

An adaptive algorithm for super-peer selection considering peer's capacity in mobile peer-to-peer networks based on learning automata

Nahid Amirazodi¹ · Ali Mohammad Saghiri² · Mohammadreza Meybodi²

Received: 8 February 2016 / Accepted: 16 August 2016
© Springer Science+Business Media New York 2016

Abstract Mobile peer-to-peer (MP2P) networks refer to the peer-to-peer overlay networks superimposing above the mobile ad-hoc networks. Heterogeneity of capacity and mobility of the peers as well as inherent limitation of resources along with the wireless networks characteristics are challenges on MP2P networks. In some MP2P networks, in order to improve network performances, special peers, are called super-peers, undertake to perform network managerial tasks. Selection of super-peers, due to their influential position, requires a protocol which considers the capacity of peers. Lack of general information about the capacity of other peers, as well as peers mobility along with dynamic nature of MP2P networks are the major challenges that impose uncertainty in decision making of the super-peer management algorithms. This paper proposes an adaptive super-peer selection algorithm considering peers capacity based on learning automata in MP2P networks, called SSBLA. In the proposed algorithm, each peer is equipped with a learning automaton which is used locally in the operation of super-peer selection by that peer. It has been shown that the suggested algorithm is superior to the existing algorithms. The results of the simulation show that the

proposed algorithm can maximize capacity utilization by minimum number of super-peer and improve robustness against failures of super-peers while minimizing selection communication overhead.

Keywords Mobile peer-to-peer network · Super-peer · Learning automata

1 Introduction

Mobile Peer-to-Peer (MP2P) network refers to an overlay network superimposing above a mobile ad-hoc network. The main purposes of these networks are sharing resources (i.e., data, CPU cycles, memory, storage space and bandwidth) among mobile peers as well as coordination of the services. Consider the following scenario: we have a mobile phone with 1GB memory storage and a couple of different radios. While sitting at a restaurant or walking in a shopping mall, one can subscribe and join a local P2P network to see what other people around have offered to share in their mobile devices. The shared content could be a MP3 file or a movie, or even a game. MP2P networks have good scalability, robustness, and low-cost self-organization. Also they can admit the user's request of connection to the network at any time. However, MP2P network has additional challenges, such as: capacity heterogeneity, strong mobility of peers, inherent limitation of resources and wireless network characteristics [1–3]. The topology of the whole network can become different cause the highly dynamic property. It can also lead to lack of consistency between the overlay network and the topology of the underlying physical network. Therefore, it can result worse network performance and system instability [4].

The pure form of mobile P2P architecture can cause poor performance, because all peers irrespective of their capacities

✉ Ali Mohammad Saghiri
a_m_saghiri@aut.ac.ir

Nahid Amirazodi
n.amirazodi@gmail.com

Mohammadreza Meybodi
mmeybodi@aut.ac.ir

¹ Department of Electrical, Computer and IT Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

² Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran

act equal in all operations. Specifically, when the size of the network increases it may lead to overloading low-capacity peers [5, 6]. Super-peer topologies are developed in order to increase a further scalability and reduce the network traffic in MANET. Query response (resource discovery) in these systems is much faster than any other P2P systems and it reduces the time needed for search. The peers in the systems are classified into super-peers and ordinary-peers. Super-peers, having relatively higher capability, assume special responsibilities and manage their neighboring ordinary-peers. The network communications in the systems are done mostly among super-peers. When a pair of neighboring super-peers is too far to communicate, one or two of their ordinary-peers bridge the super-peers. All other peers are called ordinary-peers. They submit their queries to their super-peers and receive the results from them [7–9].

Super-peers have a supervising position in super-peer topologies, therefore, selecting a set of peers as super-peers require an approach that considers capacity of peers to provide the required services. Peers may be dispersed well throughout the peer-to-peer overlay network, or may be concentrated in a particular zone. The challenges of super-peer selection are: wide variety in distributions of peers characteristics, such as uptime, bandwidth, or available storage space. Such heterogeneity among peers is considered as the capacity of these networks. This problem is made more challenging cause the lack of global information about the capacity of other peers and the dynamic nature of MP2P networks [3]. Furthermore, the mobile environment raise additional challenges to super-peer selection. The topology of the super-peer layer may change with super-peer selection, because of the incompatibility or transmission/reception limitations imposed by the natural heterogeneity of super-peer networks. The newly-selected super-peer, unlike previous ones, may not be able to directly communicate with the same super-peers [9].

Therefore, to select super-peer, we need a distributed algorithm that is able to adapt with dynamic conditions. It also tries to take advantage of the heterogeneous capacities (e.g., bandwidth, processing power, etc.) of the participating peers to improve performance and reliability for the entire network [10]. In MP2P networks, peers span a wide range of mobile device with different underlay communication protocols. These peers may use WIFI(802.11) [11, 12] or WPA N(802.15.4) [12, 13] in the underlay network which leads to heterogeneity of radios of send (and receive) of peers. Each peer runs a process by which it makes relevant local observations in a distributed decision making and, on the basis of these observations, makes a judgement on which reconfiguration action is to be adopted. In an ideal condition, local observations of each peer would be entirely representative of a more global scenario, and the peers would easily obtain a final decision based on those observations. It should be noted that, because of heterogeneity of radios of send (and receive) of

Learning algorithm (LALG) for each peer p

```

01  For each peer S in under-loaded peers of p's local view
02    the RoleLA of P chooses an action
03    If capacity(S) < capacity(P)
04      If RoleLA selected action is super-peer
05        update action probability vector by equation (1)//
          rewarding
06      Else If RoleLA selected action is ordinary-peer
07        update action probability vector by equation (2)//
          penalizing
08      End If
09    Else If capacity(S) >= capacity(P)
10      If RoleLA selected action is super-peer
11        update action probability vector by equation (2)//
          penalizing
12      Else If RoleLA selected action is ordinary-peer
13        update action probability vector by equation (1)//
          rewarding
14      End If
15    End If
16  End For
17  set the last action as output of learning algorithm -> OLALG

```

Fig. 1 Learning phase for each peer p

peers supporting a connection oriented communications protocols for MP2P network may results in high collisions and also creating many control messages in the lower layers of the underlay networks.

In this paper, we propose an adaptive super-peer selection algorithm that uses learning automata as an adaptive decision-making mechanism. Learning automata [14, 15] have been found to be useful in dynamic environments with uncertain decision-making [3]. The theory of learning automata was not used in MP2P networks yet. Note that, Gholami and et al. [3] used learning automata in a super-peer selection algorithm on wire-line IP network. The results of the simulation show effectiveness of its proposed protocol in terms of capacity utilization and behavior towards super-peer failure and communication cost. We intend to improve capacity utilization,

DoTransfer (super-peer p, super-peer s)

```

Assumptions:
Rp: p's region
01  C = min (capacity(s) - load(s), load(p))
02  For C peer do
03    If P.ordinary-peer is in S.Rp
04      Transfer P.ordinary-peer from P to S
05  End for
06  If load(P) = 0
07    If load(S) < capacity(S)
08      P becomes ordinary-peer of S
09      P comedown to AOP role
10    Else P comedown to USB role
11  Else
12    R = pick highest capacity peer from P.HigherSet
13    If R is a USP
14      DoTransfer(P, R)
15  End If

```

Fig. 2 DoTransfer (p,s) method

(a)	unattached-ordinary-peer algorithm
Assumptions: P: local ordinary-peer , Rp: p's region S:remote peer , Found = false	
01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27	If P is UOP then P Performs learning algorithm according to Fig. 2 If P.OLALG = super then P promotes to SP role Else If P.OLALG = ordinary-peer then IF recive AttachProposeMessage from S then P become ordinary-peer of S Else Form P.HigherSet from local view While P is UOP and P.HigherSet is not empty S = pick highest capacity peer from P.HigherSet If S is USP then P send AttachRequestMessage to S If recive AcceptMessage from S then P become ordinary-peer of S Else if S is UOP or AOP then P Send PromoteProposeMessage to S End If End While If P is AOP If Super-peer Not Available then P comedown to UOP Else If PromoteProposeMessage is recived then Performs its learning phase If P.OLALG = super P promotes to SP role End If
(b)	super-peers algorithm
Assumptions: P: local super-peer , Rp: p's region X , S: remote peer in P's region	
01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23	P performs learning algorithm according to Fig. 2 If P.OLALG = super P forms LowerSet While P.Lowerset is not empty and P is USP X = pick lowest capacity peer from P.LowerSet If X is UOP Send AttachProposeMessege to X End While While Receive AttachRequestMessage from X and P is USP If P.OLALG = super X becomes ordinary-peer of P End While Else If P.OLALG = ordinary-peer While Load(P) > 0 and P.HigherSet 0is not empty S = pick highest cpacity peer from P.HigherSet If S is a USP DoTransfer(P, S) Else if S is UOP or AOP Send PromoteProposeMessage to S End If End While End If End If End if

Fig. 3 Algorithm corresponding to (a): unattached-ordinary-peer, (b): super-peers

robustness at the time of super-peer failure, as well as overhead traffic production according to current network condition. To do that, learning automata can be used to solve problems in MP2P networks.

The rest of the paper is organized as follows. In Section 2, we review the related literature. Learning automata will be discussed in section 3. Section 4 represents our proposed protocol. Section 5 gives the performance evaluation of the proposed protocol, and finally section 6 concludes the paper.

2 Related work

Researches on mobile ad-hoc networks have recently been linked to P2P concepts. Examples of this link from different perceptions can be found in [16]. This relation can be extended to include super-peer networks. In the rest of this section, we study the existing algorithms and then we focus on super-peer selections of MP2P networks that invest in peers capacities.

The system administrators in [17] and [18] select super-peers manually. Solutions [4, 7, 9, 19–21] consist of systems where the population of peers is divided into groups, and super-peers are selected within each group independently. The grouping is usually based on such peer properties as physical location, network proximity, and semantic content. The heterogeneity of the peers capacity is observed in [4, 7, 9, 19–21]. They make use of the difference between peers capacity. Solutions [9] and [20] use static values as the thresholds to select super-peers. A peer will be a super-peer candidate if its superiority ratio exceeds a certain threshold. Manual Selection requires a global view of the network which opposes

Table 1 Simulation parameters

Parameters	Values
Simulation time	20 min
Maximum number of peers	200
Network area	1000 m × 1000 m
Maximum capacity(c_{max})	15
Maximum communication range	$c_{max} * 10$
Maximum speed of a peer	0 ~ 5 m/s
α parameter in power-law	2
λ parameter in Poisson	1
a (reward parameter)	0.5
b (penalty parameter)	0.2
Channel type	Wireless Channel
Network interface	WirelessPhy

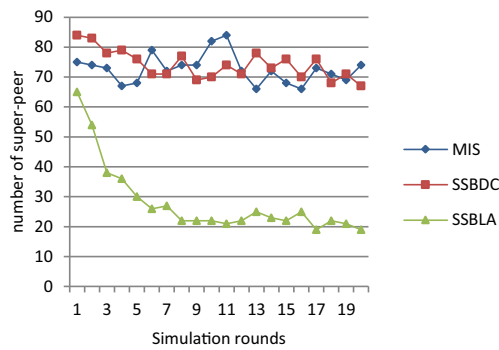


Fig. 4 Number of selected super-peers as simulation proceeds

the scalability of the network. The grouping of the peers does not allow the system to actively control and adapt the number of super-peers with the change in the system conditions. The number of the super-peers, which assigned according to the current groups, is not changeable at the runtime. Moreover, the maximum number of groups is fixed in many systems. Using capacity heterogeneity by fixed thresholds can only be applied to networks where the distribution of system-wide peer characteristics does not change significantly in time and is well known by the system designer or administrator. The static threshold does not allow the system to control the number of super-peers in dynamic populations of peers. If peer properties change, the super peer sets changes accordingly. This can lead to extreme cases, where no super-peers are selected, if all peers fall below the threshold, or where every peer is a super peer, if all peers are above the threshold.

In [8], a super-peer based MP2P system is proposed, in which super-peers are selected based on their mobility pattern in order to enhance the system stability and reliability. Also [22] share the same concern, but they believe that the peer energy level should be taken into consideration along with the mobility factor. Authors in [23] utilizes a fuzzy method with both mobility and energy as the fuzzy inputs. During each reconfiguration of the network, the proposed system selects super-peers adaptively using the fuzzy method to more effectively reflect the current environment [24] propose a

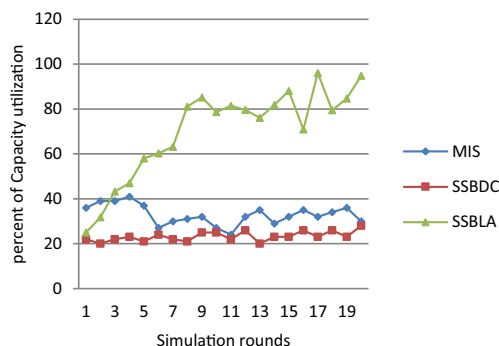


Fig. 5 Capacity utilization

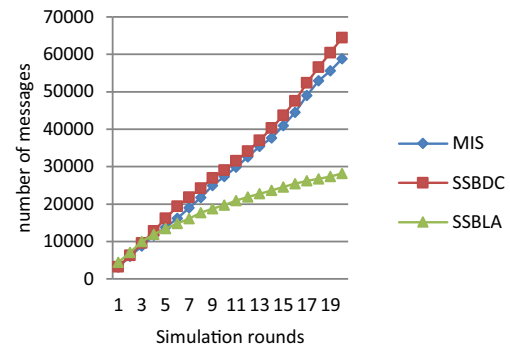


Fig. 6 Total Overhead

super-peer topology construction and maintenance scheme based on network coordinates.

All proposed algorithm in [8, 22, 23] and [24] don't consider peers capacity. In addition to, [8, 22, 23], believe If super peers move too fast, clusters may be broken because their ordinary-peers cannot communicate with their own super-peers. Therefore, they select low speed peers to play super-peer role. This idea doesn't answer in network with high speed peers.

Three selection algorithms are presented in [4, 7], and [1] that dynamically consider the capacity. they are described in the rest of this section.

In [1], a super-peer selection algorithm considering peers capacities is given. A main drawback of this algorithm is that it gathers much information about peers. Since MP2P networks are dynamic, this approach forces the management algorithms to continually gather information about peers which leads to drastically increase the traffic of the network that is not appropriate.

In [7], authors put forward two systems. The first system chooses super-peers based on a greedy method. Greedy system selects a peer with the highest degree of neighbor in the network greedily as super-peer and let its neighbor peers become its ordinary-peers. The second system chooses super-peers based on a maximal independent set (MIS) algorithm. This system discovers a maximal independent set and the peers that belong to the MIS become super-peers. The

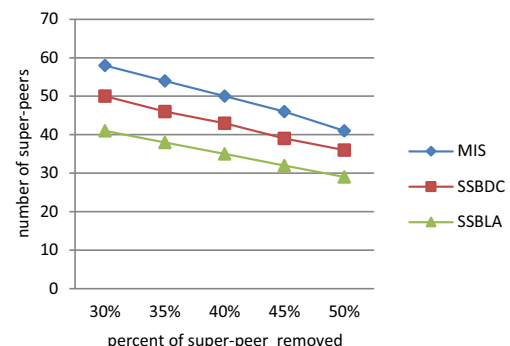


Fig. 7 Number of super-peers in the face of super-peers' failure

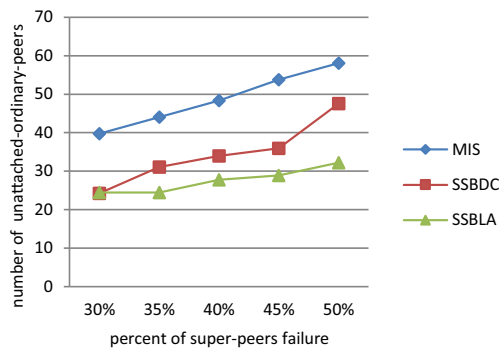


Fig. 8 number of unattached-ordinary-peers in the face of super-peers' failure

algorithm prepares each peer with a random number and then compares it with its neighbor peers, and a peer that has the largest number among its neighbor peers is put into MIS.

The difference between the above two system is that the first system needs a server to maintain the super-peers, however, the second system can be carried out in a distributed way and therefore can be fit for the mobile environment. On super-peer selection, the greedy method only takes the peer's degree into account, and the MIS algorithm prepares each peer in the network with a random value, that makes the selected super-peers too random. These systems are not combined with the characteristics of mobile peer-to-peer network. In addition, the above systems adopt a reselecting super-peer manner in lack of super-peer that will lead to longer query delay on searching files.

In [4], a Super-peer Selection Based on Dynamic Performance (SSBDC) algorithm is presented. This algorithm tries to select peers with high dynamic performance as super-peer, considering peers capacity and stability, and adopt an alternative backup super-peer. SSBDC selects a new super peer just when a super peer lack or leaves its ordinary-peers. Therefore super-peer service time to its ordinary-peers is more and super-peer should be able to handle the additional load. Also, if a peer with better conditions be arrived, the system cannot take advantage of its conditions. As a result, such system cannot adapt to network environment changes.

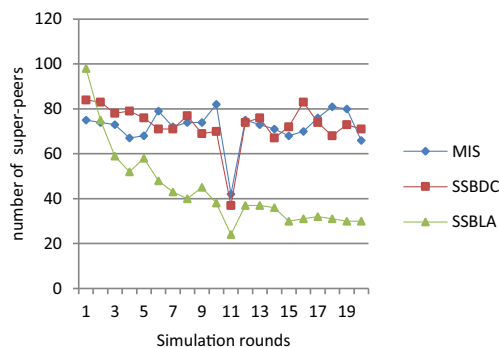


Fig. 9 Number of super-peers in the face of 50 % super-peers' failure

Furthermore protocols used in these applications have no adaptive solutions to exploit capacity heterogeneity of peers.

Main drawback of the existing algorithms is the lack of an adaptive mechanism for managing the peers role considering network condition. To solve this problem an adaptive algorithm will be proposed in the next section.

3 Overview of learning automata

learning automaton (LA) is an adaptive decision-making unit that has been shown to perform well in computer networks [25]. LA can improve its performance by learning how to choose the optimal action from a finite set of allowed actions through repeated interactions with a random environment. At each iteration, the action is chosen at random based on a probability distribution kept over the action-set and at each instant the given action is served as the input to the random environment.

The environments in which the reinforcement signal can only take two binary values 0 and 1 are referred to as p-model. Each LA belongs to one of fixed or variable structure category [3]. Later case are represented by a triple $\langle \beta, \alpha, L \rangle$ where β is the set of inputs, α is the set of actions, and L is learning algorithm that is a recurrence relation which is used to modify the action probability vector. Let $\alpha_i(k) \in \alpha$ and $p(k)$ indicate the action selected by LA and the probability vector defined over the action set at instant k , respectively. Let a and b indicate the reward and penalty parameters and determine the amount of increases and decreases of the action probabilities, respectively. Let r be the number of actions that can be taken by LA. At each instant k , the action probability vector $p(k)$ is updated by the linear learning algorithm given in (1), if the selected action $\alpha_i(k)$ is rewarded by the random environment, and it is updated as given in (2), if the taken action is penalized. If $a = b$, the recurrence equations (1) and (2) are called linear reward-penalty (L_{RP}) algorithm. If $0 < a < b < 1$, (L_{REP}) algorithm.

where a and b are reward and penalty parameters, respectively. If $a = b$, the automaton is called L_{RP} . If $b = 0$, the automaton is called L_{RI} and, if $0 < b < a < 1$, the automaton is called L_{REP} .

More information can be found in [14]:

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1-p_i(n)] \\ p_j(n+1) &= (1-a)p_j(n) \quad \forall j, j \neq i \end{aligned} \quad (1)$$

$$\begin{aligned} p_i(n+1) &= (1-b)p_i(n) \\ p_j(n+1) &= \frac{b}{r-1} + (1-b)p_j(n) \quad \forall j, j \neq i \end{aligned} \quad (2)$$

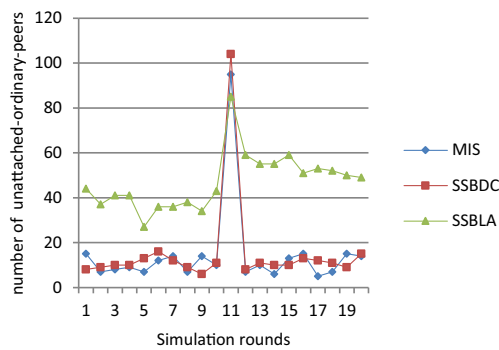


Fig. 10 number of unattached-ordinary-peers in the face of 50 % super-peers' failure

In the recent years, LA have been arisen to different network applications such as wireless sensor networks [26], WiMAX networks [27], network security [28], wireless mesh networks [29], mobile video surveillance [30], vehicular environment [31, 32], Peer-to-Peer networks [33, 34], cognitive network [35], wireless data broadcasting systems [36–38], smart grid systems [39], grid computing [40] and cloud computing [41], to mention a few.

4 Proposed method

In this section, first, the functionality of the proposed method is briefly explained, then required notations are introduced, and finally the detailed description of the proposed method is given.

In this method, each peer may play one of three roles: unattached-ordinary-peer, ordinary-peer and super-peer. The initial role of each peer is equal to unattached- ordinary-peer. Each peer uses a learning automaton to select its role. The action set of each learning automaton consists of two actions: super-peer and ordinary-peer. The steps of the proposed method are briefly described as follows.

The proposed method consists of three steps. The first step begins with presence of peers in the network area and forming underlying network. At this step, each peer with performing a neighboring management algorithm, identify its neighbors

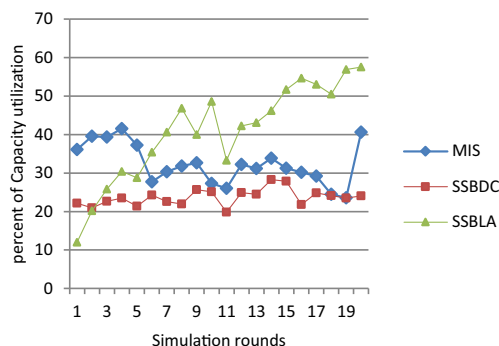


Fig. 11 Capacity utilization in the face of 50 % super-peers' failure

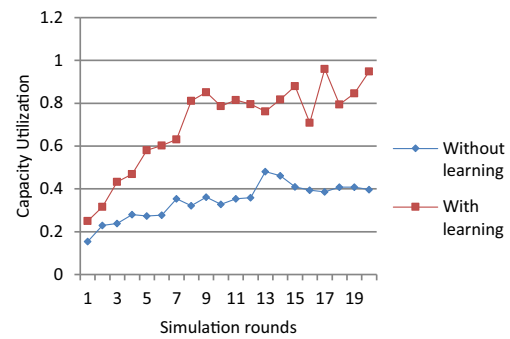


Fig. 12 Impact of learning on capacity utilization in SSBLA

within itself communication range. At the end of this step, peer achieves a local view of its neighbors. In the second step, each peer performs its learning algorithm based on its learning automata. Peer, relying on its learning automata selected action, chooses an appropriate role: super-peer or ordinary-peer. In the third step, the peer performs an algorithm corresponding to its role. In the rest of this section some required notations are given and then the proposed method is described in more details.

4.1 Required notations

We have a network with n peers that Each peer p is associated with a capacity (p) parameter. Capacity (p) shows the maximum number of ordinary-peers that if peer p is selected as super-peer, can handle them. Also, load (p) shows the current number of ordinary-peers is managed by super-peer p at this time. If load(p) is less than capacity (p), p would set as under-loaded super-peer. If and only if a super-peer is under-loaded and has more capacity, it is allowed to absorb new ordinary-peer. When every peer joins overlay it obtains a local view from its neighbor.

Notations used to refer the peers is given as follow: SP: super-peer, USP: under-loaded super-peer, AOP: attached ordinary-peer, UOP: unattached-ordinary-peer. p .Higherset/ p .Lowerset consists of connected peers in local view whose capacity is higher/lower than that of p . Neighboring items, also called peer descriptors, include the identifier, capacity,

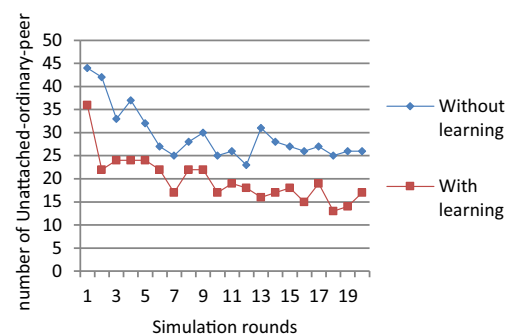


Fig. 13 Impact of learning on unattached-ordinary-peer in SSBLA

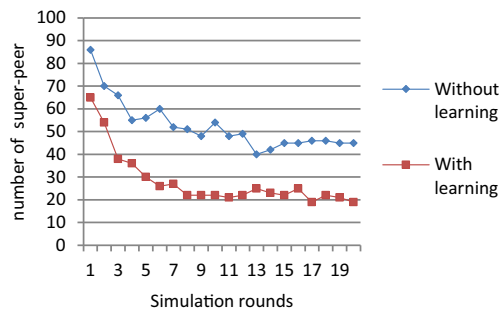


Fig. 14 Impact of learning on super-peer in SSBLA

current role (super-peer, ordinary-peer or unattached- ordinary-peer), and other functional properties of peer. Local view is maintained by an underlay peer neighboring manager.

Each peer p in the network is equipped with a learning automaton $RoleLA_p$. $RoleLA_p$ uses L_{REP} algorithm to help the peer for adopt a decision about its proper role according to network condition. The action set of each $RoleLA$ consists of two actions: super-peer and ordinary-peer. It should be noted that “action” term and “role” term are different from each other. Each peer in the network introduces itself in accordance with its current role and takes the corresponding duties. However, the action set of each peer is related to its own $RoleLA$. Each peer uses the selected action of its learning automaton locally, in order to be able to adopt a decision about its role and admit roles: ordinary-peer, unattached-ordinary-peer and super-peer. For this purpose, we use a learning algorithm called LALG. OLALG indicates the output of LALG algorithm. $p.OLALG$ indicates the output of $LALG$ algorithm running by peer p .

Three types of messages are used in the proposed algorithm: introductory, super-peer selection, and search file. Each of these messages contains the source ID, destination ID, message type, and a payload. Introductory messages include hello and identify message. The hello message has an empty payload, and Payload of identify message covers capacity, current role (super-peer, ordinary-peer or unattached-ordinary-peer) and other useful properties of a peer. super-peer selection include AttachProposeMessage, AttachRequestMessage,

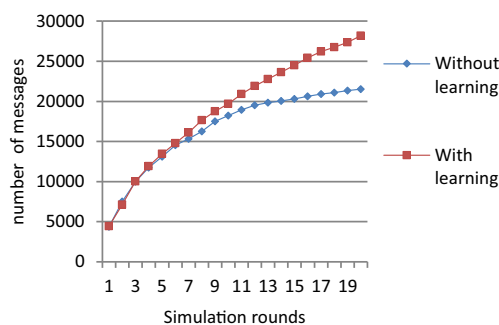


Fig. 15 Impact of learning on overhead in SSBLA

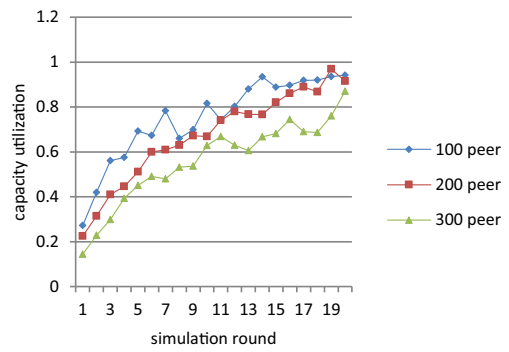


Fig. 16 Impact of peer number on capacity utilization in SSBLA

PromoteProposeMessage, RegisterFilesMessage, TransferPeersMessage. Payload of AttachProposeMessage, AttachRequestMessage, PromoteProposeMessage is empty. RegisterFilesMessage Payload consist of shared files list of ordinary-peer to register in its corresponding super-peer. At last TransferPeersMessage is used to transferring abandoned ordinary-peers between super-peers and it's payload consist of ordinary-peers information.

Now we describe the proposed protocol in more details.

4.2 Proposed algorithm: SSBLA

When each peer p joins the system, it takes unattached-ordinary-peer role. After joining the system, p calls neighboring manager and creates its local view. Then it performs its learning algorithm according to Fig. 1. In learning algorithm, each peer p updates its action probability vector in accord with “its selected action” and “the comparison between its capacity and neighbors capacity in its local view”. This process is repeated for each neighbor in local view of peer p . Then peer p chooses an action according to its action probability vector. It is clear that our policies for giving reward or penalty are based on achieve maximizing capacity of selected super-peers aim. The last selected action is considered as the output of $LALG$ algorithm, indicated as $p.OLALG$.

Each super-peer maintains ordinary-peers currently associated with it. Super-peers periodically check members

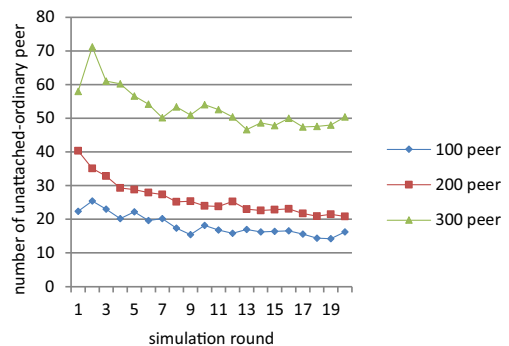


Fig. 17 Impact of peer number on unattached-ordinary-peer in SSBLA

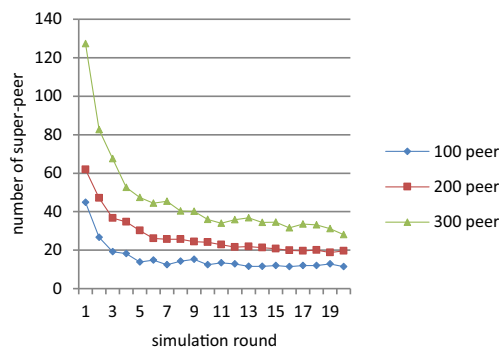


Fig. 18 Impact of peer number on super-peer in SSBLA

in their local view set and try to find a appropriate candidate to absorb ordinary-peer. Also if super-peer p needs to transfer its ordinary-peers to another super-peer s , transferring process is done from p to s , according to Fig. 2 (here C indicates the number of ordinary-peers in transmission range of s that are allowed to be accepted by s). If all of p 's ordinary-peers were transferred and s was exhausted, peer p becomes ordinary-peers of s and come-down to AOP role. Otherwise, p remains in super-peer role and examines members in its local view set again.

Each peer performs learning algorithm in each round, except attached ordinary-peers that performs this phase on demand, as far as local view of each peer is updated periodically. In other words, learning algorithm is the starting step in our designed algorithms for unattached-ordinary-peers and super-peers. But attached ordinary-peers perform this algorithm if a request received. When peer joins the system, it performs an algorithm periodically according to its current role. As shown in Fig. 3: attached and unattached-ordinary-peers perform an algorithm shown in part (a) and super-peers act as part (b). We introduce notations and give a detailed description of our algorithms in the following lines.

There are some key rules applied in our algorithms:

1. Each UOP or AOP can promote itself to super-peer role if and only its OLALG is super-peer.

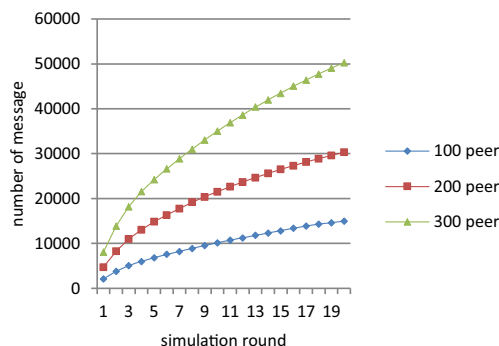


Fig. 19 Impact of peer number on overhead in SSBLA

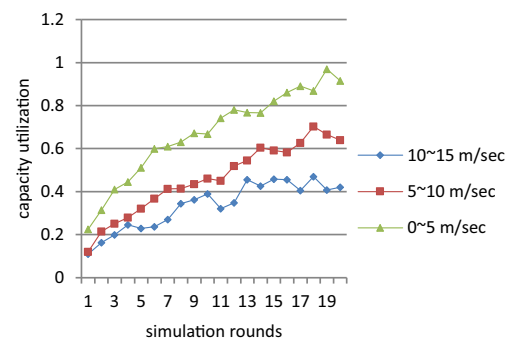


Fig. 20 Impact of peer speed on capacity utilization in SSBLA

2. Each USP is allowed to accept more load if and only if its OLALG is super-peer and USP be in ordinary-peer transmission range.
3. Each UOP can become AOP if and only if its OLALG is ordinary-peer.
4. Dotransfer(p, s) method can be performed if and only if p and s are SPs, s has higher capacity and s .OLALG is super-peer.

All of these rules can be easily satisfied in practice because all of them use local information and operations. It should be noted that both algorithms used in LALG and DoTransfer are implemented based on local information about peers. According to part (a) of Fig. 3, in first step, each UOP such as p performs Neighborhood manager by sending and receiving introductory messages. Then p performs learning algorithm. According to rule (1), if p .OLALG is super-peer, p itself promotes to super-peer role. Otherwise if p .OLALG is ordinary-peer and receives a AttachProposeMessage from a super-peer to attach, p becomes its ordinary-peer. Else, p considers its HigherSet, In each iteration, picks its highest capacity neighbor, s , and sends it a AttachRequestMessage. This message is made s to use its own LALG. If s is an USP and rule (2) is obeyed by s , p becomes its ordinary-peer. If s is an AOP or UOP, p proposes promotion to s by PromoteProposeMessage. Hereupon s utilizes its own LALG, if rule (1) is followed by s , firstly s promotes as super-peer then according to rule (2), p becomes its ordinary-peer. Above

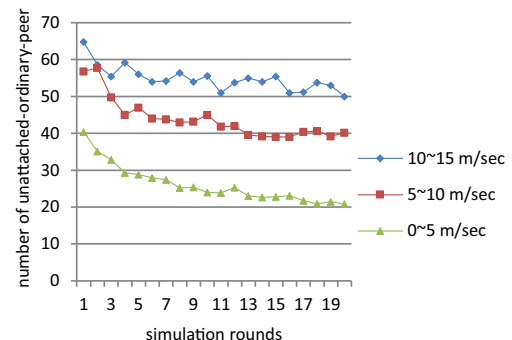


Fig. 21 Impact of peer speed on unattached-ordinary-peer in SSBLA

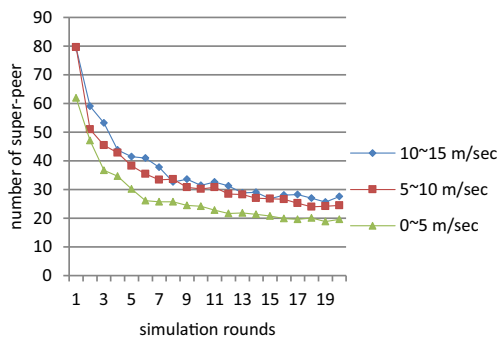


Fig. 22 Impact of peer speed on super-peer in SSBLA

process continues until p becomes AOP or no more peer exists in HigherSet of p .

According to part (b) of Fig. 3, at first step, each SP such as p update its view and form p .HigherSet/ p .LowerSet. Then p performs its learning algorithm and utilizes its LALG to check rule (2). If p .OLALG is super-peer, peer p is eligible to absorb more ordinary-peers. So until p is USP and receives AttachRequestMessage pick sender as its ordinary-peer. Then if p is still USP, it picks the lowest capacity neighbor x from p .LowerSet. If x is an UOP, it uses its own LALG, if x follows rule (3), x is attached to p . This loop continues until p becomes fully-loaded or no more peer exists in LowerSet of p . If p .OLALG isn't super-peer, it search in highest capacity neighborS to transfer its ordinary-peers. p picks s from HigherSet. If s is a SP, rule (4) is investigated to perform DoTransfer(p,s). Otherwise If s is an AOP or UOP, p proposes it to promote to super-peer. The above-mentioned process continues until p transfers all ordinary-peers.

5 Performance evaluation

In this section we have developed our protocol using OMNeT++ simulator [42] to evaluate efficiency of SSBLA in comparison with SSBDC [4] and MIS [7] algorithms. OMNeT++ is a discrete event-based simulator. Among super-peer selection algorithms in mobile peer-to-peer networks, MIS is a well-known algorithm. SSBDC algorithm is

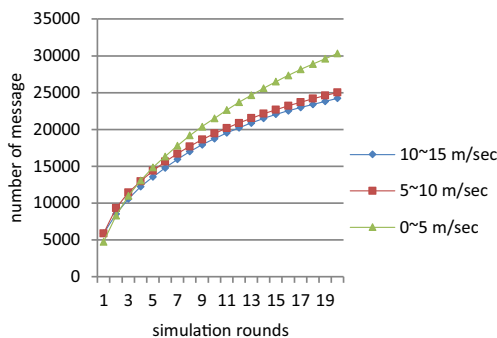


Fig. 23 Impact of peer speed on overhead in SSBLA

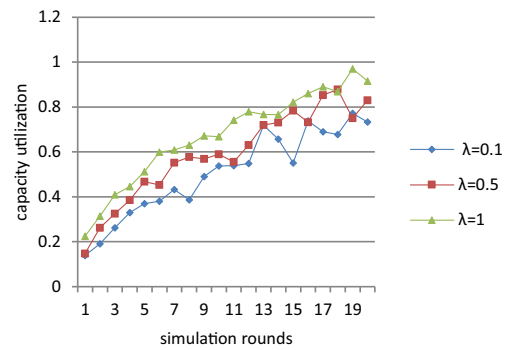


Fig. 24 Impact of arrival rates on capacity utilization in SSBLA

the super-peer selection algorithm in mobile peer-to-peer network that dynamically considers peers capacities without any threshold.

The parameters of the experiments are given in Table 1. Results are averaged over 20 experiments. In proposed algorithm, each peer is equipped with a variable structure learning automaton of type $L_{R \in P}$. Reward and penalty parameters of learning automaton respectively is equal to $a = 0.5$, $b = 0.2$. Peers are located in a 1000×1000 m network area. The departure and arrival of peers from/to the network area are governed by a Poisson distribution. Parameter λ indicates the parameter of the Poisson distribution. The movements of peers are based on the random waypoint model in [43]. The maximum speed of each peer is specified in each experiment and maximum 12 s stay, and all peers are able to move anywhere. The capacities of peers are based on power-law distribution. In the power-law distribution, the probability $P[cn = x]$ that a peer n has a capacity x , with x contained in a bounded range $[1; c_{max}]$, is equal to $x^{-\alpha}$, where α is the distribution exponent [44]. The maximum capacity available within the network is called c_{max} . Capacity of a peers does not change during the peer presence in the network.

To evaluate the result of experiment, one way ANOVs and tukey statistical tests are used. Result of statistical tests are provided in Appendix and its analysis has been added to each experiment.

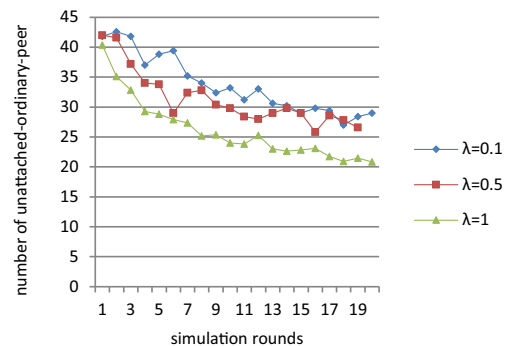


Fig. 25 Impact of arrival rates on unattached-ordinary-peer in SSBLA

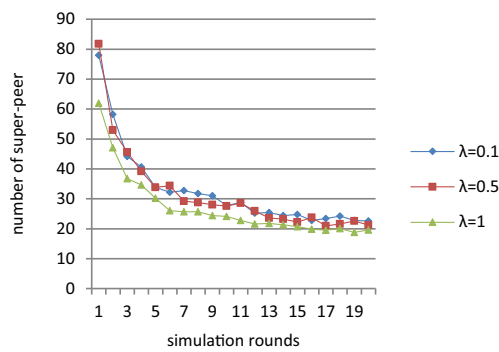


Fig. 26 Impact of arrival rates on super-peer in SSBLA

5.1 Experiment 1

This experiment is conducted to study the capacity utilization in super-peer selection. The capacity utilization (CU) is defined as the ratio of current number of attached ordinary-peer to total capacity provided by super-peers, as given in (3) (let S indicates the number of selected super-peers).

$$\text{capacity utilization} = \frac{\text{number of attached ordinary-peer}}{\sum_{i=1}^S \text{capacity}(i)} \quad (3)$$

ALL parameters have been assigned under section 5 and Table 1.

In SSBLA, each peer learns about its eligibility in its learning algorithm. Therefore, high capacity peers are able to take super-peer role. But, SSBDC makes each decision by only one capacity comparison. As shown in the Fig. 5, in the third round, SSBLA obtained more capacity utilization than SSBDC and MIS and until the end of the simulation, it obtained utilization toward 100 %, while MIS and SSBLA did not exceed 40 %. An interesting point concerning the evaluation of capacity utilization of MIS protocol is that its efficient capacity is more than SSBDC, despite disregarding the capacity factor in selection process.

High utilization of SSBLA, despite the lack of high capacity peers in power-law distribution as well as low number of selected super-peers, represents high performance of protocol in utilizing the existing capacity. The approach that SSBLA

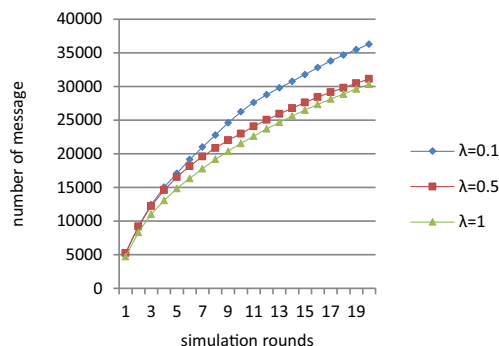


Fig. 27 Impact of arrival rates on overhead in SSBLA

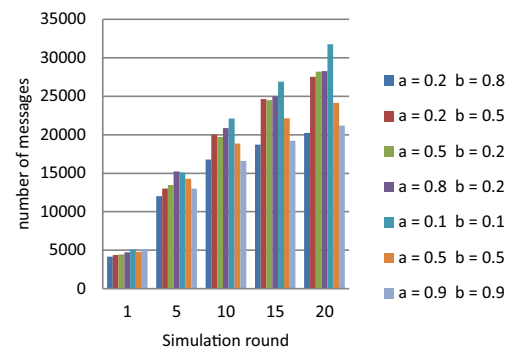


Fig. 28 Impact of Learning Automata reward and penalty parameters on overhead in SSBLA

protocol takes in learning algorithm enables peers to learn network conditions and use network facilities correctly. According to Fig. 5 In SSBLA, each peer learns its eligibility in its learning algorithm. Therefore, high capacity peers are able to play super-peer role. However, SSBDC makes each decision by only one capacity comparison.

SSBDC at the beginning of group formation, only one time selects a peer with highest capacity. Selected super-peer remain in its role until leaving the network. With arrival of a high capacity peer to communication range of current super-peer or change of topology for peers movement, likewise, doesn't change current super-peer. Just in lack of super-peer, the algorithm is called. Although, MIS selects super-peers in a greedy manner and does not consider capacity in super-peer selection process. In both algorithm may super-peer role assign to low capacity peers. it may reduce performance and Experiment results show it.

Until round 5, MIS tries to get most CU then decreases. As it is shown in Fig. 4, SSBLA is reduced in the number of super-peers to cover all other peers according to current network condition, while CU in Fig. 5 is increased. Altogether, capacity utilization in SSBLA is higher than SSBDC and MIS and number of super-peers is lower to cover all other peers.

According to Appendix Table 2 about Fig. 4, column value of the P- value, assuming equal means of protocols cannot be accepted. Therefore protocols significantly are different from

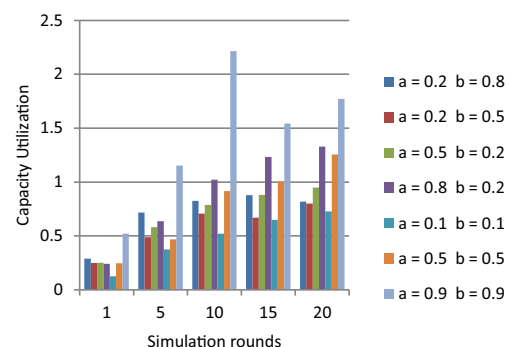


Fig. 29 Impact of Learning Automata reward and penalty parameters on capacity utilization in SSBLA

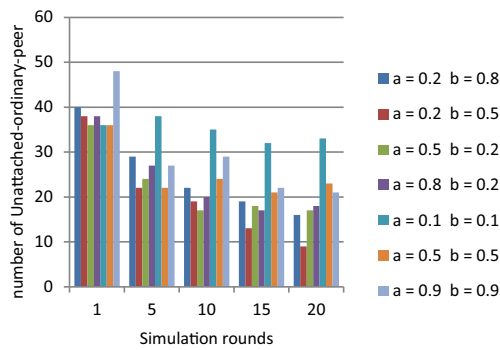


Fig. 30 Impact of Learning Automata reward and penalty parameters on unattached-ordinary-peer in SSBLA

each other. According to Appendix Table 3, MIS and SSBDC have same function and SSBLA are significantly different from other protocol.

According to the results presented in Appendix Table 2 about Fig. 5, protocols significantly are different. Upon the Appendix Table 3, all three protocols are significantly different and none of them are similar to each other.

5.2 Experiment 2

This experiment is conducted to investigate the impact of the proposed protocol on communication overhead. communication overhead indicates the number of messages that are transmitted between peers during super-peer selection to ask for information. ALL parameters have been assigned under section 5 and Table 1.

SSBDC and MIS algorithms in super-peer lack will proceed to the selection. SSBLA on top of lack, calls the selection algorithm when a more eligible peer arrival. With this explanation, it is expected SSBLA overhead be more than other two algorithms. According to Fig. 6, since in SSBLA each peer makes its decisions based on the action selected in its learning algorithm, in primary rounds more message are transferred. After passing through learning phase in round 4, the number of messages is reduced in comparison to SSBDC and MIS.

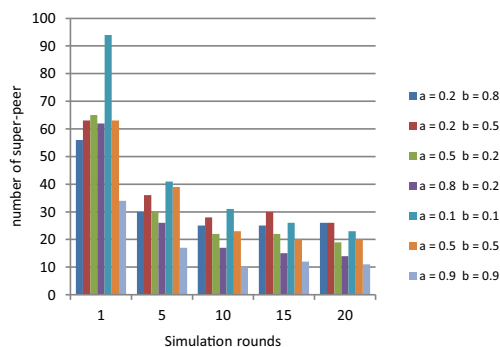


Fig. 31 Impact of Learning Automata reward and penalty parameters on super-peer in SSBLA

It can be concluded from these experiments, Whole Proceeding of SSBDC and MIS doesn't lead to an overhead reduction. Using newly joined eligible peers and learning environment condition, help to select more stable super-peer. It also reduce frequent super-peer selection and overhead.

Considering the presented results in Appendix Table 2 about Fig. 6, protocols significantly are different. Based on Appendix Table 2, SSBDC and MIS are similar to each other and SSBLA and SSBDC have significant differences.

5.3 Experiment 3

This experiment is conducted to study the robustness of the proposed protocol. ALL parameters have been assigned under section 5 and Table 1.

To evaluate the robustness of our protocol in super-peers failure, as in Figs. 7 and 8, a catastrophic scenario is shown: 30 %, 35 %, 40 %, 45 %, 50 % of the super-peers are removed. After the initial stage, when all ordinary-peers whose super-peer had been crashed, ordinary-peers become unattached and the protocol behaves as usual. It repairs the overlay topology by selecting new super-peers among the peers. Figure 7 shows the number of remaining super-peers and Fig. 8 shows the percentage of unattached-ordinary-peers respectively. According to Figs. 7 and 8 after crash, the percentage of unattached-ordinary-peers in SSBLA is lower than other algorithm. It shows that SSBLA is more stable in super-peers failure.

Figure 9 show number of super-peers in the scenario in which 50 % of super-peers removed at round 10. Figures 10 and 11 respectively show effect of crash on the number of unattached-ordinary-peers and percent of Capacity utilization. According to Figs. 9 and 10, Although the number of super-peers are less than other algorithms, SSBLA have unattached-ordinary-peers less than other too.

On the other hand, after removing 50 % of super-peers, SSBLA still has the highest Capacity utilization towards other protocols. In the first round after the failure, SSBLA obtains nearby 20 % of capacity utilization. It is result of using previous learning in SSBLA. the experiment results show, SSBLA is more self-healing than other algorithms.

According to the results presented in Appendix Table 2 about Fig. 9, tree protocols significantly are different. With considering Appendix Table 3, MIS resembles SSBDC and there is a significant difference between SSBLA with these two protocol. Fig. 9. Also confirms these results.

Based on the results presented in Appendix Table 2 about Fig. 10, tree protocols significantly are different. With considering Appendix Table 3, MIS resembles SSBDC and there is a significant difference between SSBLA with these two protocol.

As shown in the Appendix Table 2 about Fig. 10, this protocols significantly are different. Considering Appendix Table 3, they doesn't have any meaningful resemblance with each other.

5.4 Experiment 4

This experiment is conducted to study the impact of learning on our protocol. For this propose, the algorithm one time has been performed using learning automata and the next time without using learning automata. ALL other parameters have been assigned under section 5 and Table 1.

As the simulation results in Figs. 12, 13, 14, and 15 are shown, utilizing learning indicates significant improvement over evaluation parameters. As shown in Fig. 15, in the absence of learning, utilization, super-peer selection overhead is reduced. It occurs due to lack of learning messages for learning environment. It should be noted, as shown in Fig. 6, the overhead of algorithm with learning still has been lower than overhead of other algorithms. But using learning, as shown in Figs. 12, 13, and 14, with lower number of super-peer cover more peer and achieve greater capacity utilization.

5.5 Experiment 5

This experiment is conducted to study the impact of number of peers on our algorithm. In this experiment, the algorithm is tested with different peers number: 100, 200 and 300. Other parameters have been assigned under section 5 and Table 1.

With the increasing of peers number, need to super-peer And follow it, number of selection message increase. Accordingly, number of unattached-ordinary-peer also increased. capacity utilization is correlated with number of peers. growing number of peer, led to decline in capacity utilization and vice versa. Based on these experiment, the proposed algorithm has better performance in low-population environment (Figs. 16, 17, 18, and 19).

5.6 Experiment 6

This experiment is conducted to study the impact of speed of the peers on our protocol. In order to evaluate impact of speed on the network parameters, The proposed algorithm is performed at 0 ~ 5, 5 ~ 10 and 10 ~ 15 m/s speeds. Other parameters have been assigned under section 5 and Table 1.

Increasing peers speed causes rising dynamism of the network topology. It can reduce the stability of the connections among peers. Based on the results, with an increase in speed, number of super-peers and unattached -ordinary-peer is growth. In this situation, the ratio of attached ordinary-peer to super-peers and subsequently capacity utilization is reduced. While unattached-ordinary-peers increases, frequently trying to select new super-peer leads to produce selection message. So overhead in 10 ~ 15 m/s is more than others. overhead in 5 ~ 10 and 0 ~ 5 is close to each other (Figs. 20, 21, 22, and 23).

5.7 Experiment 7

This experiment is conducted to study the impact of arrival rates of the peers on our protocol. In this experiment impact of lambda is compared with 0.1, 0.5 and 1 values. other parameters have been assigned under section 5 and Table 1.

It was said the departure and arrival of peers from/to the network area are governed by a Poisson distribution. Lambda parameter of Poisson distribution determines peer arrival rates and the dynamics of network. Whatever lambda parameter is less, more peer join or leave per time unit. Thus the connections become more unstable. Then need to renewed super-peer selection will grow. Figures 24, 25, 26, and 27, show this concept.

5.8 Experiment 8

This experiment is conducted to study the impact of reward and penalty parameters of the learning automata on our protocol. For this propose, SSBLA is measured under a (reward parameter) = 0.1, 0.2, 0.5, 0.8, and 0.9, b (penalty parameter) = 0.1, 0.2, 0.5, 0.8, and 0.9. Other parameters have been assigned under section 5 and Table 1.

According to Fig. 28, 29, 30, and 31, LRP model results are not satisfactory with different values of 1.0 to 9.0. L_{RP} model with $a=0.5$ and $b=0.2$ show better result on evaluation parameters. $a=0.5$ and $b=0.2$ value in the charts are shown by in green.

6 Conclusion and future work

This paper presents SSBLA, a super-peer selection based learning automata for mobile peer to peer network. we intended to maximizing capacity utilization by minimum number of super-peer and improve robustness against failures of super-peers while minimizing selection communication overhead.

According to the simulations, SSBLA can reach capacity utilization more than SSBDC and MIS. The simulations render the mobile P2P network topology, constructed with SSBLA, more stable, less affected by super-peers failure and it is able to recover faster. On the other hand, SSBLA reduces overhead of super-peer selection compared to SSBDC and MIS. SSBLA, in spite of SSBDC, prevents relatively high capacity peers from demotion, as far as possible.

Since the proposed method depends on communication networks, In the future work, we intend to reduce its communication overhead by a mechanism inspired from an algorithm reported in [45].

Appendix

In this section, we present the results of statistical tests. To evaluate the result of experiment, one way ANOVs and tuky statistical tests are used.

Table 2 One way ANOVs test

ANOVA						
	Method	Sum of Squares	df	Mean Square	F	Sig
Fig. 4	Between Groups	27,412.433	2	13,706.217	215.501	.000
	Within Groups	3625.300	57	63.602		
	Total	31,037.733	59			
Fig. 5	Between Groups	24,255.633	2	12,127.817	84.002	.000
	Within Groups	8229.350	57	144.375		
	Total	32,484.983	59			
Fig. 6	Between Groups	1.848E9	2	9.240E8	4.115	.021
	Within Groups	1.280E10	57	2.245E8		
	Total	1.465E10	59			
Fig. 9	Between Groups	10,929.433	2	5464.717	33.811	.000
	Within Groups	9212.750	57	161.627		
	Total	20,142.183	59			
Fig. 10	Between Groups	13,939.300	2	6969.650	21.514	.000
	Within Groups	18,465.300	57	323.953		
	Total	32,404.600	59			
Fig. 11	Between Groups	2993.033	2	1496.517	23.478	.000
	Within Groups	3633.300	57	63.742		
	Total	6626.333	59			

Table 3 Tuky test

Multiple Comparisons							
	method Tukey HSD					95 % Confidence Interval	
	(I) Allscenario	(J) Allscenario	Mean Difference (I-J)	Std. Error	Sig.	Lower Bound	Upper Bound
Fig. 4	MIS	SSBDC	−1.45000000	2.52193883E0	.834	−7.5188394	4.6188394
		SSBLA	44.60000000*	2.52193883E0	.000	38.5311606	50.6688394
	SSBDC	MIS	1.45000000	2.52193883E0	.834	−4.6188394	7.5188394
		SSBLA	46.05000000*	2.52193883E0	.000	39.9811606	52.1188394
	SSBLA	MIS	−4.46000000E1*	2.52193883E0	.000	−50.6688394	−38.5311606
		SSBDC	−4.60500000E1*	2.52193883E0	.000	−52.1188394	−39.9811606
Fig. 5	MIS	SSBDC	9.65000000*	3.79966527	.036	.5064165	18.7935835
		SSBLA	−37.00000000*	3.79966527	.000	−46.1435835	−27.8564165
	SSBDC	MIS	−9.65000000*	3.79966527	.036	−18.7935835	−.5064165
		SSBLA	−46.65000000*	3.79966527	.000	−55.7935835	−37.5064165
	SSBLA	MIS	37.00000000*	3.79966527	.000	27.8564165	46.1435835

Table 3 (continued)

Multiple Comparisons

		method Tukey HSD						
		(I) Allscenario	(J) Allscenario	Mean Difference (I-J)	Std. Error	Sig.	95 % Confidence Interval	
							Lower Bound	Upper Bound
Fig. 6	MIS	SSBDC	46.65000000*	3.79966527	.000	37.5064165	55.7935835	
		SSBDC	-2370.10000000	4738.46128182	.872	-1.3772819E4	9032.6192591	
		SSBLA	10,407.30000000	4738.46128182	.080	-995.4192591	21,810.0192591	
	SSBDC	MIS	2370.10000000	4738.46128182	.872	-9032.6192591	13,772.8192591	
		SSBLA	12,777.40000000*	4738.46128182	.025	1374.6807409	24,180.1192591	
Fig. 9	SSBLA	MIS	-10,407.30000000	4738.46128182	.080	-2.1810019E4	995.4192591	
		SSBDC	-12,777.40000000*	4738.46128182	.025	-2.4180119E4	-1374.6807409	
		MIS	SSBDC	-65000000	4.02028846	.986	-10.3244951	9.0244951
	SSBDC	SSBLA	28.30000000*	4.02028846	.000	18.6255049	37.9744951	
		MIS	.65000000	4.02028846	.986	-9.0244951	10.3244951	
Fig. 10	SSBLA	SSBLA	28.95000000*	4.02028846	.000	19.2755049	38.6244951	
		MIS	-28.30000000*	4.02028846	.000	-37.9744951	-18.6255049	
		SSBDC	-28.95000000*	4.02028846	.000	-38.6244951	-19.2755049	
	MIS	SSBDC	-85000000	5.69168368	.988	-14.5465710	12.8465710	
		SSBLA	-32.75000000*	5.69168368	.000	-46.4465710	-19.0534290	
Fig. 11	SSBDC	MIS	.85000000	5.69168368	.988	-12.8465710	14.5465710	
		SSBLA	-31.90000000*	5.69168368	.000	-45.5965710	-18.2034290	
		MIS	32.75000000*	5.69168368	.000	19.0534290	46.4465710	
	SSBLA	SSBDC	31.90000000*	5.69168368	.000	18.2034290	45.5965710	
		MIS	SSBDC	8.75000000*	2.52471989	.003	2.6744682	14.8255318
Fig. 11	SSBLA	SSBLA	-8.55000000*	2.52471989	.004	-14.6255318	-2.4744682	
		MIS	-8.75000000*	2.52471989	.003	-14.8255318	-2.6744682	
		SSBLA	-17.30000000*	2.52471989	.000	-23.3755318	-11.2244682	
	SSBLA	MIS	8.55000000*	2.52471989	.004	2.4744682	14.6255318	
		SSBDC	17.30000000*	2.52471989	.000	11.2244682	23.3755318	

- systems,” Master of Electrical Engineering and Computer, Department of Electrical Engineering, the Faculty of Sciences and Technology New University of Lisbon
11. Fitzek FHP, Charaf H (2009) Introduction to WLAN IEEE802.11 communication on mobile devices. In: *Mobile Peer to Peer (P2P): A Tutorial Guide*. John Wiley & Sons
 12. Schiller JH (2003) *Mobile communications*, 2nd ed.: Addison-Wesley
 13. Okdem S (2015) A cross-layer adaptive mechanism for low-power wireless personal area networks. *Comput Commun*
 14. Najim K, Poznyak AS (1994) *Learning automata: theory and application*. In: Tarrytown
 15. Thathachar KSNMA L (1989) *Learning automata: An introduction*. Prentice-Hall
 16. Castro MC, Kessler AJ, Chiasserini C-F, Casetti C, Korpeoglu I (2010) Peer-to-peer overlay in mobile ad-hoc networks. In: Shen X, Yu H, Buford J, Akon M (eds) *Handbook of peer-to-peer networking*. Springer, New York, pp 1045–1080
 17. Seet BC (2005) “Mobile P2Ping: a super-peer based structured P2P system using a fleet of city buses,” presented at the IEEE3th International Conference on Pervasive Computing and Communications Workshops, Kauai Island, Hawaii
 18. Tang B, Zhou Z, Kashyap A, Chiueh TC (2005) “An integrated approach for P2P file sharing on multi-hop wireless networks,” presented at the IEEE International Conference on Wireless And Mobile Computing, Networking And Communications, Montreal, Canada
 19. Gerla M, Tsai JTC (1995) Multiclustet, mobile, multimedia radio network. *Wirel Netw* 1:255–265
 20. Wang TI, Tsai KH, Lee YH (2004) “Crown: an efficient and stable distributed resource lookup protocol distributed resource lookup protocol,” presented at the International Conference EUC 2004, Aizu-Wakamatsu, Japan
 21. Wei Y, Xie G, Li Z (2007) A hierarchical cross-layer protocol for group communication in MANET,” presented at the Telecommunications and Malaysia International Conference on Communications, Penang, Malaysia
 22. Kim SK, Lee KJ, Yang SB (2011) An enhanced super-peer system considering mobility and energy in mobile environments,” in 6th International Symposium on Wireless and Pervasive Computing (ISWPC) pp. 1–5
 23. Lee K-J, Choi J-H, Yang S-B (2012) Fuzzy inference-based super peer selection for a practical double-layered mobile peer-to-peer system. *Ad Hoc Sensor Wirel Netw* 21:327–351
 24. Merz P, Priebe M, Wolf S (2008) Super-peer selection in peer-to-peer networks using network coordinates,” In: *Internet and Web Applications and Services, 2008. ICIW’08. Third International Conference on*, pp. 385–390
 25. Torkestani JA, Meybodi MR (2010) An intelligent backbone formation algorithm for wireless ad hoc networks based on distributed learning automata. *Comput Netw Int J Comput Telecommun Netw* 54(5):826–843
 26. Safavi S, Meybodi MR, Esnaashari M (2014) Learning automata based face-aware Mobicast. *Wirel Pers Commun* 77: 1923–1933
 27. Misra S, Chatterjee SS, Guizani M (2015) Stochastic learning automata-based channel selection in cognitive radio/dynamic spectrum access for WiMAX networks. *Int J Commun Syst* 28:801–817
 28. Krishna PV, Misra S, Joshi D, Gupta A, Obaidat MS (2014) Secure socket layer certificate verification: a learning automata approach. *Secur Commun Netw* 7:1712–1718
 29. Kumar N, Lee J-H (2015) Collaborative-learning-automata-based channel assignment with topology preservation for wireless mesh networks under QoS constraints. *IEEE Syst J* 9:675–685
 30. Kumar N, Lee J-H, Rodrigues JJ (2015) Intelligent mobile video surveillance system as a bayesian coalition game in vehicular sensor networks: learning automata approach. *IEEE Trans Intell Transp Syst* 16:1148–1161
 31. Kumar N, Misra S, Obaidat M, Rodrigues J, Pati B (2014) Networks of learning automata for the vehicular environment: a performance analysis study. *IEEE Wirel Commun* 21:41–47
 32. Kumar N, Misra S, Obaidat MS (2015) Collaborative learning automata-based routing for rescue operations in dense urban regions using vehicular sensor networks. *IEEE Syst J* 9:1081–1090
 33. Saghir AM, Meybodi MR (2015) A distributed adaptive landmark clustering algorithm based on mOverlay and learning automata for topology mismatch problem in unstructured peer-to-peer networks. *Int J Commun Syst*
 34. Saghir AM, Meybodi MR (2015) A self-adaptive algorithm for topology matching in unstructured peer-to-peer networks. *J Netw Syst Manag* 1–34
 35. Saghir AM, Meybodi MR (2016) An approach for designing cognitive engines in cognitive peer-to-peer networks. *J Netw Comput Appl* 70:17–40
 36. Polatoglou M, Nicopolitidis P, Papadimitriou GI (2014) On low-complexity adaptive wireless push-based data broadcasting. *Int J Commun Syst* 27:194–200
 37. Nicopolitidis P, Chrysostomou C, Papadimitriou GI, Pitsillides A, Pomportsis AS (2014) On the efficient use of multiple channels by single-receiver clients in wireless data broadcasting. *Int J Commun Syst* 27:513–520
 38. Nicopolitidis P (2015) Performance fairness across multiple applications in wireless push systems. *Int J Commun Syst* 28:161–166
 39. Misra S, Krishna PV, Saritha V, Obaidat MS (2013) Learning automata as a utility for power management in smart grids. *IEEE Commun Mag* 51:98–104
 40. Hasanzadeh M, Meybodi MR (2015) Distributed optimization Grid resource discovery. *J Supercomput* 71:87–120
 41. Morshedlou H, Meybodi MR (2014) Decreasing impact of sla violations: a proactive resource allocation approach for cloud computing environments. *IEEE Trans Cloud Comput* 2:156–167
 42. Varga A, Hornig R An overview of the OMNeT++ simulation environment. In: *The 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*
 43. Broch J, Maltz DA, Johnson DB, Hu Y-C, Jetcheva J (1998) A performance comparison of multi-hop wireless ad hoc network routing protocols. In: *The 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom’98)*, pp. 85–97
 44. Montresor A (2004) A robust protocol for building superpeer overlay topologies. In: *Fourth International Conference on Peer-to-Peer Computing*, pp. 202–209
 45. Meng W, Wang X, Liu S Distributed load sharing of an inverter-based microgrid with reduced communication. *IEEE Trans Smart Grid*. doi:10.1109/TSG.2016.2587685



Iran

Nahid Amirazodi received the B.S. and M.S. degrees in computer engineering from Islamic Azad University of Qazvin in 2011 and 2014, respectively. Her research interests include peer-to-peer networks, distributed systems, artificial intelligence, learning automata, reinforcement learning, parallel algorithms, and soft computing.

Faculty of Computer and Information Technology Engineering, Qazvin Branch, Islamic Azad University, Qazvin,



Ali Mohammad Saghiri received the B. Sc. and M. Sc. degrees in computer engineering in Iran, in 2008 and 2010, respectively. He is currently the Ph.D. student of computer engineering in AmirKabar University of Technology, Tehran, Iran. His research interests include peer-to-peer networks, distributed systems, artificial intelligence, learning automata, reinforcement learning, parallel algorithms, and soft computing.

Soft Computing Laboratory, Computer Engineering and Information Technology Department, Amirkabir University of Technology (Tehran Polytechnic), 424 Hafez Ave., Tehran, Iran



Mohammad Reza Meybodi received the B.S. and M.S. degrees in Economics from Shahid Beheshti University in Iran, in 1973 and 1977, respectively. He also received the M.S. and Ph.D. degree from Oklahoma University, USA, in 1980 and 1983, respectively in Computer Science. Currently, he is a full professor in Computer Engineering Department, Amirkabir University of Technology, Tehran, Iran. Prior to current position, he worked

from 1983 to 1985 as an assistant professor at Western Michigan University, and from 1985 to 1991 as an associate professor at Ohio University, USA. His research interests include wireless networks, fault tolerant systems, learning systems, parallel algorithms, soft-computing and software development.

Soft Computing Laboratory, Computer Engineering and Information Technology Department, Amirkabir University of Technology (Tehran Polytechnic), 424 Hafez Ave., Tehran, Iran