Multi-Platform Coordination in Command and Control

Patrick Beaumont and Brahim Chaib-draa
Computer Science & Software Engineering Dept
Laval University
Ste-Foy, Quebec, Canada.
{`beaumont;chaib`}`@damas.ift.ulaval.ca`

Abstract. The use of agent and multiagent techniques to assist human in its daily routine has been increasing for many years, notably in Command and Control (C2) systems. In this article, we focused on multiagent coordination techniques for resources management in real-time C2 systems. The particular problem we studied is the design of a decision-support for anti-air warfare on Canadian frigates. In the case of the several frigates defending against incoming threats, multiagent coordination is a complex problem of capital importance. Better coordination mechanisms are important to avoid redundancy in engagements and inefficient defence caused by conflicting actions. We present different task sharing coordination mechanisms with their evaluation.

1 Introduction

Coordination is the process by which agents avoid superfluous actions, by managing the interdependencies to minimise the conflicting actions and goals. In most systems, this coordination must be carried out in an environment constrained on time, available bandwidth, etc.

In the case of the defence of several frigates, multiagent coordination is a very complex problem of capital importance. The environment imposes strong real-time constraints, which is mostly due to the threats becoming smarter and the Anti-Air Warfare (AAW) situations happening more and more on the littoral rather than in open sea. In a standard AAW situation, operators have few seconds available, in which they must identify threats, choose and apply defence plans. Furthermore, in a multi-frigate system, it is also necessary to coordinate defence actions between the ships, which is obviously more complex. As the reaction time is usually very short, it is often not possible for the operators to coordinate their actions with the other members of the group. This can result in 1) redundancy in the engagements, using more resources than necessary, 2) inefficient defence and, 3) an increase in the cost of the global defence solution. Another impact of the lack of coordination is the negative interactions that can take place when certain resources are used in parallel, which creates a degradation of the global solution. This prompts for an increasing need for cooperation between frigates. Indeed, good coordination mechanisms for the optimal use of the resources of a group of frigates become essential during a military deployment.

In our on-going project called NEREUS, in collaboration with Lockheed Martin Canada (LMC) and Defence R&D Canada, the coordination is cooperative and is intended to organise the individual actions toward a common goal, which is the efficient defence of the complete fleet. This problem is very complex, since we focus on a problem close to real-life situations, as we have strict deadlines (a limited amount of time to coordinate) and communications are not free.

2 Coordination Mechanisms

In the light of those elements, the first step to a successfully coordination of a set of frigates (considered here as agents) consists of developing efficient coordination mechanisms. The approach considered is to distribute the threats among agents before starting the planning process. This threat repartition problem is usually referred to as *task sharing* in the literature and is a divide-and-conquer approach to multiagent coordination.

We consider a general approach to coordination in AAW, which consists of:

1. Detecting and identifying threats (STA).
2. Distributing threat among frigates.
3. Creating individual plans.
4. Managing positive and negative interactions between frigates' resources.
5. Performing the determined plan.

Thus, each threat will be allocated to agents, then each agent will only plan on threats it has been allocated. The *task sharing* alleviates the burden of the agents, as it significantly reduces the complexity of the planning process. Indeed, it is easier to plan for a few threats than for the complete list of threats. Another advantage of this approach is that we eliminate conflicting actions where two agents engage the same threat while another is left unimpeded. Furthermore, since we are not in a system with free communications, it takes several milliseconds for a message to reach its recipient. Thus, reducing the conflicts greatly accelerate the coordination process since much less messages will be exchanged.

In task sharing, the following mechanisms are commonly used to distribute tasks among agents: market mechanisms, Contract Net, multiagent planning and organisational structure. Inspired by these mechanisms, we developed coordination mechanisms to distribute threats among frigates, which we will present in the following sections.

2.1 Probability of Success

Before going on, we need to define what is exactly the probability of success (P_S) and how it can be computed. The P_S of a frigate for a threat is a value that represents the probability, evaluated by the frigate, to destroy this threat. A P_S list (LP_S) for a specific frigate contains its P_S evaluation for each threat. Finally, a P_S matrix (MP_S) contains the LP_S of every frigate.

Obviously, it is possible to compute the exact P_S of a frigate against a threat only when the complete plan has been constructed. Since we need to determine the P_S *before* planning, it is imperative that we use heuristics to determine probabilities of success.

The current P_S evaluation implemented was inspired by Brown [2] and is based on the CPA of a threat relating to a frigate. The CPA is the shortest possible distance between the trajectory described by a threat and the evaluation point.

The CPA-based method to evaluate the P_S is given by the following formula:

$$P_S = P_K \cdot (1 - \frac{CPA}{R_{max}}) \tag{1}$$

where P_K is the *probability of kill* for this specific threat and R_{max} is the *maximum distance* at which a threat can be engaged (in our case, this is the range of the SAM, i.e. 50 km). The probability of kill P_K is relative to the prior number of threat engagements contracted. To determine the P_K, we did empiric testing and compiled the results. Then, we extrapolated the equation 2, which gives the estimated value of P_K, in function of the number of threats already targeting the frigate. We do not believe that this curb reflects the actual survivability in function of the number of threats, but it is accurate for any number of threats below eight, since the results and the expected value do not diverge by more than 0.75 % in any case.

$$P_K = (-0.00424414 \cdot nb_threats^2) - (0.0020234375 \cdot nb_threats) + 1 \tag{2}$$

2.2 Central Coordination Mechanism

The *central coordination* mechanism is based on communications, with a centralised coordinator. The concept of the central coordination is that a central frigate-agent is responsible to collect the information, and decide of a task distribution according to this information. In this case, the information transmitted is the LP_S of every agent.

The central coordination process is described in the following way:

1. The fleet chooses a coordinator.
2. When one or more threats are detected, every ship computes its LP_S and sends it to the coordinator.
3. The central coordinator constructs a capability matrix (MP_S), which is a matrix of LP_S for each frigate.
4. The coordinator decides how to assign the incoming threats to frigates, using the capability matrix and an optimization algorithm.
5. The coordinator sends notification messages to chosen ships.

Note that the choice of an appropriate coordinator is done prior to detecting any threat. In our case, the coordinator is the frigate with the highest ranking. The process of choosing the coordinator is facilitated by the assumption that every agent has the same common view. This, combined with the fact that our agents are fully cooperative, makes it possible for any agent to determine which one is the coordinator without the need for communication or negotiation.

Optionally, one or more backup coordinators can be chosen in the first step. Those backup coordinators will receive the same information as the central coordinator, and will take the role of coordinator if the central coordinator is destroyed or become unable to accomplish its tasks.

The tasks assignation by the coordinator can be done with any optimization algorithm such as a greedy algorithm or even a complete lookup of the solution set. The greedy algorithm is a fast heuristic, while the complete solution lookup takes more time, since it looks over every possible solution to choose the best one. Usually, the central coordinator allocates only one threat per frigate. However, if there are more threats than frigates, the coordinator has two choices. The first is to allocate one threat per frigate as usual and then start the process over by demanding another LP_S evaluation for the remaining threats. When it has received every P_S list, it then assigns the threats to the agents. The process is iterated as long as there are threats left unassigned.

A disadvantage of the central coordination mechanism is that it is centralized. This can be dangerous, since a single point of failure can make the whole coordination process abort. Indeed, if the coordinator becomes unresponsive for any reason (damaged communication system, the ship is sunk, etc.), the coordination process must be started over with a new coordinator, at the expense of several important seconds. Of course, the use of backup coordinators can alleviate this problem, but will also significantly increase the use of communication channels, which can become overloaded.

2.3 Contract Net

The Contract Net mechanism is similar to the central coordination mechanism, as it relies on a central coordinator and the use of communications. The difference between the two mechanisms is the number of threats assigned at one time. In the central coordination, we want to assign all threats at once, while we will allocate one threat at a time in the Contract Net mechanism. The following describes the Contract Net process:

1. The fleet chooses a coordinator.
2. For each threat detected, one at a time:
    - (a) The coordinator asks every ship to send an estimated P_S value for this threat.
    - (b) Each ship returns its estimated P_S value for this specific threat, considering already assigned threats.
    - (c) The central coordinator chooses the best frigate to engage this threat and informs the agent.

The protocol is adapted from the FIPA Contract Net ([6]). It is interesting to compare the iterative process in this protocol to the one in the central coordination. We find that the only difference is the number of threats we allocate at a time.

The problems associated to the central coordination mechanism also apply to the Contract Net protocol, since both mechanisms are centralized coordination methods. However, the communication channels are more solicited in the Contract Net protocol than in the central coordination protocol, since more messages are exchanged. In this case, if the communications are unsafe or unstable, the chances increase that they become elements of failure. Therefore, the central coordination will give more "reactive" (i.e., faster) responses, while the Contract Net approach will give better results but will also take longer. In addition, as the communication bandwidth decreases, the expected quality and timeliness of the coordinated solution obtained with this mechanism will decrease faster than with the central coordination method. Therefore, Contract Net is applicable only when there is sufficient time to coordinate. A more detailed comparison of these two mechanisms is provided in Section 4.

2.4 ∼Brown (Similarly Brown)

Another mechanism based on communication is the mechanism proposed by Brown ([3] and [2]). This method can be used in a centralized or decentralized way.

This mechanism closely resembles the central coordination with added parameters. The main difference between the central coordination and the ∼*Brown* mechanism is that the threats are ordered by a priority evaluation before being distributed. This priority is based on three factors: the certainty that a threat is aimed at a specific ship, the relative importance of each ship and the fleet engagement capability for each threat.

To transform the original centralized mechanism in a decentralized mechanism, the following assumptions must hold: 1) the agents must be entirely cooperative, 2) the agents must be homogeneous, 3) the protocol must be used in the same way for every agent and 4) the agents must be aware that the three preceding assumptions hold. These assumptions are required to make sure that every agent, receiving the same information, will evaluate the situation in the same way. Firstly, if agents are not entirely cooperative (e.g., if they are from different nationalities), there is the possibility that one agent might defect, which is unacceptable. Furthermore, if the agent are not homogeneous, or if any part of the protocol uses information specific to a ship (as ship's own ranking), the allocation might be evaluated differently among different ships. The fourth assumption is self-explanatory.

In the case where such assumptions hold, a simple way to use the mechanism in a decentralized fashion is to broadcast all the information to every agent. Thus, since each agent receives the same information and reasons the same way, each agent can deliberate and come up with an allocation solution. Moreover, this solution does not need to be sent to other agents since each one will found the same solution. Therefore, once a solution is obtained, an agent only needs to act on its assigned threats, as it is sure that the other agents will take care of their own threats.

The following process describes the steps of the decentralized version of this mechanism:

1. Before any threat is detected, each agent determines the relative weight of each ship and puts in a list of weights (W). The generic formula to compute the weights is:

   $$weight = rank \cdot x + y$$

   The *rank* is a simple value of the relative importance of each ship. Thus, a ship with a ranking of 10 is more important than a ship with a ranking of 5. Since any agent knows the ranking of every ship, it also knows what is the lowest and highest ranks. Also, an adjustable parameter is known: the *maximum weight deviation* (Dev_{max}), which is the desired difference between the highest and lowest ranking frigates' weights. Knowing this, we have that:

$$1 - Dev_{max} = lowest \cdot x + y$$

$$1 = highest \cdot x + y$$

Thus,

$$x = \frac{Dev_{max}}{highest - lowest}$$

$$y = 1 - \frac{highest \cdot Dev_{max}}{highest - lowest}$$

2. When threats are detected, a matrix of threats' targeting probability (T) is created. This is an evaluation of which ship, each threat might be targeting.

3. Each agent determines its P_S list (LP_S) and broadcasts it to the other agents.

4. A threat-weight matrix ($T \cdot W$) is calculated, which is a multiplication of the weight list with the targeting matrix.

5. Once each LP_S is received, the fleet engagement capability (P_F)[1] for each threat is determined. The fleet engagement capability (P_F) for a threat is computed by multiplying every P_S for this threats.

6. Each agent computes the fleet engagement priority matrix, which is $\frac{T \cdot W}{P_F}$, for each threat.

7. Each agent constructs a capability matrix which is the multiplication of the the matrix composed of the LP_S of every frigate by the *fleet engagement priority* matrix ($\frac{T \cdot W}{P_F} \cdot MP_S$).

8. Using an allocation algorithm, as in the central coordination mechanism, each agent determines the assignation of threats to ships.

The final step before allocating the threats is to construct the fleet capability matrix ($\frac{T \cdot W}{P_F} \cdot MP_S$). Once the fleet capability matrix is obtained, the allocation is done as in the central coordination process, assigning one threat per frigate and iterating the process if necessary.

3 Results of Preliminary Experiments

To evaluate the previous task distribution mechanisms, we ran two different types of tests. The first set serves to analyse the adjustment of coordination parameters while the second compares the different coordination mechanisms. The tests were conducted on a dual Xeon 2.6 GhZ, with 4 GB of RAM.

For the first set, we ran 1,000 tests, for each value of the parameter to test. The average time required to run a coordination test is 900 milliseconds.

---

[1] referred to as *force performance* by Brown

We present here five of the parameters that we tested: the *ships formation*, the *inter-ship distance*, the *bandwidth*, the *number of frigate engaging each threat* and the *capability matrix evaluation*.

The default coordination mechanism used for testing is the central coordination. When testing individual parameter values, we used the default settings presented in Table 1 for the other parameters, unless otherwise specified:

Coordination Mechanisms	*Parameters*	*Default Values*
All	Ship formation	Layout 1
	Inter-ship distance	500 m
	Bandwidth	1,024 k/s
Contract Net	Frigates per threat	1
∼Brown	Priority evaluation	$T \cdot W/P_F$

Table 1: Default values for coordination parameters.

We will now detail each parameter, give results for different values evaluated and discuss those results. In the presentation of the results, we use the term *scenario* which is a typical naval battle; it starts when threats are detected and ends when each threat has been either destroyed or has hit the ship. When we specify the number of hits per scenario, it represents the average number of hits (on any ship) during a scenario.

3.1 Ships Formation

The *ships formation* is the way the ships are positioned relatively to each other. The different formations are presented in Figure 1.

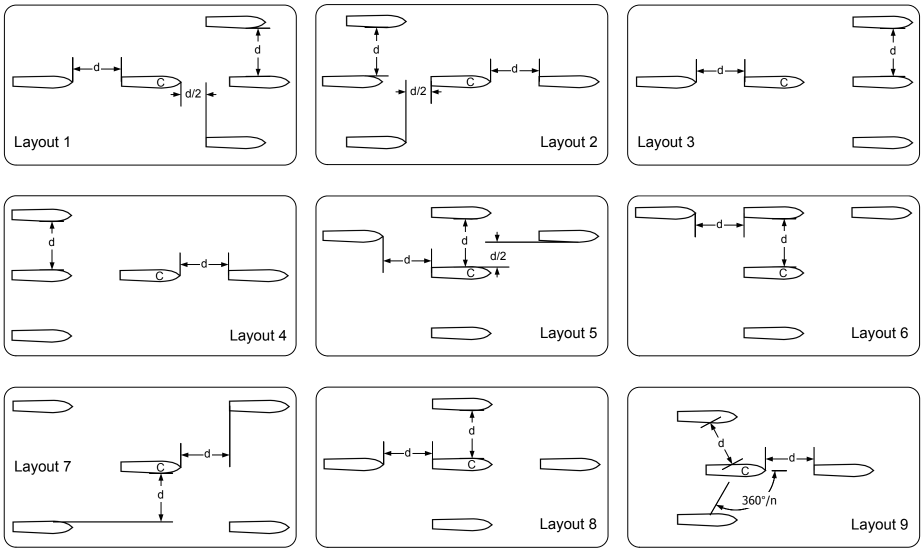

Figure 1: Coordination layouts.

Figure 2 presents the complete results for the different formations. In this figure, we can see that layout 7 is the best, almost with every number of threats.

However, this is not conclusive in itself, since a typical frigate in our project NEREUS is still a simplification of a real frigate. It might be interesting to point out that the layout 1 is one of the commonly used formation in AAW, while being the most inefficient formation tested. It is also interesting to compare the similar layouts: (1 and 3), (2 and 4) and (5 and 6). It seems, that each time, the formation with three ships aligned gives better results than the formation where they are placed in a triangular disposition. Another interesting observation that can be made is that in the *reverse* formations: (1 and 2) and (3 and 4), it seems that the configuration where the three grouped ships point toward the centre is more efficient.

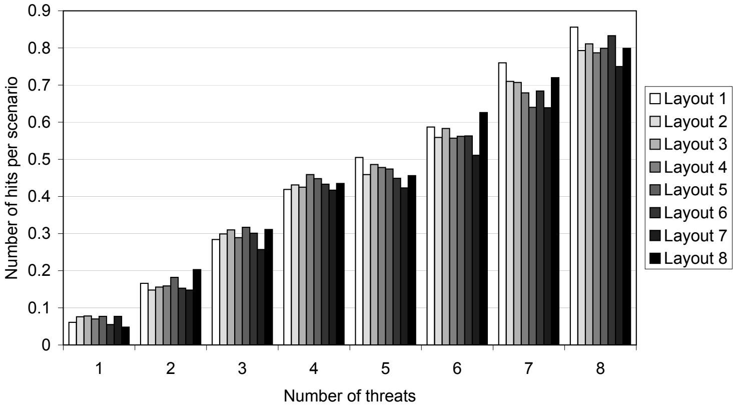

Figure 2: Number of hits per scenario, considering different formations.

3.2 Inter-ship Distance

Since the effective range of the various weapon systems varies greatly, it is interesting to consider the standard distance between ships. Analysing the data presented in Figure 3, which is the number of hits per scenario considering the inter-ship distance, it is clear that any distance up to 1,000 metres is roughly equivalent, while the efficiency decreases rapidly past 1,500 metres. Furthermore, this difference becomes clearly marked beyond three threats in a scenario. This can be explained by the fact that the farther the ships are, the longer an illuminators (STIR) is used to guide a SAM intercepting a threat aimed at another ship. While a STIR is occupied, no other gun or SAMs can be used on this side of the ship. Analysing the average of the results of each distance, we see that it is best to be at a distance of 1,000 metres, since being closer than that will impede the movement used for positioning the frigate (see [1], [5] and [4]).

3.3 Bandwidth

In our approach. the bandwidth of the system's communication channel is fixed for the length of the simulation. Before starting simulations, the bandwidth can be reduced to represent background noise or degraded communication conditions. Figure 4 shows the average number of hits per scenario, for tests

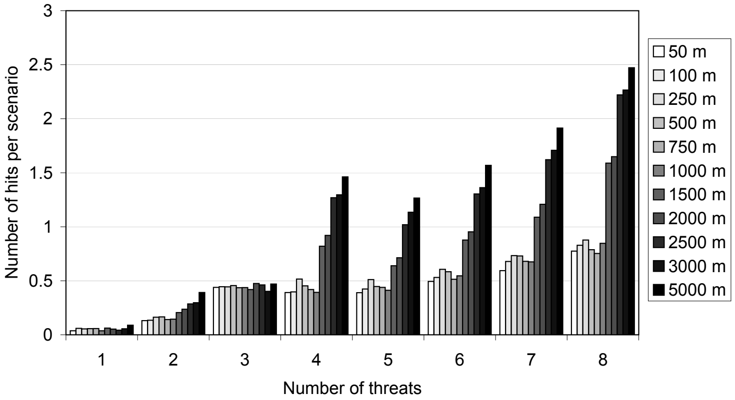

Figure 3: Number of hits per scenario, considering different inter-ship distances.

considering different bandwidth values from 1k/s to 8192k/s. In this figure, we see that even at 1k/s, which is low by modern standards, the number of hits per scenario is much the same. The average weight of a message is around 1040 bytes in size, and does not vary much depending of the number of threats, since the message core is relatively small, the rest being the security and transport headers. This explains why the results are not very conclusive: at the lowest bandwidth tested (1k/s), it takes only 1 second to send a message. This means that in most scenarios, the agents have enough time to communicate before a soft deadline is met. This is due to the fact that after the coordination has been done, each frigate only intercepts a limited number of threats (usually 1 or 2). Thus, it is possible to intercept the incoming ASMs a little later without important degradation in the plan quality, as the STIRs will not be used after the incoming threats are destroyed. However, we used the central coordination, which is not the most communication intensive mechanism, to evaluate the impact of this parameter. Thus, having a high communication preparation delay in Contract Net coordination could significantly reduce the efficiency as the number of threat increases. This is due to the fact that the Contract Net mechanism considers threats one at a time, and therefore uses a lot of communications.

3.4 Number of Frigates per Threat

This parameter controls the number of frigates that will engage each incoming threat. The minimum number of frigates to intercept a threat is one, while the maximum is the number of frigates in the scenario. This parameter was tested with the Contract Net protocol, which makes it trivial to assign more than one frigate for each threat.

Interesting results are shown in Figure 5, which presents the efficiency, given the number of frigates per threat. This efficiency is the percentage of threats destroyed, divided by the number of SAMs used to destroy them ($1 - \%$ *of hits / nb SAMs used*). In this figure, we see that the most efficient ratio is one frigate/threat. However, in our case, the survivability is far more important than the total of resources used. Thus, if we transform the efficiency calculus by adding an α parameter to stress the fact that it is bad to be hit by threats, the efficiency becomes: ($1 - \alpha \cdot \%$ *of hits / nb SAMs used*). Figure 6 shows the

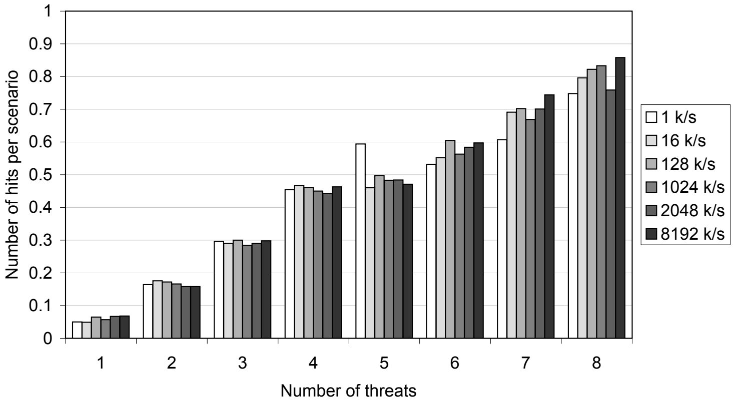

Figure 4: Number of hits per scenario with one threat, considering different bandwidth values.

efficiency calculated with an α of 10. In this case, we see that the best ratio becomes two frigates per threat.

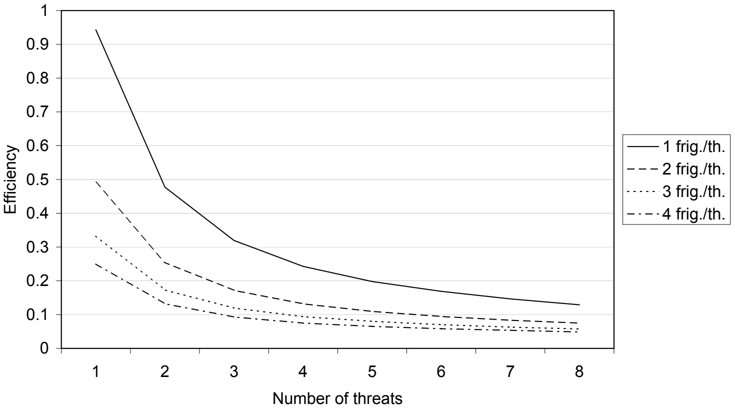

Figure 5: Efficiency, considering different number of frigates per threat.

3.5 Capability Matrix Evaluation

In the ∼*Brown* mechanism, the capability matrix is evaluated by each ship (step 7 of the ∼Brown process), before using the allocation algorithm to determine how to assign the threats. What we call the *fleet engagement priority* matrix is presented by [3] as the “prioritized force level threat table” and is evaluated in the following way:

$$\frac{T \cdot W}{P_F}$$

Once this matrix is obtained, the multiplication of $\frac{T \cdot W}{P_F}$ by the P_S matrix (MP_S) will gives what we call the *capability matrix*. The standard capability matrix is therefore: $\frac{T \cdot W}{P_F} \cdot MP_S$. In addition, we tried four different other ways to

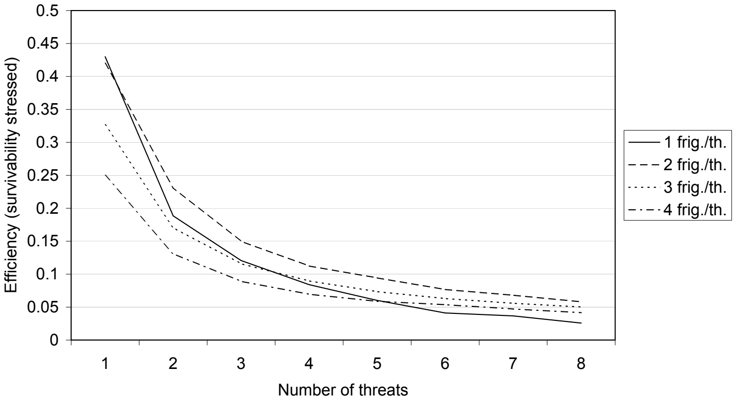

Figure 6: Efficiency with survivability stressed, considering different number of frigates per threat.

determine the capability matrix, which are shown in Figures 7 and 8, which present respectively the average number of hits per scenario and the average rank of destroyed ships. Note that the MP_S alone method is the one used to build the capability matrix of the central coordination mechanism.

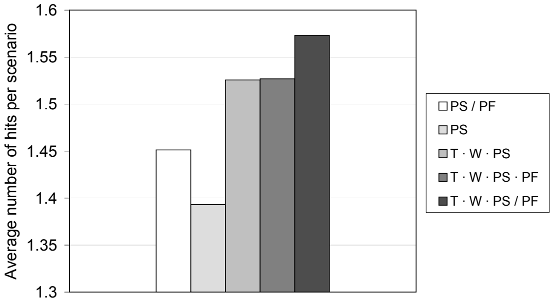

Figure 7: Average number of hits per scenario, considering different capability matrix evaluation.

At first, we believed that $\frac{T \cdot W}{P_F} \cdot MP_S$ evaluation would give the best results as suggested by Brown himself. However, experimentation demonstrates (Figure 7) that the best capability matrix would be the MP_S/P_F matrix. While this is surprising, we can explain those results. Since the cargos, with the highest ranking, cannot defend themselves, they have to be defended by other frigates. As the defence of an other ship consummates more resources that defending itself, less actions can be planned on the overall when a ship with higher ranking is prioritized. Thus, when weights are entered in the computation, the defence of the cargos is rated higher and less actions are planned.

Furthermore, the comparative results of $\frac{T \cdot W}{P_F} \cdot MP_S$ and $T \cdot W \cdot P_F \cdot MP_S$ in Figure 8 require explanations. When we *multiply* the $T \cdot W$ by P_F, we give greater importance to threats that the fleet is more confident of being able to intercept. On the contrary, dividing $T \cdot W$ by P_F tends to prioritise threats harder to intercept. Thus, these results show that it is better to invest time and

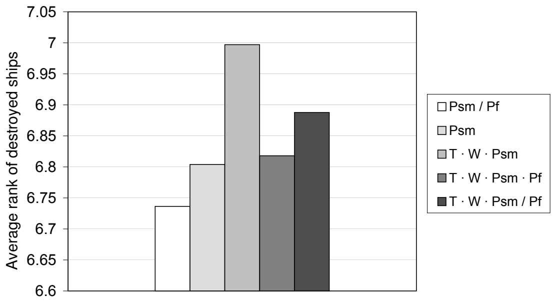

Figure 8: Average rank of destroyed ships, considering different capability matrix evaluation.

resources on threats which have more chances of being intercepted.

4 Comparison

We have seen how different parameters values can influence the coordination mechanisms. Thus, the fine tuning of these parameters is crucial to develop a good mechanism. However, we can analyse the different protocols independently of the results to determine their relative strengths and weaknesses.

The following is a non-exhaustive list of coordination mechanism attributes:

- *Communications:* As we have discussed earlier, the communications are important in many coordination mechanisms. Thus, knowing the number and importance of communication is important to know which mechanisms will be more sensible to degradation in the communication environment.

- *Centralized:* Some mechanisms use centralized information and decision-making. In multiagent systems, it is usually believed that a centralised method is less robust than an equal but decentralised method. Indeed, in centralised mechanisms, the failure of a single agent (the coordinator) can make the process abort, or at least significantly reduce the quality of the solution. Furthermore, a centralised mechanism implies a certain hierarchy and authority structure. While this structure is present in most military contexts, there are systems where having an authoritative agent might be unwelcome.

- *Ship importance:* We have seen that in some cases some ships are more important than other. A ship with higher importance could be a commanding ship, an escorted supply ship, a coalition ship, etc. Some mechanisms deal with the relative importance of ships, while others do not take into account these ranking methods.

- *Backup plan:* In a stochastic environment such as our project NEREUS, it is important to be able to implement and use backup plans, in case where an agent is unable to take care of its assigned tasks. Thus, this attribute represents the possibility of integrating such backup plans in the mechanism.

	Central coord.	Contract Net	∼Brown
Number of comm.	$m + n\%(m+1)$ $+2(m-1) \cdot \left\lceil \frac{n}{m} - 1 \right\rceil$	$n \cdot (2m-1)$	$m(m-1) \left\lceil \frac{n}{m} \right\rceil$
Centralised	✓	✓	
Ship Importance			✓
Backup plans	✓	✓	✓
Completeness	✓	✓	✓

Table 2: Comparing coordination mechanisms.

- *Completeness:* In project NEREUS, it is unacceptable to let threats reach ships unimpeded. Therefore, the completeness of a solution (whether every task is distributed) is important.

The evaluation of these factor for each coordination mechanism is presented in Table 2, which compares the different coordination mechanisms. In this table, m is the number of frigates, n is the number of threats and p is the number of times a contingency arises.

4.1 Metrics

The metric used to evaluate the different coordination mechanisms is the *efficiency*, which uses the *total number of communications*, the *total number of SAMs used* and the *survival rate*. As discussed in Section 3.4, we can add an α parameter to the efficiency evaluation to stress the importance of survivability versus the used resources. We looked at two different efficiency measure: the efficiency according to the SAMs launched and according to the total size of messages sent.

In figures below, we compare the results for different coordination mechanisms. We used many different values for the various parameters described earlier, and we averaged the results to get a good idea of the performances of the mechanisms. However, since the *frigate per threat* parameter had a too great impact on the results of the Contract Net mechanism, we also included the results for the Contract Net protocol with only one frigate per threat, which is noted "Contract Net*"in the figures. The relative differences between "Contract Net*" and "Contract Net" provide an idea of the performances that could be obtained if we adapted the other mechanism to permit assigning a threat to more than one frigate.

Figure 9 presents the *efficiency* of the mechanisms, according to the utilization of SAMs, while Figure 10 presents the efficiency according to the utilization of communications. In the first figure, Contract-Net provides the best efficiency since it considers only one threat a time while it consumes less communications than others. In the second figure the central coordinator has a good efficiency, which can be explained by the fact that it does not use much communications. However, we should introduce an α parameter in the calculus of efficiency, as presented in Section 3.4. Indeed these results are not biased enough toward the importance of survivability. Thus, applying an α factor would allow to correctly put the emphasis on the survivability, which is our primary concern.

We should note here that the *zone defence* is just an organizational way based on social laws where there is very few communication and it is given here

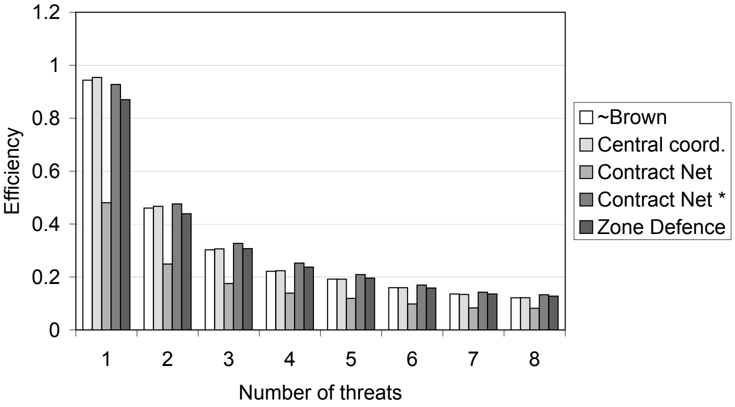

Figure 9: Efficiency relative to SAM use, considering different coordination mechanisms.

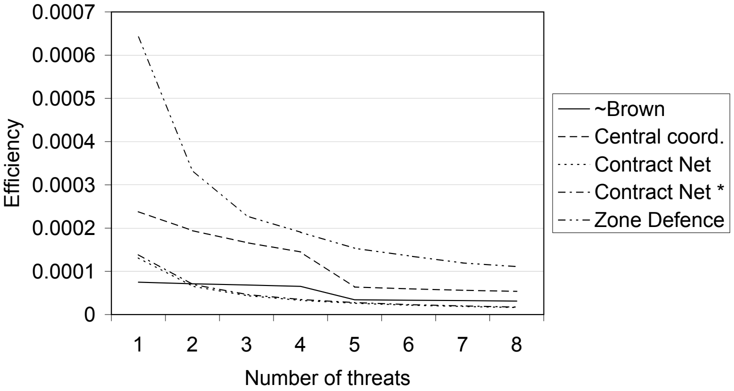

Figure 10: Efficiency relative to communications use, considering different coordination mechanisms.

just as a comparative mechanism.

5 Discussion

Inthis article, we have seen different coordination mechanisms to address the specific problem of task distribution. We have less conflicts by using task sharing. However, we have deliberately left open the question of *managing interactions*. What is the best way to avoid negative interactions while promoting positive interaction? An method based on social rules is a possible way to solve this problem. In fact, social laws coordination mechanisms can obtain very good results without as much communications as in communication-based approaches. Moreover, the navy already uses standard operating procedures, doctrines and rules of engagement. These are all social laws, with more or less importance via their firmness and sanctions that apply for breaking them.

Other open aspects are the evaluation of P_S. We presented in this chapter an heuristic to evaluate those probabilities of success. However, it is a very simple heuristic, and we have reasons to believe that is quite inaccurate. First, the evaluation of the first parameter, P_K, is an extrapolation that is probably too simple. Secondly, the parameter based on the CPA is oversimple as it cannot appropriately model the complexity of the different systems on a typical frigate. An heuristic considering each system independently would probably be closer to the real probabilities of success and allow for better coordination.

Furthermore, there is still researches to be done on the use of backup plans. How could we introduce such safeguard in the planning? What would be the importance of backup plans? Is it a good idea to sacrifice actions in the constructed plan to keep some backup actions in case another agent fails? Incorporating safeguard actions in plans is time consuming and blocks resources that could have been used elsewhere. Another interest of backup is to plan actions to protect *ourselves*. In the case where a threat directed at us in engaged by agent-x, how far do we trust agent-x to be able to defend us? An agent may be cooperative but still fail in its tasks.

We showed in this chapter that the communications channels were not as used as we first expected them to be. Therefore, the communications seem not to be as problematic as we first imagined. However, communication-based mechanisms are especially hard to scale up. Will these mechanisms be as successful if we double the number of threats and frigates? On the other hand, it is usually easier to scale up mechanisms with social laws approaches; thus, we could develop new coordination mechanisms based on social laws.

Moreover, an interesting modification that could be done is on the calculus of the fleet priority. Currently, we consider the threat's target and the weight of this target. However, in real-life, there are various kind of threats. Some missiles are far more sophisticated than others, and therefore more dangerous.

Finally, we have presented results where more than one frigate engage the same threat. The results were pointing that this behavior is desired as it increases the survivability. Thus, still in a *task sharing* setting, we would simply assign the same task to more than one agent. However, there is work still to be done on that redundancy in engagements. There need to be some way to prioritize the threats differently when constructing plans, so to make sure that the threats are engaged uniformly (i.e., so that not every resource are pitted against some threats while other threats get engaged with only few actions).

6 Acknowledgement

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada, by Lockheed Martin Canada and by Defence R&D Canada.

References

[1] Dale Blodgett, Pierrick Plamondon, Brahim Chaib-draa, Peter Kropf, and Eloi Bossé. A method to optimize ship maneuvers for the coordination of hardkill and softkill weapons within a frigate. In *7th International Command and Control Research and Technology Symposium (7th ICCRTS)*, Québec, QC, September 2002.

[2] Chris Brown, Peter Fagan, Angela Hepplewhite, Bob Irving, Dick Lane, and Emma Squire. Real-time decision support for the anti-air warfare commander. In *6th International Command and Control Research and Technology Symposium (6th ICCRTS)*, U.S. Naval Academy, Annapolis, Maryland, June 2001.

[3] Chris Brown and Dick Lane. Anti-air warfare co-ordination - an algorithmic approach. In *Command and Control Research and Technology Symposium*, Monterey, CA, USA, October 2000.

[4] Jean-François Morissette, Brahim Chaib-draa, Pierrick Plamondon, and Dale Blodgett. A method for reducing complexity of frigate movement for resource planning in anti-air warfare. Forthcoming.

[5] Pierrick Plamondon. A frigate survival approach based on real-time multi-agent planning. Master's thesis, Computer Science & Software Engineering Department, Laval University, 2003.

[6] The Foundation for Intelligent Physical Agents. Fipa contract net interaction protocol specification. Component 00029, FIPA, December 2002. Interaction Protocols.