has been implemented with Swarm, a simulation system under development by the Swarm Development Group [10]. It represents a very helpful framework to design and implement agent-based simulations, independently of the particular nature of the simulation itself.

Swarm is a package of Java classes (originally implemented in Objective-C) containing all the necessary tools to manage agents, to monitor the built model and to collect and analyze the acquired data; from a conceptual point of view, is a tool that allows researchers to build models using a common language, providing explicit control over event scheduling for understanding event causality. Swarm offers an important hierarchical organization as well, based on the concept of "swarm": a swarm is a collection of objects and a schedule of activity over those objects. Schedules can be used to regulate agents' activities inside the swarm. Swarms can be encapsulated in hierarchical orderings: in this way, swarm activities can be controlled, run, paused and terminated by higher level swarms[10].

Because hierarchical ordering is a crucial characteristic of a military group, and tracing event causalities is particularly important in conflict analysis, we think Swarm is particularly fit to model a conflict as a CAS.

4. Model environment and wind management

In the simulation agents move on a 300 x 300 square grid, each containing one ship at most. The grid intends to map a square region of sea of about 16 x 16 nautical miles. Eight possible directions are allowed on the grid, from N to NE.

There exist two classes of wind: a global and a local wind. In the input LISP file, user specifies the mean direction, the mean speed and the variability of global wind: at each cycle the speed and the direction of global wind are extrapolated from normal distributions having the specified wind mean speed and mean direction as mean values, and the variability as standard deviations.

In addition, a local wind also exists: each cell of the grid has a wind speed and a wind direction associated to it, which are the actual values ships on the grid are subjected to. At the first, the program gives each cell global wind direction and position; then, through a second scan, ships on the grid are detected and their perturbation effect is computed. In fact, each agent determines a wind decay through the first five lee cells, and their local wind speed is modified following this relation:

$$w_{lee} = w(1 - s^{|d|})$$

where w_{lee} is the modified wind speed in the lee cell, w is the local wind speed of a lee cell, s is the amount of sail opened by the ship determining the wind decay, and d is the distance between the lee cell and the cell of the ship. In other words, each agent on the grid cause a modification of the conditions of neighbor cells in a way that depends on its own state and on the distance between the cells.

5. Agents

5.1. Ship classes

Each agent on the grid represents a different ship. We thought it could have been useful to consider a reference ship to shape all the rest compared to it and so determine their main characteristics. We found the English flagship Victory –admiral Nelson's ship- to be the right candidate to do this, because of the abundance of historical data regarding this particular ship. This permitted us to find three classes of ships with common main characteristics, as shown in the following table:

Table 1. Ship classes

Ship class	Crew number	Cx	Max speed
Flag	850	1.0	10.0
I	680	0.8	9.0
II	520	0.6	8.0
III	340	0.4	6.0

Each of the 60 ships belongs to one of the classes presented in Table 1.

5.2. Ship movements

Ships can move in eight different directions upon the grid. Each ship has a specific direction, that changes across the simulation and which represents the direction of its bow, and a specific speed, which can have any value belonging to the range [0, Max speed]. Ship speed is the result of a product of four basic factors: local wind speed in the cell occupied by the ship; ship's relative ($c(\theta)$) and absolute (cx) efficiency; ship's amount of opened sail (sail coefficient).

Ship relative efficiency varies in the [0, 1] range and is affected by the angle between wind and bow direction in the way shown if Figure 1. This dependence can be considered a good approximation of the actual efficiency of square-rigged vessels: we have highest efficiency (therefore highest speed and maneuvering capacity) in broad reach conditions, lowest in reaching, no movement capability when the ship's bow is against wind's direction.

The absolute efficiency coefficient is assigned by the user to the ship –according to the ship class the agent belongs to – and can change during the simulation if the element is hit by an enemy shoot, in a way we'll see later: it holds all the information about the intrinsic maneuvering capability of the ship, and its damage state due to the fire in the battle.

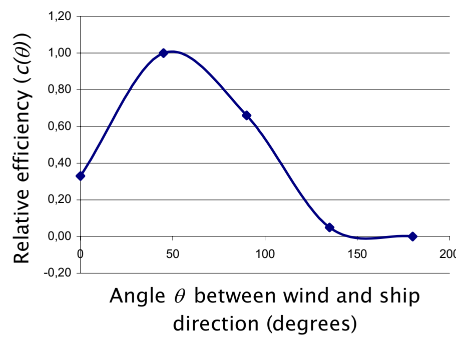

Figure 1. Ship's relative efficiency

The sail coefficient can vary in the [0, 1] range and represents the percentage of sail the ship has opened. Agents can unfurl or furl sails –therefore increase or decrease the sail coefficient – in order to reach higher or lower speeds. The more the sail is opened, the higher the perturbation effect on local wind speeds of neighboring cells.

Ship speed is the number of cells a ship can move by in a single cycle. A specific algorithm rules the speed tuning of each ship: agents modify their speeds according to the particular target they have to reach on the grid, depending on the specific strategy they're following.

A movement caused by drift also is present: each ship moves in the direction of wind by one cell with a probability proportional to wind speed.

5.3. Ship vision

Each agent is able to have a local information about the environment he moves within, thanks to its vision capabilities: it can see every cell surrounding it within a range of 40 cells. It can distinguish friends and enemies and gather information about their states.

5.4. Fights among ships

At each cycle, every single ship engages two different fights with, respectively, the starboard and port nearest enemy. Fighting can happen in two different ways: shooting or boarding.

Shooting fight is a kind of non-local interaction between agents. It takes place if the distance between the ships is less than 2. To take into account of the different gun series each ancient vessel had on board, each agent can perform three types of shoots against an enemy: short-gun, middle-gun and long-gun shoot.

Table 2. Different shoot types

	Short gun shoot	Middle gun shoot	Long gun shoot
Max range	12	16	24
Men casualty	[0, 2.0]	[0, 1.50]	[0, 1.25]
Gun damage	[0, 1.0]	[0, 0.75]	[0, 0.25]
Cx damage	[0, 0.04]	[0, 0.03]	[0, 0.02]

A shoot consists in modifying enemy's status, specifically enemy's crew number, gun number and cx coefficient. The farthest reachable distance as well as the amount of the inflicted damage depends on which kind of shoot is performed, as shown in Table 2. All casualties and damages are real random numbers belonging to a specified range.

All the agents have 42% probability to fire a short-gun, 28% to fire a middle-gun, 30% to fire a long-gun. At each cycle, all agents execute a number of shoots against the starboard or port nearest enemy, proportional to the half of number of guns the ship has and to the effectiveness coefficient. This coefficient is the average ratio of guns that will be actually fired by each agent per each cycle: i.e. a .2 effectiveness coefficient means that an agent will be able to fire 20% of its guns per cycle. Effectiveness coefficients are common to ships belonging to the same fleet and don't change during the simulation. In our model, English and Franco-Spanish effectiveness coefficients are set, respectively, to .22 and .022. The great difference between these two values can be explained by the much higher fire power of English vessels historians report[4].

Boarding fight consists just in a men loss: casualties are real random numbers belonging to the [0, 5.0] range.

5.5. Agent states

Agents can be in four different states: normal, escaping, surrendered and sunk. The rules that condition the state transitions are based upon the achieving of certain thresholds (crew and gun number, cx) and the existence of specified conditions.

It is remarkable that these conditions depend on what an agent actually sees around itself: if the number of escaping, surrendered or sunk friends an agent sees is equal or greater to the half of all visible friends, its state is switched to escaping. The presence of this local information generates a kind of "panic spreading effect" (see Figure 2), that leads an entire fleet to quit if a sufficiently large number of ships have withdrawn, yielding unexpected results. British escaping ships try to reach the western limit, Franco-Spanish ones the eastern one, to put themselves out of bounds. Escaping ships still interact with other boats, fighting with them. Surrendered ships have no sailing and fighting capabilities; sunk ships just stay on the same cell until the end of simulation (crew and guns are lost).

5.6. Agent strategies

The user can associate a list of sequential goals to each single agent. Goals are coded in a specified syntax and are sequential in the sense that the agent tries to achieve the first of the list: if this appears to be impossible, the next is considered and so on, until the end of the list. If none of the goals are reachable, no action is taken, and the agent remains in its present state.

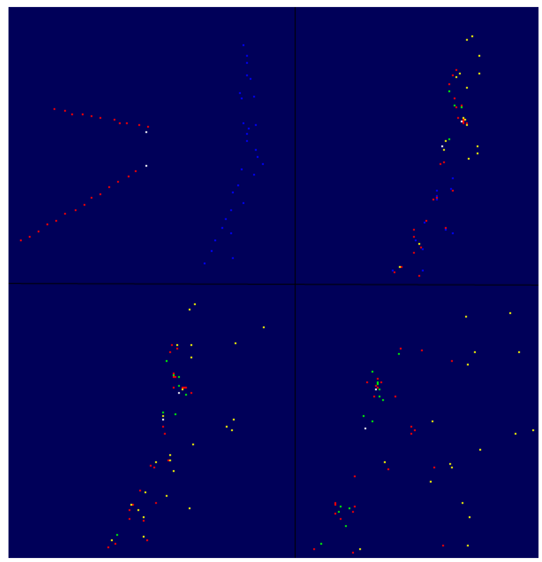

Figure 2. Evolution of the system.
Blue dots are allied ships, red are English, two white are English Admirals' vessels, yellow are escaping ships, green surrendered. "Panic spreading effect" is notable in 2nd and 3rd quarter

5.7. The end of simulation

The simulation ends if one of the two fleets is completely formed by surrendered or sunk ships; in this case, if the other faction is also formed by surrendered or sunk ships, a stalemate occurred.

6. Results

We have made three different simulations. Each of them consists of 100 runs of the same model to obtain statistically significant results.

6.1.Historical simulation

In this first simulation we have used historical data. Initial positions of single ships have been reproduced on the grid of the simulation. According to historical data, English exploited the wind coming from N-NW to enter in the battle presenting themselves in two parallel columns, perpendicularly to the single Franco-Spanish line, heading east. The two columns were composed by 12 and 15 ships: the first moved towards the 13^{th} enemy ship, the second towards the 18^{th}. English strategy was to separate the enemy line, generating two different sub-fights and a "pell-mell battle", to say it with Nelson's words. Franco-Spanish fleet was ordered to stay and wait for the enemy to come as near as possible. What actually came up was a number of local fights among groups of relatively few ships. In many cases, these fights are historically reconstructable, and this helped us to give each ship a specific and historically accurate strategy.

The results show a very good agreement with historical data, except the number of English casualties. This little discrepancy can be considered as a level of detail of the simulation: refining the strategies of the single agents in a more and more strict accord with the historical one, a reduction of the discrepancy is observed.

Table 3. Results of historical simulation

	Obtained	Expected
Eng casualties	$5.6\cdot10^2\pm70$	449
F-S casualties	$4.4\cdot10^3\pm300$	4408
Eng escaped ships	0.03±0.17	0
Eng surrendered ships	0.03±0.17	0
F-S surrendered ships	18±2	17
F-S escaped ships	15±2	16

When the simulation output data contained the right number of Franco-Spanish escaped ships (16), a further check has been made about the names of the escaped ships. It is remarkable that the simulation gives a list of names which is the same of the historical one by the 75% (on the average 12 of 16 escaped ships are correct).

6.2.Counterfactuals experiments

Two counterfactual simulations have been made in order to check the possibility that the outcome of the battle could have been different from that historically known, and to study the conditions for these different scenarios to happen.

Counterfactual experiments turn out to be very interesting even for a comparison of simulation results with Lanchester's forecasts. As mentioned before, Lanchester used a subtle argument regarding Trafalgar to use this battle as a kind of validation of his own model; but, instead of applying it on the actual battle, he used it to justify the strategy Nelson intended to follow, but was unable to do in the reality. According to Nelson's plan, British expected to be outnumbered, 40 ships against 46; he also expected to find the enemy in a single line, so he planned to break this line in three places: at the center (23^{rd} ship), with a column of 16; at the ¾ point (12 ships from the rear) with a column of 16; at about the 20^{th} ship, with a column of 8. Nelson's intention was to isolate the leading half of Franco-Spanish fleet, preventing to join the main battle in the center.

Lanchester claimed this strategy was in perfect agreement with his theory: assuming all the ships had the same strength (therefore effectiveness coefficients in [I] are the same) he found (using [II]) that in the rear $(32^2-23^2)^{1/2}=(495)^{1/2}$ English ships would emerge victorious. Then, they would encounter the $(23^2-8^2)^{1/2}=(465)^{1/2}$ Franco-Spanish ships coming from the fight in the vanguard; consequently, at the end, $(495-465)^{1/2}=5.5$ British ship would survive. Nelson wanted to split the battle in two sub-battles, lose one of them and win the final fight exploiting the relative outnumber of the enemy. According to Lanchester's theory, had Nelson decided to face the enemy in a single line, he would have been defeated, and about 23 Franco-Spanish ships (46^2-40^2) would have emerged victorious.

In the first counterfactual experiment we made we decided to change the English strategy, presenting the British fleet in line against the combined fleet.

Table 4. Results of the first counterfactual simulation

	Obtained	Historical
Eng casualties	$5.5\cdot10^2\pm76$	449
F-S casualties	$4.4\cdot10^3\pm400$	4408
Eng escaped ships	0.5±0.6	0
Eng surrendered ships	0.03±0.17	0
F-S surrendered ships	17±2	17
F-S escaped ships	17±4	16

Table 4 shows that results are compatible with historical data, except for the English casualties number and the possible escaped English ships.

In the second counterfactual experiment, we investigated the possibility for the combined fleet to win. We changed both environment parameters and Franco-Spanish fleet strategy: wind global direction was set at E (against British ships), Franco-Spanish ships had the goal to move towards the nearest enemy and fight against it, concentrating forces. In this case we obtained completely different data from the historical ones (see Table 5).

English casualties increased by almost 2.9 times over the value obtained in the historical simulations, and the expected number of British escaped ships is now 19, while no English ship actually escaped in the real battle.

Table 5. Results of the second counterfactual simulation

	Obtained	Historical
Eng casualties	$1.6 \cdot 10^3 \pm 200$	449
F-S casualties	$2.5 \cdot 10^3 \pm 500$	4408
Eng escaped ships	19±1	0
Eng surrendered ships	8±1	0
F-S surrendered ships	3±2	17
F-S escaped ships	10±9	16

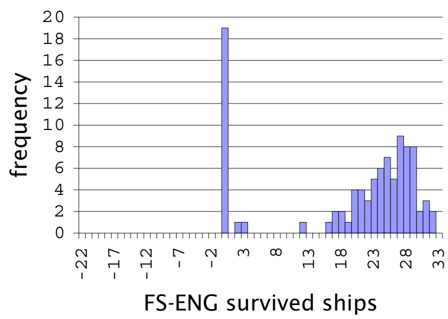

Figure 3. Outcomes of the second counterfactual simulation

Figure 3 shows the battle outcomes for 100 runs, plotting the difference between Franco-Spanish and English survived ships: positive values represent English victories, negative values combined fleet victories, 0 is a stalemate. It is interesting to note that Franco-Spanish fleet won 76 times and that, even if in none of the runs an English victory occurred, 19 stalemates took place. This is due to the "panic spreading effect" mentioned before, that leads ships to quit in critical global situations instead of been captured or destroyed.

7. Conclusions

Our agent-based computational model of the Trafalgar battle has produced interesting results in very good agreement with historical data at a high level of detail. Simulations, implemented using Swarm, are particularly flexible to use, and this made us able to investigate the nature of the conflict, comparing the results obtained with the solutions of Lanchester's model. This analytical model is unable to describe even a qualitative evolution of the system and is based on assumptions that are over-simplified. In particular, according to Lanchester's model, the English victory in Trafalgar is substantially due to the particular strategy adopted by Nelson, because a different plan would have led the outnumbered British fleet to lose for certain. On the contrary, our counterfactual simulations showed that English victory always occur unless the environmental variables (wind speed and direction) and the global strategies of the opposed factions are radically changed, which lead us to consider the British fleet victory substantially ineluctable. The reason of this assertion lies in the fact that not the strategy, but the (historically documented) higher English firepower represented the main cause of victory for the Royal fleet. And, while Lanchester underestimates this effect supposing a=b in [I], it is striking in our simulations, where different fleets' firepower is reflected by different effectiveness coefficients. Moreover, the dynamics of the system results to be of fundamental importance for a correct analysis of the conflict: in fact, analyzing the system evolution in detail, it is clear that Nelson's plan adopts the safest possible strategy, leading the Franco-Spanish vanguard to be substantially out of the fight for a long time, so as to minimize the risk for the English of losing ships and men.

8. References

[1] Alan Schom, *Trafalgar, Countdown to Battle 1803-1805*, Oxford University Press, New York, 1990

[2] F.W.Lanchester, *Aircraft in Warfare: The Dawn of the Fourth Arm*, D.Appleton & Co., New York, 1916

[3] J.M.Epstein, *Nonlinear Dynamics, Mathematical Biology, and Social Science*, Addison-Wesley, 1997

[4] B. McCue, *Combat Analysis – Lanchester and the Battle of Trafalgar*, Available at http://www.mors.org/publications/

[5] J.A. Dinges, *Exploring the Validation of Lanchester Equations for the Battle of Kursk*, Thesis, Naval Postgraduate School Monterey, California, 2001

[6] R. Darilek, W. Perry, J. Bracken, J. Gordon, B. Nichiporuk, *Measure of Effectiveness for the Information-Age Army*, RAND, 2001

[7] J.G. Taylor, *Lanchester Models of Warfare*, Military Applications Section, Operation Research Society of America, 1983

[8] S. Weinberg, "Is the Universe a Computer?", *The New York Review of Books*, October 24, 2002

[9] P.K. Davis, *Aggregation, Disaggregation, and the 3:1 Rule in Ground Combat*, RAND, 1995

[10] G.E. Ropella, S.F. Railsback, S.K. Jackson, "Software Engineering Considerations for Individual-based Models", *Natural Resource Modeling*, Volume 15, Number 1, Spring 2002