Defence Science Journal, Vol. 72, No. 1, January 2022, pp. 49-55, DOI : 10.14429/dsj.72.17039

Task Scheduling in Fog Node within the Tactical Cloud

Biji Nair* and S. Mary Saira Bhanu

Department of Computer Science and Engineering, National Institute of Technology, Tiruchirappalli - 620 015, India
**E-mail: nairbiji@rediffmail.com*

ABSTRACT

Fog computing architecture competent to support the mission-oriented network-centric warfare provides the framework for a tactical cloud in this work. The tactical cloud becomes situation-aware of the war from the information relayed by fog nodes (FNs) on the battlefield. This work aims to sustain the network of FNs by maintaining the operational efficiency of the FNs on the battlefield at the tactical edge. The proposed solution monitors and predicts the likely overloading of an FN using the long short-term memory model through a buddy FN at the fog server (FS). This paper also proposes randomised task scheduling (RTS) algorithm to avert the likely overloading of an FN by pre-empting tasks from the FN and scheduling them to another FN. The experimental results demonstrate that RTS with linear complexity has a schedulability measure 8% - 26% higher than that of other base scheduling algorithms. The results show that the LSTM model has low mean absolute error compared to other time-series forecasting models.

Keywords: Tactical cloud; Network-centric warfare; Fog computing; Fog nodes; Randomised task scheduling; Long short-term memory

NOMENCLATURE

abs_err	Mean absolute error
X^i_{real}	Actual value of CPU usage
$X^i_{predicted}$	Predicted value CPU usage
$F^{'}_{BW}$	Buddy FN
F_W	FN pair of Buddy FN $F^{'}_{BW}$
T	Task list of size M
F	FN list of size N
P_T	Preference list of tasks in T
$Allot$	Final scheduling list
A_{ij}	$T_i \rightarrow F_j$ Task-FN allocation

1. INTRODUCTION

The information revolution has transformed warfare doctrines from platform-centric warfare (PCW) to network-centric warfare (NCW)[1]. There is a paradigm shift in warfare strategy from centralised to distributed command and control (C2)[2,3] tactics. The modus operandi of the PCW is the physical attrition of the opponent through a centralised C2 using autonomous, closed, self-sustaining and independent weapon systems or platforms, leading to a lack of situational awareness and making it impossible to use other platforms to engage the target.

NCW is a decentralised combat model realised by the advances in technology and communication systems and operates through robust networking of forces engaged in a mission. The situational awareness of the battlefield through information sharing aids in self-synchronisation and speed of combat, which increases the precision of action and mission effectiveness. NCW dramatically improves the speed of the observe, orient, decide and act (OODA) cycle of the force. It renders a digital battlefield to achieve high performance in terms of maximum enemy damage while incurring minimal loss through precise and fast decision-making. NCW integrates the technologies on the battlefield (sensor grid, communication technology) and those thousands of miles away (commander centre, satellite) to make the real-time integrated common operational picture (COP) available to every stakeholder of the battle enabling timely decision making.

Cloud computing is a solution for decentralised combat operations[3]; however it does not envisage the vision of NCW that demands the execution of precise and highly time-critical mission-centric operations. Cloud processes the data at its data centre in a distant location, making it incapable of meeting the critical needs of combat operations. Extending cloud services near the tactical end through fog computing[4] assures the operational requirement of NCW.

The term 'tactical cloud'[5, 6, 7, 8] or 'combat cloud' used in this paper is the fog computing framework that implements NCW in defence. Fog nodes (FNs)[9] in the battlefield are instrumental in providing computing, storage and networking services. FNs on tactical edge systems capture real-time data on the battlefield as well as disseminate and share information. The tactical edge sensing devices generate enormous amounts of data from various technologies, such as the smart sensor network, unmanned aerial vehicles (UAV) and autonomous multi-agents. FNs aggregate the data from sensors, perform elementary data analysis and forward relevant information to the fog server (FS). The FS relays the information from FNs to the C2 unit at the tactical cloud for obtaining a COP for higher-level decision-making. FSs connect the tactical

Received : 05 May 2021, Revised : 03 October 2021
Accepted : 02 November 2021, Online published : 05 January 2022

edge to the centralised military-owned cloud for real-time and near real-time data processing and quick information sharing. The operational efficiency of FNs is critical for the realisation of network-centric operations. Maintaining FNs in optimal operational conditions is crucial for the lethality and survivability of the combat units.

This work aims to maintain the FN in optimal operational condition to ensure the sharing of quality information. In the proposed architectural framework of tactical cloud, FSs are at a safe location near the battlefield, usually in the communication base (signal corps infrastructure)[10,11]. The FS hosts a buddy FN for every FN deployed on the battlefield. The purpose of the buddy FN is to monitor the operational parameters (CPU, memory, disk, network, power consumption) of its FN pair in real-time. A buddy FN in the proposed work employs the long short-term memory (LSTM) model trained for predicting the operational efficiency of the FN pair. The LSTM model at the buddy FN uses the CPU utilisation history recordings of the FN pair for predicting its working condition. An FN underperforms and eventually gets overloaded if its CPU utilisation exceeds the threshold value. Since an underperforming FN cannot provide real-time services, the buddy FN considers relieving the workload of its FN pair by pre-empting one or a few of its tasks to avoid overloading. On likely overloading of an FN on the battlefield, its buddy FN sends scheduling requests to the FS for tasks that would be pre-empted to relieve the FN. This work proposes a randomised task scheduling (RTS) policy for the FS to schedule such tasks to FNs with a lighter workload.

This paper proposes a solution for maintaining the fog network in a highly dynamic setup of the battlefield to ensure NCW. The proposed work is of high relevance as it provides a framework for the potential application of artificial intelligence[12, 13, 14] in defence for efficient data acquisition and precise decision making.

2. MATERIAL AND METHODS

2.1 Architecture of Tactical Cloud

Figure 1 elaborates the architecture of the tactical cloud. Figure 1(a) provides the deployment of force. Subsequently, the FNs, FSs and C2 are deployed accordingly on the battlefield, forming the tactical cloud, as seen in Fig. 1(b). Figure 1(c) shows the functional structure of the tactical cloud in hierarchical form.

The tactical cloud is primarily a 3-tier system and the layers are:

(i) Tactical edge:- It is the bottom-most tier comprising the disaggregated network of sensors, data collecting devices and the weapon systems deployed on the battlefield.
(ii) Fog layer:- FNs, buddy FNs and FSs form the fog layer, connecting the tactical edge to C2 and the defence cloud.
(iii) The tactical cloud:- It forms the C2 of the combat force and is at a safe distance from the point of action. Relevant data reach the tactical cloud in real-time through the fog layer from the tactical edge.
(iv) Defence cloud:- Above all is the centralised cloud owned by the Department of Defence for the collaboration and coordination of various departments, forces and defence activities.

C2 decides the type, the number of the sensors and the initial deployment of the sensors on the battlefield using a strategic model. This model provides the layout for the distribution of FNs on the battlefield such that they cover the sensors optimally within their range. The battlefield situation awareness depends upon the coverage and control of the FNs over the sensors. C2 assigns the authority and control of FNs in a well-defined area to an FS. C2 controls and directs the FNs through the FS.

FS instils in itself a worker host called buddy FN for each FN deployed and pairs them. The FN updates its status to its buddy FN. The buddy FN has the image of its FN pair. FN being vulnerable, the buddy FN acts as the backup of its FN pair. In the event of loss or damage of an FN, the FS replaces it with a new FN using a copy of the virtual image[15] from its buddy FN. The FS manages the FNs through their buddy FNs.

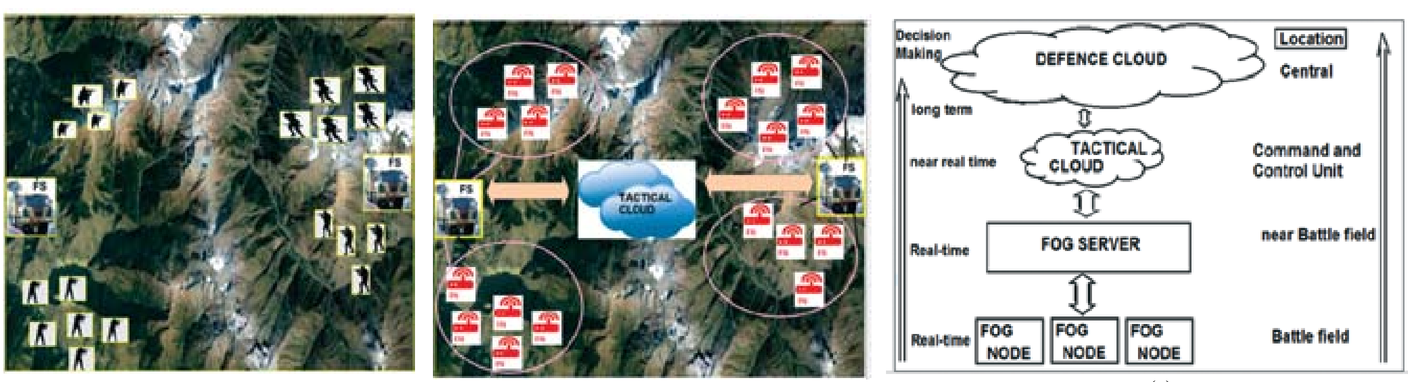
2.2 Problem Statement

Given the pivotal role of FNs in the mission-critical operation of NCW, the operational efficiency of FNs is critical. Hence, the problem this paper addresses is:

'to sustain the FNs in the tactical edge by maintaining them in optimal working conditions by ensuring their operational efficiency'.

Following are the assumptions made while designing the fog computing framework and the solution:

(i) The buddy FN is a virtual machine in the FS that hosts an image of its FN pair.



(a) (b) (c)

Figure 1. Architecture of the tactical cloud: (a) Deployment of force, (b) Deployment of the tactical cloud, and (c) Hierarchical structure of tactical cloud.

(ii) The FN updates its status to its buddy FN.
(iii) This work limits its span to a mission-centric operation in NCW.
(iv) The role of the defence cloud is overall supervision of the mission operation and, hence, is not visible here.

2.3 Proposed Solution

There are several existing task scheduling solutions for near-end services at the network edge. Qin[16], *et al.* have provided reconnaissance utility-based task scheduling in multi-access edge computing enabled by UAV. Tang[17], *et al.* have conceived functions (monitoring, scheduling) as microservices for data sharing among aircraft nodes. Various aspects of task scheduling in different warfare scenarios, such as operational coordination in combat clusters[18], cross-platform task scheduling in warship networks[19] and route finding for information transmission in mobile ad hoc networks formed using UAV[20] are under study. There are also hybrid solutions[20] of autonomous intelligent combat systems for C2, including manned and unmanned systems. Solutions for modelling the tactical edge[21,22] for information processing and decision making in military operations have also drawn interest. Task scheduling in fog computing is an active area of research, with solutions that focus on parameters[23] such as response time, delay, latency, cost and load balancing. However these task scheduling solutions become impractical for the highly dynamic and critical environment of the tactical cloud for NCW. The FN network at the tactical edge must be reliable, operationally efficient and fault-tolerant. NCW demands collection and rapid analysis of data, relay of quality information, low latency response, interoperability among devices, low bandwidth consumption and scheduling and processing of tasks. A solution for the efficient execution of a mission-critical operation at the tactical edge must incorporate these features. In the time-critical mission of NCW, the optimal functioning of the fog network is vital. The FNs must operate under optimal conditions to deliver maximum efficiency for real-time effectiveness.

This paper presents the solution in two phases: monitoring FNs for operational efficiency and then pre-empting and scheduling tasks in overloaded/underperforming FNs to maintain their performance. Fig. 2 provides the workflow of the proposed solution.

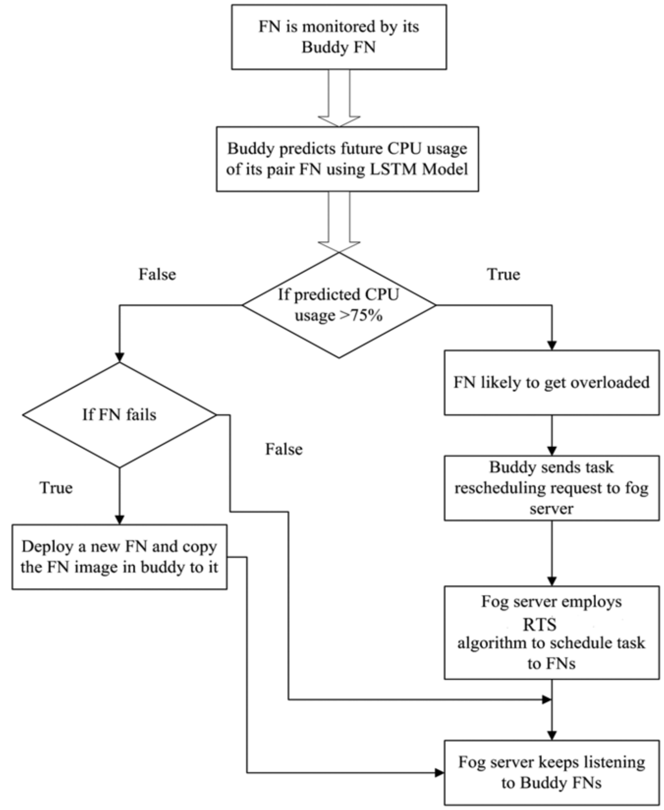


Figure 2. Workflow of the proposed solution.

Phase I: Monitoring And Predicting the Operational Efficiency of FNs

The buddy FN employs LSTM, the recurrent neural network, for predicting the CPU utilisation of its FN pair. The LSTM is efficient in handling sequential time-series data, such as the CPU utilisation parameter of the FN pair. The LSTM has an internal memory that stores the data history and offers superior performance and high forecasting accuracy when compared withits counterparts such as the ARIMA[24] model. The buddy FN monitors and profiles the CPU utilisation of its FN pair on the battlefield and uses the profiled history to predict the future CPU utilisation of the FN. CPU utilisation above the optimal level (the CPU utilisation threshold value) lowers the efficiency of the FN dramatically. The LSTM successfully captures long-term dependencies and provides an accurate prediction of CPU utilisation, with a minimal error rate when compared with other neural networks. This work evaluates the mean absolute error of the predicted value with the actual value to assess the accuracy of the model. Eqn (1) defines the mean absolute error mathematically.

$$abs\_err = \frac{\sum_{i=1}^{n} | X^{i}_{predicted} - X^{i}_{real} |}{n} \tag{1}$$

where abs_err is the mean absolute error, X^{i}_{real} and $X^{i}_{predicted}$ are i^{th} actual and predicted value, respectively.

The ability of the LSTM to process time-series data step-by-step, along with the feature to remove or add information to the cell state, precisely regulated by the gates, efficiently predicts the operational efficiency of the FNs. The LSTM model is the most appropriate one for predicting the performance of FN in a dynamic environment, such as a battlefield, because it learns from the most relevant and recent past behaviours of the FN for predicting its future performance under the current workload.

Phase II: Workload Scheduling

Two cases trigger workload scheduling. One is when an FN turns hostile or fails; in this case, the FS has to replace it with another FN. The FS loads the new FN with the image from the respective buddy FN, pairs it with the buddy and deploys it on the battlefield. The other is when a buddy FN predicts the probable overloading of its FN pair. To avert the overloading of the FN pair, the buddy FN pre-empts one or more tasks in the FN pair to maintain it in optimal working conditions. The buddy FN places scheduling requests for the pre-empted tasks to the FS. Furthermore, the buddy FN provides the FS with the list of the FNs that each of these preempted tasks prefer in the descending order of their preference. The FS executes the procedure of scheduling these tasks to a nearby capable FN. The FS assesses the FNs regularly and prepares a list of FNs with CPU utilisation below the threshold value. Buddy FNs use this list to prepare the preference list for the tasks to be scheduled. The FS follows the RTS method in Algorithm 1 to schedule the tasks without hindering the operational efficiency of the destination FNs.

Algorithm 1 includes both cases: the FN failure and the underperformance of FN due to overloading. The availability of resources such as CPU, memory, disk and network defines the capability of an FN. An FN can serve a task if it can guarantee the resource requirement of the task and serve it within its threshold CPU utilisation value. The FS forms a set T of the tasks received at any instant t for scheduling and passes T as an input to the RTS algorithm. The FS executes the RTS algorithm to select a task from T for scheduling and schedules it to the FN that the chosen task has first on its preference list. The FS removes the FN scheduled from the list of preferences of all the other tasks. The FS repeats these steps until the preference list and the list of available FNs are empty. Eqn (2) provides the FN priority value of a task. The FN preference list for a task is the descending order of FN priority as provided by the buddy FN. The buddy FN selects the FN for the preference list only if it is within the proximity threshold value of the FN pair and has CPU utilisation below the threshold value.

$$Priority = \frac{cpuTh - [PredCPU + TaskCPU]}{ProxTh} \quad (2)$$

where $cpuTh$ is the CPU utilisation threshold value, $PredCPU$ is the predicted CPU utilisation value of the FN, $TaskCPU$ is the CPU utilisation required by the task and $ProxTh$ is the proximity range provided by the buddy FN. The criteria for choosing the proximity threshold value depend on the diameter of the transmission range of the sensors providing data for the task.

Algorithm 1 (RTS)

Input : Buddy FN $F^{'}_{BW}$ sends request to FS to replace or relieve the FN F_W .

Output: Replacing or relieving FN F_W by FS.

1. FS listens($BuddyRequest \quad F^{'}_{BW}$)
2. while ($BuddyRequest == True$)
   - 1.1 if ($BuddyRequestNo == 1$)
     - i. Action is to replace FN F_W .
     - ii. FN with equivalent capability of F_W is selected.
     - iii. Image of F_W is loaded in new FN from $F^{'}_{BW}$.
     - iv. New FN is deployed in battlefield.
   - 1.2 else if ($BuddyRequestNo == 2$)
     - i. Buddy FN $F^{'}_{BW}$ forwards list T of the task in F_W to be scheduled in another FN.
     - ii. FS creates a list F of FNs in the transmission range of sensors monitored by F_W .
     - iii. FS also creates a preference order of FNs for each task T_i using Equation 2.
     - iv. Confirm Resource requirements (CPU, memory, disk, network) of task less than Resource Available in each of FNs in F .
     - v. FS executes $RTS(T, F)$.
   - 1.3 else Check $BuddyRequest$.
3. End

procedure $RTS(T, F)$

4. $T = \{T_1, T_2, ..., T_M\}$ P_T denotes the list of tasks for FN allocation at the FS.
5. $F = \{F_1, F_2, ..., F_N\}$ denotes the list of tasks for available FNs for allocation.
6. $P_T = \{P_{T_1}, P_{T_2}, ..., P_{T_M}\}$ denotes the preference list of each task in T .
7. $Allot$ denotes the final task allocation to FNs
8. while (P_T, F) are not empty do
   - 8.1 Randomly choose a task in T_i in T and schedule it to the FN F_j of its first preference in P_{T_i} .
   - 8.2 Place the scheduled task-FN pair $A_{ij} = T_i \rightarrow F_j$ for allocation in the list $Allot$.
   - 8.3 Remove the task T_i from T , FN F_j from F and preference list P_{T_i} from P_T .
9. end while
10. return task scheduling list $Allot$.

3. RESULTS AND DISCUSSION

The evaluation of the proposed task-scheduling algorithm is done in a laboratory setup similar to another research works in this area of research[25]. A standard desktop computer with Ubuntu OS simulates the FS. The VMs in different computers execute CPU and memory-intensive tasks to mimic the FNs on the battlefield. Xen hypervisor creates VM to simulate the FN pair and buddy FN in the experimental set of FS. SysBench[26] workload benchmark provides CPU and memory-intensive tasks as workload to the FNs. Each FN profiles its CPU and memory utilisation, which is updated to its buddy FN every 5 ms. The buddy FN predicts the CPU utilisation of the FNs based on the profiled history of the FN pair and moves on to the workload scheduling phase if the FN is likely to be overloaded. The buddy FN pre-empts the tasks one by one from its underperforming FN pair until its CPU utilisation is below the threshold value. In the experiments, the task preemption follows the descending order of the CPU utilised by the tasks.

3.1 Phase I: Monitoring And Predicting the Operational Efficiency of the FNs

The result of the LSTM model evaluation at the buddy FN shows that it accurately predicts the future working condition of the FN pair. Each FN samples its CPU and memory usage/utilisation over 5 ms, while the FN serves CPU-intensive and memory-intensive tasks as workloads. The buddy FN monitors its FN pair from the profiles and predicts its future CPU utilisation using the LSTM model. Here, 70% of the data sampled trains the LSTM model, 20% validates the model and the remaining 10% tests it.

Figure 3 shows the profiling data of CPU and the memory usage parameters of an FN sampled every 5 ms. Figure 4 represents the successful forecasting of the CPU utilisation

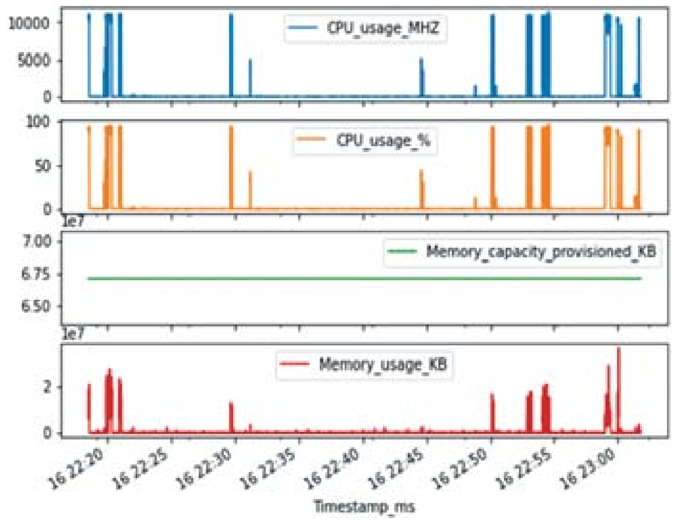


Figure 3. Monitoring of CPU and memory utilisation of an fog node (FN).

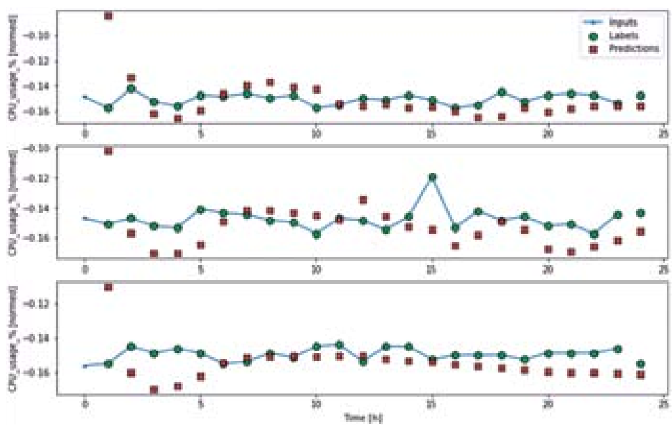


Figure 4. CPU utilisation prediction using LSTM.

of the FN pair based on the history of the CPU utilisation parameter received from the FN pair.

Figure 4 predicts the CPU utilisation (in percentage) 1 h into the future using the LSTM model. It takes a window of 24-hour time-series input data from the dataset with all its features to predict the CPU utilisation 1 hour into the future. Figure 4 illustrates the prediction of the CPU utilisation of the FN pair in eachy time step of 1 hour from its profiled data by its buddy FN. The CPU utilisation value depicted by filled circles in Fig. 4 denotes the target prediction value in response to the previous input step. The cross symbol is the actual CPU utilisation value predicted by the LSTM model trained over the dataset, while the continuous line in the graph of Fig. 4 is the actual CPU utilisation profiled at that hour. The y-axis of the graph in Fig. 4 shows the CPU utilisation value scaled using normalisation, which is usually done while training the model. Fig. 4 confirms that the CPU utilisation value of an FN predicted for a moment in the future is close to the actual value recorded later at that moment. Hence, the LSTM model adopted by the buddy FN to predict the working condition of its FN pair using the history of the profiled CPU utilisation of the FN is reliable.

Figure 5 compares the performance of the LSTM model with other time-series forecasting models, such as baseline, linear, dense, multi-step dense and convolutional models, in

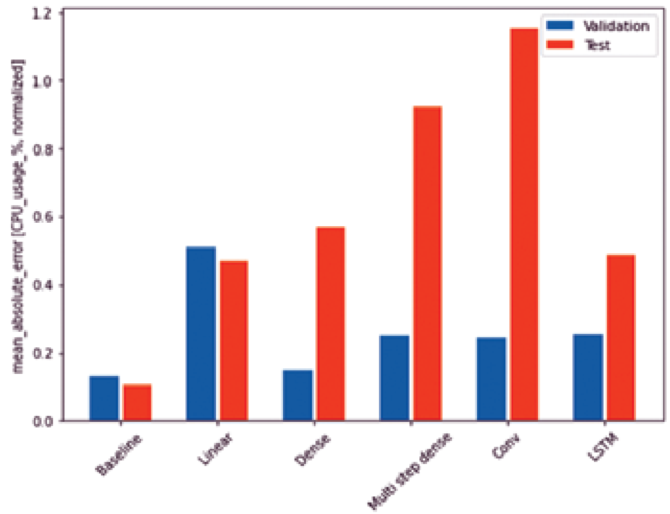


Figure 5. Comparison of the LSTM model with other algorithms.

terms of the absolute mean error in the predicted value. The LSTM model has low absolute mean error as per Fig. 5. Accordingly, the baseline model has the least absolute mean error. However, the baseline model is unrealistic as it predicts the next step from just the previous step and does not perform well if predictions are to be made further into the future.

3.2 Phase II: Workload Scheduling

The result of the experiments in the second phase of the proposed algorithm, RTS, has lower time complexity than the other base scheduling methods such as First Come First Serve (FCFS), Shortest Job First (SJF), laxity and greedy. Fig. 6 demonstrates RTS has linear time complexity while the other methods have polynomial time complexity. These methods spend a substantial amount of time pre-processing the list of tasks as per their policy. The buddy FN prepares a task preference list for each task and forwards it along with the scheduling request to the FS, because of which the RTS takes much less time to implement random policies while scheduling the tasks.

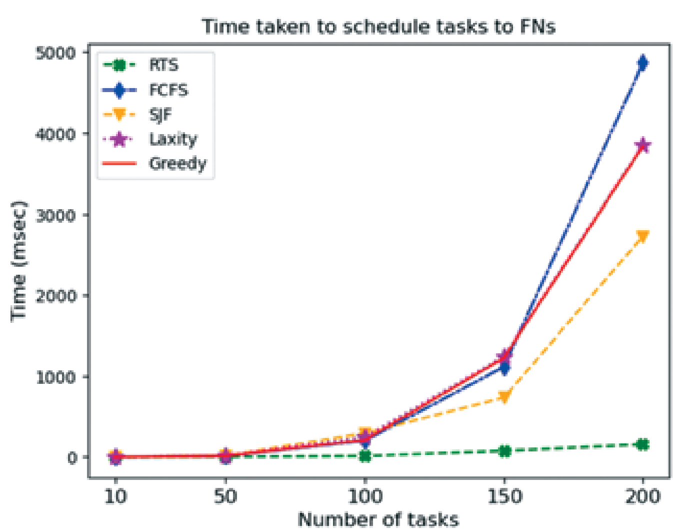


Figure 6. Time complexity comparison of the scheduling algorithms.

Table 1 compares the scheduling solutions as per their schedulability (i.e. the ratio of the number of tasks an algorithm successfully schedules to the number of task requests it receives). The first column gives the total number of task requests the scheduling algorithm receives and the cells provide the number of tasks successfully scheduled by them. Table 1 demonstrates that the RTS algorithm possesses the highest schedulability even when there is an increasing number of the task scheduling requests at the FS. RTS schedules more number of the tasks than the other scheduling algorithms with an increase of 8% - 26% in schedulability.

Table 1. Schedulability comparison

No. of tasks	Scheduling algorithms				
	RTS	FCFS	SJF	Laxity	Greedy
200	85	90	77	82	76
150	60	52	56	52	58
100	40	37	34	37	32
50	18	16	18	16	13
10	3	2	2	3	2

The proposed RTS algorithm has linear time complexity, which is in contrast to the polynomial time complexity of the other scheduling algorithms. Solutions with linear complexity can serve highly dynamic, critical and time-sensitive scenarios of NCW. The proposed solution is capable of sustaining the fog network on the battlefield through the buddy-FN pair strategy. Unlike a battleground, the experimental evaluation of the proposed solution in a lab set-up or a simulator provides an ideal condition for communication. Hence, there is a need to test the two-phased RTS solution under low bandwidth, uneven terrain and volatile conditions of the battlefield for further improvement, such as using relay networks with UAV[27]. A battlefield-like situation can also be used to test the agility of the fog network. There are several challenges to overcome the implementation of tactical cloud and the defence department of various countries is working towards it in various sphere[28,29,30].

4. CONCLUSION

NCW has a decisive war-fighting advantage over its enemy through a lethal combination of quick decision-making, tactics and techniques. The presented tactical cloud framework integrates real-time information for NCW through the networking forces involved in the mission. The framework provides quality information to create situational awareness in C2 for undertaking tactical operations. The LSTM model employed by the buddy FNs and the RTS policy at the FS meet the time dynamics required for NCW operation. The tactical cloud is a promising platform for networking various combat units in mission-centric operations and offers an edge over the enemies through a robustly connected network. Nonetheless, this work must be evaluated in the real-life scenario of tactical networking environment. Future studies should be aimed at incorporating all other complex scenarios on a battlefield, such as self-destructing FNs in case of node compromise.

REFERENCES

1. Guha M. Technical ecstasy: Network-centric warfare redux. *Security Dialogue*. 2021. doi:10.1177/0967010621990309.
2. Mathiassen, K.; Schneider, F.E.; Bounker, P.; Tiderko, A.; Cubber, G.D.; Baksaas, M.; Główka, J.; Kozik, R.; Nussbaumer, T.; Röning, J. and Pellenz, J. Demonstrating interoperability between unmanned ground systems & command and control systems. *Int. J. Intell. Def. Support Syst.*, 2021, **6**(2). doi:10.1504/IJIDSS.2021.10037444.
3. Snyder, D., Lynch, K. F., Steiner, C. P., Drew, J. G., Hura, M., Marlier, M. E., & Milonopoulos, T. Command and Control of US Air Force Combat Support in a High-End Fight. *RAND Corp Santa Monica CA.*, 2021.
4. Iorga, M.; Feldman, L.; Barton, R.; Martin, M.; Goren, N. & Mahmoudi, C. The nist definition of fog computing. *National Institute of Standards and Technology*, 2017, 800-191. doi:10.6028/NIST.SP.500-325
5. Choi, S. O. & Kim, J. B. National Defense Cloud Strategy. *In* Proceedings of the SNPD-Winter, IEEE, 2021, 87-89. doi: 10.1109/SNPDWinter52325.2021.00026
6. Yego, K.; Thyer, M.; Jones, T.; & Davidson, R. A heterogeneous tactical cloud architecture for emergency services search and rescue operations. *In* Proceedings of the Military Communications and Information Systems Conference, IEEE, 2020, 1-6. doi:10.1109/MilCIS49828.2020.9282376
7. Vidal, J. M. & Monge, M. A. S. Denial of sustainability on military tactical clouds. *In* Proceedings of the 15th Int. Conf. on Availability, Reliability and Security, 2020, 1-9. doi:10.1145/3407023.3409226
8. Magar, A. Assessing the use of tactical clouds to enhance warfighter effectiveness. *Sphyrna Sec. Inc. Kanata*, *ON Canada*, 2014.
9. Marín-Tordera, E.; Masip-Bruin, X.; García-Almiñana, J.; Jukan, A.; Ren, G. J. & Zhu, J. Do we all really know what a fog node is? Current trends towards an open definition. *Computer Communications*, 2017,**109**, 117-130. doi:10.1016/j.comcom.2017.05.013
10. Anderson, B. W. & Garfinkel, G. S. Maintaining the Information Flow: Signal Corps Manpower and Personnel Requirements for the Battlefield. Army Research Lab Fort Monmouth N, 2004. https://apps.dtic.mil/sti/pdfs/ADA465971.pdf [Accessed on 24-05-2020].
11. Department of The Army, United States of America: FM 6-02 Signal Support to Operations, September 2019. 1-22. https://fas.org/irp/doddir/army/fm6-02.pdf [Accessed on 24-05-2020].
12. Toth, S. & Hughes, W. The journey to collaborative AI at the tactical edge (CATE). *Artificial Intell. Mach. Learning for Multi-Domain Operations Applications III,* 2021, 11746, 117460O. doi:10.1117/12.2586096
13. Sturzinger, E. M.; Lowrance, C. J.; Faber, I. J.; Choi, J. J. & MacCalman, A. D. Improving the performance of AI models in tactical environments using a hybrid cloud architecture. *Artificial Intell. Mach. Learning for*

Multi-Domain Operations Applications III, 2021, 11746, 1174607.
doi:10.1117/12.2587932

14. Schwartz, P.J.; O'Neill, D.V.; Bentz ,M. E.; Brown, A.; Doyle, B.S.; Liepa, O. C.; Lawrence, R. & Hull, R. D. AI-enabled wargaming in the military decision making process. Int. Conf. on Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications. *Int. Society for Optics and Photonics*, 2020, p. 114130. doi:https://doi.org/10.1117/12.2560494
15. Zhang, K.; Li, L.; Li, Y.; Xie, B.; Fei, C. & Wang, Z. Resource-Visualization-based Cooperative Architecture and Mechanisms within the Tactical Clouds. *In* Proceedings Int. Conf. on Computer and Information Science (ICIS), *IEEE*, 2019, 512-517. doi: 10.1109/ICIS46139.2019.8940217
16. Qin, Z.; Wang, H.; Wei, Z., Qu; Y., Xiong, F.; Dai, H. & Wu, T. Task Selection and Scheduling in UAV-Enabled MEC for Reconnaissance with Time-Varying Priorities. *IEEE Internet Things J.,* 2021. doi: 10.1109/JIOT.2021.3078746
17. Tang, L.; Hu, H.; Wang, Z.; Wang, J. & Li, Y. Microservice Architecture Design for Big Data in Tactical Cloud. *In* Proceedings Int. Conf. on Big Data and Security (ICBDS). Springer, 2019, Dec 2019, 402-416. doi: 10.1007/978-981-15-7530-3_31
18. Zhao, L.; Wang, B. & Shen, C. A multi-objective scheduling method for operational coordination time using improved triangular fuzzy number representation. *Plos One,* 2021, **16**(6). doi: 10.1371/journal.pone.0252293.
19. Wu, S.; Wen, Y.; Xia, Y.; Wang, H. & Li, H. Cross-platform task scheduling algorithm of warship formation network under bandwidth constraints. *Chinese J. Ship Res.*, 2020. **15**(6), p.83. doi: 10.19693/j.issn.1673-3185.01899
20. He, H.; Wang, W.; Zhu, Y.; Li, X. & Wang, T. Function chain-based mission planning method for hybrid combat SoS. *IEEE Access*, 2019, 100453-66. doi:10.1109/ACCESS.2019.2928524.
21. Du, B.; Shan, N. & Zhou, S. An Efficient Mobile Cloud Service Model or Tactical Edge. *In* Proceedings Int. Conf. on Intelligent Networking and Collaborative Systems. Springer, 2019. 422-430. doi: 10.1007/978-3-030-29035-1_41.
22. Nithya, S.; Sangeetha, M.; Prethi, K.A.; Sahoo, K.S.; Panda, S.K. & Gandomi, A.H.;. SDCF: A software-defined cyber foraging framework for cloudlet environment. *IEEE Trans. Network Service Management*. 2020, 2423-2435. doi:10.1109/TNSM.2020.3015657
23. Kaur, N.; Kumar, A. & Kumar, R. A systematic review on task scheduling in Fog computing: Taxonomy, tools, challengesand future directions. *Concurrency and Computation: Practice and Experience*, 2021, 6432. doi: 10.1002/cpe.6432
24. Siami-Namini, S.; Tavakoli, N. & Namin, A. S. A comparison of ARIMA and LSTM MODEL in forecasting time series. *In* Proceedings Int. Conf. on Machine Learning and Applications (ICMLA), *IEEE*, 2019, 1394-1401. doi:10.1109/ICMLA.2018.00227
25. Nair, B. & Somasundaram, M.S.B.. Overload prediction and avoidance for maintaining optimal working condition in a fog node. *Comput. Elect. Eng.*, 2019, **77**, 147-162. doi: 10.1016/j.compeleceng.2019.05.011
26. https://wiki.gentoo.org/wiki/Sysbench#Using_the_CPU_workload, [Accessed on 25-09-2021].
27. Dong, J.; Wu, G.; Yang, T. & Jiang Z. Battlefield situation awareness and networking based on agent distributed computing. *Physical Commun.,* 2019, 178-86. doi: 10.1016/j.phycom.2019.01.002.
28. Gros, P. The 'tactical cloud', a key element of the future combat air system. Note 19/19, 2nd Paris, Fondation pour la recherche stratégique. 2019. https://www.frstrategie.org/en/publications/notes/tactical-cloud-key-element-future-combat-air-system-2019. [Accessed on 30-09-2021].
29. https://media.defense.gov/2019/Feb/04/2002085866/-1/-1/1/DOD-CLOUD-STRATEGY.PDF, [Accessed on 30-09-2021].
30. https://tasdcrc.com.au/wp-content/uploads/2021/02/JSPS_4.pdf, [Accessed on 30-09-2021].

CONTRIBUTORS

Ms Biji Nair, Lecturer on Computer Engineering at Govt. Polytechnic College, Neyyattinkara, Kerala, India. She is currently pursuing a PhD degree at the National Institute of Technology, Tiruchirappalli, India. Her research area is fog computing. She is working on the problem of maintaining FN in optimal working conditions for serving latency-sensitive tasks. She has extended her research findings in defence application of the tactical cloud.
She has contributed from proposing the solution for the problem, its evaluation and the presentation as a research paper.

Prof. S. Mary Saira Bhanu is a Professor in the Department of Computer Science and Engineering, National Institute of Technology, Tiruchirapalli. Her current research interests include Operating Systems, Distributed Systems, Real-time systems, Grid Computing.
Her contribution to this work is relentless guidance throughout the research until the scripting of the research paper.