FFI-rapport 2013/00861

Modelling battle command with context-based reasoning

Rikke Amilde Løvlid, Anders Alstad, Guro Skogsrud,
Solveig Bruvoll, Ole Martin Mevassvik
and Karsten Bråthen

FFI Forsvarets forskningsinstitutt

FFI-rapport 2013/00861

Modelling battle command with context-based reasoning

Rikke Amilde Løvlid, Anders Alstad, Guro Skogsrud, Solveig Bruvoll,

Ole Martin Mevassvik and Karsten Bråthen

Norwegian Defence Research Establishment (FFI)

27 August 2013

FFI-rapport 2013/00861

1233

P: ISBN 978-82-464-2286-2
E: ISBN 978-82-464-2287-9

Keywords

Agenter

Modellering og simulering

Approved by

Karsten Bråthen	Project Manager
Anders Eggen	Director

English summary

An important aspect of simulation based training is the need for realistic computer generated forces. In typical systems for computer generated forces the entities can be instructed to perform simple tasks like “move along route” and “move into formation”. Our objective is to make a simulation system that is capable of simulating the execution of a higher level operational order autonomously. In order to do this, the simulation system will have to understand and plan how to execute higher level commands like "seize area x" or "support unit y by fire", and be able to react to unplanned events according to doctrine. Such a system can be used both for training purposes and as a support tool when planning military operations.

The simulation system consists of a multi-agent system together with a commercial off the shelf system for computer generated forces. Knowledge about tactics and doctrine is modelled in the multi-agent system, where the agents are organized in a hierarchy representing military leaders and staff. The focus in this report is to explore how we can use the human behaviour modelling paradigm context-based reasoning to model the behaviour of the battle command agents. Three challenges not addressed by previous work on context-based reasoning were identified, and possible solutions are suggested in the report. The three challenges concern how to define contexts for battle command agents, how the agents plan their tasks and how higher level tasks are decomposed through the agent hierarchy.

A behaviour model based on an example military operation was developed with assistance from subject matter experts in order to illustrate the concept and to reveal challenges and further research questions. Our main conclusion is that context-based reasoning seems well suited for modelling the behaviour of battle command agents, both because it seems easy to gradually expand the model with different or more complex behaviour, and because the resulting model appears easy to understand and validate by subject matter experts.

Sammendrag

En viktig del av et simuleringsverktøy for trening og planlegging er simulerte enheter med realistisk oppførsel. Typiske systemer for datagenererte styrker er i stand til å simulere enkel oppførsel hvor enhetene utfører lavnivåoppgaver som “følg rute”, “gå i formasjon”, osv. Vårt mål er å lage et simulseringssytem som er i stand til autonomt å simulere utførelsen av ordre på høyere nivå. Dette vil kreve at simuleringssytemet kan forstå og planlegge utførelsene av høyere nivå oppgaver som “ta område x” eller “støtt enhet y med ild”, og at systemet er i stand til å reagere doktrinelt på uventede hendelser. Et slikt system kan brukes både for trening og øving, men også som et støtteverktøy under planlegging av militære operasjoner.

Simuleringssystemet vi utvikler består av et multi-agent system sammen med et kommersielt tilgjengelig system for datagenererte styrker. Kunnskap om taktikk og doktrine modelleres i multi-agent systemet. Agentene er organisert i et hierarki og representerer ledere og staber på forskjellige nivåer. Fokuset i denne rapporten er å studere hvordan stridsledelsesagentenes oppførsel kan modelleres med adferdsmodelleringsparadigmet kontekstbasert resonnering. Spesielt har vi identifisert tre utfordringer som ikke er behandlet i tidligere arbeider med kontekstbasert resonnering. De tre utfordringene omfatter hva som er fornuftige kontekster for en stridsledelsesagent, hvordan oppgaver planlegges og hvordan oppgaver på høyere nivå kan brytes ned gjennom agenthierarkiet. Mulige løsninger på disse utfordringene er foreslått i denne rapporten.

For å illustrere konseptet og belyse videre utfordringer har vi med hjelp fra offiserer modellert handlingsmåter som er nødvendige for å utføre en eksempeloperasjon. Hovedkonklusjonen vår etter dette arbeidet er at kontekstbasert resonnering egner seg godt til å modellere stridsledelse, både fordi det tilsynelatende er enkelt å utvide modellen gradvis med ny eller mer kompleks adferd, og fordi resultatet synes å være enkelt å forstå for militære eksperter.

Contents

Acknowledgements

We would like to direct a special thanks to Ltcol Jan Harry Pay and Ltcol Geir Karslen for providing subject matter expertise. They were always available to answer our questions and without them the agents behaviour would have been much less realistic. Also we would like to thank Professor Avelino Gonzalez from University of Central Florida for introducing us to context-based reasoning and assisting us in applying this modelling paradigm.

1 Introduction

The use of simulators, computer generated forces (CGFs) and serious games for military training and operational planning is becoming increasingly important [1]. Simulation-based training can be cost effective and time efficient. Additionally, it makes it possible to train in scenarios that would not be feasible in real life, either because they require large areas, lots of employment, because they could be highly dangerous or because they are politically sensitive. Simulation-based training is also easier to set up, repeat and control than real life training.

An important aspect of simulation-based training is the need for realistic computer controlled entities. In systems for computer generated forces the entities can be directed to perform simple tasks like “move along route” and “move into formation”, and it is possible to make scripts that prescribe predetermined actions upon a specific set of events [2]. This for example makes it possible for one operator to control an entire company in an Army Computer Assisted Exercise (CAX).

The introduction of digitized plans, orders, reports and requests, i.e. the Coalition Battle Management Language (C-BML) [3], calls for autonomous simulation of military tasks used at the Army battalion level and above. In order to do this the simulation system will have to interpret, plan and execute higher level commands like “seize area x” or “support unit y by fire”, and be able to react to unplanned events according to doctrine. Such a system can be used, not only for training purposes, but as a support tool when planning military operations and to improve communication between leaders and subordinates [4].

The objectives of the work documented in this report were to explore how to use a multi-agent system to simulate battle command and how to model tactics and doctrine within the artificial intelligence (AI) framework context-based reasoning (CxBR). CxBR is a modelling paradigm specifically designed for representing human tactical behaviour [5, 6, 7], and it is based on the idea that humans only use a small portion of their knowledge at any given time. Which actions an agent can select from and what sensory input it should care about depend on which context it is in. The appropriate context is decided based on the overall goal for the agent and the current situation.

In our research we wanted to use CxBR to model battle command in a hierarchy of agents representing military leaders and staff. The attempt to apply CxBR to a hierarchy of battle command agents led to three main challenges that had not been addressed by previous applications of CxBR. First, what are the contexts for a battle command agent? Second, how can a battle command agent make a plan for its assigned task in the CxBR way of thinking? Third, how can we utilize CxBR to decompose higher level military tasks through a hierarchy of agents? These are the questions we will try to answer in this report. Also, in order to demonstrate and test the proposed method, we have modelled key decisions made by company and platoon commanders based on an example battalion order and used the results to reveal challenges and further research questions.

An explanation of the overall architecture and goal of the system is given in section 2. Section

3 describes the artificial intelligence techniques CxBR and intelligent agents, while section 4 explains how these techniques are used focusing on the three challenges stated above. An example scenario is presented in section 5 together with a detailed documentation of the behaviour models. Finally, section 6 discusses the model and considers limitations and future work.

2 Simulation in support of planning

The process of planning a military operation consists of five steps [8]. First, a preliminary analysis of the situation makes sure necessary preparations are initiated, defines guidelines for the proceeding planning process, informs the affected personnel and establishes a schedule for the remaining planning process. Step 2 consists of determining *what to do and why* based on what is known about the enemy, own forces, weather, time frame, etc. The results from step 2 form the basis for the course of action (COA) analysis, i.e. *how* to solve the mission. The different COAs are evaluated in step 3 based on alternative expected COAs for the enemy, a comparison of strengths and weaknesses of own and enemy forces, available resources, etc. In step 4, finalizing the order, wargaming is an important part. Wargaming means going through the order step by step to make sure all parts of the plan are coordinated and synchronized. "Fast wargaming" can also be conducted during the COA analysis in step 3, but the general use of wargaming is limited because current methods is time and personnel consuming. When the plan has been developed, the mission is rehearsed. In the last and final step the plan is verified by taking into consideration how the situation has developed during the planning process.

A simulation system that is able to interpret a battalion order and carry out a simulation of the operation autonomously in a synthetic, natural environment can be used for COA. It will enable the battalion commander to simulate the execution of his plan multiple times during step 3 of the planning process, trying out different COAs against different enemy COAs. The idea is that simulation-based COA analysis will be much faster and require fewer resources (like personnel) than traditional wargaming, which means that it can be applied more frequently than current methods. Simulation can also be used in step 4 to support wargaming of the whole plan or part of the plan, and as a tool to perform mission rehearsal. Results from the simulation, e.g. a video, might be included as attachments to the final order illustrating the commanders intent.

C-BML is a technology that will provide a seamless interface between simulation systems and command and control information systems (C2IS). It covers digital plans/orders in addition to reports and requests created during mission execution. C-BML is under development by Simulation Interoperability Standards Organization (SISO) [9]. Military Scenario Definition Language (MSDL) is a complementary technology that can be used for coherent initialization of systems [10].

One application of C-BML is to serve as an interface between a C2IS and a simulation system for simulation-based COA analysis. If the C2IS is used to capture the main aspects of a COA (a plan), the simulation system will be easy to use, the battalion commander and his staff will not

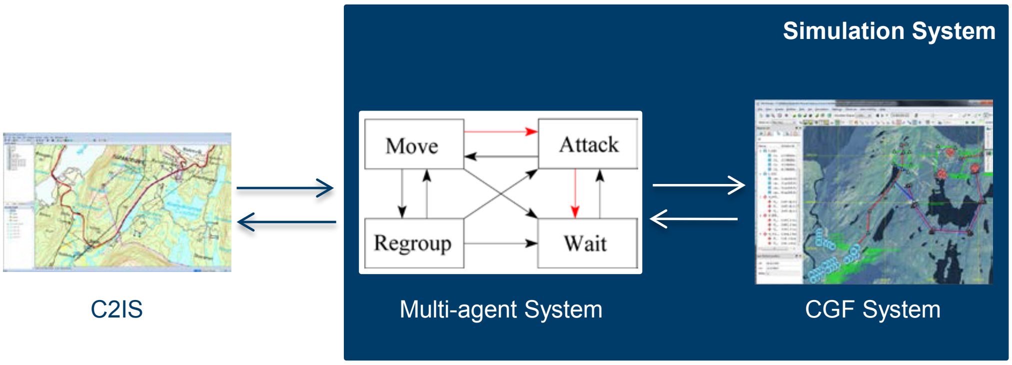


Figure 2.1 The MAS gets as input an operational order at the battalion level and produces commands to the CGF system. The agents' actions are influenced by reports from the CGF system. Reports are sent back to the C2IS.

have to learn a new system, and the final plan does not have to be transferred to another system. When the simulation system has received the order[1], no human interaction should be needed. The simulation system interprets the order, decomposes the ordered task into lower level tasks and reacts to unplanned events, and sends reports back to the C2IS. The battalion commander can observe the simulation in the C2IS and evaluate the results in order to improve his plan.

The architecture of the total system, consisting of a C2IS and a simulation system, is shown in figure 2.1. The simulation system consists of a commercially available CGF system, VR-Forces, together with a FFI-developed multi-agent system (MAS) based on CxBR. The MAS commands the entities in the CGF system and makes decisions based on reports from the CGF system. Knowledge about Norwegian tactics and doctrine is modelled in the MAS, so that the CGF system only needs to include models of low level behaviour. The motivation behind building a MAS separate from VR-Forces is to make it easy to replace VR-Forces with another CGF system. The interface between the CGF system and the MAS is documented in [11]. A description of earlier work by FFI, with focus on BML interfacing in existing C2IS, can be found in [12].

3 Applied techniques from artificial intelligence

Decomposition and simulation of tasks given in a battalion order can be performed in three steps. First, the simulation system has to make sure the tasks in the order begin and end as scheduled. Second, each task must be planned based on the current, perceived situation and terrain information. Third, the simulation system must be able to react to unplanned events and possibly replan, within the constraints given by the order, as more information is received during mission execution. To realize these requirements, two techniques from the field of artificial intelligence have been used, *intelligent agents* and *context-based reasoning*.

[1]Plan and order is used interchangeably in this report, order is sometimes used to not confuse a military plan with planning made by agents.

3.1 Intelligent agents

An *agent* is an autonomous entity that observes through sensors and acts upon its environment using actuators in order to meet its design objectives. To be called intelligent, an agent also has to be reactive, proactive and social; meaning it must be able to react to changes in the environment, pursue goals and be able to communicate with other agents [13].

A multi-agent system (MAS) consists of a number of agents that communicate with each other. Each agent will influence different parts of the environment, and the agents are linked by some kind of organizational relationship. The agents in a MAS can be identical (homogeneous MAS) or different (heterogeneous MAS), and they can be cooperative or self-interested. Motivations for using a MAS can be to solve problems that are too large for a centralized agent alone, to allow interconnection and interoperation of multiple legacy systems, or to offer conceptual clarity and simplicity of design.

In the MAS developed by FFI, a hierarchy of intelligent agents decomposes the order from company level tasks to low level CGF commands. The MAS consist of one agent for the battalion, one for each company in the battalion and one for each platoon. The agents represent the commanders and possibly staff of these military units, and model the planning and decision making done by these leaders. Using a MAS for this task makes the design clear and simple, and it becomes easy to understand for military experts. Simulation of the real chain of command also prepares the system to be used for other task, like studies of communication in the hierarchy.

3.2 Context-based reasoning

Context-based reasoning (CxBR) is used to model the behaviour of the intelligent agents. The motivation behind CxBR is the realization that people only use a fraction of their knowledge at any given time. The idea is to divide the knowledge into contexts in order to limit the number of possibilities for the action selection process. The following gives a short introduction to CxBR, including explanations of some essential concepts. A more extensive description can be found in [5].

Contexts are organized in a hierarchy consisting of a *mission context* and *major* and *minor contexts* as illustrated in figure 3.1. The mission context is a purely descriptive context, meaning it does not describe behaviour. A mission context contains a goal and a plan for reaching it together with parameters associated with the mission like objective area, phase-line, route, etc. It also contains a *context map*, as illustrated in figure 3.2, where all possible transitions between the major contexts are defined. A plan consists of a sequence of major contexts along with objectives.

Major contexts constitute the next level in the context hierarchy and are the ones controlling the agent. There is only one major context in control of the agent at any time, called the *active context*. A major context basically contains three kinds of knowledge: *action knowledge*, *transition knowledge* and *declarative knowledge*.

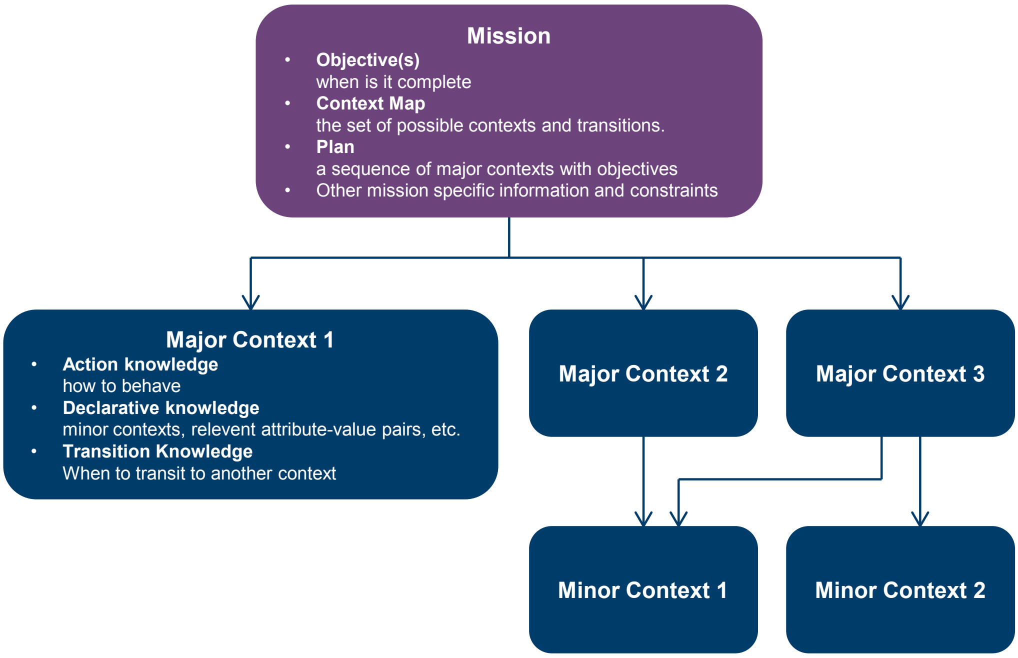


Figure 3.1 A hierarchy consisting of a mission context, major and minor contexts describes the behaviour of the agent.

Action knowledge concerns how an agent should behave in this context. Since the agents are battle command agents, the actions are commands to the subordinates, reports to their superior and possibly reports and/or requests to other agents at the same echelon. If a part of the behaviour is shared with other major contexts, this behaviour should be expressed as a *minor context*. A minor context only controls the agent for a short period of time. There can be unlimited levels of minor contexts, but one or zero should be sufficient. Minor contexts are not used in the example model presented in this report.

Knowledge on when to switch to another context is collected in the transition knowledge. This includes recognition of a situation leading to deactivation of the active context and activation of a better suited context. This knowledge can be contained in transition rules, with criteria for when the agent makes the transitions defined in the context map. The transition rules consist of both planned transitions and general doctrinal reactions, and should include transition to a default context when no other context is applicable.

Declarative knowledge includes other properties of the context, e.g. parameters like route and preferred formation, and a list of possible minor contexts.

All agents can access information in a global fact base with information available to all agents, and a local fact base containing information which only this agent is aware of. The global fact base can for example include a map of the operation area or the general situation in the battlefield (i.e. intelligence information). Information that should not be available unless explicitly communicated are stored in each agent's local fact base. This could for example be the agent's active context,

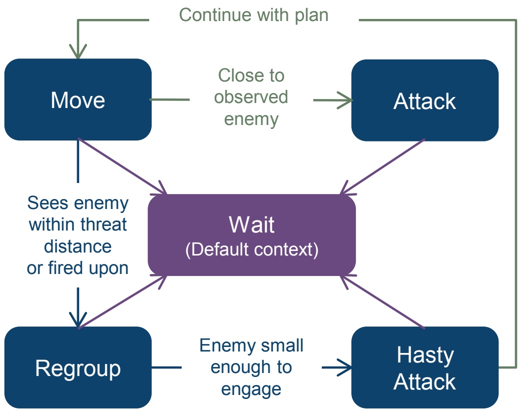


Figure 3.2 A context map defines all possible contexts and the transitions between them.

damage state, fuel level etc. Note, that what information should be available for all agents and what should be stored locally is up to the designer of the system and depend on the need for explicit simulation of communication channels.

How we have used CxBR to model the behaviour of a hierarchy of battle command agents is the topic of the following section.

4 CxBR for a hierarchy of battle command agents

The agent hierarchy is illustrated in figure 4.1. The battalion agent receives an order with company tasks from the C2IS and makes sure the companies receive their tasks as scheduled in the order. Each company agent makes a plan for its received task and commands its subordinate platoons. The platoon agents break their tasks further down into simple commands for the platoon aggregates in the CGF system. Units below platoons (i.e. squads, vehicles, soldiers) are not represented in the MAS, and the decomposition of platoon tasks to tasks for single entities is handled by the CGF system, which means all units in the platoon will get equivalent tasks. It is possible to extend the MAS to include agents also at lower levels or to implement more sophisticated models in the CGF system.

The CGF platoon aggregates report back to the platoon agents about observed enemies, their task status, position, damage, fuel level, etc. This makes it possible for the agents to monitor the simulation and react to events in the simulation. For example if an unexpected enemy is observed, the agents will change their plans according to the situation, not blindly follow through with the received plan. The agents also send reports to their superiors, and perceived truth of both own and enemy position are reported back to the C2IS [2].

Former applications of CxBR have been limited to lower level entities with concrete actions like “drive”, “stop”, “turn right”, and major contexts like “suburban driving”, “free-way driving” etc.,

[2]In addition ground truth is sent from the simulation system to the C2IS. This is possible because it is a simulation and everything is available. One can then decide in the C2IS what should be visible for different users.

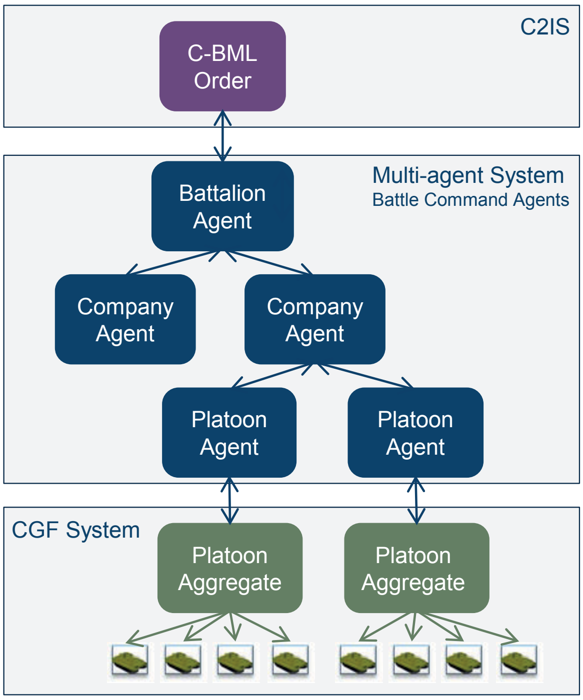


Figure 4.1 The agent hierarchy.

and to our knowledge, the paradigm has not been used in a MAS with more than two agents. In the introduction we described three main challenges that needed to be addressed in order to apply CxBR for a MAS of higher level battle command agents organized in a hierarchy. The challenges were defining what are the contexts of battle command agents, developing a strategy for how the agents can plan their mission context as a sequence of major contexts, and figure out how higher level tasks can be decomposed through the agent hierarchy. In this section we will present our model with focus on how we have solved these issues.

The basic idea behind our model is that *the behaviour at different levels in the agent hierarchy is basically the same*. This is based on the assumption that a military task is carried out in similar patterns at different levels in the military hierarchy, only at different scales in time and space. For example the tasks "seize" and "attack" mean the same for both a company and a platoon commander. They will both assign different task (e.g. reconnoitre, support etc.) to the different subordinates, or divide the attack area into smaller areas for each of the subordinates. How they carry out the task and how they divide it between the different subordinates depend on the terrain and their resources, e.g. the relative strength and manoeuvrability of the subordinates, and not whether they are commanding a company or a platoon.

We have tried to keep the whole CxBR model as simple as possible by implementing equivalent behaviours at different levels and reusing major contexts for different mission contexts. Also we have related contexts to military terms in order to make it easy for subject matter experts to understand and validate the model. All task verbs in the operational order have a corresponding mission context in CxBR terminology. The mission context is planned as a sequence of major

Figure 4.2 The figure illustrates what is communicated between a battle command agent and its superior, subordinates and peers.

contexts, usually by the agent who receives the mission. The major contexts represent tactics, or more precisely management of tactics, meaning how to organize the subordinates, which tasks to give them and how they should be synchronized in order to realize a tactical pattern. The agents react to unplanned events by firing transition rules, which change the agents' contexts to more appropriate contexts according to doctrine.

4.1 Major contexts

As explained in section 3.2, major contexts consist of action knowledge, transition knowledge and declarative knowledge. The actions for a battle command agent is commanding subordinates, sending reports to the superior and communicate with agents at the same echelon. The commands can be missions commands, resulting in new mission contexts for the subordinates, or coordinating instructions like changing formation or rules of engagement. Reports include status updates, mission completed reports and reports about observed enemies. Communication with agents at the same level is for example necessary when one is tasked to support another. The communication between a command agent and its superior, its subordinates and its peers in the current model, are illustrated in figure 4.2.

Since major contexts represent tactical management, a command agent will transit to a new major context when the tactical pattern needs to change. The transition will result in new missions to all the subordinates. The agent might of course send a mission command to a subordinate independent of transiting to a new major context, as sending a command is considered an action, but major synchronization points will be represented as context transitions in the missions plan.

However, what to consider a "tactical pattern" and thus a major context, is not obvious. Imagine one agent having three subordinates. The agent's unit is approaching an enemy unit, and the agent orders one subordinate to "fix"[3] two enemy entities, which are separated from the rest of the

[3] The military task "fix" is defined as "prevent an enemy from moving any part of his forces from a specified location for a specific period of time" [8].

enemy unit. Another subordinate is ordered to move to a lookout position and be ready to support the third subordinate with fire as it moves in and attacks the enemy unit. What is the agent's major context? The question is how to define an agents major context when its subordinates are performing different types of mission contexts.

The small scenario described above can be viewed as one attack-tactic. Since major contexts are related to tactics, one can say that the agent is in major context *Attack*, and that this context contains action rules that are used to realize different types of attacks. Alternatively one could define a major context *Fix-attack-with-support*, which implies that the subordinates will be given three different roles. How specific the contexts should be depends on the level of detail in the simulation and is a trade-off between context complexity and planning complexity. Few contexts make planning easy, but the intention of CxBR is to divide the behaviour in order to limit the number of decisions the agents need to consider at any time, which means the contexts should not be too large. Splitting the context into more specific contexts will on the other hand require more from the planning and re-planning mechanisms. The challenge is to find the right balance.

We have kept both mission and major contexts independent of agent types, which means the content of the contexts are identical whether the contexts are used for platoon or company agents. Using the same contexts at all levels is in accordance with the assumption that the same behavioural patterns, i.e. tactics, are applicable at all levels of the military hierarchy. However, there might be contexts that are relevant only for one type of agent.

Only major contexts are used in the current model, but as the contexts will grow in complexity we will look into expressing parts of the behaviours as minor contexts. However, using a minor context is only beneficial when a behaviour is shared among two or more contexts.

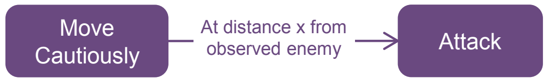
4.2 Planning

When an agent receives a new mission context, that mission is planned as a sequence of one or more major contexts. The major contexts represent sub-goals, and an agent will proceed to the next major context in the plan when the goal of its active context is fulfilled. A basic plan was predefined for each possible mission context, and based on these basic plans, the planning procedure can be carried out in two or three steps: 1) Retrieve the basic plan for the current mission context, 2) adapt the basic plan to the current situation, and possibly 3) replace the general major contexts from the basic plan with more specific tactical patterns. These three steps are illustrated in figure 4.3.

4.2.1 Step 1: retrieve a basic plan

Our planning strategy is based on the assumption that it is possible to define a basic plan for all types of missions contexts, e.g. one for *Seize*, one for *Support by Fire*, etc. This basic plan expresses what the agent has to do to complete the given type of mission context and must be applicable in all situations. The basic plans currently consists of sequences of major contexts,

(a) Planning step 1: Retrieve a basic plan for mission *Seize*



(b) Planning step 2: Adapt the basic plan to the current situation.

(c) Planning step 3: Replace general contexts with more specific versions

Figure 4.3 The figure illustrates an example of making a plan for mission context Seize.

which can be interpreted as an ordered list of sub-goals or higher level actions. When the model becomes more complex, the basic plan might need to be expanded into a graph with different possible paths to success, but for now a sequence is sufficient.

We intend the basic plans to be the same for different agent types. For example, the basic plan for mission context *Seize* is the same whether it is for a platoon or a company agent. This is again based on the assumption that the behaviour at different echelons in the military hierarchy is equivalent.

4.2.2 Step 2: adapt the basic plan

Step 2 consists of adapting the basic plan to the current situation. At the moment that means pruning away major contexts that are not necessary and creating the transition rules for the remaining sequence of major contexts. If we were to have a more complex graph as a starting point, we would first have to find the best path through this graph given the current situation.

The basic plan includes some general transition rules that need to be made specific for the current situation during planning. For example could the general rule "when within attack distance of the enemy" be changed to "when on top of peak x". Alternatively, the planned transitions can be realized by adding goals to each context, specifying when it is complete, and then use a general transition rule "when current major context is completed, then go to next in plan". We have chosen the latter.

4.2.3 Step 3: replace general contexts with more specific versions

The major contexts used in the basic plan have to be fairly general to be applicable for all agents in all situations. The basic plans only specify what to do, not how to do it. The purpose of this last step is to figure out *how* to do it.

After pruning away the irrelevant contexts and specifying the transition rules, we consider the possibility of replacing some or all of the major contexts with more specific versions based on the agents capabilities, available resources, enemy observations, and terrain. An example is replacing *Attack* with *Fix Attack*. In *Attack*, all subordinates will be commanded to attack, which will work no matter how many subordinates the agent has. The context *Fix Attack* on the other hand presuppose at least two subordinates, one for fixing a part of the enemy and on for attacking another part of the enemy.

4.3 Reactive behaviour and replanning

For the agents to be useful, they must be able to handle a set of different possible events. We distinguish between reactive behaviour, replanning and mission failure.

In the current model the agents are able to react when enemies are detected or units are fired upon. Upon receiving message of such an incident, an agent will regroup, which involves ordering the subordinates to stop and possibly pull back while the agent decides how to handle the situation. How the agent should handle the situation depends on what has been observed; a small enemy unit, a large enemy unit or an unidentified unit. Whether an enemy unit is categorized as “small” or “large” depends on the size of own forces and possibly other factors like whether the terrain is more familiar to own forces than the enemy. If the enemy is considered small enough for the agent’s units to take out, the agent will order an attack. If the enemy is too large, the agent will ask its superior to handle the situation. If the observed unit is unidentified, the agent should try to obtain more information about the unit by organizing reconnaissance, but this is out of scope for the current version.

A blocked road or a destroyed bridge are possible events that can make an agent unable to continue its plan. It then needs to re-plan. If the agent is able to generate a new plan that will complete its mission, it will do so and carry on. If it cannot find a new, suitable plan, the superior agent will try to re-plan. If a company agent is unable to generate a new plan, it will just command all subordinates to *Wait*, and an updated order (fragmentary order, FRAGO) from the human commander using the system is required. We call this mission failure.

In the first version of the MAS we will not consider replanning or FRAGOs, but the MAS should be able to handle enemies which are small enough for a platoon or too large for the platoon but small enough for its company to engage. The platoon will not be able to continue its mission before the enemy is defeated, possibly with support from other platoons in the company.

4.4 Decomposing an order with a hierarchy of agents

While developing the model we kept major contexts and basic plans for mission contexts equal for platoon and company agents. We got positive feedback from military experts when presenting this idea, as they confirmed that the behaviours and tactics are basically the same at these two echelons.

However, this constrains how an order is decomposed through the agent hierarchy, from company tasks to low level CGF commands.

In this section we illustrate how we did the decomposition and compare it with two alternatives. One alternative is to exploit additional possibilities that arise when we allow different basic plans for platoon and company agents. The second alternative is to use different major contexts for platoon and company agents. We have not included an example where both basic plans and major contexts can be different, because the current model is too simple to make advantage of both these possibilities at the same time. We call the three alternative decompositions "different basic plans", "different major contexts" and "equal basic plans and major contexts". Table 4.1 illustrates these three decompositions for the three company tasks we have worked with.

One future improvement can be to identify factors that will distinguish platoon and company agents, and that need to be taken into consideration in the context content and the planning procedure. At this point, however, there are only two capabilities that distinguish these two agent types. First, a platoon agent has always only one subordinate, whereas a company agent has 1-3 subordinates. Second, the platoon agents can only send low-level mission commands to their subordinates (wait, attack and move), whereas company agents in principle do not have any such restrictions.

4.4.1 Different basic plans

In the illustrated example of using different basic plans, the knowledge that a platoon agent has only one subordinate is exploited. For example, the mission context *Reconnoitre* is planned with the context *Reconnoitre* for the company and *Move→Observe* for the platoon. To accomplish this mission, the agent has to move along some route, reporting observations along the way, and then set up an observation post and observe the target area. A company agent is not able to split the mission context *Reconnoitre* into two major contexts because the subordinates should start observing as soon as they reach the target destination, without waiting for each other. This means that for the company agent with more than one subordinate, there is a gradual transition from all subordinates moving to all subordinates observing. A platoon agent, on the other hand, commands only one CGF platoon aggregate and can therefore split the mission context into two major contexts.

4.4.2 Different major contexts

The second illustrated option is to allow the use of different major context implementations for platoon and company agents. In the Reconnoitre example the major context *Reconnoitre* for the company agents can send *Reconnoitre* missions to the platoon agents, whereas the platoon agents can command their subordinates to *Move* or *Wait* while in this context. Because platoon agents can only send low level commands to their subordinates, it would not be possible for them to send *Reconnoitre* missions to their subordinates, but company agents do not have this restriction. Note that the company agents did send *Reconnoitre* missions to their subordinates in the example of

Different basic plans

Company Mission Contexts	Reconnoitre		Support By Fire			Seize		
Company Major Context Plans	Reconnoitre		Support By Fire			Move→Move Cautiously→Attack		
Platoon Mission Contexts	Reconnoitre		Support By Fire			Move	Move Cautiously	Attack
Platoon Major Context Plans	Move→Observe		Move→Wait→Attack			Move	Move	Attack
CGF-commands	Move	Move, Wait	Move	Wait	Attack	Move	Move	Attack

Different major contexts

Company Mission Contexts	Reconnoitre	Support By Fire	Seize		
Company Major Context Plans	Reconnoitre	Support By Fire	Move→Move Cautiously→Attack		
Platoon Mission Contexts	Reconnoitre	Support By Fire	Move	Move Cautiously	Attack
Platoon Major Context Plans	Reconnoitre	Support By Fire	Move	Move Cautiously	Attack
CGF-commands	Move, Wait	Move, Wait, Attack	Move	Move	Attack

Equal basic plans and major contexts

Company Mission Contexts	Reconnoitre		Support By Fire			Seize		
Company Major Context Plans	Reconnoitre		Support By Fire			Move→Move Cautiously→Attack		
Platoon Mission Contexts	Move, Observe		Move, Wait, Attack			Move	Move	Attack
Platoon Major Context Plans	Move	Observe	Move	Wait	Attack	Move	Move	Attack
CGF-commands	Move	Move, Wait	Move	Wait	Attack	Move	Move	Attack

Table 4.1 The table illustrates three strategies for decomposing a high level company task to low level CGF commands with a hierarchy of agents. The three tasks Reconnoitre, Support by Fire and Seize *are expressed as company mission contexts, and broken down to a basic plan consisting of company major contexts. Basic plans are shown as major contexts connected with arrows. In each of these major contexts mission contexts are sent to the subordinate platoon agents. When a company major context includes action rules for more than one type of platoon mission context, the platoon mission contexts are expressed as a list (e.g. Move, Observe), and basic plans for each of these mission contexts are illustrated in separate columns in the next row. The CGF commands are low-level commands for the CGF expressed as mission contexts. The differences between the different strategies are whether the company and platoon mission contexts are the same and/or company and platoon major contexts are the same.*

different basic plans as well, but in that example, the platoon would never be in major context *Reconnoitre*, because of how the mission context *Reconnoitre* was planned for platoon agents.

4.4.3 Equal basic plans and major contexts

The third option is to use the same major contexts and the same basic mission context plans for all agents, i.e. the option “equal contexts and plans”. We have chosen this solution, because we believe the behaviour and decision making at company and platoon level have very much in common, and that this strategy will avoid duplication. Also the two differences between platoon agents and company agents at the moment, i.e. the possible number of subordinates and limited number of commands available for the platoon agent, might change in the future.

The downside of using the same major contexts and basic plans for all agents is that the major contexts and the planning procedures must contain more complex rules, which consider the resources and capabilities of the agent. Some of these resources and capabilities (like the number of subordinates) could have been assumed if the major context or the planning procedure were specially designed for a company, a platoon, or maybe even for a specific type of platoon or company, e.g. tank or mechanized infantry. However, if we include reasoning about resources and capabilities in the planning procedures and major contexts, the planning procedures and major contexts will still work when a unit is damaged or has lost some of its resources.

As for now the model is very simplistic, and all options would work, but it is hard to predict how these strategies will work as the complexity of the model increases. Because of this we do not rule out having to use different basic plans or major contexts in later versions.

5 Modelling behaviour based on an example scenario

An example scenario was used to limit the behaviours that had to be modelled for a first version of the simulation system. The scenario was also used as a guide to what functionality is needed in the agent framework and the CGF system in order to handle a simple military operation. The example scenario is explained in the succeeding section, and section 5.2 will present the corresponding behaviour model.

5.1 Scenario

The example scenario covers the first part of an offensive military battalion operation, in which a vanguard of the enemy force is taken down. The scenario consists of the task organization shown in figure 5.1 describing the entities and the structure of the battalion, an order consisting of a set of synchronized tasks shown in table 5.1, a map of the area, shown in figure 5.2 and scripted red forces, i.e. the enemy. The agent hierarchy and the entities in VR-Forces will be created from the task organization, and the CxBR model is designed to handle the tasks in the order. The resulting agent behaviour depends on the terrain and the behaviour of the red forces.

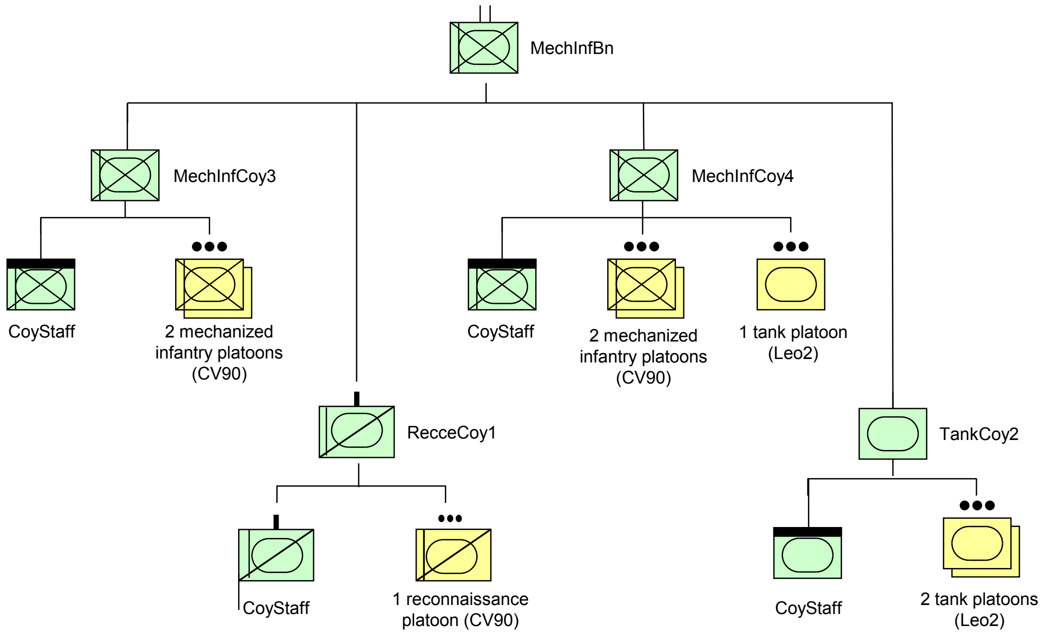


Figure 5.1 The task organization.

5.1.1 Task organization

The task organization includes the blue forces that are to conduct the operation. The example order is a battalion operation, and the task organization shown in figure 5.1 contains the structure of the Mechanized Infantry Battalion (MechInfBn), which will execute the operation. The task organization defines the companies in the battalion, and the platoons of each company. The agents in the MAS represent commanders and their staff, while the CGF system simulates the combat units. For simplicity dismounted infantry is not represented in the CGF system nor is the company staff. The combat units are marked yellow in figure 5.1.

The MechInfBn consists of four companies, of which three play a role in the scenario. The first company is Reconnaissance Company 1 (RecceCoy1), which consists of one infantry fighting vehicle (CV90) platoon with four vehicles equipped for Reconnaissance. The second company is Tank Company 2 (TankCoy2), consisting of two tank platoons with four main battle tanks (Leo2) in each platoon. TankCoy2 is only used as backup in this part of the operation. The third company is Mechanized Infantry Company 3 (MechInfCoy3), which has two mechanized infantry platoons, each consisting of four CV90s. The last company is Mechanized Infantry Company 4, which has three platoons with four vehicles in each. Two of the platoons are mechanized infantry platoons consisting of CV90s, and the third is a tank platoon consisting of Leo2s.

5.1.2 Operation order

The basic concept of the operation order is to approach the enemy along the road that exits the map near the upper right corner expressed in figure 5.2. There is an enemy vanguard at area 102. This area must be seized before own forces can continue towards the main enemy position.

Company	Task		
Recce-Coy1	**Reconnoiter** axis from SL to 102	Keep 102 under surveillance	
MechInf-Coy3		**Seize** objective area 101	**Support by fire** MechInfCoy4 (towards 102)
MechInf-Coy4			**Seize** objective area 102

Table 5.1 The synchronization matrix.

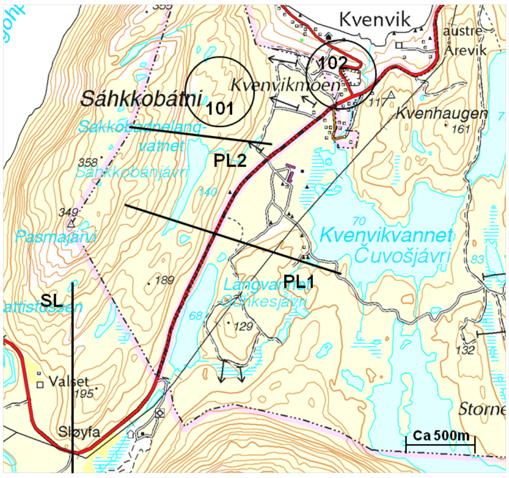


Figure 5.2 The map with defined areas and phase-lines from the OPORD.

The overall goal of the example order is therefore to seize area 102 in the map in figure 5.2. This operation is to be carried out by the MechInfBn consisting of the companies RecceCoy1, TankCoy2, MechInfCoy3, and MechInfCoy4, as depicted in the task organization in figure 5.1. TankCoy2 is not directly involved in this operation, but is available for backup if necessary.

The battalion operation order [4] with tasks for each of the three involved companies is shown in table 5.1. The operation starts with reconnaissance. RecceCoy1 is tasked to look for activity along the road leading to the objective area 102. The objective is to obtain an overview of the area and the situation before initiating an attack. When the reconnaissance is finished, the company moves to a secure position north of area 101, from where area 102 can be observed. Thereafter MechInfCoy3 moves to area 101 and seizes this area. When area 101 is secured, MechInfCoy3 should find a suitable position in area 101 from where it can support MechInfCoy4 in seizing the objective area 102. Finally, MechInfCoy4 moves along the road towards area 102 and seizes the area with support from MechInfCoy3.

5.1.3 Red forces

The operation order does not include a detailed task organization for the red forces or information about their location. However, an order is based on assumptions about the composition of enemy forces, their location and expected actions. Although these assumptions are not explicitly expressed in the order, tasks like *Seize* and *Support By Fire* imply that enemies presence are expected in the objective areas (OAs) tied to these tasks, and that the enemy forces located there are small enough for the tasked unit to engage.

Currently the red forces are created in the CGF system, and their behaviour is scripted. In the future a separate MAS may be used to simulate red forces. Variations of the composition and behaviour of the red forces can be used to reveal strengths and weaknesses of the behaviour model.

[4] Only tasks for manoeuver units was used by the simulation (mission execution part of an operation order).

5.1.4 Desired behaviour for planned events

In cooperation with subject matter experts (SMEs) we have determined how the three companies should execute the ordered tasks. This solution is based on the terrain and available resources, i.e. the number of platoons and the platoon types.

To compensate for the lack of automatic terrain analysis in the current system, more control measures must be included in the order than what is normally done. The order includes routes to all the platoons and phase-lines to indicate where to start spreading out before a planned attack.

The following outlines the desired behaviour if there are no unplanned events, e.g. no enemies outside the target areas 101 and 102.

5.1.4.1 RecceCoy1

RecceCoy1 reconnoiters the axis from the start line (SL) to the objective area 102 by following its route through area 101. From this route in the hillside, the company has a good overview of the road and the surrounding area. Because of the terrain, the company moves in two columns formation instead of in line formation. The company looks for enemy activities in the area and reports to the other companies. RecceCoy1 also checks the terrain for accessibility, and the findings can be used by other companies for route planning. When the company reaches the end of the route, it creates an observation post at a suitable place for keeping area 102 under surveillance throughout the rest of the operation.

5.1.4.2 MechInfCoy3

MechInfCoy3 follows the same route as RecceCoy1. Knowing that the route has been checked for enemies and obstacles, they move rather fast in column formation. When MechInfCoy3 approaches the objective area 101, it crosses phase line PL2. This is the sign for preparing to attack potential enemies positioned in the objective area 101. After seizing area 101, it secures the area and moves into position for overviewing the objective area 102, waiting for a signal to support MechInfCoy4 by fire during their attack on area 102.

5.1.4.3 MechInfCoy4

MechInfCoy4 moves along the road at high speed knowing that RecceCoy1 has reconnoitred the axis without seeing any enemies along the road. When MechInfCoy 4 has passed the canalizing terrain along the lake, i.e. reached PL1, it spreads from column formation to line formation with the tank platoon in the middle. Each platoon changes to two columns formation. When area 102 is within weapon range, the attack begins, with support from MechInfCoy3. The tank platoon can attack from a larger distance than the mechanized infantry platoons, which continue moving towards 102 at each side of the tank platoon while firing. All platoons move into the objective area 102 when they cannot affect it from outside.

	Name	Used by Company Agents	Used by Platoon Agents	CGF Units
Mission Contexts	Reconnoitre	yes	no	no
	Seize	yes	no	no
	Support by Fire	yes	no	no
	Observe	no	yes	no
	Move	no	yes	yes
	Attack	no	yes	yes
	Wait	no	yes	yes
Major Contexts	Reconnoitre	yes	no	*not applicable*
	Support by Fire	yes	no	
	Move	yes	yes	
	Move Cautiously	yes	no	
	Attack	yes	yes	
	Hasty Attack	yes	yes	
	Observe	yes	yes	
	Regroup	yes	yes	

Table 5.2 Mission and major contexts.

5.2 CxBR model

Section 4 explained the method for utilizing CxBR to model a hierarchy of battle command agents. In this section we will give a complete description of the missions, contexts and transition rules of the behaviour model.

First of all, there is one mission context for each of the tasks in the order. This yields three mission contexts: *Reconnoitre*, *Seize* and *Support by Fire*. Available low level CGF commands expressed as missions contexts are *Move*, *Wait* and *Attack*. The mission contexts are directly mapped to sequences of Low-level BML tasks that are sent to the CGF system. An example of Low-level BML tasks for mission context *Move* are "set-rules-of-engagement→move-into-formation→move-along-route". Further description of the Low-level BML can be found in [11]. The reason for expressing CGF commands as mission contexts is to keep the major contexts agent independent. Thus the battalion, the companies and the platoons all send mission contexts to their subordinates. The decomposition in between have been done according to the strategy "equal contexts and plans", as discussed in section 4.4.

Table 5.2 lists all the mission and major contexts we have defined. Both mission and major contexts are modelled independent of agent type (platoon or company), but not all of them are used for both platoons and companies. The table reflects which missions and contexts are used for which agents.

5.2.1 Mission context descriptions

The next pages contain detailed descriptions of all the different mission contexts including context hierarchies, context maps and parameters.

Mission Context Reconnoitre

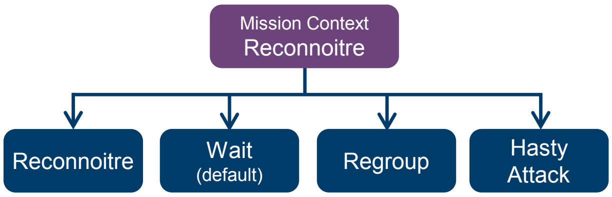
Military Description The mission Reconnoitre represents the task "reconnoitre an axis" from the order. From the field manual *"a route reconnaissance is conducted to obtain detailed information about one route and all the adjacent terrain to locate sites for emplacing obstacles. A route reconnaissance is oriented on a road; a narrow axis, such as an infiltration lane; or a general direction of attack"* [14].

Model Description The mission requires a route and implies moving along that route, reporting observations along the way.

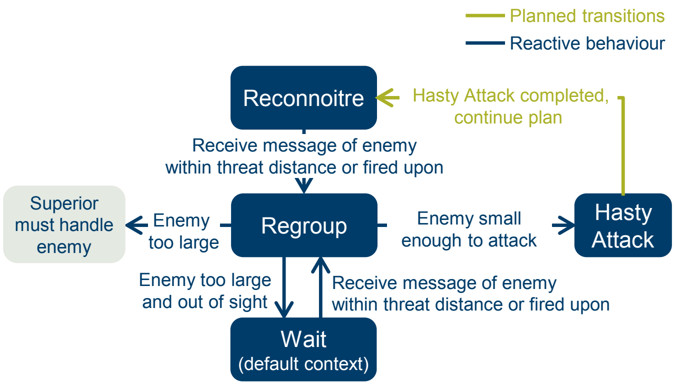
Parameters Route(s) - mandatory at this point
Rules of engagement (ROE) - optional, default "fire when fired upon"

Basic Plan *Reconnoitre*

Context Hierarchy



Context Map



Mission Context Seize

Military Description

The task "seize" is defined as *"a tactical mission task that involves taking possession of a designated area by using overwhelming force"* [15].

Model Description

In the model the mission context *Seize* includes three steps, 1) move towards the objective area (OA), 2) spread out and move more cautiously, and when the OA is getting closer, 3) attack observed enemies in the OA. When to spread out depends on the terrain and is simplified by requiring a phase line in the order. When to start the attack, also depends on the terrain, but is currently defined as a fixed distance. The mission is completed when the enemies in the OA are defeated.

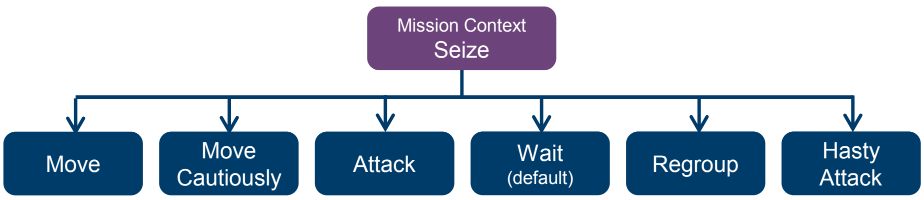
Parameters

Route(s) - mandatory at this point
Phaseline - mandatory at this point
OA - mandatory

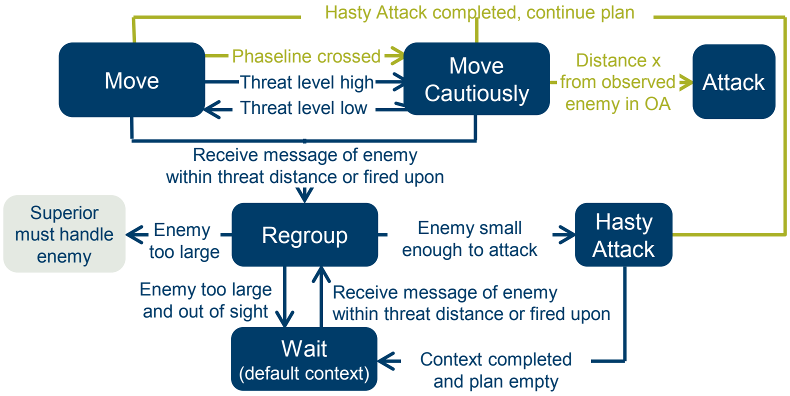
Basic Plan

$\textit{Move} \xrightarrow{\text{Phaseline crossed}} \textit{Move Cautiously} \xrightarrow{\text{X meter from observed enemy in OA}} \textit{Attack}$

Context Hierarchy



Context Map



Mission Context Support by Fire

Military Description

The tactical task "support by fire" is defined as *"a tactical mission task in which a maneuver force moves to a position where it can engage the enemy by direct fire in support of another maneuvering force"* [15].

Model Description

Move all subordinates to suitable support position and command them to *Observe* until the agent to support changes context to *Attack*, then command all subordinates to engage enemies in the OA.

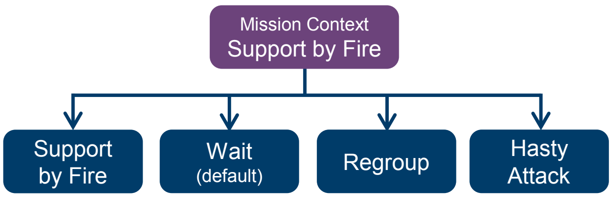
Parameters

OA - mandatory
Agent to support - mandatory

Basic Plan

Support by Fire

Context Hierarchy



Context Map

Mission Context Observe

Military Description

Observe is not a recognized military task, but our use of this task is similar to the military task "Screen" which is described as "observe, identify, and report information on threats to the main force. Only fight in self-protection" [8].

Model Description

It is assumed the agent's unit is located at the observation position. To complete the mission the agent must command its subordinates to move into a wide formation heading towards the OA.

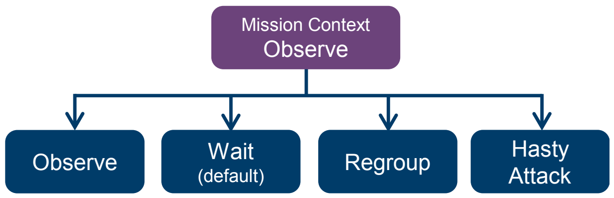
Parameters

OA (to observe) - mandatory

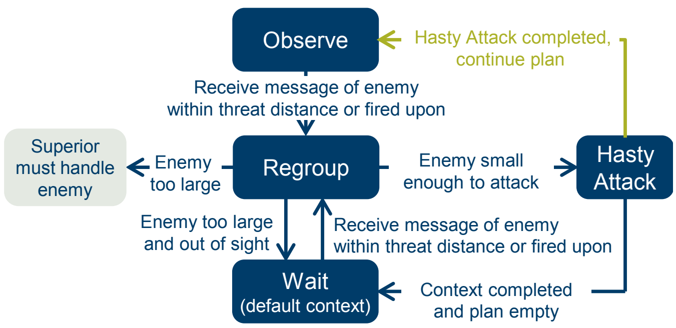
Basic Plan

Observe

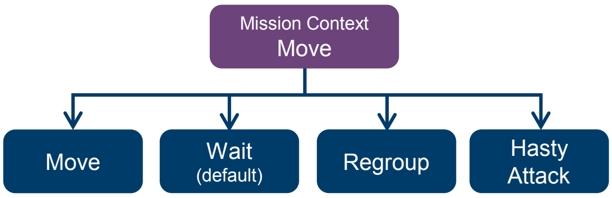
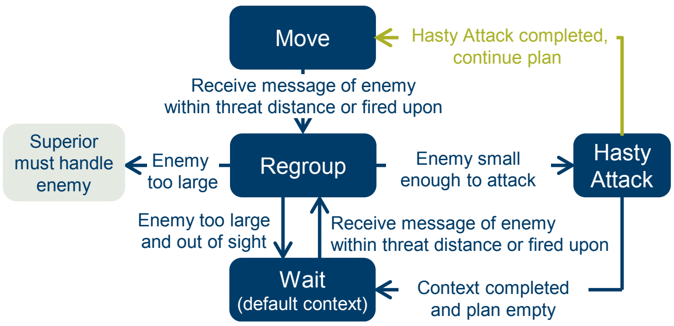
Context Hierarchy



Context Map



Mission Context Move

Military Description	Military terminology includes specific movement tasks like *Advance to Contact* and *March*.
Model Description	The move mission in this model is a more general movement task which is only used for platoon level agents and below and implies moving along a route in a given formation.
Parameters	Route(s) - mandatory at this point Formation - mandatory ROE - optional (Defalut "fire when fired upon")
Basic Plan	*Move*
Context Hierarchy	
Context Map	

Mission Context Attack

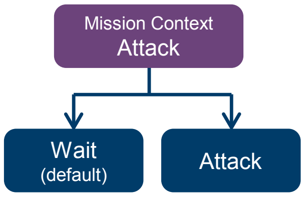
Military Description The general task "Attack" is defined as "*Take offensive action against a specified objective*" [8]. Usually a more specific task statement is used, but for now we only need this general term as the task is modelled very simplistically.

Model Description Move towards the OA or enemy position with rules of engagement (ROE) fire at will.

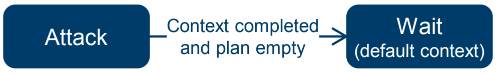
Parameters OA or Enemy position

Basic Plan *Attack*

Context Hierarchy



Context Map



Mission Context Wait

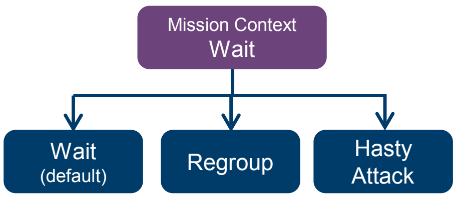
Military Description Wait is not an explicit military task.

Model Description Do nothing.

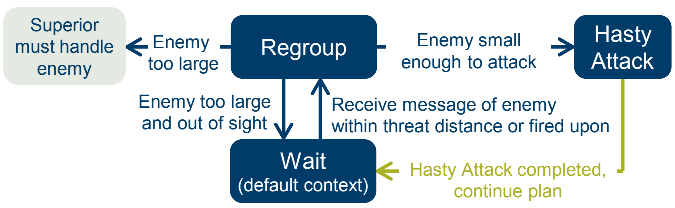
Parameters none

Basic Plan *Wait*

Context Hierarchy



Context Map



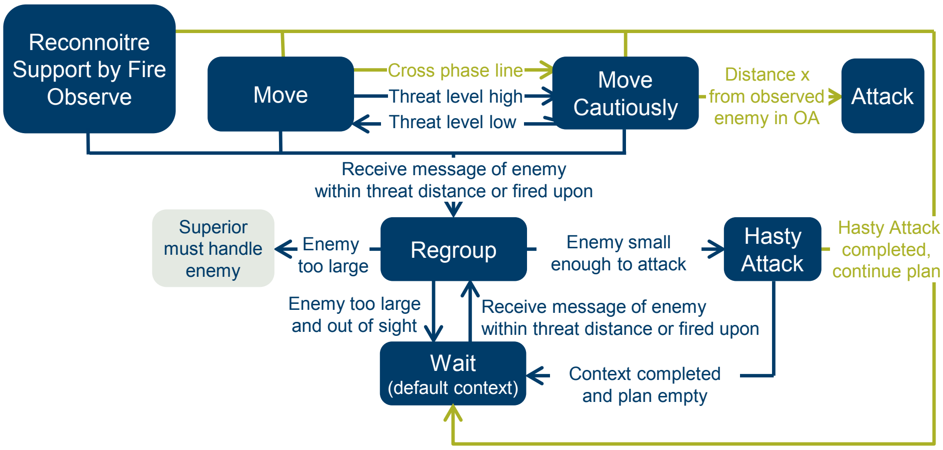


Figure 5.3 The context map includes all company and platoon major contexts and is applicable for all company and platoon mission contexts.

5.2.2 Major context descriptions

The descriptions of the different mission contexts contained a context map for each mission. These context maps are fully compatible and are merged together in figure 5.3, illustrating all major contexts and transition rules for both company and platoon agents. Transitions that are a part of the planned actions are shown in yellow-green. In the detailed context descriptions that follow, planned transitions are united as one universal transition rule, *Context state completed* → *Next context in plan* .

The next pages contain detailed descriptions of all major contexts applicable for company and platoon agents. As explained in section 4.1, these are management contexts, e.g. when an agent is in major context *Attack*, it means that the agent organizes an attack by ordering its subordinates to realize different parts of the attack. In addition to traditional description of action knowledge, transition knowledge and declarative knowledge, we have included a figure illustrating which mission contexts can be sent to the subordinates when an agent is in a given major context. These figures should not be confused with context hierarchies or contexts maps, but illustrates the action rules that lead to new mission contexts for the subordinates. We call these actions *mission commands* and the illustrations *mission command diagram*.

The model contains two important simplifications because of the lack of automatic terrain analysis. First, many of the missions and contexts require routes. In the future either the MAS or the CGF system should be able to calculate these routes as they are needed, and the only input required in the order created by the C2IS should be regular tactical graphics like seize- and reconnoitre arrows and control measures like OAs and phase lines[5]. Second, the transition rules implies that there are defined fixed distances for when an enemy is considered a threat (transition rule *Observed enemy within threat distance* → *Regroup*) and when to start a planned attack (planned transition from

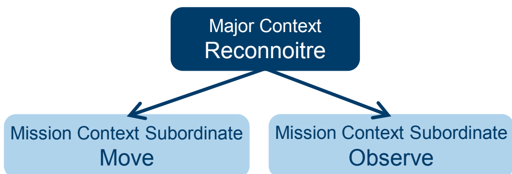
[5]In special occasions the battalion commander might want to specify a specific route as a control measure. This should be possible, but not mandatory.

Move Cautiously to *Attack*). These distances depend on the terrain, and during implementation we will use trial and error to decide fixed distances that will work for now. The only requirement is that the distance for when to start a planned attack must be larger than the threat distance to avoid transition from *Move Cautiously* to *Regroup* when an attack is planned. With automatic terrain analysis we will not have to define fixed distances. Our plans for future automatic terrain analysis are documented in [16].

Universal for all Contexts

Description	These action and transition rules are common for all contexts.	
Action Knowledge	When	Received instruction change ROE.
	Then	Send instruction change ROE to all subordinates.
	When	All subordinates mission completed.
	Then	Change context state to *Completed*.
Transition Knowledge	Context state *Completed* → Next context in plan	
	Context state *Completed* AND Plan empty → *Wait*	
Declarative Knowledge	Context state: *Not initialized, Initialized, Completed, Interrupted*	

Major Context Reconnoitre

Description	The context Reconnoitre is modelled as cautious movement along an axis. Preliminary a route is included for this task, next step could be to define the axes as a couple of way-points, but eventually the task should be accomplished with only the information extracted from the regular tactical graphics. It is assumed that the agent observes everything within its line of sight.	
Action Knowledge	When	Entering the context.
	Then	Command all subordinates to *Move* along route in a wide formation.
	When	Received report end of route.
	Then	Command all subordinates to *Observe*.
Transition Knowledge	Received report fired upon → *Regroup*	
	Received report observed enemy within threat distance → *Regroup*	
Declarative Knowledge	Route	
	Rules of engagement (default "fire when fired upon")	
Mission Command Diagram		

Major Context Support by Fire

Description Command all subordinates to move to support position and observe. When the agent to support enters the context *Attack*, command all subordinates to attack.

Action Knowledge

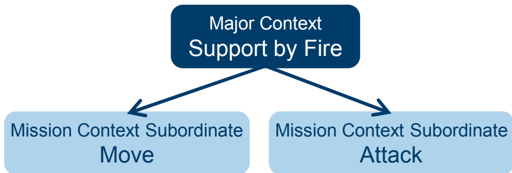
When Subordinate units are not at support position AND `isMovingTowardSupportPosition` is false.
Then Send mission command Move to all subordinates.

When Agent to support is attacking and `isAttacking` is false.
Then Send mission command Attack to all subordinates.

Transition Knowledge Only universal transition rules

Declarative Knowledge
Agent to support
Support position
`isAttacking`
`isMovingTowardSupportPosition`

Mission Command Diagram



Major Context Move

Description Move along a route as fast as possible in the prescribed formation. With formation we mean the platoon formation, no support for company formations has been implemented in the first version.

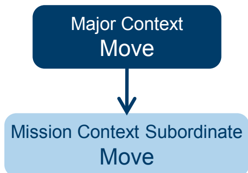
Action Knowledge

When Entering the context.
Then Command all subordinates to *Move* along route (in the prescribed formation if it is set).

Transition Knowledge
Received report fired upon → *Regroup*
Received report observed enemy within threat distance → *Regroup*

Declarative Knowledge
Route(s)
Formation

Mission Command Diagram



Major Context Move Cautiously

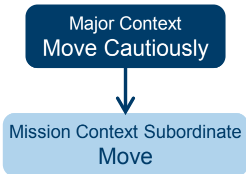
Description Move in a formation as wide as possible and avoid roads (the routes are not on roads).

Action Knowledge
When Entering the context.
Then Command all subordinates to *Move* along route in a formation as wide as the terrain permits.

Transition Knowledge
Received report fired upon → *Regroup*
Received report observed enemy within threat distance → *Regroup*

Declarative Knowledge
Route(s)
Formation

Mission Command Diagram



Major Context Attack

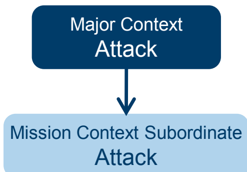
Description The context attack refers to a "deliberate attack" which means *"preplanned coordinated employment of firepower and manoeuvre to close with and destroy or capture the enemy" [8]*. The modelling of this behaviour is relatively simple, it means move toward the objective area and fire at will.

Action Knowledge
When Entering the context.
Then Command all subordinates to *Attack* while moving along a route toward the OA.

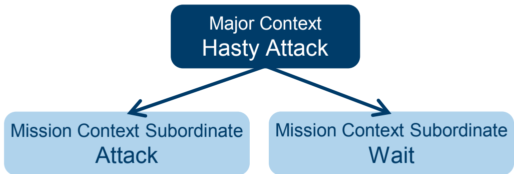
Transition Knowledge Only universal transition rules

Declarative Knowledge
OA
Route(s)

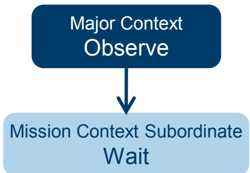
Mission Command Diagram



Major Context Hasty Attack

Description	A "hasty attack" is *"an attack in which preparation time is traded for speed in order to exploit an opportunity" [8]*. The modelling of this behaviour is relatively simple, it means move toward the enemy and fire at will.	
Action Knowledge	When	Entering the context.
	Then	Calculate route toward enemy and command all subordinates to *Attack* along that route.
	When	Enemies are destroyed.
	Then	Set context state *complete*.
	When	Distance to enemy is less than x meter.
	Then	Command all subordinates to *Wait* (but without changing rules of engagement, meaning they will still fire at will).
Transition Knowledge	Only universal transition rules	
Declarative Knowledge	Enemy Position	
Mission Command Diagram		

Major Context Observe

Description	Move into formation heading toward the objective area to observe. It is assumed that the agent already is at the observation position.	
Action Knowledge	When	Entering the context.
	Then	Command all subordinates to *Wait* AND move into formation toward OA.
Transition Knowledge	Only universal transition rules	
Declarative Knowledge	OA	
Mission Command Diagram		

Major Context Regroup

Description As the forces are advancing towards their target areas they must handle enemies along the way. When an enemy is observed within threat distance or one of its units is fired upon, the agent will enter the regroup context. In this context they will deliberate on what to do with the enemy.

Action Knowledge

When	Entering the context.
Then	Command all subordinates to "Wait".
When	The enemy is too strong to attack.
Then	Command all subordinates to Move away from the enemy AND set `isRetreating` to true AND ask superior for assistance.

Transition Knowledge

Enemy small enough to attack → *Hasty Attack*

Enemy too large and out of sight → *Wait*

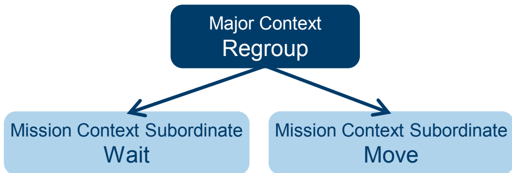
Remember that if the enemy is too large to attack, the superior agent will transition to the context Regroup.

Declarative Knowledge

boolean `isRetreating`

Threat distance

Mission Command Diagram



Major Context Wait

Description Do nothing, wait for new orders.

Action Knowledge

When	Entering the context.
Then	Command all subordinates to "Wait".

Transition Knowledge

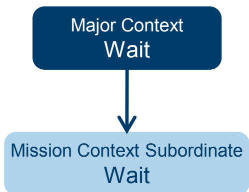
Received report fired upon → *Regroup*

Received report observed enemy within threat distance → *Regroup*

Declarative Knowledge

Threat distance

Mission Command Diagram



In all context descriptions a mission command diagram is included, illustrating the possible mission contexts the subordinates can be given while the agent is in that context. In figure 5.4

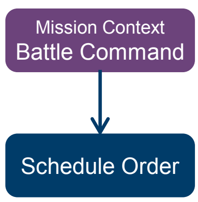
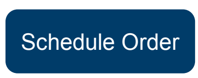
all the mission command diagrams for company and platoon agents are united. The figure also includes the reports that the agents are currently explicitly sending to the superior. Position reports are omitted because all agents have access to a common situation picture including perceived truth for own and enemy forces.

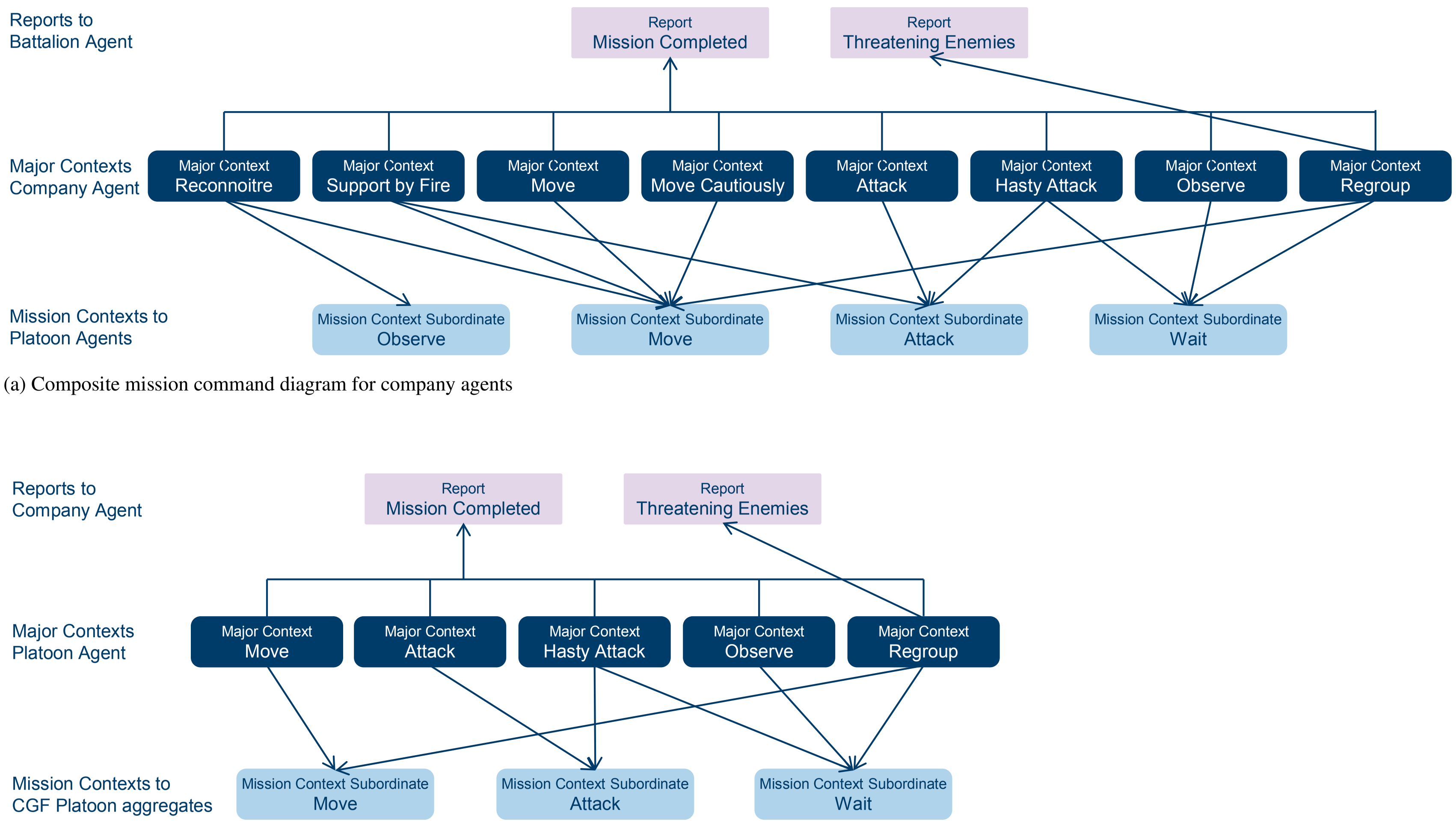
Currently there is no explicit communication between agents at the same level, e.g. one company agent with another company agent. In the future the agents will communicate their context to peer agents, as indicated in figure 4.2, in order to facilitate cooperation. At this point cooperation is limited to major context *Support by Fire*, where the agent orders its superior to *Attack* when the agent to support starts attacking.

5.2.3 Mission and context for the battalion agent

Up until now we have only discussed the behaviour of the company and platoon agents, but the model also includes a battalion agent. However, the operational order consists of company tasks, which means the battalion agent does not have to decompose any tasks. The purpose of the battalion agent is to synchronize the company tasks. In this first version that means making sure the companies receive the appropriate tasks at the (relative) times specified in the synchronization matrix. For the battalion we consequently use only one type of mission context, which we have called *Battle Command* and is carried out by one major context, *Schedule Order*. These contexts are described next.

Mission Context Battle Command

Military Description	Schedule order is not a military task.
Model Description	Deliver the task from the order to the right companies according to the schedule in the synchronization matrix.
Parameters	none
Basic Plan	*Schedule Order*
Context Hierarchy	
Context Map	

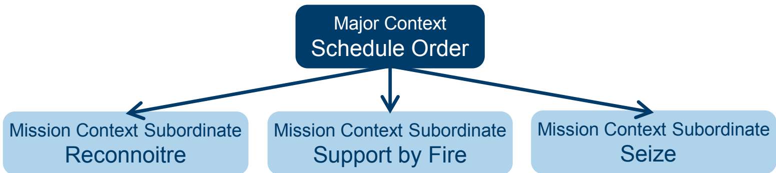


(a) Composite mission command diagram for company agents

(b) Composite mission command diagram for platoon agents

Figure 5.4 *The figure illustrates what (a) a company agent and (b) a platoon agent in the current model can communicate to its subordinates and superior in the different major contexts. Note that the diagram for the platoon (b) is a sub-set of figure (a) as the platoon agents currently use a sub-set of the major contexts for the company agents.*

Major Context Schedule Order

Description	Synchronize the company task in the order
Action Knowledge	When: A start rule is matching Then: Send the mission context associated with the start rule to the appropriate company.
Transition Knowledge	No other contexts applicable for battalion agents are defined.
Declarative Knowledge	Company missions - list with mission contexts for all the companies, generated from the order Start rules - a list of rules which states when to start each mission in the order
Mission Command Diagram	

6 Discussions and conclusions

The objective of the research presented in this report is to create a simulation system that can autonomously simulate a battalion operation by applying common tactics and follow current doctrine. The simulation system consists of a MAS and a commercially available CGF system. The entities in the CGF system can handle simple commands like “move along route” and “move into formation x”, but higher level tactics and doctrinal knowledge is represented in the MAS that controls the CGF system.

This report has focused on how to model the behaviour of the agents in the MAS. We have explored using the CxBR paradigm for a hierarchy of battle command agents implementing basic tactics. The objective was to build a model complex enough to illustrate the concept and to reveal challenges and further research questions.

From our experience the CxBR paradigm seems suitable for modelling battle command as well as lower level behaviour. One advantage with CxBR is that it is easy to explain to non-experts, and we expect organizing the behaviour in this way will make it easy to implement a user interface that illustrates and explains the decisions the agents make along the way.

We intended to decompose the ordered task and model the behaviour in a way the SMEs can relate to. This will be beneficial for validation and makes the model more credible for the intended users. For example, when talking with SMEs we were repeatedly told that what is done on the different echelons in the military hierarchy is basically the same. We have gone so far as to use the same mission contexts with the same basic plans and the same major context implementation for company and platoon agents. The feedback from SMEs has been positive, but it is hard to say whether this approach is feasible as the model becomes more complex.

We have tried to relate mission contexts to military tasks, major contexts to implementation of tactics and transition rules to doctrinal procedures. Still, it is not easy to define when a battle command agent, who has subordinates with different mission contexts, actually changes major context. The idea to keep the major contexts the same for companies and platoons have guided the proposed solution of how tasks are decomposed through the agent hierarchy, which makes implications for what is a major context and which major contexts are needed to complete a mission context. This strategy looks promising, but must be re-evaluated when we make a more complex model.

Further investigation is also needed regarding how one agent plans a mission contexts. We suggested a three step planning strategy: 1) retrieve a basic plan for the current mission context, 2) adapt the basic plan to the current situation by removing redundant major contexts and specifying when a context in the plan is completed, and 3) replace the general major contexts from the basic plan with more specific versions. There are several questions concerning this strategy. Can we make a basic plan that takes into consideration all possible, relevant knowledge? At the moment, the basic plan for the mission "Seize" for example, does not consider enemy observations outside objective areas. Do we want the plan to be a sequence of major contexts, or would we want the possibility to plan for more than one course of action? We could generate the plan as a direct acrylic graph rather than a sequence, but the same result could be achieved by reconsider and replan as new information are available.

The current model does not include replanning at all. This is a topic for further studies. Currently the model can only handle unplanned events through the reactive behaviour ensured by standard transition rules. Other subjects we will have to address more thoroughly in future work are communication and coordination between agents at the same echelon in the hierarchy.

Before we can make a more complex model, at least for the land domain, we will need automatic terrain analysis as most tactical decisions depend on the terrain. We also plan to look into using machine learning to generate behaviour from examples. Hard-coding action rules is time consuming and requires a lot of domain expertise. It is possible we could make better models and/or generate behaviour faster with machine learning.

Good methods for evaluation and validation of the model will become even more important if we try to generate behaviour automatically with machine learning, but in any case this is an important topic that needs to be addressed further. Building good test cases to simulate and having SMEs observe the simulation and give feedback on the behaviour might be the best option.

References

[1] J. Fletcher, "Education and training technology in the military," *Science*, vol. 323, pp. 72–75, 2009.

[2] VT MÄK VR-Forces. [Online]. Available: www.mak.com/products/simulate/computer-generated-forces.html

[3] S. Carey *et al.*, "Standardizing battle management language - a vital move towards the army transformation," in *Proceedings of the 2001 Fall Simulation Interoperability Workshop*, no. 01F-SIW-067, 2001.

[4] O. M. Mevassvik and A. Alstad, "Stridsledelsesspråk for koalisjonsoperasjoner - en teknologi for å integrere kommando og kontroll og simulering," Forsvarets forskningsinstitutt, FFI-rapport 2012/00176, 2012.

[5] A. J. Gonzalez, B. S. Stensrud, and G. Barret, "Formalizing context-based reasoning: A modeling paradigm for representing tactical human behavior," *International Journal of Intelligent Systems*, vol. 23, pp. 822–847, 2008.

[6] A. Gallagher, A. J. Gonzalez, and R. DeMara, "Modeling platform behaviors under degraded states using context-based reasoning," in *Proceedings of the 2000 Interservice/Industry Training, Simulation and Education Conference (I/ITSEC-2000)*, 2000.

[7] A. J. Gonzalez and R. Ahlers, "Context-based representation of intelligent behavior in training simulations," *Transactions of the Society for Computer Simulation International*, vol. 15, no. 4, pp. 153–166, 1998.

[8] G. P. S. Opedal and O. K. J. Støve, "Stabshåndbok for Hæren - plan- og beslutningsprosessen," 2010.

[9] SISO, *Standard for: Coalition Battle Management Language (C-BML)*, 2012, SISO-STD-011-20128-DRAFT.

[10] SISO, *Standard for: Military Scenario Definition Language (MSDL)*, 2008, SISO-STD-007-2008.

[11] A. Alstad *et al.*, "Low-level battle management language," in *Proceedings of the 2013 Spring Simulation Interoperability Workshop*, no. 13S-SIW-032, 2013.

[12] A. Alstad, "Norwegian Nato MSG-048 technical documentation," Forsvarets forskningsinstitutt, FFI-notat 2010/01261, 2010.

[13] M. Wooldridge, *An Introduction to MultiAgent Systems*. John Wiley & Sons Inc., 2002.

[14] Headquarters Department of the Army, "Field manual no. 7-8," 1992. [Online]. Available: www.globalsecurity.org/military/library/policy/army/fm/7-8/.

[15] Headquarters Department of the Army , "Field manual no. 3-90, tactics," 2001. [Online]. Available: www.globalsecurity.org/military/library/policy/army/fm/3-90/.

[16] S. Bruvoll, "Preliminary study of terrain analysis and path planning for computer generated forces," Forsvarets forskningsinstitutt, FFI-notat 2013/00688, 2013.

Abbreviations

AI	Artificial Intelligence
C2IS	Command and Control Information System
C-BML	Coalition Battle Management Language
COA	Course of Action
CGF	Computer Generated Forces
CxBR	Context-based Reasoning
FRAGO	Fragmentary Order
IEEE	Institute of Electrical and Electronics Engineers
I/ITSEC	Interservice/Industry Training, Simulation and Education Conference
MAS	Multiagent System
MSDL	Military Scenario Definition Language
MSG	Modelling and Simulation Group
NATO	North Atlantic Treaty Organization
OA	Objective Area
OPORD	Operational Order
ORBAT	Order of Battle
PL	Phase Line
ROE	Rules of Engagement
SISO	Simulation Interoperability Standards Organization
SL	Start Line
SME	Subject Matter Expert
VR-Forces	Virtual Reality Forces