

# An Adaptive Algorithm for Super-Peer Selection Considering Peer's Capacity in Mobile Peer-to-Peer Networks based on Learning Automata

Nahid Amirazodi <sup>a</sup>, Ali Mohammad Saghiri <sup>b</sup>, Mohammadreza Meybodi <sup>b</sup>

<sup>a</sup> Department of Electrical, Computer and IT Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran

<sup>b</sup> Department of Computer Engineering and Information Technology, Amirkabir University of Technology, Tehran, Iran

## 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 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, 8]- [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 WPAN(802.15.4) [13],[12] 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 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, 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 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 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 [7] [4], 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 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. 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 a adaptive mechanism for managing the peers role whit 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  $\beta$  is the set of inputs,  $\alpha$  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  $a_i(k) \in \alpha$  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  $a_i(k)$  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]:

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

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

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] [37] [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 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.

#### A. 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, 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, 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.

#### B. 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 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 comedown to AOP role. Otherwise,  $p$  remains in super-peer role and examines members in its local view set again.

#### 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  $p$

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

```

Assumptions:
Rp:  $p$ 's region
01  $C = \min(\text{capacity}(s) - \text{load}(s), \text{load}(p))$ 
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      $R = \text{pick highest capacity peer from } P.\text{HigherSet}$ 
13     If  $R$  is a USP
14       DoTransfer( $P$ ,  $R$ )
15   End If

```

Fig. 2. DoTransfer ( $p, s$ ) method

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:

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

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

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

27 End If

(b)	super-peers algorithm
	Assumptions: P: local super-peer , Rp: p's region X , S: remote peer in P's region
01	P performs learning algorithm according to Fig. 2
02	If P.OLALG = super
03	P forms LowerSet
04	While P.Lowerset is not empty and P is USP
05	X = pick lowest capacity peer from P.LowerSet
06	If X is UOP
07	Send AttachProposeMesge to X
08	End While
09	While Receive AttachRequestMessage from X and P is US
10	If P.OLALG = super
11	X becomes ordinary-peer of P
12	End While
13	Else If P.OLALG = ordinary-peer
14	While Load(P) > 0 and P.HigherSet 0is not empty
15	S = pick highest cpacity peer from P.HigherSet
16	If S is a USP
17	DoTransfer(P, S)
18	Else if S is UOP or AOP
19	Send PromoteProposeMessage to S
20	End If
21	End While
22	End If
23	End if

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

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 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  $L_{ReP}$ . Reward and penalty parameters of learning automaton respectively is equal to  $a=0.5$ ,  $b=0.2$ . Peers are located in a  $1000m \times 1000m$  network area. The departure and arrival of peers from/to the network area are governed by a Poisson distribution. Parameter  $\lambda$  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 12sec 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  $\alpha$  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.

**Table 1** Simulation parameters

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

To evaluate the result of experiment, one way ANOVs and tuku statistical tests are used. Result of statistical tests are provided in Appendix 1 and its analysis has been added to each experiment.

#### A. 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).

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

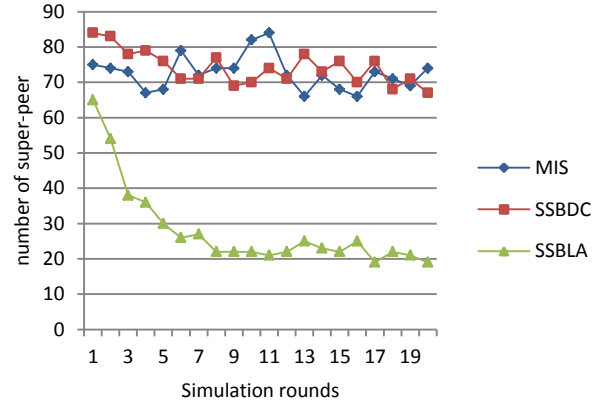
ALL parameters have been assigned under section 5 and Table 1.

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 percent. 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 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.



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

According to Appendix 1, Table 1 about Fig.4, column value of the P- value, assuming equal means of protocols cannot be accepted. Therefore protocols significantly are different from each other. According to Appendix 1, Table 2, MIS and SSBDC have same function and SSBLA are significantly different from other protocol.

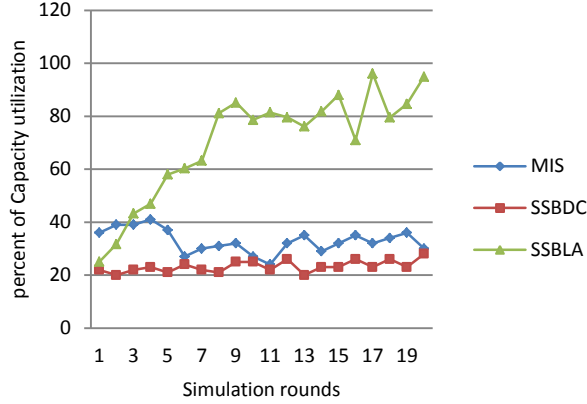


Fig. 5. Capacity utilization

According to the results presented in Appendix 1, Table 1 about Fig.5, protocols significantly are different. Upon the Appendix 1, Table 2, all three protocols are significantly different and none of them are similar to each other.

## B. 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.

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.

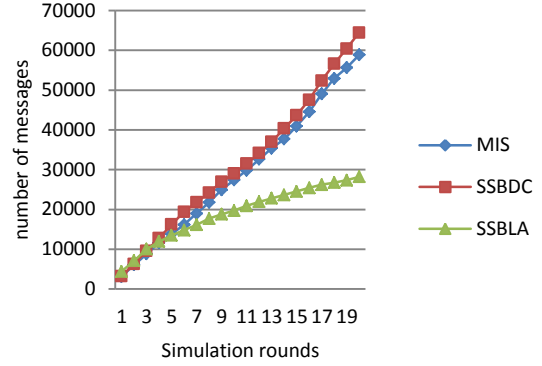


Fig. 6. Total Overhead

Considering the presented results in Appendix 1, Table 1 about Fig.6, protocols significantly are different. Based on Appendix 1, Table 1, SSBDC and MIS are similar to each other and SSBLA and SSBDC have significant differences.

## C. 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 Fig. 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. Fig. 7 shows the number of remaining super-peers and Fig. 8 shows the percentage of unattached-ordinary-peers respectively. According to Fig. 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.

Fig. 9 show number of super-peers in the scenario in which 50% of super-peers removed at round 10. Fig. 10 and 11 respectively show effect of crash on the number of unattached-ordinary-peers and percent of Capacity utilization. According to Fig. 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.

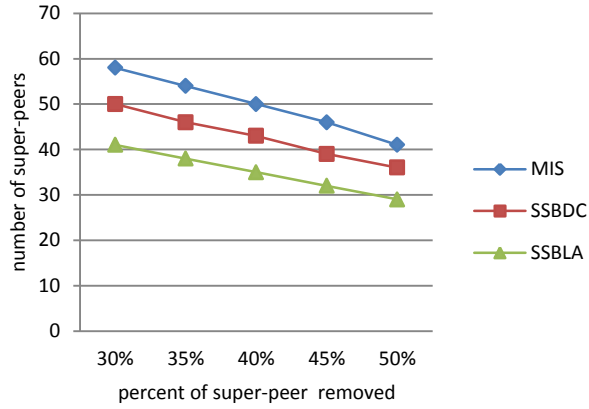


Fig. 7. number of super-peers in the face of super-peers' failure

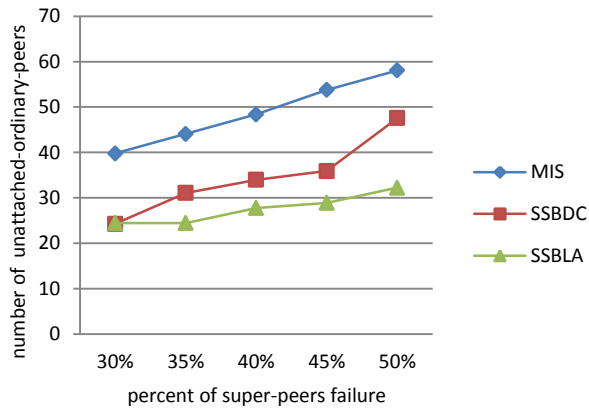


Fig. 8. number of unattached-ordinary-peers in the face of super-peers' failure

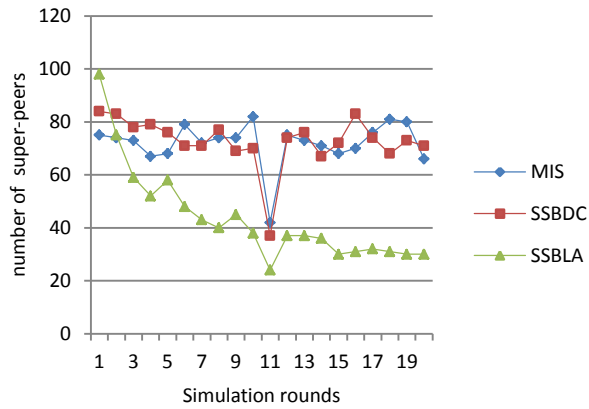


Fig. 9. number of super-peers in the face of 50% super-peers' failure

According to the results presented in Appendix 1, Table 1 about Fig.9, tree protocols significantly are different. With considering Appendix 1, Table 2, MIS resembles SSBDC and there is a significant difference between SSBLA with these two protocol. Fig. 9. Also confirms these results.

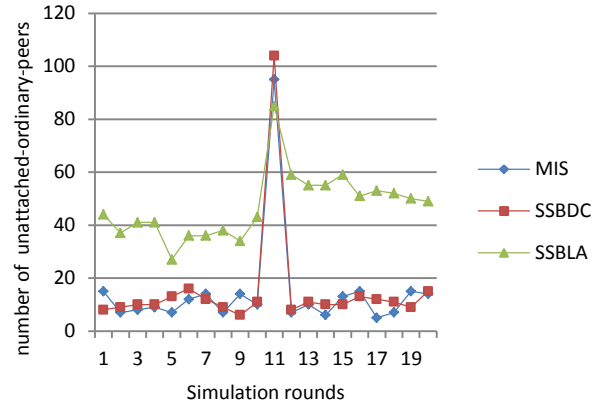


Fig. 10. number of unattached-ordinary-peers in the face of 50% super-peers' failure

Based on the results presented in Appendix 1, Table 1 about Fig.10, tree protocols significantly are different. With considering Appendix 1, Table 2, MIS resembles SSBDC and there is a significant difference between SSBLA with these two protocol.

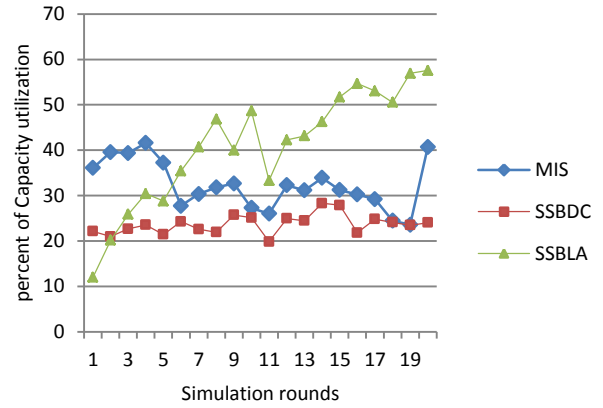


Fig. 11. Capacity utilization in the face of 50% super-peers' failure

As shown in the Appendix 1, Table 1 about Fig.10, this protocols significantly are different. Considering Appendix 1, Table 2, they doesn't have any meaningful resemblance with each other.

#### D. 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 Fig. 12 to Fig. 15 are shown, utilizing learning indicates significant improvement over evaluation parameters. As shown in Figure 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 Figure 6, the overhead of algorithm with learning still has been lower than overhead of other algorithms. But using learning, as shown in Figure 12, 13, 14, with lower number of super-peer cover more peer and achieve greater capacity utilization.

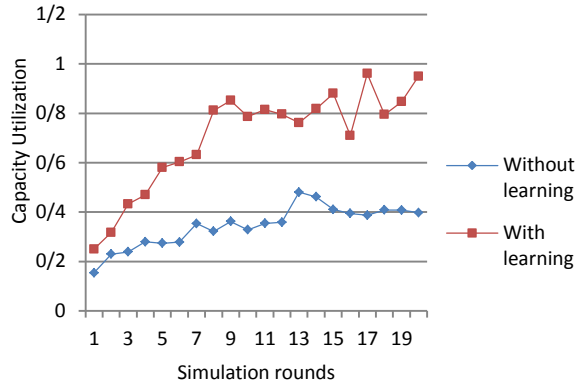


Fig. 12. impact of learning on capacity utilization in SSBLA

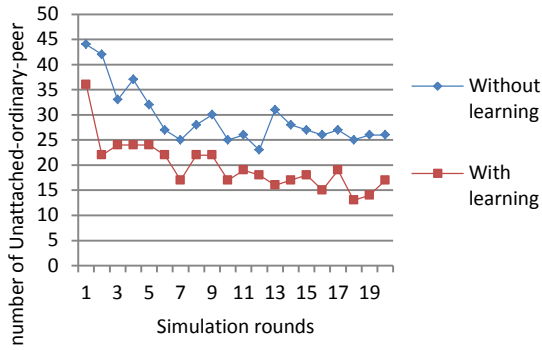


Fig. 13. impact of learning on unattached-ordinary-peer in SSBLA

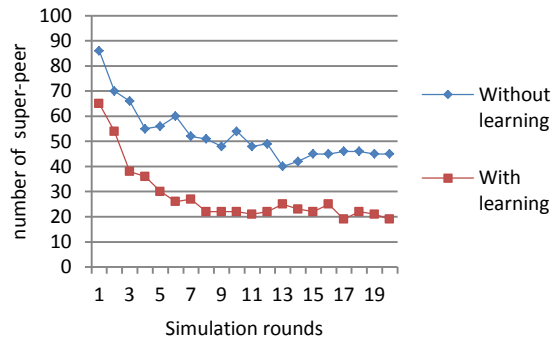


Fig. 14. impact of learning on super-peer in SSBLA

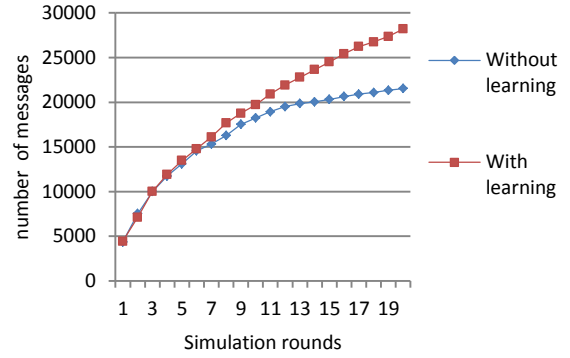


Fig. 15. impact of learning on overhead in SSBLA

#### E. 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.

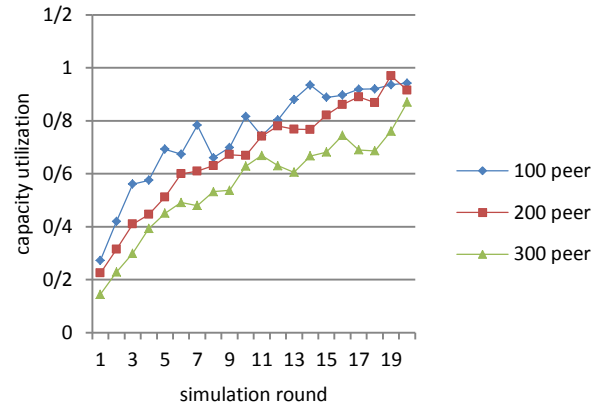


Fig. 16. impact of peer number on capacity utilization in SSBLA

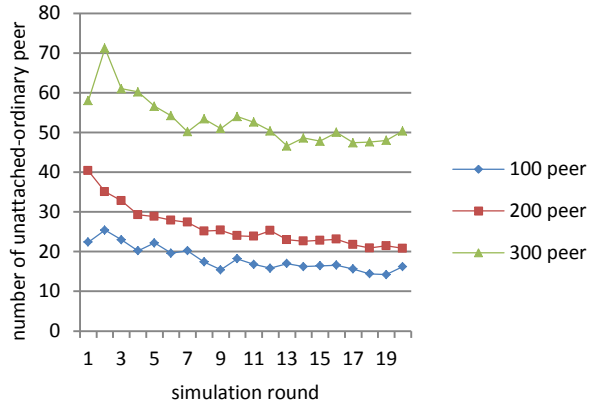


Fig. 17. impact of peer number on unattached-ordinary-peer in SSBLA

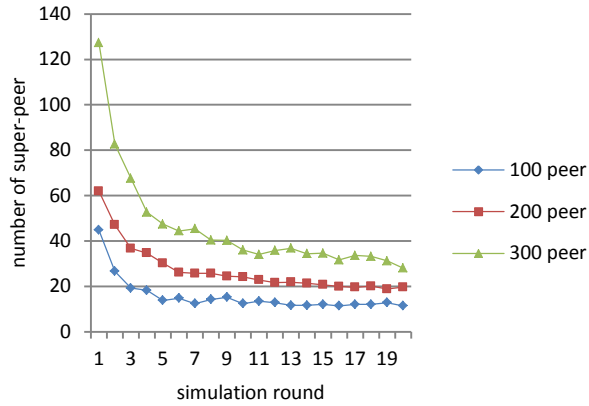


Fig. 18. impact of peer number on super-peer in SSBLA

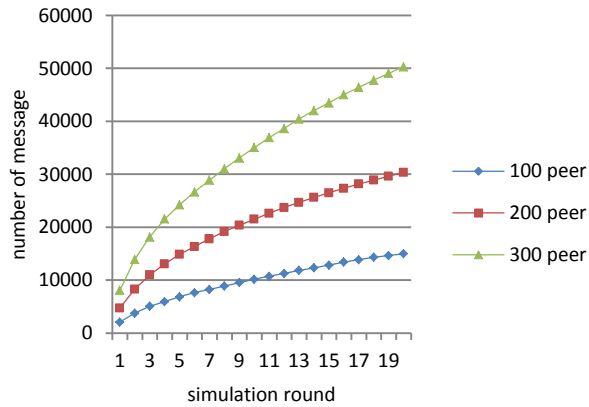


Fig. 19. impact of peer number on overhead in SSBLA

## F. 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/sec

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/sec is more than others. overhead in 5~10 and 0~5 is close to each other.

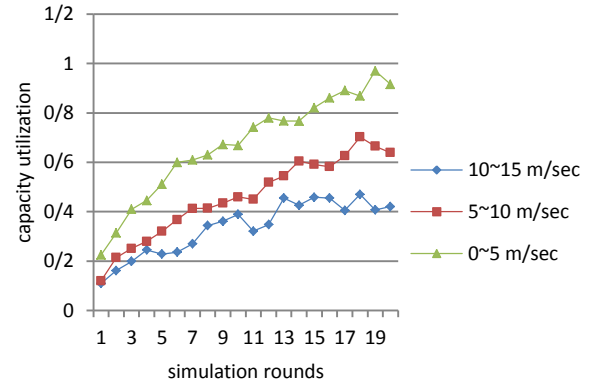


Fig. 20. impact of peer speed on capacity utilization in SSBLA

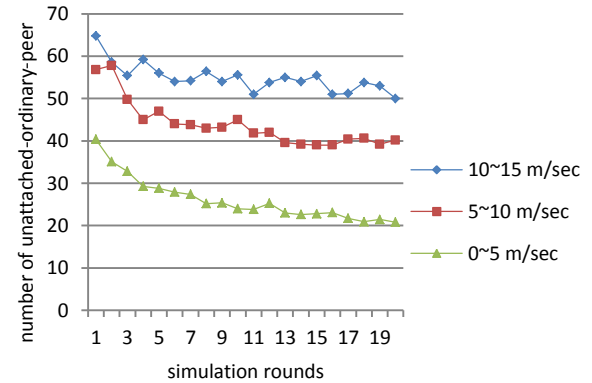


Fig. 21. impact of peer speed on unattached-ordinary-peer in SSBLA

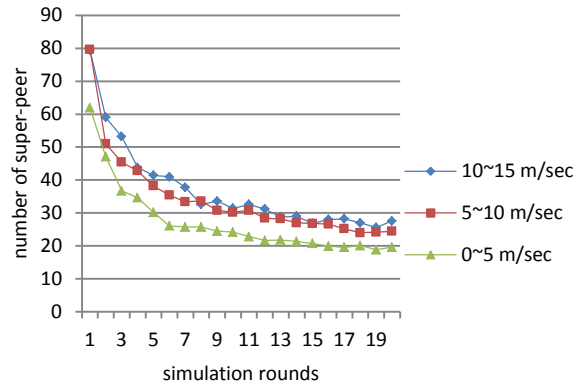


Fig. 22. impact of peer speed on super-peer in SSBLA

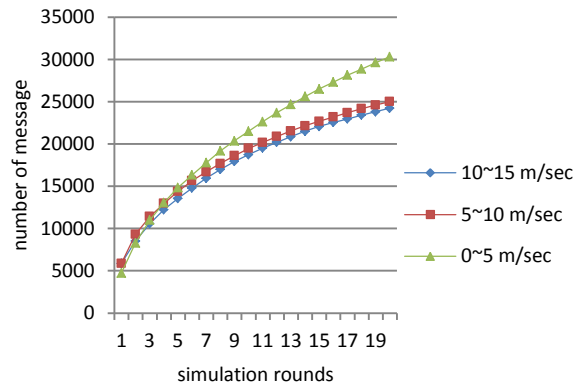


Fig. 23. impact of peer speed on overhead in SSBLA

### G. 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. Fig. 24, 25, 26, show this concept.

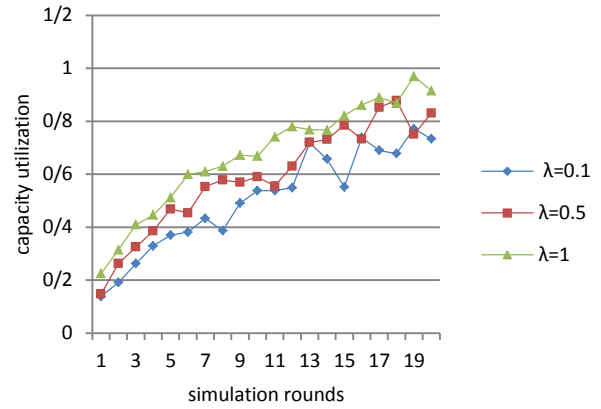


Fig. 24. impact of arrival rates on capacity utilization in SSBLA

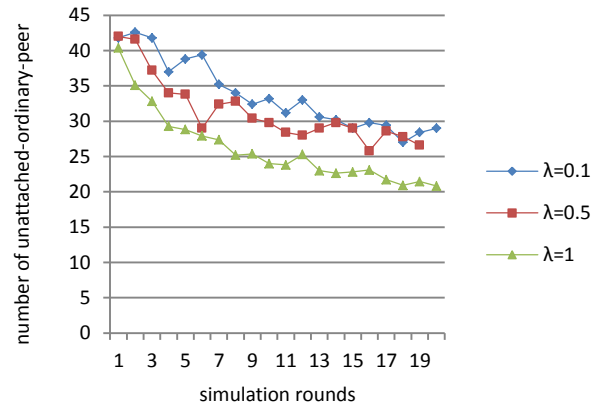


Fig. 25. impact of arrival rates on unattached-ordinary-peer in SSBLA

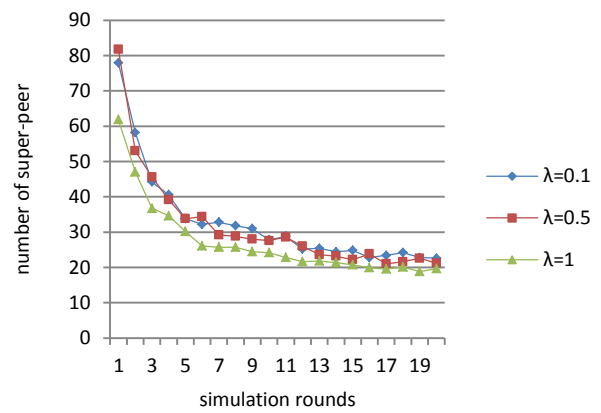


Fig. 26. impact of arrival rates on super-peer in SSBLA

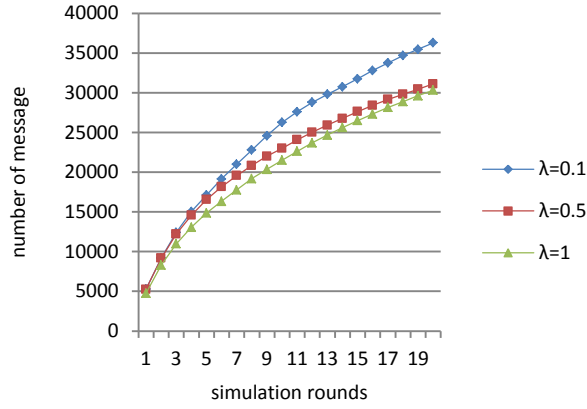


Fig. 27. impact of arrival rates on overhead in SSBLA

#### H. 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, 31, LRP model results are not satisfactory with different values of 1.0 to 9.0.  $L_{REP}$  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.

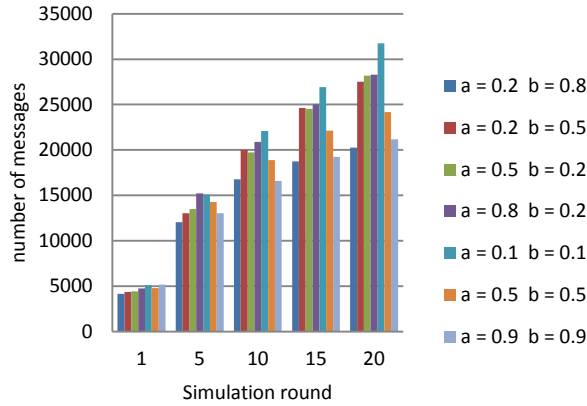


Fig. 28. impact of Learning Automata reward and penalty parameters on overhead in SSBLA

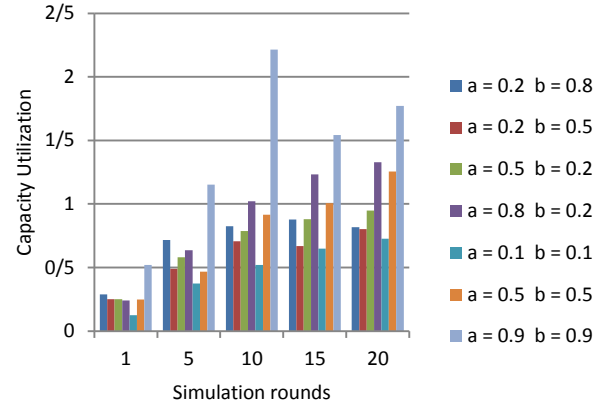


Fig. 29. impact of Learning Automata reward and penalty parameters on capacity utilization in SSBLA

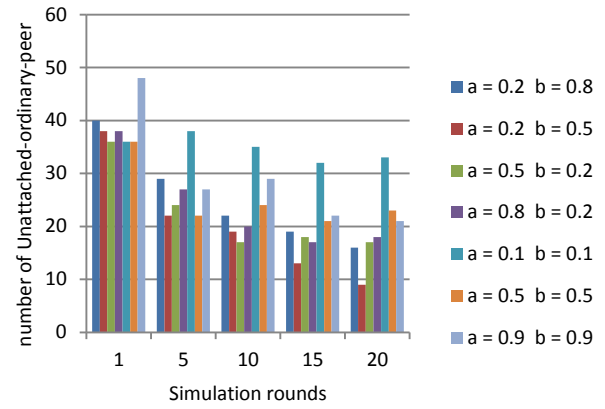


Fig. 30. impact of Learning Automata reward and penalty parameters on unattached-ordinary-peer in SSBLA

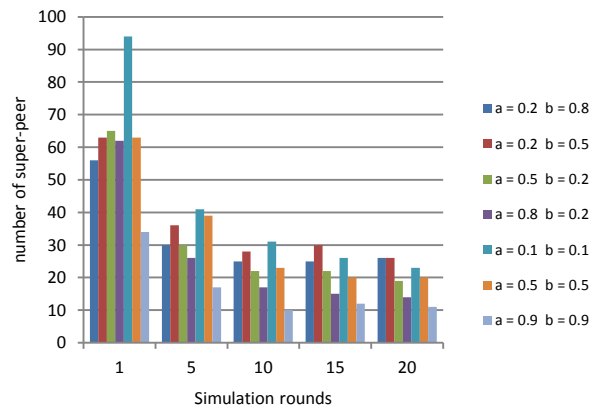


Fig. 31. impact of Learning Automata reward and penalty parameters on super-peer in SSBLA

## 5. 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].

## References

- [1] K. Elgazzar, W. Ibrahim, S. Oteafy, and H. S. Hassanein, "RobP2P: A Robust Architecture for Resource Sharing in Mobile Peer-to-Peer Networks," presented at the The 4th International Conference on Ambient Systems, Networks and Technologies (ANT 2013), 2013.
- [2] F. A. Guo and J. B. Xu, "Searching Methods of MP2P Based on Diffusion Strategy of Resources Index," *Computer Network and Information Security*, vol. 3 pp. 26-33, 2011.
- [3] S. Gholami, M. R. Meybodi, and A. M. Saghir, "A Learning Automata-based Version of SG-1 Protocol for Super-peer Selection in Peer-to-Peer Networks," in *the 10th International Conference on Computing and Information Technology (IC2IT2014)*, 2014, pp. 189-201.
- [4] L. Bin, X. Y. Huan, Z. Yan, C. B. Ting, and W. Jian, "The Algorithm of Super-peer Selection Based on Dynamic Performance in Mobile Peer-to-Peer Network," presented at the EICE2012, Macau, China, 2012.
- [5] S. Androutsellis-Theotokis and D. Spinellis, "A survey of peer-to-peer content distribution technologies," *ACM Computing Surveys (CSUR)*, vol. 36, pp. 335-371 December 2004.
- [6] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A Survey and Comparison of Peer-to-Peer Overlay Network Schemes," *IEEE Communications Surveys and Tutorials*, pp. 72-93, MARCH 2005.
- [7] J. S. Han, K. J. Lee, J. W. Song, and S. B. Yang, "Mobile Peer-to-Peer systems using Super Peers for Mobile Environments," presented at the The International Conference on Information Networking 2008, Busan, Korea 2008.
- [8] J.-H. Kim, J.-W. Song, T.-H. Kim, and S.-B. Yang, "An Enhanced Double-Layered P2p System For The Reliability In Dynamic Mobile Environments," *Ad Hoc & Sensor Wireless Networks*, vol. 30, p. 467, June 2011.
- [9] A. M. Mahdy, J. S. Deogun, and a. J. Wang, "A Dynamic Approach for the Selection of Super Peers in Ad Hoc Networks " presented at the the Sixth International Conference on Networking Martinique, France, 2007.
- [10] P. M. d. S. Henriques, "A Lightweight Distributed Super Peer Election Algorithm for Unstructured Dynamic P2P Systems," Master of Electrical Engineering and Computer, Department of Electrical Engineering, the Faculty of Sciences and Technology New University of Lisbon, 2011.
- [11] F. H. P. Fitzek and H. Charaf, "Introduction to WLAN IEEE802.11 Communication on Mobile Devices," in *Mobile Peer to Peer (P2P): A Tutorial Guide* ed: John Wiley & Sons, 2009.
- [12] J. H. Schiller, *Mobile Communications*, 2nd ed.: Addison-Wesley, 2003.
- [13] S. Okdem, "A cross-layer adaptive mechanism for low-power wireless personal area networks," *Computer Communications*, 2015.
- [14] K. Najim and A. S. Poznyak, "Learning automata: theory and application " in *Tarrytown*, 1994.
- [15] K. S. N. M. A. L. Thathachar, *Learning automata: An introduction*: Prentice-Hall, 1989.
- [16] M. C. Castro, A. J. Kessler, C.-F. Chiasserini, C. Casetti, and I. Korpeoglu, "Peer-to-Peer Overlay in Mobile Ad-hoc Networks," in *Handbook of Peer-to-Peer Networking*, X. Shen, H. Yu, J. Buford, and M. Akon, Eds., ed New York: Springer, 2010, pp. 1045-1080.
- [17] B. C. Seet, "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, 2005.
- [18] B. Tang, Z. Zhou, A. Kashyap, and T. c. Chiuieh, "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 2005.
- [19] M. Gerla and J. T. C. Tsai, "Multicliuser, mobile, multimedia radio network," *Wireless Networks*, vol. 1, pp. 255-265, 1995.
- [20] T. I. Wang, K. H. Tsai, and Y. H. Lee, "Crown: An Efficient and Stable Distributed Resource Lookup Protocol Distributed Resource Lookup Protocol," presented at the International Conference EUC 2004, Aizu-Wakamatsu, Japan, 2004.
- [21] Y. WEI, G. XIE, and Z. LI, "A Hierarchical Cross-Layer Protocol for Group Communication in MANET," presented at the Telecommunications and Malaysia International Conference on Communications, Penang, Malaysia, 2007.
- [22] S. K. Kim, K. J. Lee, and S. B. Yang, "An enhanced super-peer system considering mobility and energy in mobile environments," in *6th International Symposium on Wireless and Pervasive Computing (ISWPC)*, 2011, pp. 1-5.
- [23] K.-J. LEE, J.-H. CHOI, and S.-B. YANG, "Fuzzy Inference-based Super Peer Selection for a Practical Double-layered Mobile Peer-to-peer System," *Ad Hoc & Sensor Wireless Networks*, vol. 21, pp. 327-351, May 15 2012.
- [24] P. Merz, M. Priebe, and S. Wolf, "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*, 2008, pp. 385-390.
- [25] J. A. Torkestani and M. R. Meybodi, "An intelligent backbone formation algorithm for wireless ad hoc networks based on distributed learning automata,"

- Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 54, no. 5, pp. 826–843, 2010.
- [26] S. Safavi, M. R. Meybodi, and M. Esnaashari, "Learning automata based face-aware Mobicast," *Wireless Personal Communications*, vol. 77, pp. 1923–1933, 2014.
- [27] S. Misra, S. S. Chatterjee, and M. Guizani, "Stochastic learning automata-based channel selection in cognitive radio/dynamic spectrum access for WiMAX networks," *International Journal of Communication Systems*, vol. 28, pp. 801–817, 2015.
- [28] P. V. Krishna, S. Misra, D. Joshi, A. Gupta, and M. S. Obaidat, "Secure socket layer certificate verification: a learning automata approach," *Security and Communication Networks*, vol. 7, pp. 1712–1718, 2014.
- [29] N. Kumar and J.-H. Lee, "Collaborative-learning-automata-based channel assignment with topology preservation for wireless mesh networks under QoS constraints," *Systems Journal, IEEE*, vol. 9, pp. 675–685, 2015.
- [30] N. Kumar, J.-H. Lee, and J. J. Rodrigues, "Intelligent mobile video surveillance system as a bayesian coalition game in vehicular sensor networks: learning automata approach," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, pp. 1148–1161, 2015.
- [31] N. Kumar, S. Misra, M. Obaidat, J. Rodrigues, and B. Pati, "Networks of learning automata for the vehicular environment: a performance analysis study," *Wireless Communications, IEEE*, vol. 21, pp. 41–47, 2014.
- [32] N. Kumar, S. Misra, and M. S. Obaidat, "Collaborative learning automata-based routing for rescue operations in dense urban regions using vehicular sensor networks," *Systems Journal, IEEE*, vol. 9, pp. 1081–1090, 2015.
- [33] A. M. Saghir and M. R. Meybodi, "A distributed adaptive landmark clustering algorithm based on mOverlay and learning automata for topology mismatch problem in unstructured peer-to-peer networks," *International Journal of Communication Systems*, 2015.
- [34] A. M. Saghir and M. R. Meybodi, "A Self-adaptive Algorithm for Topology Matching in Unstructured Peer-to-Peer Networks," *Journal of Network and Systems Management*, pp. 1–34, 2015.
- [35] A. M. Saghir and M. R. Meybodi, "An approach for designing cognitive engines in cognitive peer-to-peer networks," *Journal of Network and Computer Applications*, vol. 70, pp. 17–40, 2016.
- [36] M. Polatoglou, P. Nicopolitidis, and G. I. Papadimitriou, "On low-complexity adaptive wireless push-based data broadcasting," *International Journal of Communication Systems*, vol. 27, pp. 194–200, 2014.
- [37] P. Nicopolitidis, C. Chrysostomou, G. I. Papadimitriou, A. Pitsillides, and A. S. Pomportsis, "On the efficient use of multiple channels by single-receiver clients in wireless data broadcasting," *International Journal of Communication Systems*, vol. 27, pp. 513–520, 2014.
- [38] P. Nicopolitidis, "Performance fairness across multiple applications in wireless push systems," *International Journal of Communication Systems*, vol. 28, pp. 161–166, 2015.
- [39] S. Misra, P. V. Krishna, V. Saritha, and M. S. Obaidat, "Learning automata as a utility for power management in smart grids," *Communications Magazine, IEEE*, vol. 51, pp. 98–104, 2013.
- [40] M. Hasanzadeh and M. R. Meybodi, "Distributed optimization Grid resource discovery," *The Journal of Supercomputing*, vol. 71, pp. 87–120, 2015.
- [41] H. Morshedlou and M. R. Meybodi, "Decreasing impact of sla violations: a proactive resource allocation approach for cloud computing environments," *Cloud Computing, IEEE Transactions on*, vol. 2, pp. 156–167, 2014.
- [42] A. Varga and R. Hornig, "An overview of the OMNeT++ simulation environment," in *the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*.
- [43] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "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)*, 1998, pp. 85–97.
- [44] A. Montresor, "A Robust Protocol for Building Superpeer Overlay Topologies," in *Fourth International Conference on Peer-to-Peer Computing*, 2004, pp. 202–209.
- [45] W. Meng, X. Wang, and S. Liu, "Distributed Load Sharing of an Inverter-Based Microgrid with Reduced Communication," *IEEE Transactions on Smart Grid*, DOI: 10.1109/TSG.2016.2587685

## Appendix 1

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 1** one way ANOVs test

ANOVA						
	method					
		Sum of Squares	df	Mean Square	F	Sig.
Fig 4	Between Groups	27412.433	2	13706.217	215.501	.000
	Within Groups	3625.300	57	63.602		
	Total	31037.733	59			
Fig 5	Between Groups	24255.633	2	12127.817	84.002	.000
	Within Groups	8229.350	57	144.375		
	Total	32484.983	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	10929.433	2	5464.717	33.811	.000
	Within Groups	9212.750	57	161.627		
	Total	20142.183	59			
Fig10	Between Groups	13939.300	2	6969.650	21.514	.000
	Within Groups	18465.300	57	323.953		
	Total	32404.600	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 2** tucky test

**Multiple Comparisons**

		method					
		Tukey HSD					
	(I) Allscenario	(J) Allscenario	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval	
						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
		SSBDC	46.65000000*	3.79966527	.000	37.5064165	55.7935835
Fig6	MIS	SSBDC	-2370.10000000	4738.46128182	.872	-1.3772819E4	9032.6192591
		SSBLA	10407.30000000	4738.46128182	.080	-995.4192591	21810.0192591
	SSBDC	MIS	2370.10000000	4738.46128182	.872	-9032.6192591	13772.8192591
		SSBLA	12777.40000000*	4738.46128182	.025	1374.6807409	24180.1192591
	SSBLA	MIS	-10407.30000000	4738.46128182	.080	-2.1810019E4	995.4192591
		SSBDC	-12777.40000000*	4738.46128182	.025	-2.4180119E4	-1374.6807409
Fig 9	MIS	SSBDC	-.65000000	4.02028846	.986	-10.3244951	9.0244951
		SSBLA	28.30000000*	4.02028846	.000	18.6255049	37.9744951



	SSBDC	MIS	.65000000	4.02028846	.986	-9.0244951	10.3244951
		SSBLA	28.95000000*	4.02028846	.000	19.2755049	38.6244951
	SSBLA	MIS	-28.30000000*	4.02028846	.000	-37.9744951	-18.6255049
		SSBDC	-28.95000000*	4.02028846	.000	-38.6244951	-19.2755049
Fig 10	MIS	SSBDC	-.85000000	5.69168368	.988	-14.5465710	12.8465710
		SSBLA	-32.75000000*	5.69168368	.000	-46.4465710	-19.0534290
	SSBDC	MIS	.85000000	5.69168368	.988	-12.8465710	14.5465710
		SSBLA	-31.90000000*	5.69168368	.000	-45.5965710	-18.2034290
	SSBLA	MIS	32.75000000*	5.69168368	.000	19.0534290	46.4465710
		SSBDC	31.90000000*	5.69168368	.000	18.2034290	45.5965710
Fig 11	MIS	SSBDC	8.75000000*	2.52471989	.003	2.6744682	14.8255318
		SSBLA	-8.55000000*	2.52471989	.004	-14.6255318	-2.4744682
	SSBDC	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