International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-8 Issue-6, March 2020

Provisional Access of Workflow Scheduling With Mobile Agents in Agricultural Application

N. Priyadharshini, V. Narayani

Abstract: ***In mobile computing, if agent experiences are assessed and available among end communication for environmental modelling, it helps in improving exploration load for unknown or unvisited circumstances. Therefore, it may speed learning procedure. As, building an accurate and effectual model with constraint time is also an essential factor, specifically for difficult conditions, this work initiates reinforcement model base learning approach based on work flow scheduling to acquire lesser memory consumption and effectual modelling. Here, two methods have been compared for attaining a real time experience and to produce virtual experiences as elapse time in learning process is reduced. However, this two modelling is appropriate for knowledge sharing. This analysis is inspired with knowledge sharing concept in multi agent based systems where agents has the competency to generate global modelling from scattering these models provided by individual agents. Subsequently, it may increase accuracy modelling; therefore it may offer valid experience for learning at earlier learning stage. To reduce make span process, anticipated model uses cost, reward and action techniques to grafting workflow scheduling need and resourceful experience from experienced system indeed of merging entire model. Simulation outcomes depict that anticipated scheduling model can acquire sample learning and efficiency model based acceleration in Multi-agent application objectives. Here, MATLAB environment is used for simulation. Metrics like cost, Make span is evaluated for agricultural dataset. Comparison is done with anticipated Dense mobile Network and Deep Q Network. Here, DMN shows better trade off than DQN model and more appropriate for agricultural dataset.***

Keywords: Multi-agent, Mobile computing, agriculture, Dense Mobile network, Deep Q network, work flow scheduling, global modelling, Agent based learning

I. INTRODUCTION

Mobile computing is considered to be a growing high performance computing environment with heterogeneous, large scale accumulation of autonomous system and appropriate computational framework [1]. It offers technologies and tools to construct computational intensive or data for parallel applications with huge amount of affordable cost in contrary to conventional parallel computational approaches [2]. Therefore, there is some enormously growing number of active investigations in mobile computing such as energy management, placement, scheduling, security, privacy and policy and so on [3].

In Mobile computing environment, workflow scheduling is measured to acquire higher attentions to provide wider applications in both economic and scientific fields [4]. Workflow is generally designed as directed acyclic graph model with numerous tasks that has to fulfil certain predominant constraints. Workflow scheduling in mobile is provided as matching tasks over associated computational resources, that is, virtual machines connected to cloud resources [5]. In Multi-objective based scheduling, certain objectives are determined to be more conflicting. For instance, minimizing execution time, quicker VM are more appropriate than those of slower machines. Moreover, faster VMs are generally most costly and therefore minimized execution time is contradictory for cost reduction based objective [6]. It is extensively acknowledged that scheduling multi-task workflow over distributed platforms are NP hard constraints. Therefore, it is widely time consuming process to acquire schedules via traversal based approaches [7]. More specifically, Meta heuristic and heuristic approaches with polynomial difficulties are competent to offer near or approximation scheduling outcomes as acceptable optimality loss based cost. Finest instances of these algorithms are extremely multi-objective swarm optimization approach and non-dominance based genetic algorithm for sorting [8]. Even though, these approaches offer satisfactory solutions, they need huge amount of human interventions and prior expert's knowledge, generally in encoding strategies [9]. It is observes that diverse functionality over game theoretic approaches are extremely competent of handling workflow based scheduling model in mobile based environments [10].

In recent times, novel machine learning based approaches are considered to more powerful and versatile, it is considered to be research oriented and efforts are provided based on reinforcement learning (RL) and Q based learning model [11]. This is to determine near optimal workflow based scheduling model with service level agreement limitations [12]. Even though, there are numerous Multi-agent based reinforcement learning approaches and techniques for decentralized network model, multi-robotic control, electronic auctions, distributed load balancing and traffic control crisis Multi agent based reinforcement learning approaches are still under investigation [13].

Manuscript received on February 10, 2020.
Revised Manuscript received on February 20, 2020.
Manuscript published on March 30, 2020.
* Correspondence Author
N. Priyadharshini*, Assistant Professor, Department of Computer Science at Sree Saraswathi Thyagaraja College, Pollachi, Tamil Nadu. Email: priya.samy@gmail.com
Dr. V. Narayani, Assistant Professor, Department of Computer Science at St. Xavier's College, Palayamkotai, Tamil Nadu. Email: narayaniv79@rediffmail.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

With the above mentioned observations, this work formulates a scheduling crisis in a Multi-criteria based interaction and discrete event driven model and compares two algorithms with reinforcement learning Known as Deep Q Network model and dense Mobile Network model for fulfilling Multi-objective workflow scheduling that aims at optimization of both workflow based cost and time [14]. DQN and DMN models are trained with Multi-agent reinforcement learning circumstances and provide data from legacy system as heuristic model [15]. Here, agricultural based dataset is considered for computation. Data extraction, accumulation are performed with this agricultural dataset. Here, the anticipated agents monitor all actions of other agents and reward and select it with joint distribution with environmental updation.

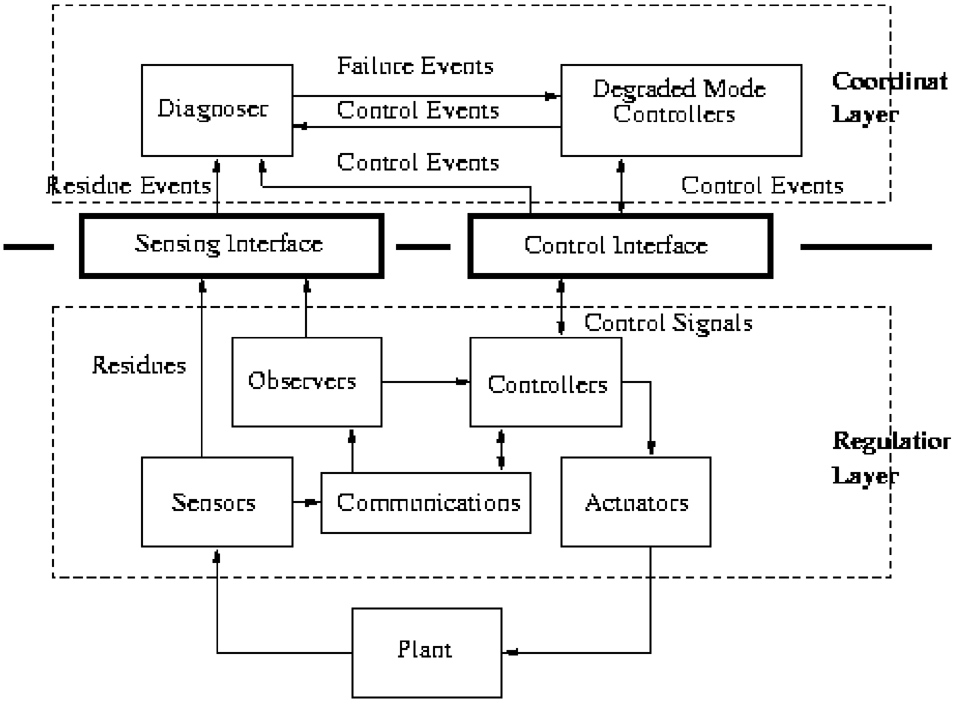

Fig 1: Provisioning mobile agent for real time applications

The anticipated model are computed with subsequent strengths: 1) Mobile agents are trained with workflow scheduling with diverse kinds of processing approach and heterogeneous VMs with diverse resource configurations; 2) planning of destination scheduling can be attained devoid of prior experts knowledge and human interventions. Here, a comparative analysis has been performed among DQN and DMN model using MATLAB simulation environment for examining multiple scientific workflow over simulation environment and to preserve those agricultural data over cloud. Experimental outcomes clearly recommend that DMN model outperforms existing DQN model in terms of cost optimization and with Make span computation.

Finally, Dense M Network model based Mobile agent reinforcement learning framework is considered for assisting workflow scheduling. Mobile agent based workflow over mobile environment is optimized and observed with Multi-flow completion time and computed subsequently. Fig 1 depicts agent based model for application in real time scenario. However, weight based item set is needed for scheduling workflow of mobile application and to compute input state of agriculture based application. Frequent item set is measured as an element of correlation function among makespan computation devoid of expert's knowledge and merges its standard with dynamic real time environment.

The essential contribution of this work is provided as: Section II depicts related works and background study, Section III compared DQN and DMN model for workflow scheduling. Section IV explains numerical computation of anticipated model where DMN performs well. Finally section V shows conclusion with future works.

II. RELATED WORKS

In this section, a brief mobile agent based overview and prevailing agent based model for distributed rule mining has been discussed. Generally, software agents specify intelligent program that carry out certain functions with respect to user. These agents are competent with mobility property termed as Mobile agents [16]. MA is considered as an autonomous transportable program that may migrate with host or own control from one node to another in heterogeneous network to carry out certain task. Subsequently, functions running at host may postpone arbitrary point based execution, transfer to another host or host request to move to other destination and re-generate execution from suspension point [17]. Once if agents are launched, it may continue its function when user is disconnected from other network. MA can move only from one host to another however it spawns to other new agents; it communicated with other stationary agents and recognizes resources/services [18]. Agents may assist and improve knowledge discovery procedure in numerous methods.

For example, agents may provide data extraction, selection, pre-processing and integration and they offer superior choice for P2P computation, multi-source mining and distribution. Agents shows finest match for offering interactive mining, DM based service delivery, human centered DM and customer services. For deploying various investigations, MA can mine association rules for distributed sites. Certain essential contribution of these agents is provided below:

In [19], author anticipated distributed approach for association rule mining with Apirori procedure and MA technology. To enhance effectual functionalities when determining frequent tem sets. In [20], author depicted IDMA framework which facilitates increased association rule mining and mobile agent based distribution for heterogeneous and distributed database system. This kind of system may provide knowledge based discovery sub-system and distributed knowledge discovery based management system. This system may dispatch mobile agent related to every site. Mobile agents may migrate from one system to other and perform data mining functions. Some local item set is acquired with local association rules which may be attained and local knowledge are refreshed. Some large local item set and related counts may provide distributed mining with mobile agents. When every mobile agent comes from this mining system, probable maximum support counts and minimal counts of potential item set can be acquired. Some system was implemented with IBM agents. In [21], author recognized association rules with distributed database with intelligent agents and uses loosely coupled incremental model for effectual distribution of association based rule mining.

In [22], author anticipated an extensible agent based enriched data mining system execute with agent based development environment termed as Extendible Multi agent based mining system. System operations are illustrated and depicted based on two knowledge discovery data based scenario: a) Meta association based rule mining and classifier generation. Meta mining is depicted as combination process that may individually acquire outcomes with data mining based application activities. Every local frequent item set trees are merged and collected with global frequent item set. In [23], author executes data partition approach for distributed and parallel ARM. Some system may distribute data between agents in horizontal and vertical partition and utilizes Apriori procedure to mine association rules. The ultimate objective is to provide that MADM model is competent of providing advantages over parallel computing. In [24], author depicted that bit table based multi agent association rule procedure merges multi-agent approaches and association rules with bit table data structures to reduce time essential for performing counted processes and candidate generation.

Central security agency also enables legal certificate to each mobile agent before launching and when agent moves to node in its authenticity itinerary of those certificate is validated again, therefore no malicious agent may influence local node [25]. There exists five agents in this work, three are multi-agent and other two are intellectual stationary agents to carry out diverse tasks. Mobile agents are considered to be more frequent item set, total frequent itemset, local knowledge based generator agent, collector agent. These kinds of agents preserves dynamic itinerary, whenever needed. This may be revised at any time and in any node in itinerary models.

III. PROPOSED METHODOLOGY

This section provides comparison of two diverse agent modelling known as Dense M architecture and DQM model. With the observation generated from the proposed model provided in previous research work this comparative study has been done. Based on the previous investigation, Dense M Network architecture is our research work. Here, agricultural dataset is considered for Multi-agent based scheduling and association mining. Data can be extracted, processed and stored for computation. Metrics like cost, make span are computed for determining the system performance.

A. Dense M Network architecture

This section explains in detail about the proposed Dense M Network architecture (DMN). DMN architecture is a mobile agent based approach which is an upgraded version of reinforcement learning architecture that comprises internal world model and policy learning. M-agent model utilizes comprises of four states: current state (C_s), action (a_s) as input, next state (C_{s+1}) and rewards (r_{s+1}) as outputs. Learning agent uses real time experience of gathering data from various environments to construct virtual model and update its corresponding value function. This approach is termed as direct learning model as in Fig 2. After approximation of virtual model over the environment, it should replicate virtual experiences to attain additional policy learning, termed as indirect planning or RL. DMN architecture co-ordinates reinforcement learning algorithm to generate DMN framework comprising action, planning, model learning and direct RL. Direct RL is one-step learning process, where model learning is executed to validate experiences $(C_s, a_s, C_{s+1}, r_{s+1})$. Mobile agent randomly chooses state-action pairs as inputs and foresees next state and offers reward as virtual experiences.

The anticipated model is based on Dense M Mobile network architecture however utilizes tree structure for modelling agent based environment. Tree model is to produce virtual experiences to increase added iteration values. Henceforth, the anticipated agent comprises of two inter-leaving process, indirect learning and direct learning, that is, planning. In direct learning process, mobile agent observes environmental information termed continuous states (C_{s+1}) and chooses action (a_s). Then, agent transits to subsequent continuous state (C_{s+1}) and attains (r_{s+1}) reward. Action and continuous state pairs (C_{s+1}, a_s) are inserted to set I_s.

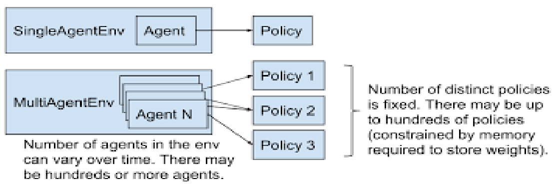

Fig 2: Multi agent group policies

The dense Mobile agent aggregates concurrent states, C_{s+1}, C_{s+2}..to discrete states C_s. Thereby, uses this experience ($C_s, a_s, C_{s+1}, r_{s+1}$) to approximate mobile environment directly. Whereas in case of indirect learning as in fig 2, tree model substitutes environment and action-continuous state pairs, (C_{s+1}, a_s) are attained from set I_s.

When retrieved action-continuous state pairs are provided to tree model, variation in subsequent continuous states and rewards has been identified. Next simultaneous continuous states are attained from $i_{-s+1} = i_{-s} + \Delta i_a$ and virtual experience (i_s, a_s, i_{s+1}, r_s) to speed up learning process. Fig. 1b depicts state space as four states. Difference action in every region will offer difference variation of continuous states, rewards r_s and i_a.

This work anticipated three diverse evolution types of DMN model for sharing data using multi-agent system based on heterogeneous tree structure. The anticipated structure is provided in Fig. 1b. DMN comprises of homogeneous multi-agent with mobile network models implemented with decision tress. Agents possess two models while in learning process: sharing mode and learning model. In former, agents share information with one another from model, whereas in latter, agents understands policy and construct its own model for the planning purpose (indirect learning).

Model learning approaches from RL have been illustrated very often. In [26-27], mobile agents learns policy in stochastic environment, therefore this model learning should consider transition probability amongst two states after an action is taken. This model considers variation amongst current and subsequent continuous states after performing action. Three diverse data sharing approaches are modelled to carry out planning in Dense M network based on diverse mobile communication protocols and its corresponding procedures.

B. Mobile Agents' state action

Alike of communication between other networking model, mobile agent perceives in continuous state and carry out action process. This is referred as continuous state-action pairs. If agent carries out pairing, it introduces huge error for temporal difference learning, and then it transmits state action pair to other agents. When subsequent agent receives that action pair, it categorizes it over leaf node in its own tree, it verifies whether there is any sufficient knowledge about area related to fitted leaf. If it is not done, agent tags that data as unknown data and transmits a request to other agents for help. This mode of communication is depicted in Fig. 3. Unknown area must fulfil one condition given below in Eq. (1) and Eq. (2):

$$N_{total} = \sum_{j=1}^{C_s} N_j^a < cN \quad (1)$$

Continuous state $C_s = (x_1, x_2, \dots, x_n)$

$$\bigcup_{j=1}^{C_s}\left(\left\{x_i | x_i \in [\mu^a_{i,j} - \sqrt[n]{\left(\sigma^a_{i,j}\right)^2}, \mu^a_{i,j} + \sqrt[k]{\left(\sigma^a_{i,j}\right)^2}], i = 1,2,\dots,n\right\}\right) \quad (2)$$

In Eq. (1), 'C' is considered as constant, 'N'' is threshold, that is, sum of samples in leaf node has to be fulfilled. Eq. (2) is confidence ranges, where $\mu^a_{i,j}$ and $\sigma^a_{i,j}$ is mean and variance of continuous state in dimension 'i'. This mean and variance computation is specifically for evaluation of sum of continuous states and square sums of continuous states of every cluster in leaf nodes. If agent is not under confidence range of mobile clusters, then Eq. (2) is satisfied.

Mobile agent then broadcasts request generated from continuous state-action pair of other mobile agents. Then, agent which receives this state-action pair request and returns information of leaf nodes whose area is overlapped by other pair.

Algorithm 1: DMN for scheduling

Input: Parameter selection

Output: Next state, stationary policies, reward

1. Initialization of memory for performing agent functionality with random weight
2. Initialize action pair and state
3. Sense initial state
4. While S ≠ max. Data scheduling
   If probability ∈ then
   Choose random action of agent
   Else
   Select $a \in f$;
5. Perform action pairing
6. Sense reward of present state and subsequent state (next state)
7. Save state transition of state and subsequent states
8. Perform random probability transition from memory;
9. Compute target scheduling with other agents
10. Determine correlated equilibrium for every transaction
11. If s' is termination condition then
    $t = r$;
    Else
    $t = r + \gamma \max_a' Q\ (a', s')$
12. Train Dense M network with agent based schedule mapping to other agents
13. Return State value, action pair and reward
14. End

C. Agent Mapping For Frequent Itemset

Consider 'D' be an undetermined dataset which has to be composed and analyzed with set of transactions known as $D = \{T_1, T_2, \dots, T_n\}$, where $'n'$ is depicted as sum of transactions in uncertain dataset environment. Dataset 'D' is provided with finite set of distinct items $I = \{I_1, I_2, \dots, I_m\}$. For performing transactions with mobile agent, determine $T_q \in D, q \in (1,2, \dots, n)$ which subset of items. Here, T_q is termed as transaction identifier in mobile agent.

Retrieval Number: F8048038620/2020©BEIESP
DOI:10.35940/ijrte.F8048.038620
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

While performing probability computation it dataset, uncertain frequent pattern mining is formulated by existential probability $p\ (I_j, T_q)$. From this, I_j specifies item that exists in T_q with probability$p\ (I_j, T_q)$.

This probability computation is initiated with positive values '0' to '1', that is, I_j has low probability to present in T_q to I_j is surely available withinT_q. So as to schedule actions to agents, items in dataset has to be included in weighted table.

It is provided as $w_{tab} = \{w(I_1), w(I_2), \dots, w(I_n)\}$, where $w(I_j) \epsilon\ (0,1), j\ \epsilon\ (1,2, \dots, m)\}$ is weight of item I_j. It item set comprises of distinct items, then I→ item set. It is known that item set is available in T_q. Minimum supported threshold is provided as $\varepsilon\ \epsilon\ (0,1)$.

Algorithm 2: DMN for mining frequent item set

Input: Agent based transactional dataset 'D', weighted table, min expected weighted support threshold

Output: Sum of frequent weighted item set

1. Initialization
2. Initialize parameters and variables
3. Scan dataset to acquire weighted frequent item set
4. For all item I_j in 'D' do
5. Compute expected weighted support threshold
6. If $expweight\_sup(I_j) \geq |D| * \varepsilon$ then
7. $Expweight\ support\ threshold_i = expweightsupport\ threshold_i \cup \{I_j\}$ yes
8. End if
9. End for
10. $Expweightsupport\ threshold = Expweightsupport\ threshold\ U\ Expweightsupport\ thr$
 Scan 'D' and attain weighted frequent item set
11. Perform weight in descending order
12. While weight ≠ null do
13. Weight = connection (weight, support)
14. For all candidate in item set in weighted item set do
15. Scan 'D' and compute expected weight support (X)
16. If $expweightsupport\ (X) \geq |D| * \varepsilon$ then
17. Weighted item set = weighted item set U {X}
18. End if
19. End for
20. Weighted item set = weighted item set U weighted item set
21. End while
22. Return weighted item set

When a mobile user needs any specific kind of service, the appropriate user has to broadcast that service request to the corresponding mobile agent with respect to user. Followed by this, when the mobile agent receives request, it verifies service providers to handle whether it can validate the recommendation. Then, it recommends the request to the most suitable VM over the machine. If agent does not find appropriate recommendation to newly arrived item set, then agent forwards the request to other available agents. When user determines the deadline, it validates all recommendation and chooses finest broker and refuses other request. The selected broker then provides the service. Selected broker then verifies collaborative brokers or service providers for validation and then transmits feedback to users. However, direct transaction channel amongst providers and users has to be set up. Finally, providers and users should be involved in transaction and send evaluation to broker.

D. DQN Modelling

Here, DQN is a most proficient modelling in reinforcement learning approach. It assists in activating value function Q^* related to optimal policy using loss minimization as in Eq. (3) and Eq. (4):

$$L(\theta) = E_{s,a,r,s'}[Q^*(s, a|\theta) - y)^2] \quad (3)$$

$$y = r + \delta \max a'\ Q^*\ (s', a') \quad (4)$$

Where y is depicted as Q target function whose factors are updated periodically with θ which assists in stabilizing learning model. Subsequent crucial stabilizing component of DQN while experiencing replaying buffer D comprising (s, a, r, s') tuples. Agents may determine actions with neural network model and integration network output with random actions for training set samples. Generally, agents may train network that will recognize weighted, cumulative rewards for every actions.

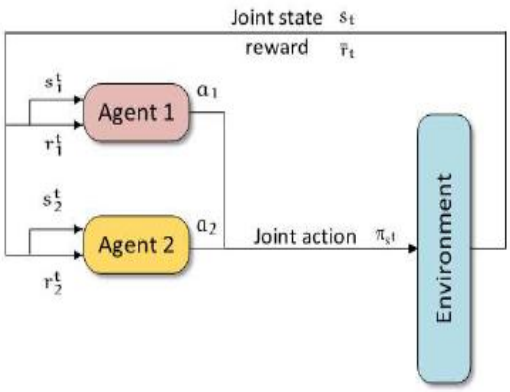

Fig 3: Agent based DQN scheduling framework

DQN based agents optimal policy not only communicates with scheduling environment directly as in fig 3, however agent's policy is not connected to it. Iterative model for evaluating global equilibrium policies depend on local Q-value updation, policies of all state. Q-values are provided with 't' time for all $i\ \in I, s\ \in S$, for $a\ \in A\ (s)$, known as $Q_i^t(s, a)$. To attain correlated equilibrium, DQN agents functions as correlated equilibrium scheme π^t, where $\pi_s^{t+1}\ \epsilon\ f\ (Q^{t+1}\ (s))$. DQN based algorithm is discussed below. With appropriate reward methods modelled with convergence of DQN based procedures in Multi-agent setting that are fulfilled. For evaluating make span agent, reward strategy is formulated as in Eq. (5):

$$R_1 = \left[\frac{ET_{k,i,j}(a) - (makespan' - makespan)}{ET_{k,i,j}\ (a)}\right] \quad (5)$$

Where, cost reward is modelled as in Eq. (6):

$$R_2 = \left[\frac{worst - ET_{k,i,j}\ (a) * P_j}{worst - best}\right] \quad (5)$$

Types	CPU	Memory
Medium	2	8
Large	2	8
Large	4	16
Large	4	16
Large	8	32
Large	8	32
Large	8	32

Where (5) depicts low make span maximization which is preferably more, i.e. worst and best specifies computational complexity measurements, P_j is probability of computing 'j' variable. As well, in (6) specifies lower cost maximization more appropriately. Fig (2), specifies convergence of DQN with cost and make span computation.

Algorithm 3: DQN

Input : Parameter initialization

Output: Q-value, rewards $'r'$ and stationary policies π^*

1. Initialize action function $'Q'$, replay function $'D'$
2. Initializing action profiles with a_p, state $'s'$
3. Determine state initialization with $'s'$
4. While max $\neq$ iteration do
5. If probability is determined them
6. Choose random action movement $'a'$
7. Else
8. Choose $a \in f$
9. Perform action function
10. Compute periodic rewards and subsequent state $'s'$
11. Store $< s, a, r, s' >$ in relay function D;
12. Iterate transition randomly from replay function $< ss, aa, rr, ss' >$
13. Compute target transition
14. If ss' at terminal state then
15. $rr = tt;$
16. $else$
17. $tt = rr + \gamma \max a' \, Q\,(ss', aa')$
18. Train dense Q network with $\left(tt - Q\,(ss, aa)\right)^2$ as loss function
19. $s = s'$
20. Return Q value, reward and action profile

IV. NUMERICAL RESULTS AND DISCUSSIONS

For validating this model, experimentation was performed with MATLAB simulation environment for determining workflow and to compute task execution scenario. This work considers agricultural based dataset for computation with DQN and DMN. Data from the above mentioned dataset is considered for validation among mobile agents. Data are being extracted from agricultural dataset and task is being scheduled by multi-agents to analyze the performance of agents. DMN model shows better trade off in contrary to prevailing approaches. DMN handles higher task scheduling for validation of workflow. Table II depicts sample attributes of agricultural dataset used for computation. Memory, CPU utilization and type of workflow have to be considered for comparing these two models, i.e. DQN an DMN. Location, crop, land, time, id are some of the attributes of agricultural dataset.

Table I: Scheduling type with Memory usage

From Table I, it is depicted that scheduling types with task associated with CPU and memory is examined. Here, two types are considered for validation with agricultural dataset, i.e. large and medium, here total CPU usage varies from 2, 4, 8 for Multi-agent computation. Here, storage is varies from 8 GB, 16 GB and 32 GB. Here, DMN and DQN methods are compared for validation. The functionality of DMN is considered to be higher in contrary to DQN model. Workflow scheduling with DMN model is higher than DQN and computation time is also lesser for performed all provided process as in Fig 4-Fig 7. The execution is carried out more parallel. Fig 8 depicts comparison of cost with make span computation of DQN and DMN model based on scheduling plan. Two metrics are considered for this approach, cost and make span computation, where DMN shows lesser cost compared to that of DQN. Make span computation clearly validates that the anticipated model is cheaper with 2.766% than DQN. There are some basic differences between proposed and the baseline model with total cost and size.

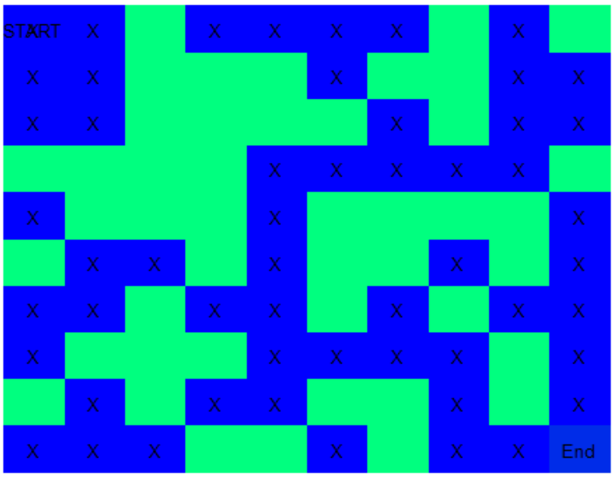

Fig 4: Data extraction

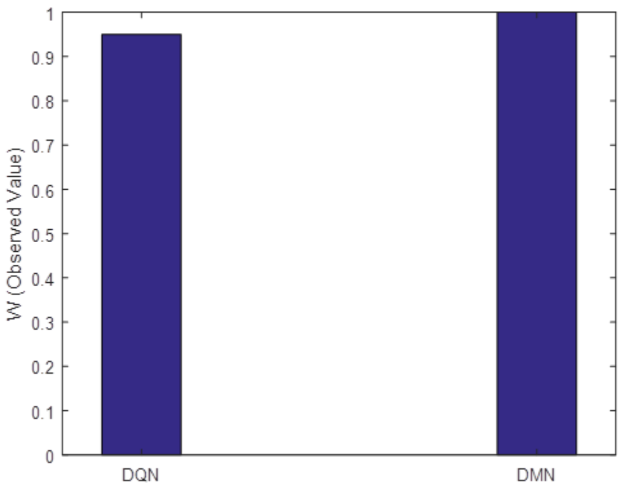

Fig 5: Task scheduling based DQN and DMN observations

Retrieval Number: F8048038620/2020©BEIESP
DOI:10.35940/ijrte.F8048.038620
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

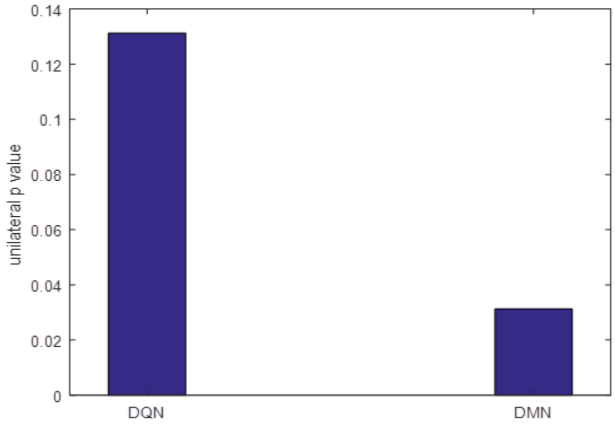

Fig 6: P value computation

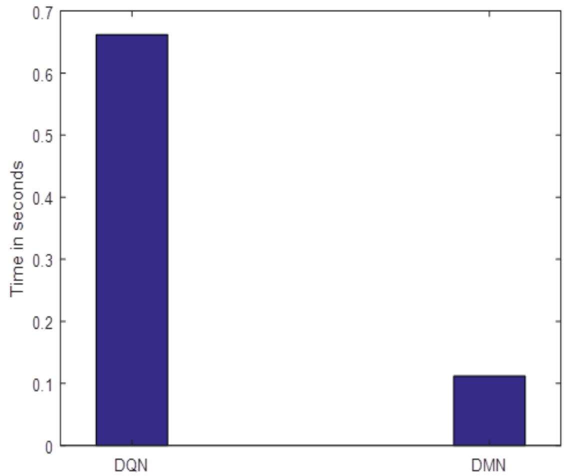

Fig 7: Scheduling time of Mobile agent

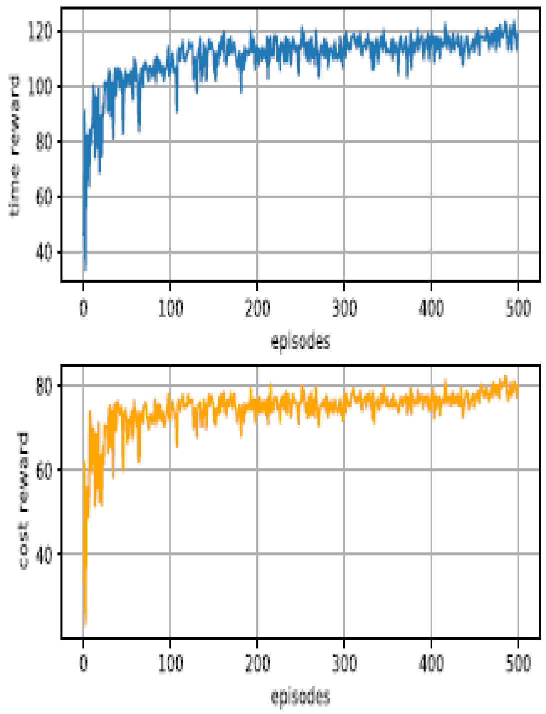

Fig 8: Time and cost reward of proposed model

Fig 3 depicts scheduling outcomes based on frequent mining to mobile agents. This scheduling is explicitly of smaller amount of total tasks considered from workflow. The anticipated DMN model is compared with existing Dyna Q network architecture model. The anticipated model outperforms the prevailing baseline approach in terms of make span. Subsequently, the major benefits of anticipated model is to acquire from the fact that the algorithm does not considers inter task dwelling time and concurrency to major exploitation of parallelism offered while mining process. In contrary, prevailing algorithm intended to trail out topological constraint of workflow initially and uncertain to carry out potential parallelism.

From fig 6 and fig 7, it is seen that the anticipated frequent item set algorithm can generate same amount of item sets alike of Multi agent PSO algorithm with provided test datasets. Moreover, number of k- item sets produced by anticipated DQN algorithm and PSO algorithm is smaller than number of k-item set depicted by DQN framework as in Fig 8 and Fig 9. This is owing to both probability and weight properties that are measured in DMN algorithm. The anticipated DMN framework holds meaningful and lesser k-item sets with prevailing approaches as in Fig 10- Fig.13. In addition, with same dataset, k-item sets offered by prevailing approaches have similar distribution trend.

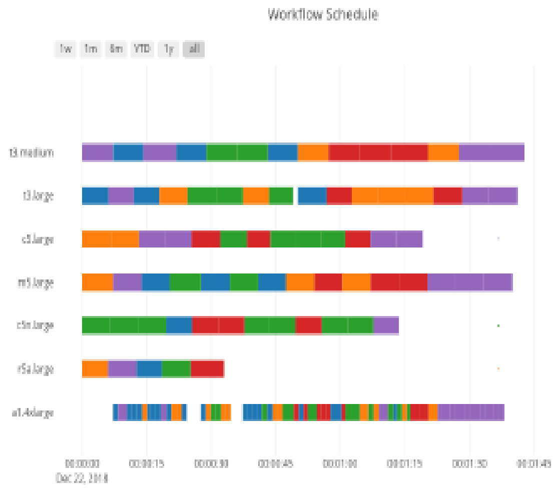

Fig 9: Workflow scheduling w.r.t agricultural dataset

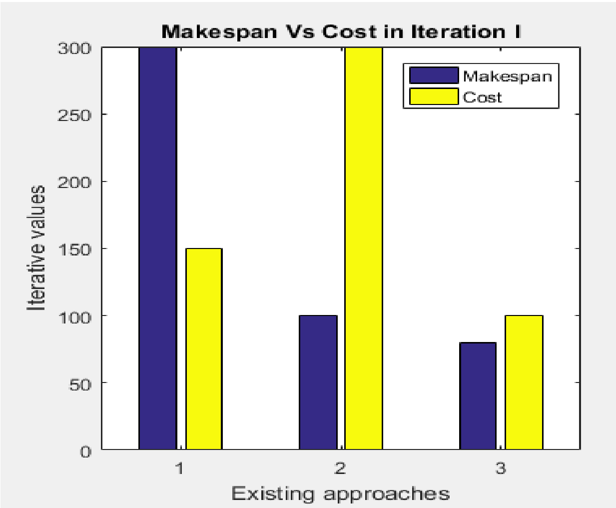

Fig 10: Makespan Vs cost computation in Iteration I

Retrieval Number: F8048038620/2020©BEIESP
DOI:10.35940/ijrte.F8048.038620
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

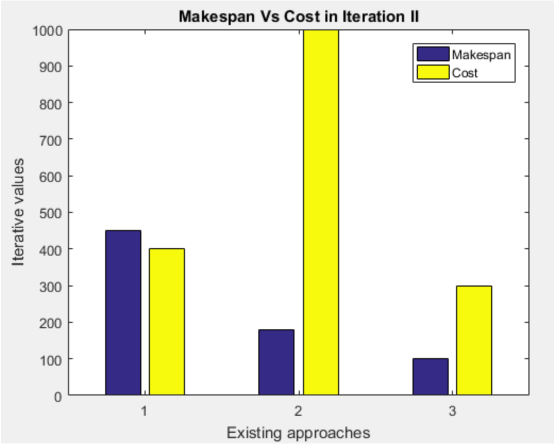

Fig 11: Makespan Vs cost computation in Iteration II

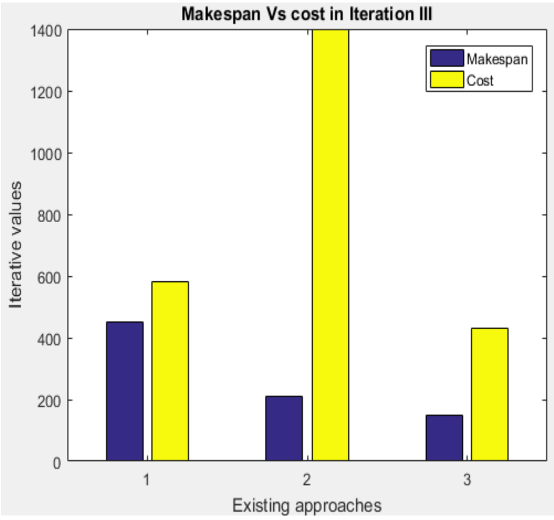

Fig 12: Makespan Vs cost computation in Iteration III

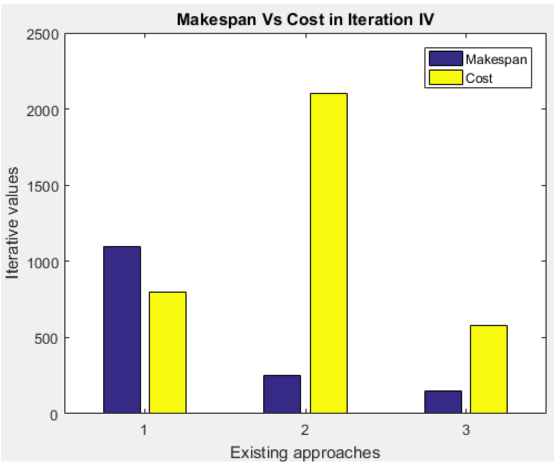

Fig 13: Makespan Vs cost computation in Iteration IV

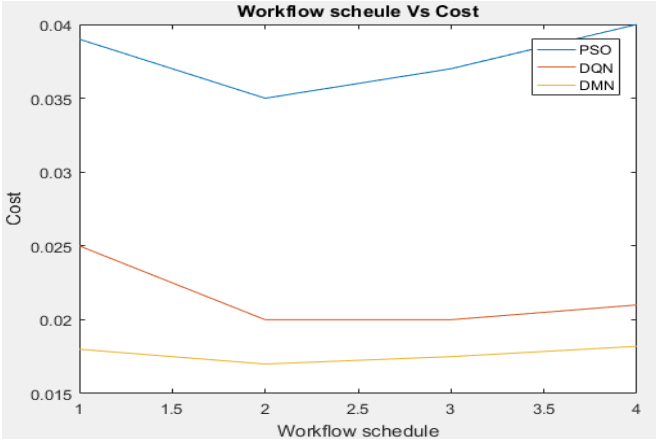

Fig 14: Workflow scheduling Vs Cost computation

Note: 1- PSO, 2- DQN, 3- DMN

V. CONCLUSION

Here, both DQN and DMN methods are used for investigation. The comparative analysis of these models is carried out with agricultural dataset for validating efficiency of these approaches.

Reinforcement based learning approach is used for both the model from real time applications and planning with certain virtual experiments. Planning and learning with DMN model is integrated with reinforcement as they sense and share data in similar methods, indeed of these experimentation. Subsequently, acting, planning, and learning model and direct learning are performed simultaneously. For execution in serial computing, some process run sequentially with time step and need lesser computational time and takes lesser amount of time. Excess time slots may be provided for planning, indirect learning that shows intensive computation inherently.With agricultural samples, agent models commence to work with empty state and filled with some experimentation. However, these models with certain sparse information are considered to be more inappropriate, specifically in case of extensive environment. In some cases, restricted amount of samples are visited in no way. When model is seems to be more inappropriate, planning procedure is so massive. Henceforth, knowledge sharing method is anticipated to provide model learning procedure that may spent most time in data communication, workflow scheduling and so on. However, consumes lesser computational cost than planning process.

Here, two works, that is, DMN and DQN are provided with successive enhancement in sharing approaches that are applied to multi-agent model. In anticipated model, agents communicate with their own local experiences to assist partners to provide more global circumstances. Using these model for learning purpose, cost and Make span is reduced such that learning with real dataset is more probable. To depict efficiency of anticipated model, simulation is carried out in MATLAB environment. As an outcome, learning time is diminished and effectual memory consumption is attained. Based on this comparative study, it seems that DMN model works more efficiently than DQN model while performing scheduling and mining frequent item sets.

Table II: Samples of agricultural dataset (online resources)

Project	ID_sample location	additional_ID	Crop	plant_analysis_ID	Date_collected	pre/post	Irrigated
Bucks	1	BD	rye	Pre_kill BBD1	12/17/2015	Pre	Dryland
Bucks	2	BD	rye	Pre_kill BBD1	12/17/2015	Pre	Dryland
Bucks	3	BD	rye	Pre_kill BBD1	12/17/2015	Pre	Dryland
Bucks	4	BD	rye	Pre_kill BBD2	12/17/2015	Pre	Dryland
Bucks	5	BD	rye	Pre_kill BBD2	12/17/2015	Pre	Dryland
Bucks	6	BD	rye	Pre_kill BBD2	12/17/2015	Pre	Dryland
Bucks	7	BD	rye	Pre_kill BBD3	12/17/2015	Pre	Dryland
Bucks	8	BD	rye	Pre_kill BBD3	12/17/2015	Pre	Dryland
Bucks	2	BD	rye	BBD2	########	Post	Dryland
Bucks	6	BD	rye	BBD6	########	Post	Dryland
Bucks	7	BD	rye	BBD7	########	Post	Dryland
Bucks	1	BD	rye	BBD1	########	Post	Dryland
Bucks	3	BD	rye	BBD3	########	Post	Dryland
Bucks	5	BD	rye	BBD5	########	Post	Dryland
Bucks	4	BD	rye	BBD4	########	Post	Dryland
Bucks	1	BBI	rye	BBI1	########	Post	irrigated
Bucks	2	BBI	rye	BBI1	########	Post	irrigated
Bucks	3	BBI	rye	BBI2	########	Post	irrigated
Bucks	4	BBI	rye	BBI2	########	Post	irrigated

REFERENCES

1. H. Chen, Q. Shi, R. Tan, H. V. Poor, and K. Sezaki, “Mobile element assisted cooperative localization for wireless sensor networks with obstacles,” IEEE Trans. Wireless Commun., vol. 9, no. 3, pp. 956–963, Mar. 2010.
2. S. Zhu and Z. Ding, “Distributed cooperative localization of wireless sensor networks with convex hull constraint,” IEEE Trans. Wireless Commun., vol. 10, no. 7, pp. 2150–2161, Jul. 2011.
3. Y. Ding, N. Krislock, J. Qian, and H. Wolkowicz, “Sensor network localization, Euclidean distance matrix completions, and graph realization,” Optim. Eng., vol. 11, no. 1, pp. 45–66, Feb. 2010.
4. Y. Dieudonné, O. Labbani-Igbida, and F. Petit, “Deterministic robotnetwork localization is hard,” IEEE Trans. Robot., vol. 26, no. 2, pp. 331– 339, Apr. 2010.
5. M. Z. Win et al., “Network localization and navigation via cooperation,” IEEE Commun. Mag., vol. 49, no. 5, pp. 56–62, May 2011.
6. Y. Xia, M. Zhou, X. Luo, S. Pang, and Q. Zhu, “A stochastic approach to analysis of energy-aware dvs-enabled cloud datacenters,” IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, pp. 73–83, Jan 2015.
7. J. Yu, Z. Kuang, B. Zhang, W. Zhang, D. Lin, and J. Fan, “Leveraging content sensitiveness and user trustworthiness to recommend fine-grained privacy settings for social image sharing,” IEEE Trans. Information Forensics and Security, vol. 13, no. 5, pp. 1317–1332, 2018.
8. A. Choudhary, I. Gupta, V. Singh, and P. K. Jana, “A GSA based hybrid algorithm for bi-objective workflow scheduling in cloud computing,” Future Generation Computer Systems, vol. 83, pp. 14–26, 2018.
9. W. Li, Y. Xia, M. Zhou, X. Sun, and Q. Zhu, “Fluctuation-aware and predictive workflow scheduling in cost-effective infrastructure-as-a-service clouds,” IEEE Access, vol. 6, pp. 61488–61502, 2018.
10. R. Xu, Y.Wang, H. Luo, F.Wang, Y. Xie, X. Liu, and Y. Yang, “A sufficient and necessary temporal violation handling point selection strategy in cloud workflow,” Future Generation Comp. Syst., vol. 86, pp. 464–479, 2018.
11. E. Iranpour and S. Sharifian, “A distributed load balancing and admission control algorithm based on Fuzzy type-2 and Game theory for large-scale SaaS cloud architectures,” Future Generation Computer Systems, vol. 86, pp. 81–98, sep 2018.
12. L. Wu and Y. Wang, “Scheduling Multi-Workflows Over Heterogeneous Virtual Machines With a Multi-Stage Dynamic Game-Theoretic Approach,” International Journal of Web Services Research, vol. 15, pp. 82–96, oct 2018.
13. Y. Wei, D. Kudenko, S. Liu, L. Pan, L. Wu, and X. Meng, “A Reinforcement Learning BasedWorkflow Application Scheduling Approach in Dynamic Cloud Environment,” pp. 120–131, Springer, Cham, dec 2018
14. B.Waschneck, A. Reichstaller, L. Belzner, T. Altenmüller, T. Bauernhansl, A. Knapp, and A. Kyek, “Optimization of global production scheduling with deep reinforcement learning,” Procedia CIRP, vol. 72, pp. 1264–1269, jan 2018.
15. L. Zhang, K. Li, C. Li, and K. Li, “Bi-objective workflow scheduling of the energy consumption and reliability in heterogeneous computing systems,” Information Sciences, vol. 379, pp. 241–256, feb 2017.
16. Y. J. Chen, K. S. Hwang, and W. C. Jiang, “Policy sharing between multiple mobile robots using decision trees,” Inf. Sci., vol. 234, no. 10, pp. 112–120, 2013.
17. K. Ito, A. Gofuku, Y. Imoto, and M. Takeshita, “ A study of reinforcement learning with knowledge sharing for distributed autonomous system,” in Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom., 2003, pp. 1120–1125
18. M. Santos, H. J. A. Martín, V. López, and G. Botella, “Dyna-H: A heuristic planning reinforcement learning algorithm applied to role-playing game strategy decision systems,” Knowl.-Based Syst., vol. 32, pp. 28–36, Aug. 2012
19. M. Asadpour, M. N. Ahmadabadi, and R. Siegwart, “Heterogeneous and hierarchical cooperative learning via combining decision trees,” in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., Beijing, China, 2006, pp. 2684–2690.
20. B. G. Amidan, T. A. Ferryman, and S. K. Cooley, “Data outlier detection using the Chebyshev theorem,” in Proc. IEEE Aerosp. Conf., Big Sky, MT, USA, 2005, pp. 3814–3819.

21. A. Konar, I. G. Chakraborty, S. J. Singh, L. C. Jain, and A. K. Nagar, "A deterministic improved Q-learning for path planning of a mobile robot," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 43, no. 5, pp. 1141–1152, Sep. 2013
22. T. Hester and P. Stone, "Learning and using models," in Reinforcement Learning: State of the Art, M. Wiering and M. van Otterlo, Eds. Berlin, Germany: Springer Verlag, 2011
23. L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," IEEE Trans. Syst., Man, Cybern. C, Appl. Rev., vol. 38, no. 2, pp. 156–172, Mar. 2008.
24. B. N. Araabi, S. Mastoureshgh, and M. N. Ahmadabadi, "A study on expertise of agents and its effects on cooperative Q-learning," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 37, no. 2, pp. 398–409, Apr. 2007.
25. M. N. Ahmadabadi and M. Asadpour, "Expertness based cooperative Q-learning," IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 32, no. 1, pp. 66–76, Feb. 2002.

AUTHORS PROFILE

Ms. N. Priyadharshini, she holds a Distinction First Class Master's degree in Computer Applications (2008) and a First Class Under Graduation degree in Mathematics (2005). She has 9 years of teaching experience. She has been awarded with Silver Medal by Infosys Technologies under Inspire Faculty partnership. She has published many research papers in International and National Journals. She had presented papers in National and International Conferences. She is a reviewer in an UGC referred journal titled, "JETIR". She is a life time member of Indian Society for Technical Education (ISTE).

Dr. V. Narayani completed Ph.D Computer Application in March 2012, M.Phil Computer Science in 2005 and MCA in 2002. She has 12 years of Teaching experience and 5 years of Research experience. She has published more than 45 research papers in International Journals. She has presented many papers in International and National Conferences. She is a life member of Indian Society for Technical Education (ISTE), CSTA.