Defence Science Journal, Vol. 68, No. 1, January 2018, pp. 46-53, DOI : 10.14429/dsj.68.11375

Command Agent Belief Architecture to Support Commander Decision Making in Military Simulation

Sanjay Bisht*, H.S. Bharati, S.B. Taneja, and Punam Bedi

DRDO-Institute for Systems Studies and Analyses, Delhi - 110 054, India
**E-mail: sanhembisht77@gmail.com*

ABSTRACT

In the war, military conflicts have many aspects that are consistent with complexity theory e.g., the higher commander's decision is directed at animate entity that react under hierarchical and self-organised structure in decentralised command and control for the collectivist dynamism of decomposed elements due to nonlinear complexity of warfare on the battlefield. Agent technology have been found to be suitable for modelling tactical behaviour of entities at multiple level of resolution under hierarchical command and control (C2) structure and provide a powerful abstraction mechanism required for designing simulations of complex and dynamic battlefield situations. Intelligent agents can potentially reduce the overhead on such experiments and studies. Command agents, plan how to carry out the operation and assign tasks to subordinate agents. They receive information from battlefield environment and use such information to build situation awareness and also to respond to unforeseen situations. In the paper, we have proposed a mechanism for modelling tactical behaviour of an intelligent agent by which higher command level entities should be able to synthesize their beliefs derived from the lower level sub ordinates entities. This paper presents a role-based belief, desire and intention mechanism to facilitate in the representation of military hierarchy, modelling of tactical behaviour based on agent current belief, teammate's belief propagation, and coordination issues. Higher commander can view the battlefield information at different levels of abstraction based on concept of aggregation and disaggregation and take appropriate reactive response to any unforeseen circumstances happening in battlefield.

Keywords: C2 structure; Intelligent agents; Aggregation and disaggregation; Command agent; Team-Oriented programming; Belief propagation

1. INTRODUCTION

Realistic and cost effective combat modelling and simulations are having emerging needs of military decision makers for planning, training, analysis and solving sponsor's problem. In the recent past, there has been an increasing interest in applying evolutionary techniques and technologies of agent based molding and simulation (ABSM) for representing human operational reasoning in military simulation. Some earlier applications of ABSM[1,2] of representing military reasoning have proved to be very useful, because of the capability of intelligent agents to define and represent human operational reasoning at all level of military C2 structure and architectural advantage of agent technology. This has prompted the development of realistic and cost effective synthetic combat environments to model the complex behaviour of a combatant (considered as combat entity) in the simulation of combat scenario. It is important to model team behaviour where large number of combat entities are participating in interaction to coordinate joint activities of offensive and defensive operations as per the higher commander's mission. These complex aspects of team behaviour of military forces are required to be represented and implemented through ABMS for experimentation and analysis of team performance at all levels of aggregation under C2 hierarchy of military organisation.

Cohen & Levesque[17] in their work applied coordinated group action in multi-agent systems (MAS). Joint Intentions theory, provides a framework within which agents groups can hold joint goals and execute mutual actions. Joint Intentions also describes the cases where some form of communication is necessary to ensure coherent state of agents joint coordinated[18] actions. Jennings[20] describes the importance of commitments and conventions in co-ordination schemes. Commitments are made by agents to perform certain joint actions and they agree on a set of conventions on how to monitor their commitments in order to achieve the objective. Tambe[21] in his work successfully applied the ideas from Joint Intentions to improve the team behaviour of attack helicopters . In our study we have used Joint Intentions to determine when agents need to send messages and when to wait for other agents to complete their task(s) before continuing with the next task.

Beliefs, desires and intentions (BDI) agent model framework[14,15,19] supports, dynamic decision making at each command agent level based on propagated belief from its subordinate teams to its higher command agent team. The role of higher command agent team is to generate plans and course of actions and then propagate the same to its subordinate command agents. Horn[22] work supports group behaviour

Received: 03 May 2017, Revised: 08 December 2017
Accepted: 11 December 2017, Online published: 18 December 2017

among team agents using a framework based on multi agent system. In our previous work[23,24] we have demonstrates the team-behaviour of armoured tank troops in a tactical armour masking scenario using intelligent agent modelling.

The main objective of this study is, how an intelligent agent can perform the behaviour of a combatant as 'Command Agent', whose role and C2[3-5] responsibilities whilst maintaining communication with the higher commanders under hierarchical structure in the battlefield. In a dynamically changing environment, command agents that monitor the battle space and facilitate subordinate agents in providing relevant information, which enables the achievement of a greater degree of situational awareness for the entire network. As a proof of concept, we have developed high-level agents[6-7] to demonstrate the tactical and combat decision making behaviour in the simulation of a tactical operation. In this scenario, we have considered higher command agent and modelled teams of agents and implemented their interactions with coordinated team behaviour down to lowest entity in the C2 hierarchy using JACK Teams[11-13]. JACK teams is an extension to JACK™ intelligent agents[8-10], which is a powerful team oriented programming tool for developing agent based simulation.

This paper initially describes the concept of command agent modelling and then architecture adopted for the implementation of command agent and its capabilities. We have also proposed a concept of modelling C2 and decision making of the command agents (aggregated at level of Companies of Infantry Battalion), abstracted in the C2 hierarchy and demonstrated the experimental results for command agents in the simulation of tactical operation up to single combatant (i.e., soldier), using JACK Teams.

2. COMMAND AGENT *VS* COMBATANT AGENT

As per C2 structure of military hierarchy as depicted in Fig. 1, entities at each level of hierarchy in multiple resolution modelling and simulation, Commander at each aggregated level of hierarchy, known as 'Command Agent', generates tactical operation plans as per intend of superior commander. Each plan includes courses of actions (COA). The modelling of command agent requires a well-structured protocols, which allow an effective communication with superiors command agents at the same time managing the ground troops actions (handling enemy detection, engagements, mine cross options) and monitoring the activities of opposition forces. Generally, in such situation, a combatant develops belief towards his teammate as per actions of events under critical situation. Intention for team actions generated based on the propagated desire of higher commander. This belief propagation mechanism can be implemented as per role-based BDI[14,15] framework and can also be observed by command agents at desired level of C2 hierarchy. The mechanism facilitates the representation of role-based BDI based on agent current belief, teammate's belief propagation, and cooperation and communication issues.

Besides consideration of command agents at all aggregated levels of C2 hierarchy, aspects of a combatant (soldier) as an intelligent agent is illustrated in Fig. 1, which depicts the behaviour of a combatant in a battlefield. The soldier has a set of relevant plans and battlefield events. He also gathers dynamic information during interaction with other battlefield entities to communicate the higher command agent.

3. COMMAND AGENT MODELLING FOR MILITARY SIMULATION

Command agents modelling for the development of military tactical simulation, includes the following aspects:

3.1 Command Agent Processing Mechanism and Architecture

As we have discussed above that each command agent is a representative of team of soldiers which provides workload sharing mechanism and serves to encapsulate coordination activity within team. Each of the team of command agent type will follow observe, orient, decide and act (OODA) loop for belief propagation to next higher level command agents as shown in Fig. 2. Command agent (CA) in the simulation environment is represented at both the upper level and lower level. But human players (i.e., observers) are always at the

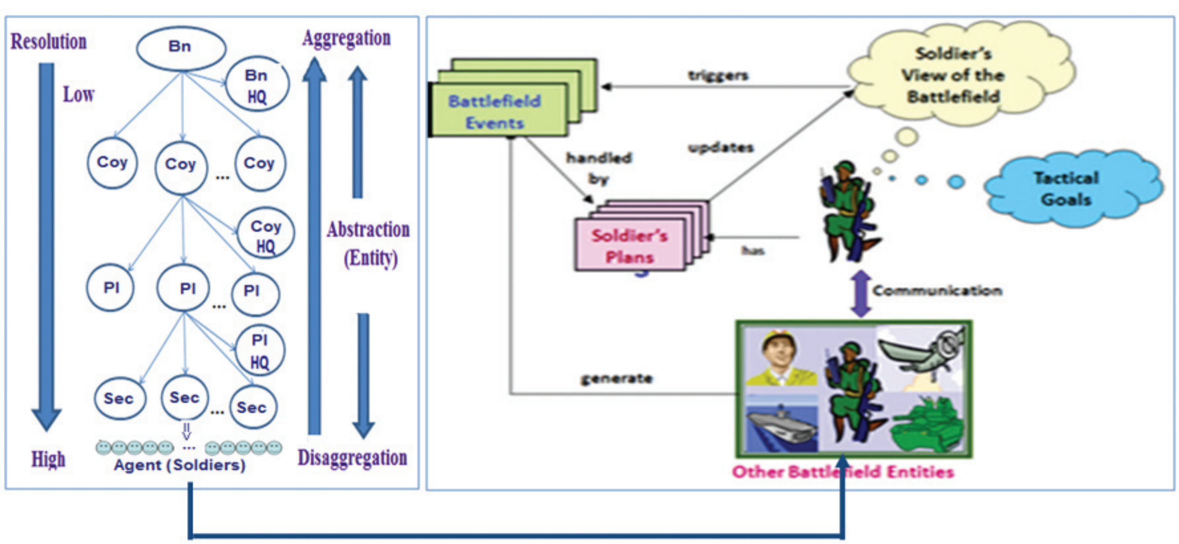

Figure 1. Tactical behaviour of a combatant under C2 hierarchy structure.

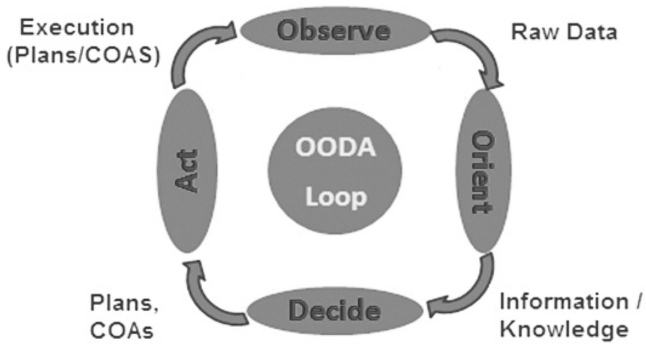

Figure 2. CA belief propagation through OODA.

upper levels and simulation entities at lower level. In the basic command agent architecture[4] as suggested in the literature and depicted in Fig. 3, Level-1 may contain command agents or simulation entities and Level+1 may contain command agents or human players. The key capabilities (i.e., reporting for action, planning for action, control for action) of command agent operate on a shared belief structure[4,2] that contains the current beliefs of command agent's (CA) regarding t he world.

The planning-for-action capability allows the command agents to take a command from the level above and by using the appropriate military doctrine, generates commands for the entities under its direct control. The progress of the resulting action is then monitored by the control-of-action capability. The reporting-on-action capability provides information back to the command level. The content of reporting information is derived from messages provided by the subordinate entities at multiple levels of resolution using aggregation and disaggregation process in multi-resolution simulation environment.

3.2 Team Aspects of Command Agent

Team oriented modelling provides workload sharing mechanism and serves to encapsulate coordination activity within team. Each subordinate team member are assigned a particular Role. These Roles are assigned during the team formation time. Each sub ordinate team member coordinates with other team member (at same level) to fulfill the main objectives assigned by its superior team member. Sub ordinate team member also perform their local objective, while they accomplish their main objective given by superior command agent entities. The command agents at each subordinate level (i.e., aggregated entity) are modelled for down the command and control military hierarchy.

Combatant agents are considered at bottom of the hierarchy as dispatcher agent to their next team command agents. Teams can be abstracted (at any aggregated level) and created in any combination for desired level of C2 structure, where each level is encapsulated within the next level and so on. In multi-resolution environment, the team encapsulates individual share of contribution by the process of aggregation, where an agent extends by associating tasks with roles as distinct entity through the process of disaggregation. Role of individual team is also very important. Role defines a relationship between teams and provides facility to play individual share of tasks.

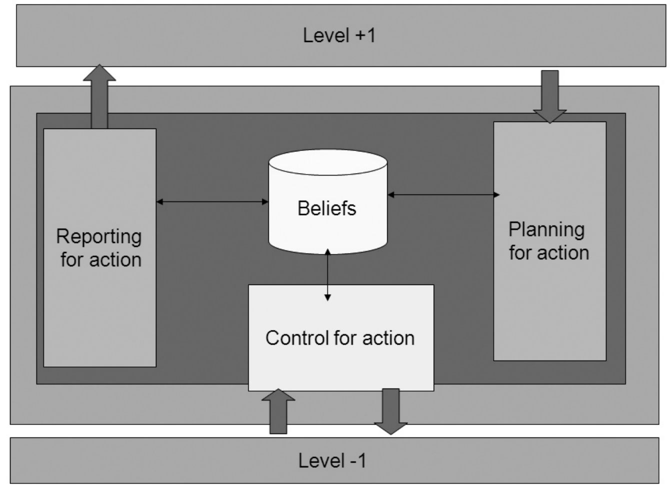

Figure 3. Command agent architecture.

4. TACTICAL SCENARIO

To model the functionality and capability of the command agent at each aggregation level of C2 hierarchy, commander at higher level (e.g., at Infantry Battalion Commander level) is considered as superior *Command Agent***,** which produces a plan and course of actions (COA) for his command agents (of subordinate entities in the C2 Hierarchy) to carry out the attack in four sub-activities, namely move towards objective in given battle formation, respond to minefield and enemy opposition, assault and capture, and communicate to respective command agent. In the scenario as depicted below in Fig. 4, Company in the Battalion organisation consists of three to four platoons. Each platoon in turn comprises of three to four sections, each of the section has ten to twelve soldiers. Command agent at company level prepares plans, defines objectives, routes, form-up positions and also coordinates for assault and fire support from firebase for subordinate subunits. Further next subordinate command agents down to Section will monitor the status of its own soldiers in terms current location, whether encountered obstacle (e.g., minefield) and casualties due to enemy opposition and also respond to situations as per battlefield environment.

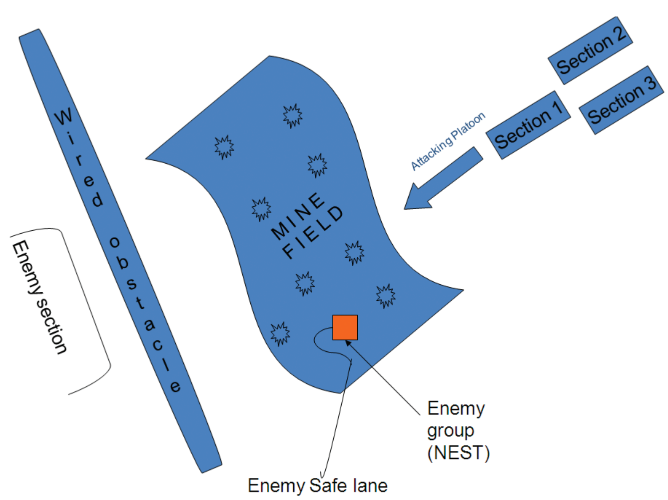

Figure 4. Platoon of infantry company attack to capture enemy position.

5. DESIGN PARADIGM OF COMMAND AGENT FOR THE TACTICAL SCENARIO

In the above scenario of attack operation, it is important to include the C2 process to achieve a realistic simulation of an entity behaviour. Inspired by various aspects of complexity theory, we have developed a representation of C2 based on a decentralised system of interacting intelligent command agents.

In Figure 5, three layers from level 2 to level 0, are considered to model command agents at each levels under C2 hierarchy. it is assumed that, higher command agent CA1 (at level 2) of company (coy) team consists of three subordinate level command agents i.e., CA11,CA12 and CA13 (at level 1) of platoon (PL) teams. Each PL team command agents from CA11 to CA13 has next subordinate level command agents i.e., CA111 to CA113, CA121 to CA123 and CA131 to CA133 (at Level 0) of section (Sec Team). Each bottom level command agent further consists of dispatcher agents (soldier). Orders and reports are exchanged between higher CA to its subordinate CAs, through dispatcher via bi-directional links. To implement the behaviour of each command agent, the tasks of lower resolution entity (aggregated level) is decomposed into smaller and manageable tasks at respective higher resolution entities (disaggregated level). Team controller assigns these sub-tasks to individual teams capable of performing their roles.

To model and simulate the scenario using team-oriented concepts, first key abstractions have to be identified, which enables us to understand the clear structure of the team. Roles and responsibilities of the team members are defined, once the team is formed. From the infantry attack scenario, we can clearly identify the team controller as the Company Commander, whose top-level goal is to move towards the assigned objective without any enemy interference. It is obvious that Coy team controller required three sub teams, namely: Platoon1_team, Platoon2_team and Platoon3_team[23], each performing its specific assigned role (assault/coordinating role) by executing the appropriate plans. For example, the first platoon takes the role of Assault, while remaining two are assigned to do the role of coordinating with assault platoon team. Each assigned Role defines a set of events, associated plans and data set (world belief) that a entity can use at a time. Using these event defined by assigned role, entity can represent different behaviour in the battlefield.

All platoon and section team will carry out movement, detection and engagement etc., by having plans for detecting enemy and fire on it and then informing status to superior team controller (Company Commander). Whenever the assault platoon casualties reach below threshold limit of the combat strength, then one of the remaining platoon team whose morale is high will take the role of assault. Table 1 describes the roles, responding events and the corresponding plans for this scenario.

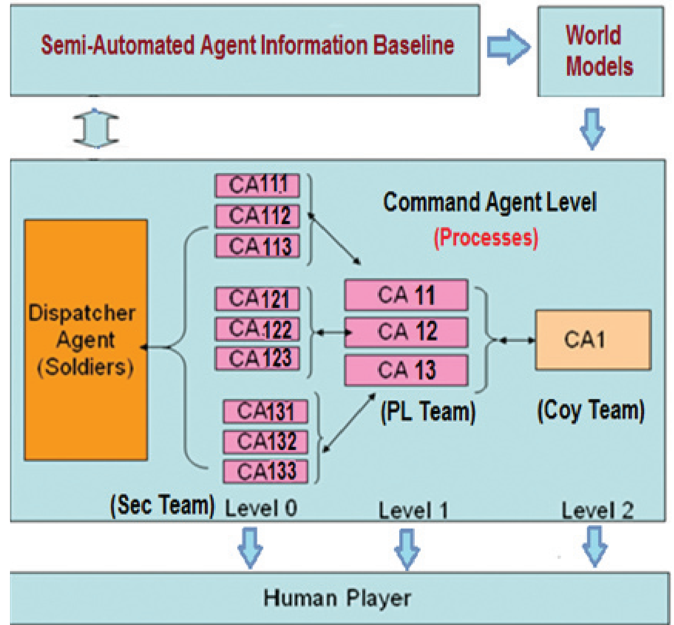

Figure 5. Implemented command agent architecture

Each platoon teams is further divided into three sections namely Sec1, Sec2, and Sec3 for performing the role of assault and coordinate role. The section team in turn consists of (1..N) soldier agents (non-team agent). We have identified two types of agents in this scenario namely the team agents (depicting coy/platoon and section) and soldier agents. These team agents have all the capabilities of agents and also encapsulate team behaviour. The soldier agents have been modeled to carry out tactical behaviour like movement along a route, avoid obstacle, patrol, fire and engage the enemy etc. A team controller (which is representing the Company Team)[23] and three sub-teams are involved. The team controller is also called the 'role tenderer'. Team controller composed of sub-teams that perform desired

Table 1. Teams / Agent and their roles, events and plans

Main Team/ Agent	Sub Team/ Agent	Roles	Responding events	Plans/Intentions (event handlers)
Team Controller (Coy)	Platoon1_team	Asslt_Pl_ ROLE	asslt_event	Asslt_plan
	Platoon2_team	Coor_Pl_ ROLE	coordinate_event	coordinate_plan
	Platoon3_team	Coor_Pl_ROLE	coordinate_event	coordinate_plan
Team Controller (Platoon)	Sec1_team	Sec1_ ROLE	move_event	move_plan
	Sec2_team	Sec2_ ROLE	move_event	move_plan
	Sec3_team	Sec3_ROLE	move_event	move_plan
Team Controller (Section)	Soldier1	-	move_event	move_plan
	:		fire_event	fire plan
	Soldier N		avoid_obs_event	avoid_obs_plan
			patrol event	petrol plan
Agent (Soldier)	-	-	move_event	move_plan
			avoid_obs_event	avoid_obs_plan
			patrol event	petrol plan
			fire_event	fire plan

roles on its behalf. The sub team performing roles for team controller are called ‘role performer’. Role is a behaviour that the ‘role tenderer’ may request the ‘role performer’ to achieve. Each role represents a unique behaviour of a team member (sub-team) participating in a particular tactical battlefield operation.

6. BELIEF PROPAGATION MECHANISM IN COMMAND AGENTS

The block schematic of team structure is given in Fig. 6.

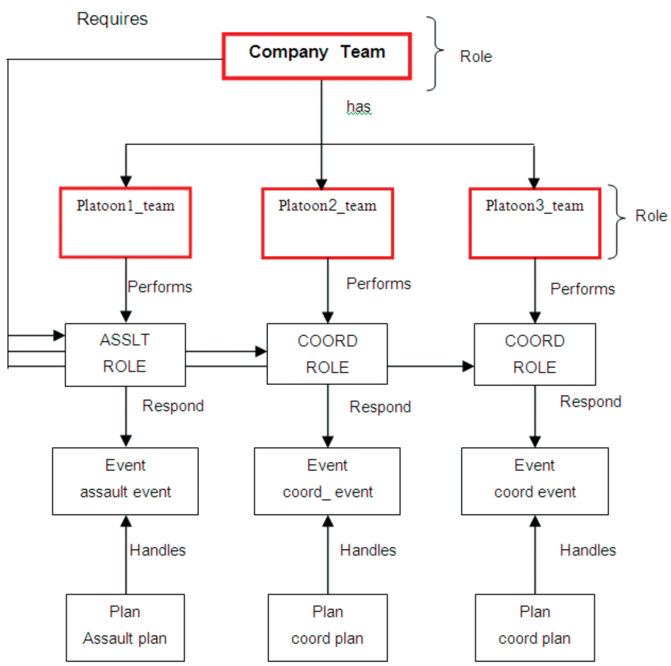

Figure 6. Command team structure for modelling infantry scenario.

Belief propagation mechanism in command agents for section-platoon-coy:

In Figures 7 and 8, belief propagation mechanism for command agent describes the relationships between teams. Platoon-team’s role is to process the raw information, to gain an adequate level of situation awareness (orientate) and consequently inform its company-team agent the status (location, morale, enemy detection, minefield status, casualty) of both friendly and opposition forces.

In illustrative tactical scenario (Fig. 9), when platoon team belief (synthesised from its subordinates Section Teams) about one of its leading subordinate team (Section Team) is encountered minefield, Platoon team changes its current moving formation by directing other two subordinate teams (following Section Teams) to align in single rod formation in order to have less casualty.

Team agents communicates with other agent via the normal message/event passing in agent-oriented programming. Teams also provides a capability for the propagation of team beliefs. This belief propagation can be both from sub-team to team (upwards) and from team to sub-team (down words). In the former case, the capability is provided within Teams to combine the propagated sub-team beliefs within the team for effective decision making at each level of command and control hierarchy. Sophisticated team behaviours of the command agent teams can be implemented by the use of Team beliefs in conjunction with the Team coordination statements.

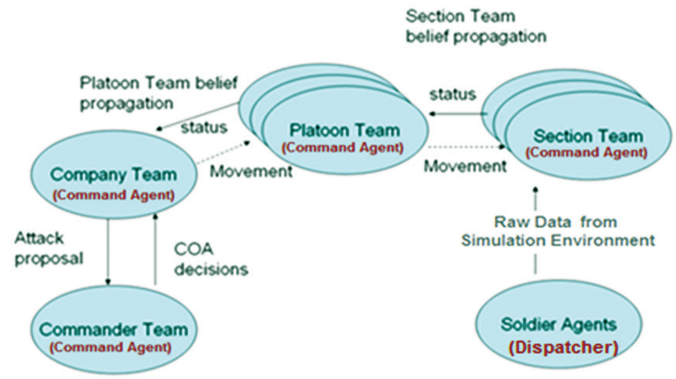

Figure 7. Belief propagation mechanism for command agent.

Similarly, the platoon acts as role tenderer, as it also requires four roles to be performed by four different section teams.

The Platoon team requires one sub-teams able to perform the SectionRole1 role, another sub-team able to perform the SectionRole2, one sub-teams able to perform the SectionRole3 role and one sub team to perform SectionRole4 role (HQtr Role). Furthermore, the platoon team is declared to be a performer of the PlatoonRole Role, which would be a role required by company team.

The Role tendering platoon team (PLATOON_1) requires four Roles as SectionRole1, SectionRole2, SectionRole3, SectionRole4 referenced by sec1, sec2, sec3 and plhqr (Platoon Head Quarter).

Each Platoon team has a belief ‘Platoon Status’, which is (as shown in Fig. 10) updated every time when the belief ‘Status’ of any of its sub ordinate teams namely sections is changes. Every Section has a private belief ‘Status’, which when updated is reflected to its platoon team. Section role performed by section team has ‘Synthesize Status’ declaration

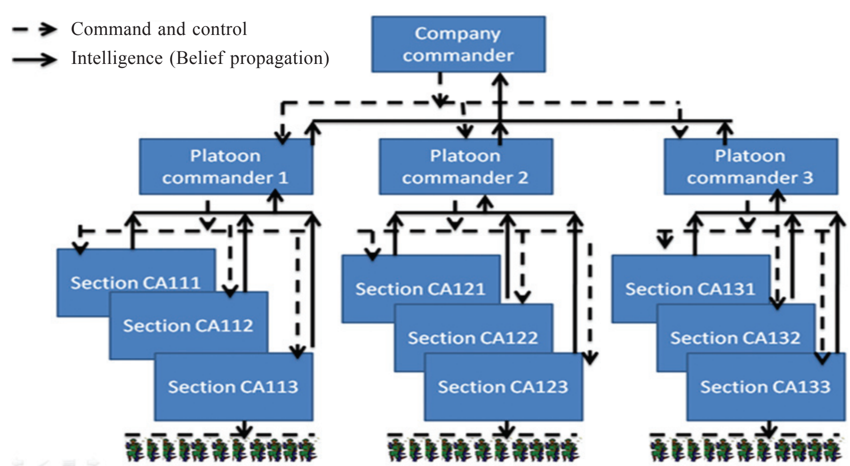

Figure 8. Belief propagation mechanism for command agent in military command hierarchy.

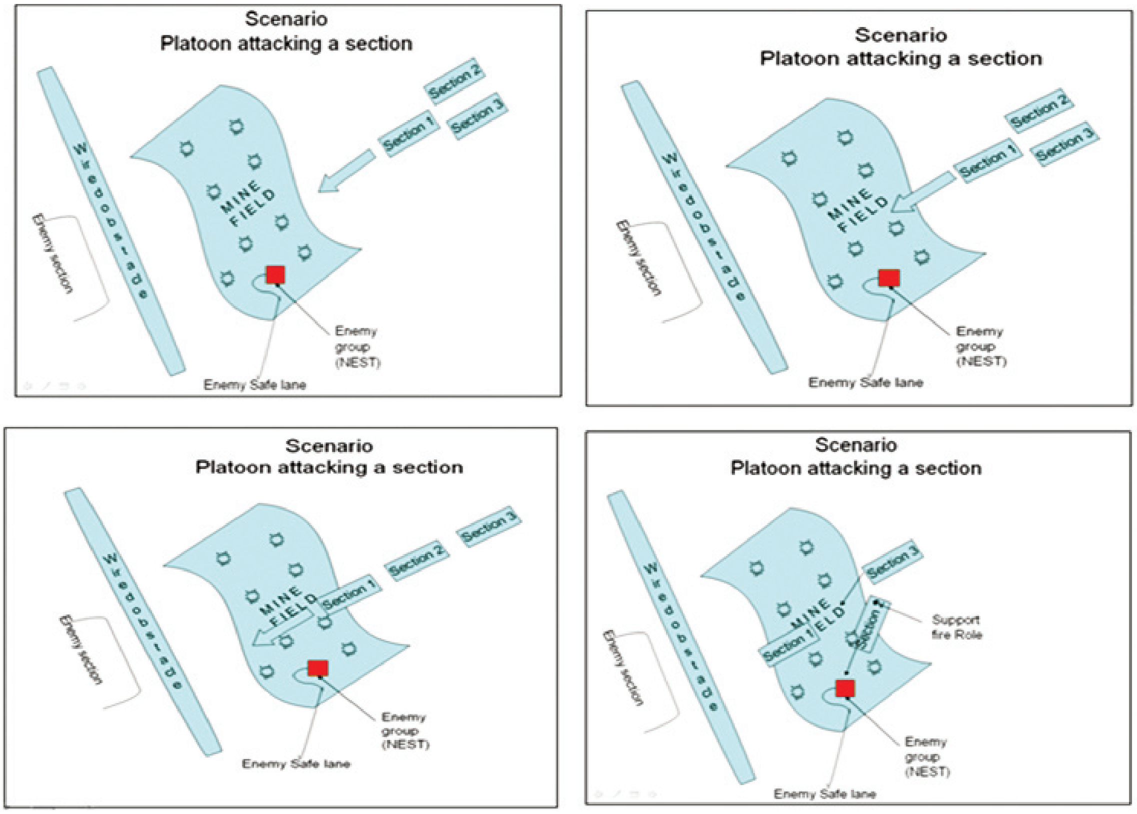

Figure 9. Platoon crossing a enemy minefield.

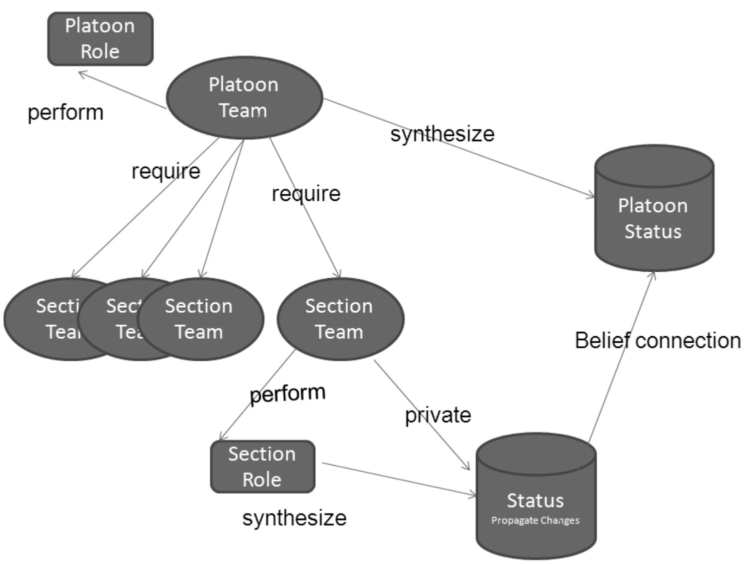

Figure 10. Belief propagation mechanism in command teams.

in its definition. Platoon team has connection method in its belief 'Platoon Status', through which the belief of its sub ordinate teams are gathered. The synthesize method in platoon status does aggregation on gathered belief of its subordinates to arrive at final platoon belief. For example, the synthesise method will perform average of section team locations to give final location of Platoon centre.

The role performer platoon team SEC_1, SEC_2 and SEC_3 and SEC_10 are acting as role filler for roles required by platoon team (PLATOON_1) namely sec1,sec2, sec3 and plhqr. The following section describes details of team belief propagation mechanism in command and control hierarchy.

6.1 Illustrative Scenario

Scenario 1: Command Agent Belief about Minefield crossing by soldiers

This example scenario depicts the decision phase of platoon command agents, when its sections are crossing the enemy mine fields. Initially the section has belief that none of its soldiers has crossed the minefield (Table 2). But during the course of time, if any soldiers encounters the minefield, it informs its respective section about mine filed crossing status. The section then reorients its soldiers position to arrange in rod formation so as to get less casualty further in battlefields. The section also informs its platoon team commander about its mine filed cross status. The platoon commander then updates (Table 3) its belief about section minefield cross section. The changed minefield status of the platoon triggers an event to other section, which eventually change their formation to rod so as to minimize causality.

Platoons team synthesize its belief 'PlatoonSectionsMinefieldCrossBelief' about the section mine cross status Info from section team status. The section team propagates its mine cross status to its platoon command team constantly. The updated belief of platoon team 'PlatoonSectionsMinefieldCrossBelief' triggers an event 'ChangeSectionFormationEvent', which is handled by its platoon team itself by plan 'ChangeSectionFormPlan'. This plan 'ChangeSectionFormPlan' further posts the

'FormationChangeEventToSection' event to its subordinate section team. The section team in turn handles this event by plan 'FormationChangePlan' and posts event 'FormationChangeEventToSoldier' to its soldier agents , which handles this event by plan 'SoldierFormationChangePlan', which is responsible for allignig the soldier in rod formation.

Each platoon has following belief structure, which stores the minefield belief crossing status of its three sections as;

Belief PlatoonSectionsMinefieldCrossBelief extends Openworld

{

Value field int section1_pre_status;

Value field int section2_pre_status;

Value field int section3_pre_status;

Value field int section1_cur_status;

Value field int section2_cur_status;

Value field int section3_cur_status;

Post event ChangeFormationtoRodEventev;

;

}

PlatoonSectionsMinefieldCrossBeliefhas value in its Tuple:

Table 2. Mine cross old belief

section1 _pre _status	section2 _pre _status	section3 _pre _status	section1 _cur _status	section2 _cur _status	section3 _cur _status
0	0	0	0	0	0

If section1 crosses the minefield then the

Platoon belief 'PlatoonSectionsMinefieldCrossBelief' is updated to:

Table 3. Current mine cross belief

section1 _pre_ status	section2 _pre_ status	section3 _pre_ status	section1 _cur_ status	section2 _cur_ status	section3_ cur_ status
0	0	0	1	0	0

In PlatoonSectionsMinefieldCrossBelief belief structure there is a callback, which is triggered as soon as a new belief about minefield is updated by any of its sections.

In this callback following condition triggers/posts the event to platoon, which in turn directs other sections to change their formation to rod

#post eventChangeFormationev;

If(section1_pre_status== 0 and section2_pre_status==0 and section3_pre_status ==0 and section1_pre_status==1 and section2_pre_status==0 and section3_pre_status==0)

{

// event ev handled by other sections to change their formation to rod

post(ev.ChangeFormation());

}

In next iteration, the belief tuple of platoon is updated to:

Table 4. Updated mine cross belief

section1 _pre _status	section2 _pre _status	section3 _pre _status	section1 _cur _status	section2 _cur _status	section3 _cur _status
1	0	0	1	0	0

Since the belief is again changed, but the above condition does not hold, so no event will be fired.

Scenario 2: Request for Arty Fire support

There is a firebase group with major weapons (MMG/ LMG/ArtyGun) situated outside the minefield but near to enemy area. This fire base group fires at enemy, while the three sections are crossing the minefield. When one of the sections (assault role) encounters the mine field, it requests the platoon commander to give arty fire support from fire base. When any section of platoon crosses, minefield, the platoon's mine cross beliefin 'PlatoonLevelStatusInfo' is changed. This belief is read by platoon plan 'Platoon Level Sec Mine Cross Plan' , This plan triggers the event 'Arty Fire Event' handled by 'Arty Fire plan' of Firebase team .

This support fire action of the support section continues until the enemy is suppresses from further firing or the support section suffers heavy causality. The assault section crosses the minefield, it request platoon commander to stop the arty firing and also issues order to support section to stop engaging the enemy ambush section further.

7. CONCLUSIONS

The issue and challenge faced by military commanders for combat entities are continuously changing battlefield environment based on enemy reaction, interaction and coordinating with friendly forces, reporting action to higher command entities under military hierarchy. The agent oriented paradigm has the ability to model complex problem through decomposition, team coordination under command and control hierarchy. Agent technology provides a mechanism for modelling tactical behaviour of combat entities at all level of hierarchy. Higher level entities will be able to synthesize their beliefs derived from the lower level sub ordinates entities. Belief from lower level entities (CA), that is status of enemy detection, weapon engagements, casualty and health, location, fuel, ammunition expenditure of sub ordinate entity are propagated to upper level entities (CA), to help effective decision making under continuously changing battlefield conditions. The command agents can model military commander decision making skill (using belief propagation mechanism), as well as represent the tactical team behaviour among friendly entities, under command and control hierarchy.

REFERENCES

1. Macal, C.M. & North, M.J. Tutorial on agent-based modeling and simulation, Part 2: How to model with agents. *In* Proceeding of Winter Simulation Conference 2006: Edited by L. F. Perrone, F. P. Wieland, J. Liu, B. G. Lawson, D. M. Nicol, and R. M. Fujimoto. Monterey, CA, Dec 2006. pp. 3-6. doi: 10.1109/WSC.2006.323040
2. Davis, P.K. & Bigelow, J.H. Agent based modeling. RAND, National Defense Research Institute, 1998, pp.1-22.
3. Mason, C. R. & Moffat, J. Representing the C2 process in simulations: Modelling the human decision-maker. *In* Proceedings of the **2000** Winter Simulation Conference Edited by J. A. Joines, R. R. Barton, K. Kang, and P. A.

Fishwick. 2000, pp. 940-49.
doi.org/10.1109/WSC.2000.899896

4. Ceruti, M.G.; Wilcox, D.R. & Powers, B.J. Knowledge management for command and control. *In* Command and Control Research and Technology Symposium, June 15-17, 2004, Loews Coronado Bay Resort, San Diego, CA
5. Mason, Colin R. & Moffat, James. An agent architecture for implementing command and control in military simulations. *In* Proceeding of the Winter Simulation Conference, Arlington, VA, USA, 2001, pp. 721-729. doi: 10.1109/WSC.2001.977360
6. Muller, Jörg P. The design of intelligent agent: A layered approach, 1996, (Lecture notes in artificial intelligence). ISBN 3-540- 62003-6. doi: 10.1007/BFb0017806
7. Lucas, Andrew & Goss, Simon. The potential for intelligent software agents in defence simulation. DSTO, 1999. pp.1-5. doi:10.1109/IDC.1999.754219
8. JACK™ Intelligent agents evaluation version. http://www.agent-software.com (Accessed on 15 January 2017).
9. JACK Intelligent agents® Agent Manual. Copyright © 1999-2011, Agent Oriented Software Pty. Ltd.
10. Busetta, Paolo; Ronnquist, Ralph; Hodgson, Andrew & Lucas, Andrew. JACK Intelligent agents–Components for Intelligent agents in Java. paolo@agent-software.com, 1999.
11. Agent Manual,Release 5.3,222 10-June-05, 1999-2009, Agent Oriented Software Pty. Ltd, Teams Manual Release 5.3, 78, 10-June-2005, 2002-2009, Agent Oriented Software Pty Ltd.
12. Agent Oriented Software Pvt. Ltd., JACK Intelligent agent, Team Manual Release 4.1, July 2003.
13. Teams Practicals, Release 5.0,32 10-June-2005, 2003-2009, Agent Oriented Software Pty. Ltd.
14. Tidhar, Gil; Heinze, Clint; Goss, Simon; Murray, Graeme; Appla, Dino & Lloyd, Ian. Using intelligent agents in military simulation or using agents intelligently. *In* IAAI-99 Proceedings. 1999,
15. Wooldridge, M. Reasoning about rational agents, MIT Press. 2000
16. Sharma, Vivek; Sagar, Anit & Taneja, S.B. A review of various agent based resolution modelling methods. *Int. J. Comput. Sci. Info. Technol.*, 2014, **5**(3) , 4469-4472
17. Levesque, H.; Cohen, P. & Nunes, J. On acting together. *In* Proceedings of the Eighth National Conference on Artificial Intelligence*,* pages 94—99, 1990.
18. Hodgson, A.; Rönnquist, R. & Busetta, P. Specification of coordinated agent behavior (The SimpleTeam Approach)', 1999. *In* Proceedings of the Workshop on Team Behaviour and Plan Recognition at IJCAI-99 (International Joint Conference on Artificial Intelligence), August 2, , Stockholm, Sweden.
19. Rao, A.; Lucas, A.; Morley, D.; Selvestrel, M. & Murray, G. Agent-oriented architecture for air-combat simulation. Australian Artificial Intelligence Institute, 1993. Technical Report Technical Note 42.
20. Jennings, N.R. Commitments and conventions: the foundations of coordination in multi agent systems. *Knowledge Engg. Rev.*, 1993, **8**(3), 223-250. doi: 10.1017/S0269888900000205
21. Tambe, M. Implementing agent teams in dynamic multiagent environments. *Appl. Artif. Intell.*, 1998*,* **12**(2-3), 189-210. doi: 10.1080/088395198117820
22. Horn, Graham S. & Baxter, Jeremy W. Co-ordinating agents in the simulated battlefield. 2000 / DERA, Defence Evaluation and Research Agency, DERA Malvern, St Andrews Road, Malvern,WR14 3PS, U.K.
23. Malhotra, Aparna; Taneja, S.B. & Bisht, Sanjay. Simplifying the tactical modelling of armour scenarios using teams of agents. *In* 3rd International Conferenceon Quality,Reliability and Infocom Technology (ICRQIT'06), 2 - 4 Dec, 2006, New Delhi.
24. Bisht, Sanjay; Malhotra, Aparna & Taneja, S.B. Modelling and simulation of tactical team- behavior. *Def. Sci. J.*, 2007, **57**(6), 853-864. doi: 10.14429/dsj.57.1824

CONTRIBUTORS

Mr Sanjay Bisht obtained his MSc (Computer Science) from DAVV University, Indore (MP) and MTech (Computer Technology and Application) from the Delhi Technological University, Delhi. Presently working as Scientist 'E' at DRDO-Institute for Systems Studies and Analyses, Delhi. He is associated with development of combat models for land combat and military simulation. His areas of interest are : Wargame, land combat modelling, genetic algorithm, heuristic optimisation, simulated annealing, and intelligent agent technology.

Dr H.S. Bharati received his MSc (Mathematics) and in Computer Science and Application. Presently working as Scientist 'F' at DRDO-Institute for Systems Studies and Analyses, Delhi and associated with design and development of combat models for constructive simulation perspective of land combat. His areas of research include : Constructive military simulation, combat modelling from micro-to-macro combat processes, multi-resolution modelling, complexity theory, AI and expert systems, algorithms, and programming paradigms.

Mr S.B. Taneja, obtained his MSc (Electronics) from University of Delhi, and MTech (Computer Science) from Indian Institute of Technology, Roorkee. Presently working as Scientist 'G' at DRDO-Institute for Systems Studies and Analyses, Delhi. He is involved in design and development of wargame simulation systems. His areas of interest and research are : Distributed simulation, wargame simulation system design, geographical information system, and agent-based models for constructive simulation.

Prof. Punam Bedi received her and MTech (Computer Science) from IIT Delhi, in 1986 and PhD (Computer Science) from the University of Delhi, in 1999. She is an Professor in the Department of Computer Science, University of Delhi. Her research interests include: Web intelligence, soft computing, semantic web, multi-agent systems, intelligent information systems, intelligent software engineering, intelligent user interfaces, requirement engineering, human computer interaction (HCI), trust, information retrieval, personalisation, steganography and steganalysis.