Peer-to-Peer Networking and Applications
https://doi.org/10.1007/s12083-018-0672-y

Asynchronous neighbor discovery with unreliable link in wireless mobile networks

Wei Li[1] **· Jianhui Zhang**[1] **· Feilong Jiang**[1] **· Zhi Li**[1] **· Chong Xu**[1]

Received: 7 September 2017 / Accepted: 27 July 2018

Abstract

In wireless mobile networks, neighbor discovery is fundamental to many useful applications. The limited energy of mobile devices stresses the need for effective and energy-saving asynchronous neighbor discovery protocols. The neighbor discovery would fail due to some uncontrollable factors such as hardware errors or sudden interruptions, which are considered as the unreliable link in this paper. Existing works do not take the unreliable link into consideration and the performances with unreliable link can still be improved. In this paper, we assume a certain probability that unreliable link would happen, and design a novel deterministic Quorum System (QS)—E-grid(k) QS and a novel probabilistic QS—Plain(k) QS and propose two algorithms based on these two QSs to solve the asynchronous neighbor discovery problem in wireless mobile networks with unreliable link. Extensive simulations are conducted to evaluate our algorithms. We use the cumulative distribution function (CDF) of the discovery latency and the Valid Overlapped Time Slots (VOTS) of QS in the evaluation. Simulation results show that Plain(k) and E-grid(k) QSs outperform most existing neighbor discovery protocols in both P2P model and clique model with unreliable or reliable link.

Keywords Neighbor discovery · Wireless mobile network · Quorum system · Unreliable link

1 Introduction

Nowadays, with more people having their own smartphones, different usage demands of smartphones rapidly grow in daily life and the communication among neighboring smartphones in wireless mobile networks has become increasingly important. For example, one may want to share his/her travel photos with his/her families via smartphone by using Zapya, an application designed to enable file sharing among smartphones. Another example is that people can shake their smartphones to get information about shops around them using WeChat. In order to enable communication among neighboring smartphones, a crucial process is how can a smartphone discover its neighboring devices, i.e. smartphones within communication range. Although central servers can help with solving this problem, these applications are able to reach greater performance by using local neighbor discovery protocols. Firstly, local neighbor discovery protocols can be used anytime while the central servers may be unavailable. Secondly, the connection between smartphones and the central server may encounter several problems, such as the delay, unexpected interruption, and weak signal. Also, detecting neighbors locally costs less money and energy. In local neighbor discovery protocols, the time of each smartphone can be different with others.

The asynchronous neighbor discovery problem is fundamental in wireless sensor networks and wireless mobile networks [8, 22]. It can be applied in many useful applications [5, 13, 25, 26]. To solve this problem and save the energy of smartphones, many neighbor discovery protocols are designed [1, 4, 6, 11, 12, 14, 15, 17–20, 22]. In these protocols, the time is divided into equal-length time slots, some of which are active time slots while others are idle time slots. Only in the active time slots, smartphones are able to find other active smartphones. Existing protocols can be divided into two groups: deterministic protocols [1, 4, 6, 11, 12, 14, 15, 17, 19, 20, 22] and probabilistic protocols [18]. The deterministic protocols establish a

This article is part of the Topical Collection: *Special Issue on Network Coverage*
Guest Editors: Shibo He, Dong-Hoon Shin, and Yuanchao Shu

✉ Jianhui Zhang
jh_zhang@ieee.org

[1] College of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou 310018 China

Published online: 13 October 2018

pattern to schedule the periodical operations of each smartphone when performing neighbor discovery. Many deterministic protocols are designed based on QSs. Gird [15] and Torus [12] are the two basic ones, which are very simple and far from the optimal quorum size. U-connect [11], Disco [6] are prime-based protocols and can improve the worst-case discovery latency. Searchlight [1] changes the regular relationship between two smartphones and performs much better in reducing discovery latency. Cyclic [14], Code-base [19] are designed based on the difference of sets in combinatorics and can reach the optimal quorum size. Nihao [20], Integer and Non-integer [4], ALOHA-like [22], Panda [17] present novel models to improve the effectiveness. Besides deterministic protocols, there are many works focusing on probabilistic protocols [21, 23]. The most representative one is the Birthday protocol [18], whose performance is stable.

However, all the existing protocols mentioned above don't take the unreliable link among smartphones into consideration. The unreliable link is caused by some uncontrollable factors, such as hardware errors and sudden interruptions, which would lead to the failure of neighbor discovery. To solve this problem, this paper aims at designing protocols dealing with unreliable link in wireless mobile networks based on QSs. There are two reasons for designing protocols based on QSs. Firstly, by the rotation closure property and intersection property of QS, the successful connectivity of a whole network is guaranteed and it does not require time synchronization among devices [28]. Secondly, although many other techniques can be used to design a distributed and asynchronous protocol, QS is simple to implement and easy to deploy in the practical environment.

In this paper, we devise a deterministic protocol E-grid(k) and a probabilistic protocol Plain(k) to solve the neighbor discovery problem with unreliable link. To evaluate our protocols, we adopt not only the CDF and the worst-case of discovery latency, which are utilized in many existing works, but also a new metric: the VOTS to estimate the protocols, which measures the average neighbor discovery ability of the protocols.

We make the following contributions in this paper:

- To the best of our knowledge, this is the first work that solves the asynchronous neighbor discovery problem with unreliable link.
- We design a novel deterministic QS—E-grid(k) QS based on the Grid QS [15] and a neighbor discovery protocol E-grid(k). The parameter k can be adjusted to meet the actual demand.
- We design a novel probabilistic QS—Plain(k) QS and a neighbor discovery protocol Plain(k). We also prove that any probabilistic QS constructed by a special way and satisfies the intersection property can also be used to solve the neighbor discovery problem.
- We evaluate our algorithms by both theoretical analysis and simulation. Experiment results show that our protocols outperform existing protocols when there are unreliable or reliable links in wireless mobile networks.

The rest of this paper is organized as follows. Section 2 introduces the definition of QS, model, and some notations. Section 3 proposes the construction and properties of three QSs. We design and analyze two quorum-based algorithms in Section 4. Section 5 provides the simulation results of these algorithms. In Section 6, we present some related works. We conclude the whole paper in Section 7.

2 System model and assumption

This paper considers the wireless mobile network with smartphone set D, where $D = \{u_0, \cdots, u_{n-1}\}$ and u represents a smartphone. Each smartphone can only discover smartphones within its communication range. This paper divides the time period T of a smartphone into m time slots, i.e. $T = \{\tau_0, \cdots, \tau_{m-1}\}$ where τ is the time slot. In order to reduce energy consumption, each smartphone wakes up in some time slots, called the active time slots, and sleeps in the remaining time slots, called the idle time slots, in each period. A smartphone sends beacons at the beginning of each active time slot and listens to other smartphones' beacons during the time slot. In each idle time slot, the smartphone does not receive or send beacons and consumes little energy. Usually, smartphones may not be able to discover their neighbors because of hardware errors, transmission interruptions and so on. So we have to consider unreliable links in the wireless mobile network when solving the neighbor discovery problem, even though these smartphones start or stop the discovery randomly. For example, the one who tries to share travel photos with his/her parents. The parents may be quite far from him/her and the wireless signal is not stable. We hope our protocols can help he/she achieve the goal even though they may start the application asynchronously. In this paper, we solve this problem by designing quorum-based protocols.

2.1 Quorum system

2.1.1 Deterministic QS

Definition 1 (**Deterministic Quorum System**) Given a universal set $U = \{0, \ldots, n-1\}$, a deterministic QS Ω_D under U is a collection of non-empty subsets of U, each called a quorum Q, which satisfies the intersection property: $\forall Q_a, Q_b \in \Omega_D : Q_a \cap Q_b \neq \emptyset$ [9].

For example, $\Omega_{Da} = \{\{0\}, \{0, 1\}, \{0, 2\}\}$ is a QS under $U = \{0, 1, 2\}$ and there are three quorums in Ω_{Da}: $Q_a = \{0\}$, $Q_b = \{0, 1\}$ and $Q_c = \{0, 2\}$. A quorum can be rotated. The rotation of a quorum Q is $R(Q, i) = \{(j + i) \bmod n \mid j \in Q\}$, where i is a non-negative integer and Q is in a QS Ω_D under $U = \{0, \ldots, n-1\}$. For instance, if $i = 2$ and the quorum is Q_b, the rotation of Q_b is $R(Q_b, i) = \{0, 2\}$.

The *rotation closure property* of a QS Ω_D under $U = \{0, \ldots, n-1\}$ is: $\forall Q_a, Q_b \in \Omega_D, i \in \{0, \ldots, n-1\} : Q_a \cap R(Q_b, i) \neq \emptyset$. For instance, QS $\Omega_{Da} = \{\{0\}, \{0, 1\}, \{0, 2\}\}$ under $U = \{0, 1, 2\}$ does not have the *rotation closure property* because $\{0, 2\} \cap R(\{0\}, 1) = \emptyset$. Another QS $\Omega_{Db} = \{\{0, 1\}, \{0, 2\}, \{1, 2\}\}$ under $U = \{0, 1, 2\}$ has the *rotation closure property*.

2.1.2 Probabilistic QS

Probabilistic QSs Ω_P are quite similar with deterministic QSs. The only difference between probabilistic QSs and deterministic QSs is that each quorum pair satisfies the intersection property with a probability in probabilistic QSs but must satisfy the intersection property in deterministic QSs. We define the probability of each quorum pair satisfying the intersection property in probabilistic QSs as P_o

$$\forall Q_a, Q_b \in \Omega_P : P_o = \mathbb{E}[sgn(|Q_a \cap Q_b|)] = \sum_i p_i \cdot sgn(|Q_a^i \cap Q_b^i|), \qquad (1)$$

where $sgn(x)$ is the sign function and $sgn(x) = 0$ when $x = 0$, $sgn(x) = -1$ when $x < 0$ and $sgn(x) = 1$ when $x > 0$. $sgn(|Q_a \cap Q_b|)$ represents whether Q_a and Q_b have overlapped elements. Then we use Q_a^i to represent the i^{th} possible case of Q_a and p_i to represent the probability that the i^{th} case appears.

Definition 2 (**Probabilistic Quorum System**) Given a universal set $U = \{0, \ldots, n-1\}$, a probabilistic QS Ω_P is a collection of subsets of U, where the probability of each quorum pair in Ω_P satisfying the intersection property is at least P_o [16].

The P_o of each quorum pair in a probabilistic QS should be the same because we use the same method to construct the quorums in a probabilistic QS. We give an example to show this point. Let $U = \{0, 1, 2, 3\}$ and we construct a probabilistic QS Ω_{Pa} consisting of three quorums, each of the three quorums is constructed by randomly selecting two elements from U. Each quorum pair in this QS has $\binom{4}{2} \cdot \binom{4}{2} = 36$ possible cases. They have overlapped elements in $\binom{4}{2} \cdot 5 = 30$ cases and do not have overlapped elements in $\binom{4}{2} \cdot \binom{2}{2} = 6$ cases. The possible cases are the same for each quorum pair. Then we have $P_o = \frac{5}{6} \cdot 1 + \frac{1}{6} \cdot 0 = \frac{5}{6}$.

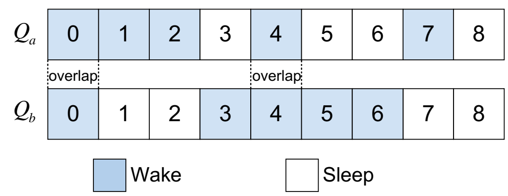

Fig. 1 Synchronous quorums

2.2 Problem formulation

As described above, a time period T of a smartphone is divided into m equal-length time slots, denoted by $T = \{\tau_0, \cdots, \tau_{m-1}\}$. Smartphones wake up in a few time slots and sleep in other time slots so as to save energy. In this paper, we employ QS to determine at which slots should smartphones wake up and perform neighbor discovery. We construct a QS Ω based on T, which corresponds to the universal set U. That is, the elements of each quorum are active time slots of a smartphone. When performing neighbor discovery, each smartphone selects a quorum Q from Ω and wakes up at time slots in Q and sleeps at other time slots during each period T.

If two smartphones can discover each other, their quorums must overlap. The time slots of the two smartphones may be synchronous or asynchronous. When their time slots are synchronous, they can communicate at the overlapped time slots. For example, as shown in Fig. 1, $Q_a = \{0, 1, 2, 4, 7\}$, $Q_b = \{0, 3, 4, 5, 6\}$, $Q_a \cap Q_b = \{0, 4\}$. The two smartphones can communicate at time slots 0 and 4. When their time slots are asynchronous, that is, each smartphone has a different time shift $t_i \in \mathbb{N}^+$. The time shift can be regarded as the time slots of one smartphone are rotated. Thus, the QS should have the *rotation closure property* so that the two smartphones can communicate. For example, as shown in Fig. 2, $Q_a = \{0, 1, 2, 4, 7\}$, $Q_b = \{0, 3, 4, 5, 6\}$, $R(Q_a, 1) \cap Q_b = \{3, 5\}$ under $U = \{0, \ldots, 8\}$.

This paper considers a pair of smartphones as neighbors when they are respectively in the communication range of each other. The link between each pair of smartphones may be unreliable due to some uncontrollable reasons (e.g., network interruption). And we assume the probability

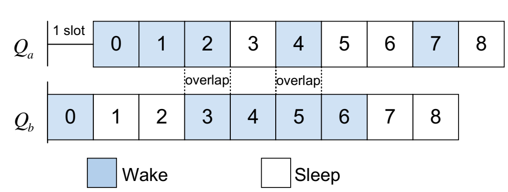

Fig. 2 Asynchronous quorums

that unreliable link appears is P_u. Therefore, the neighbor discovery problem can be described as: constructing a QS so that each pair of smartphones have a sufficiently large probability P to discover each other at least once in each time period T. Because the time slots of two smartphones may be synchronous or asynchronous, we prove that deterministic QSs having the *rotation closure property* and probabilistic QSs having the *intersection property* can be employed in neighbor discovery.

Lemma 1 *If Ω_D satisfy the rotation closure property, Ω_D is a solution to the neighbor discovery problem* [10].

Corollary 1 *If Ω_P satisfy the intersection property ($P_o > 0$) and the construction method of quorums is randomly selecting k elements, this Ω_P is also a solution to the neighbor discovery problem.*

Proof According to the construction method, Ω_P contains all the possible combinations of randomly selecting k elements from m elements. The rotations of Q_i are $Q_j \in \Omega_P$ in such QS. That is $\forall Q_i \in \Omega_P, t \in \mathbb{N}^+ : R(Q_i, t) = Q_j \in \Omega_P$. When smartphones are in an asynchronous situation, each smartphone has a time shift t_i from the real time and the quorums have to be rotated. But according to the aforementioned property of Ω_P, the rotations of quorums are still in Ω_P. Thus the influence of time shift is removed and the asynchronous situation can transformed into the synchronous situation. When smartphones are in a synchronous situation, any two quorums of smartphones have at least the probability P_o to have overlapped elements according to the intersection property in an Ω_P. Each pair of smartphones always can have overlapped time slots and solve the neighbor discovery problem. □

In order to evaluate the QS theoretically, we first introduce the Expected Quorum Overlap Sizes (EQOS) [9].

Definition 3 (**EQOS**) For a quorum system Ω, its expected quorum overlap sizes is

$$E_o = \sum_{Q_a, Q_b \in \Omega} P(Q_a)P(Q_b)|Q_a \cap Q_b|, \quad (2)$$

where $P(Q_a)$ and $P(Q_b)$ are the probability of quorums Q_a and Q_b selected by two smartphones respectively.

EQOS is an average neighbor discovery ability measurement to estimate the expected number of beacons received in each round [9]. When the wireless mobile network is a P2P model, i.e. there are only two smartphones, EQOS is calculated by Eq. 2. But when the wireless mobile network is a clique model, i.e. there are more than two smartphones, the collision of the beacons from the different neighbors cannot be ignored because there may be more than two smartphones are awake. We thus propose the Valid Overlapped Time Slots (VOTS).

Definition 4 (**VOTS**) For a quorum system Ω, its valid overlapped time slots E_v is the number of time slots where smartphones can achieve a successful discovery.

The VOTS only considers the overlapped time slots where the beacons from different neighbors do not collide. For a P2P network, the VOTS is equal to the EQOS. For a clique network, the EQOS is an upper bound of the VOTS. Because the time is asynchronous and the number of neighbors is large, the cases of the beacons are very complex. Thus, the VOTS is very hard to be calculated by theoretical analysis. In this paper, we get the value by extensive experiments. The VOTS is the average results of all experiments for each Quorum System. Note that the value of the VOTS of a specific Quorum System is constant, which is determined by its property. Thus, we can use a statistical method to calculate the value.

Most existing works use metrics such as worst-case discovery latency, CDF and so on to estimate the performance of a quorum-based method. Worst-case discovery latency and CDF only show the worst situation and the cumulation of discovery latency. Because the time slots are asynchronous among all smartphones and the number of neighbors is large, the cases of the beacons are very complex. The difference between the best and the worst case can be huge. For example, maybe there is no collision in the best case but the beacons from different neighbors may collide in every time slot in the worst case. According to the definition of the VOTS and the way we calculate the VOTS, VOTS can show the average ability of discovering neighbors [9]. In this paper, we propose the novel metric VOTS/QS $= \gamma = \frac{E_v}{|Q|}$ to evaluate QSs. It should be as large as possible and can be used as a metric to evaluate the performances of a Quorum System. The neighbor discovery problem in this paper is formulated as follows and some notations in this paper are shown in Table 1.

$$\begin{aligned}
&Objective: \\
&\text{Construct a QS } \Omega \text{ which has the maximal } \gamma \quad (3) \\
&Subject\ to: \\
&\forall Q_a, Q_b \in \Omega, 1 > P_u \geq 0, t_a \in \mathbb{N}^+ \\
&: P((R(Q_a, t_a) \cap Q_b) \neq \emptyset) > 0 \quad (4)
\end{aligned}$$

3 Quorum system construction

This section proposes two kinds of QSs, E-grid(k) QS and Plain(k) QS. We introduce their definitions, construction methods and properties respectively.

Table 1 General notations

Symbol	Description	Symbol	Description
U	Universal set	L	The length of the square array
D	Smartphone set	E_v	The valid overlapped time slots
u	Smartphone	P_o	Pr of quorum pair having overlap
T	Time period	γ	The E_v ratio
t	Time shift	P_u	Pr of unreliable link appearing
τ	Time slot	E_o	Expected quorum overlap sizes
Ω	Quorum System	$R(Q, i)$	The rotation of a quorum
Q	Quorum	$sgn(x)$	Sign function

Note: Pr means Probability

3.1 Grid quorum system

The E-grid(k) QS and Plain(k) QS are based on Grid QS, so this paper introduces the Grid QS firstly. Given an integer L, Grid QS $\Omega_G = \{Q_0, \ldots, Q_{L^2-1}\}$ is under the universal set $U = \{0, 1, \ldots, L^2 - 1\}$, where $|Q_0| = |Q_1| =, \ldots, = |Q_{L^2-1}| = 2L - 1$. To construct Ω_G, we firstly divide each time period T into $L * L$ time slots and assign them to the $L * L$ grids of the Grid QS. Each Grid quorum is constructed by randomly selecting a main element from the grids first, and all elements in the row and column of the main element compose a Grid quorum as shown in Fig. 3.

Lemma 2 *The Grid QS satisfies the rotation closure property* [28].

3.2 E-grid(k) quorum system

Definition 5 (**E-grid**(k) **Quorum System**) Given integers L, k with $1 \leq k \leq L$, the grids are generated by the universal set $U = \{0, 1, \ldots, L^2 - 1\}$ and each E-grid(k) quorum is constructed by picking k different main diagonal elements $[x_1, x_1], \ldots, [x_k, x_k]$ from the grids firstly, where $1 \leq x_k \leq L$. Then all the elements of the k "Grid quorums" corresponding to the k main diagonal elements compose an E-grid(k) quorum.

For example, as shown in Fig. 4, there is an E-grid(2) quorum under $U_a = \{0, 1, \ldots, 48\}$ and the two main diagonal elements are 16([3, 3]), 32([5, 5]). The quorum consists of all the elements in column 3, 5 and row 3, 5.

Lemma 3 *The E-grid(k) QS also satisfies the rotation closure property.*

Proof Since any E-grid(k) quorum is a super set of Gird quorums it contains, the lemma holds. □

3.3 Plain(k) quorum system

Definition 6 (**Plain**(k) **Quorum System**) Given integers L, k with $1 \leq k \leq L$, each quorum of Plain(k) QS Ω_P

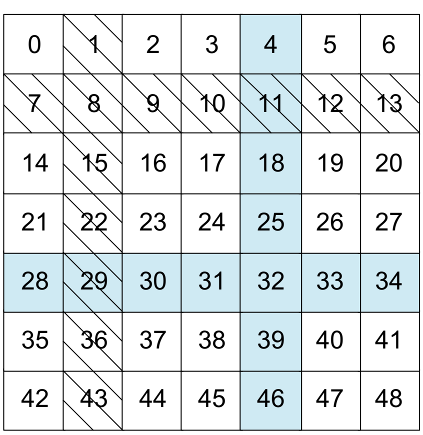

Fig. 3 There are two Gird quorums. One is indicated by blue while the other is with oblique line

Fig. 4 The two main diagonal elements of the E-grid(2) quorum are 16([3, 3]), 32([5, 5])

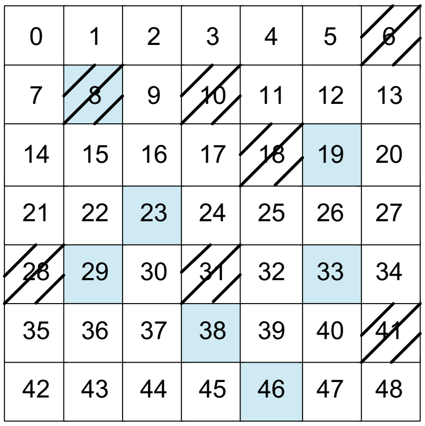

Fig. 5 There are two Plain(1) quorums that they have a overlapped time slot. One is indicated by blue while the other is with oblique line

is constructed by randomly picking kL different elements from the $L * L$ grids generated by the universal set $U = \{0, 1, \ldots, L^2 - 1\}$.

In Plain(k) QS, the grids of time slots are assigned by the same way as the Grid QS does. An example of Plain(k) QS is presented in Fig. 5. There are two Plain(1) quorums. One is Q_a={6, 8, 10, 18, 28, 31, 41} and the other is Q_b={8, 19, 23, 29, 33, 38, 46}. They have an overlapped time slot 8.

Lemma 4 *In Plain(k) QSs, the probability that each pair of quorums satisfies the intersection property is given as the following equation.*

$$P_o \geq 1 - e^{-k^2} \tag{5}$$

Proof Firstly, we give a property of combinatorial mathematics [16]

$$\text{For integers } n, c, \text{ and } i, \text{ we have } \frac{\binom{n-c}{c-i}}{\binom{n}{c}} \leq \left(\frac{c}{n}\right)^i \left(\frac{n-c}{n-i}\right)^{c-i} \tag{6}$$

According to Eq. 6,

$$P_o = 1 - \frac{\binom{L^2-kL}{kL}}{\binom{L^2}{kL}} \geq 1 - \left(\frac{L^2-kL}{L^2}\right)^{kL} \geq 1 - e^{-\frac{kL}{L^2}kL} = 1 - e^{-k^2} \tag{7}$$

□

Lemma 4 indicates that Plain(k) QSs satisfy the intersection property well and P_o is sufficiently large even though k is quite small. For instance, when $k = 2$, $P_o = 1 - e^{-4} \approx 0.982$ and $k = 3$, $P_o = 1 - e^{-9} \approx 0.999$.

4 Neighbor discovery algorithm

In this section, we design two distributed neighbor discovery algorithms to solve asynchronous neighbor discovery problem based on E-grid(k) QS and Plain(k) QS.

4.1 Algorithm construction

Algorithm 1 E-grid(k) Neighbor Discovery Algorithm

```
Input: Universal set U and parameter k, L.
Output: Neighbor set S_n of smartphone u and
        collision set S_c.
τ_i ← τ_0; S_n ← Ø;
Construct an E-grid(k) QS Ω_E;
Randomly select an E-grid(k) quorum Q_u for u;
for τ_i from τ_0 to τ_{L^2−1} do
    if τ_i ∈ Q_u then
        S_c ← Ø;
        while τ_i do
            Send a message Msg(u);
            Listen to other smartphones' messages;
            if u receives Msg(s) then
                S_c ← S_c ∪ {s};
            end
        end
        if |S_c| = 1 then
            S_n ← S_n ∪ S_c;
        end
    end
end
return S_n;
```

In the E-grid(k) neighbor discovery algorithm, as shown in Algorithm 1, the main part is to construct the QS. The first step is to set an initialization, then smartphone u constructs an E-grid(k) QS and randomly selects an E-grid(k) quorum Q_u. This selected quorum determines the time schedule of u. u is awake during active time slots and asleep during idle time slots. In the second step, u sends message $Msg(u)$ to other smartphones and listens to neighbors' messages $Msg(s)$ while it is awake and does nothing while it is asleep. If there are more than one smartphone, the collision of the beacons from the different neighbors cannot be ignored. Thus we use a collision set S_c to include the number of received messages during each active time slot. If $|S_c| > 1$, u cannot discover any neighbors due to the collision of beacons and if $|S_c| = 1$, u can discover the neighbor. This algorithm runs one period and returns a set S_n of the neighbors. Plain(k) neighbor discovery algorithm, as shown in Algorithm 2, is quite similar with the E-grid(k) algorithm. The only difference between these two algorithms is the QS they use.

Algorithm 2 Plain(k) Neighbor Discovery Algorithm

Input: Universal set U and parameter k, L.
Output: Neighbor set S_n of u and collision set S_c.

$\tau_i \leftarrow \tau_0$; $S_n \leftarrow \emptyset$;
Construct a Plain(k) QS Ω_{Pl};
Randomly select a Plain(k) quorum Q_u for u;
for τ_i *from* τ_0 *to* τ_{L^2-1} **do**
  if $\tau_i \in Q_u$ **then**
    $S_c \leftarrow \emptyset$;
    while τ_i **do**
      Send a message $Msg(u)$;
      Listen to other smartphones' messages;
      if *u receives Msg(s)* **then**
        $S_c \leftarrow S_c \cup \{s\}$;
      end
    end
    if $|S_c| = 1$ **then**
      $S_n \leftarrow S_n \cup S_c$;
    end
  end
end
return S_n;

4.2 Performance analysis

In this part, we analyze the performances of the proposed algorithms theoretically when the link is reliable and the network is a P2P model. We conduct simulations to show the performances when the network is a clique model or the link is unreliable as shown in Section 5. Also, we will show the performances of E_v through simulations. The E-grid(k) neighbor discovery algorithm is analyzed at first.

Theorem 1 *The quorum size $|Q|$ of an E-grid(k) quorum is $-k^2 + 2Lk$.*

Proof Suppose L is a positive integer and when $k = 1$, $|Q|$ is $2L - 1$. Let $k = 2, 3$, $|Q|$ is $4L - 4$, $6L - 9$. Thus, we assume $|Q|$ is $-k^2 + 2Lk$ and prove it by mathematical induction. When $k = 1$, $|Q|$ is $2L - 1$. Suppose $k = n$, $|Q|$ is $-n^2 + 2Ln$. When $k = n + 1$, the new "Grid quorum" has two overlapped time slots with each remaining "Grid quorums". So $|Q|$ is $-n^2 + 2Ln + 2L - 1 - 2n = -(n + 1)^2 + 2L(n + 1)$ and the theorem holds. □

Theorem 2 *In terms of the E-grid(k) QS, E_o is*

$$\sum_{i=max(0,2k-L)}^{k} \frac{\binom{k}{i}\binom{L-k}{k-i}}{\binom{L}{k}}(f(i) + 2(k - i)^2),$$

where $f(i) = -i^2 + 2Li$ and $L \geq k$.

Proof Suppose Q_a and Q_b are two quorums randomly selected from an E-grid(k) QS. We can get that Q_a and Q_b have i "Grid quorums" in common, where i is limited. If $2k > L$, Q_a and Q_b must have at least $2k - L$ common "Grid quorums", i.e. $2k - L \leq i \leq k$. If $2k \leq L$, Q_a and Q_b may have no common "Grid quorum", i.e. $0 \leq i \leq k$. Thus, we have $max(0, 2k - L) \leq i \leq k$.

When there are i common "Grid quorums", there are $-i^2 + 2Li$ (i.e. $f(i)$) overlapped time slots, according to Theorem 1. There are $k - i$ "Grid quorums" not in common for Q_a, each of them having two overlapped time slots with the other $k - i$ "Grid quorums" in E-grid(k) quorum Q_b. That is $2(k - i)^2$ overlapped time slots and the total amount of overlapped time slots is $f(i) + 2(k - i)^2$.

Suppose Q_a is randomly determined at first. If Q_a and Q_b have i common "Grid quorums", the occurrences of i common "Grid quorums" in Q_b are $\binom{k}{i}$. Then Q_b only can select $k-i$ not common "Grid quorums" from $T \setminus Q_a$ and the occurrences are $\binom{L-k}{k-i}$. There are $\binom{L}{k}$ possible permutations of Q_b in total. Therefore, the probability of Q_a and Q_b having i common "Grid quorums" is $\frac{\binom{k}{i}\binom{L-k}{k-i}}{\binom{L}{k}}$. Then the theorem holds. □

Lemma 5 *The EQOS E_o of Plain QS is*

$$\sum_{i=max(0,2kL-L^2)}^{kL} i\frac{\binom{kL}{i}\binom{L^2-kL}{kL-i}}{\binom{L^2}{kL}} \tag{8}$$

Proof At first, we consider the situation that Q_a and Q_b have i overlapped time slots in Ω_{Pl}. There are two cases: when $2kL \geq L^2$, Q_a and Q_b must have $2kL - L^2$ overlapped time slots; when $0 < 2kL < L^2$, Q_a and Q_b may not have overlapped time slots. Thus, $\max(0, 2kL - L^2) \leq i \leq kL$. Then we calculate the probability that Q_a and Q_b having i overlapped time slots. First, Q_a can be selected from the grids randomly. Then i overlapped time slots in Q_b only can be selected from Q_a. The occurrences of i overlapped time slots in Q_b are $\binom{kL}{i}$. Then $kL - i$ not overlapped time slots in Q_b only can be selected from $T \setminus Q_a$ and the occurrences are $\binom{L^2-kL}{kL-i}$. If there is no constraint for Q_b, the occurrences of Q_b are $\binom{L^2}{kL}$. Thus, the probability P of Q_a and Q_b having i overlapped time slots is $\binom{kL}{i}\binom{L^2-kL}{kL-i}/\binom{L^2}{kL}$. E_o of Plain(k) QS is the sum of $i * P$ for all feasible i. Then the lemma holds. □

5 Experiment results

We evaluate our protocols by simulation and consider both the P2P and clique networks. In the P2P case, there is only one smartphone within each smartphone's communication

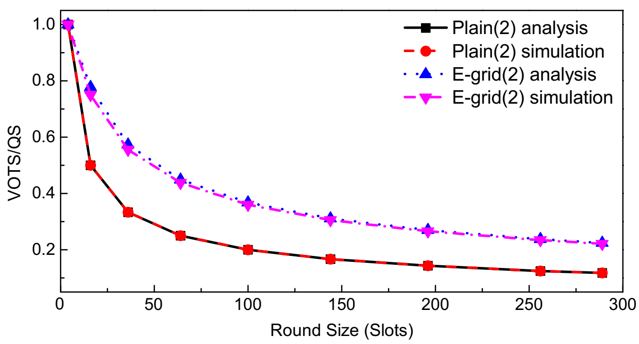

Fig. 6 Comparison between simulation and analysis with reliable link

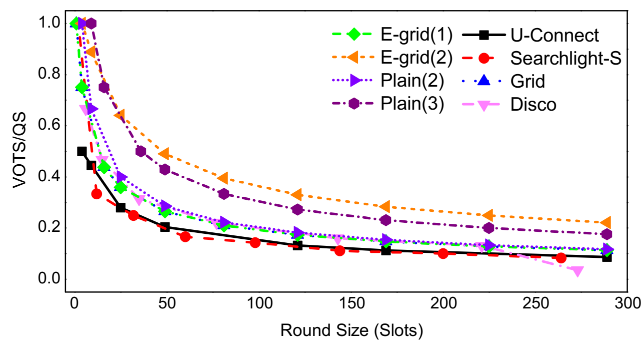

Fig. 7 Performances of VOTS/QS under eight different QSs with reliable link

range. In the clique case, there are several neighbors within each smartphone's communication range. Each smartphone sends a beacon at the beginning of the active time slot and the beacon interval has a length of 50ms. In order to utilize the unreliable link in simulations, we firstly generate a uniform distribution from 0 to 1 and then randomly get a number from this distribution. If the number is smaller than the threshold, the link is unreliable. If not, the link has no problem. For comparison, we implement quorum-based neighbor discovery protocols such as Grid, U-Connect, Searchlight-S, Disco protocols and Birthday protocol which is not a quorum-based protocol. In this section, we firstly compare VOTS/QS in theoretical analysis and simulation for E-grid(k) QS and Plain(k) QS. Then we compare all QSs mentioned above by VOTS/QS and CDF when the link is reliable and unreliable respectively. We present the trend of VOTS/QS with parameter k changing in E-grid(k) QS and Plain(k) QS with reliable link. These simulations are P2P cases and we also conduct simulations in a clique model. We compare all QSs by VOTS/QS and CDF with the link is reliable and unreliable respectively in the clique model.

In addition, when trying to get CDF or VOTS/QS, we have to traverse all the possible initialization combinations and calculate these cases. However, in lots of cases, the number of overall different combinations is too large to traverse. For instance, when the round size of Plain(1) QS is 289 in P2P simulations, the amount of all possible permutations is $\binom{289}{7}$. Instead, we calculate them by randomly selecting the initialization combinations sufficient times (e.g., 10^7 or 10^8) and using the mean of them.

5.1 P2P simulation

Figure 6 demonstrates the simulation result and the theoretical analysis result of E-grid(2) QS and Plain(2) QS. The round size of E-grid(2) QS and Plain(2) QS are from 4 to 289. For these two QSs, we can see the simulation result and theoretical result are almost the same.

Figure 7 shows VOTS/QS results of QSs. Compared with U-Connect, Searchlight-S, Grid and Disco QSs, Plain(3) QS achieves more VOTS/QS by 54.3%, 41.2%, 24.6% and 31.3% in average and 114.3%, 127.3%, 80.2% and 60.9% at most. E-grid(2) QS can improve VOTS/QS by 84.5%, 69.6%, 48.9% and 56.9% in average and 128.4%, 166.6%, 77.7% and 60.7% at most. Both Plain(k) QS and E-grid(k) QS make great improvement compared with other QSs.

Figure 8 shows VOTS/QS of eight different QSs with probability $P_u = 0.1$. The results show that Plain(k) QS reaches high VOTS/QS than other QSs when the link is unreliable. VOTS/QS of Plain(2) QS is 5.56, 4.25, 1.88 and 6.74 times that of U-Connect, Searchlight-S, Grid and Disco QSs in average cases. Overall, E-grid(1), E-grid(2), Grid, U-Connect, Searchlight-S, Disco QSs perform quite equally and not effectively. Plain(2) and Plain(3) QSs have great advantages in this case. The gap between the former six protocols and Plain(k) protocol gets smaller when round size becomes larger.

Figure 9 shows VOTS/QS results of Plain(k) QSs with different k from 1 to 5. Note that the Plain(k) QS with higher k has better VOTS/QS performances but also we should pay attention to the energy consumption. The average VOTS/QS of Plain(5) QS is 2.58 times that of Plain(1), 1.71 times that of Plain(2), 1.34 times that of Plain(3) and 1.13 times that of Plain(4). As shown in Fig. 9, when k is smaller, the VOTS/QS performances of Plain(k) QS is more easily

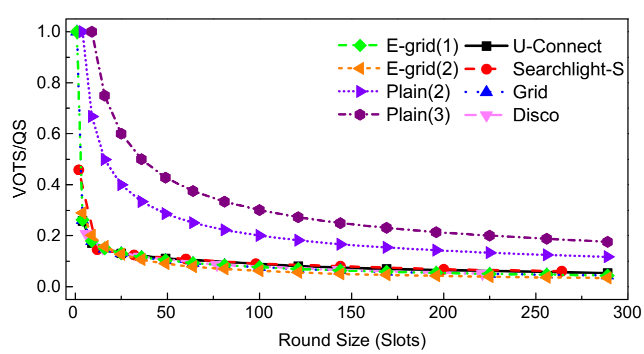

Fig. 8 Performances of VOTS/QS under eight different QSs with unreliable link

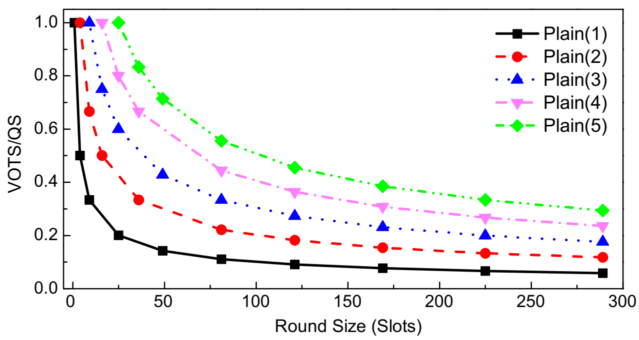

Fig. 9 Plain(k) QSs with different k and reliable link

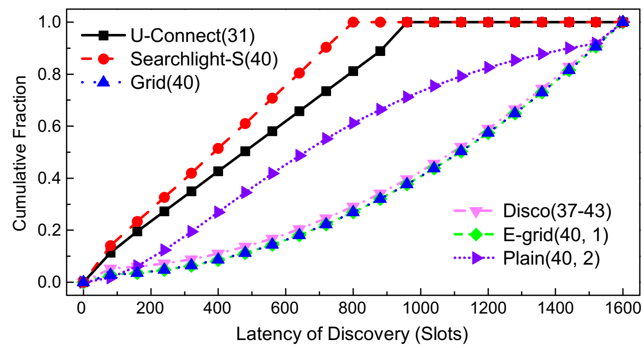

Fig. 11 P2P CDF performance with six different QSs and reliable link

affected by round size. Thus, we usually select a large k to guarantee a steady performance.

Figure 10 presents VOTS/QS performances of E-grid(k) QSs with different k from 1 to 5. An E-grid(k) QS with higher k has better VOTS/QS performances but also we should pay attention to the energy consumption. The average VOTS/QS of E-grid(5) QS is 2.34 times that of E-grid(1), 1.57 times that of E-grid(2), 1.26 times that of E-grid(3) and 1.10 times that of E-grid(4). Also, as shown in Fig. 10, when k is smaller, the VOTS/QS performances of E-grid(k) QS is more easily affected by round size. Thus, we usually select a large k to guarantee a steady performance.

Figure 11 involves the CDF performances of six different QSs. The pair of prime numbers in Disco are (37, 43), the prime number of U-Connect is 31, the probing period of Searchlight-S is 40 time slots. The side length and k in E-grid(k) are (40, 1) and in Plain(k) are (40, 2). As shown in Fig. 11, Searchlight-S QS has the best CDF performances and E-grid(k), Disco, Grid QSs have the same worst CDF performances. They have the same duty cycle 5%. The curve tendency of Plain(k) QS is different from others, which means a plenty of cases are large discovery latency in Plain(k) protocol. The minimum worst-case latency of E-grid(40, 1) and Plain(40, 1) are 1600 time slots, which is 4 times that of Searchlight-S. Plain(40, 1) QS reaches the third place overall and the result is better than that of Disco and Grid QS. In a conclusion, the P2P CDF performances of Plain(k) and E-grid(k) QSs are worse than that of U-Connect and Searchlight-S QSs.

Figure 12 shows the CDF performances of six different QSs with same duty cycle 5% and $P_u = 0.3$. The result is quite different from that of Fig. 11. Especially, Plain(40, 2) QS performs better when there exists unreliable link. In Fig. 11, discovery latency of U-Connect is always smaller than that of Plain(k) QS. However, median discovery latency of Plain(k) QS is greater than that of U-Connect in this Figure. It can be explained by the random construction method of Plain(k) QS. The randomness of construction offsets the influence of P_u in a certain degree. Plain(k) QS makes the second place overall, which is prior to its performances in Fig. 11. Then other QSs do not have significant changes.

5.2 Clique simulation

Besides P2P simulations, we also conduct clique simulations, where there is more than one neighbor within each smartphone's communication range. In this scenario, we evaluate QSs by VOTS/QS and CDF. The discovery latency in clique model is an amount of time slots for a smartphone to discover m different neighbors. With many neighbors, the beacons from different neighbors may conflict at the smartphone A resulting in smartphone A cannot deal with any of

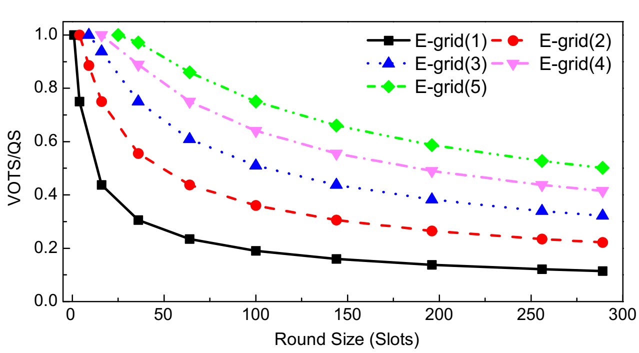

Fig. 10 E-grid(k) QSs with different k and reliable link

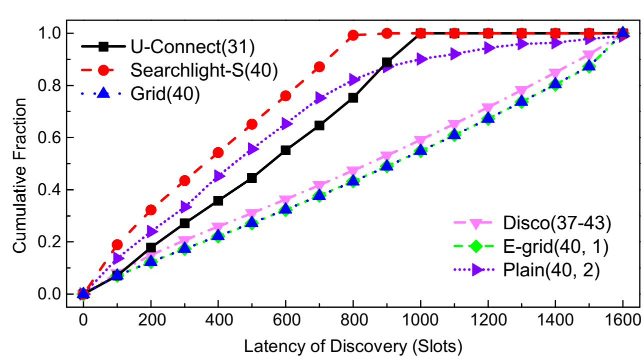

Fig. 12 P2P CDF performance with six different QSs and unreliable link

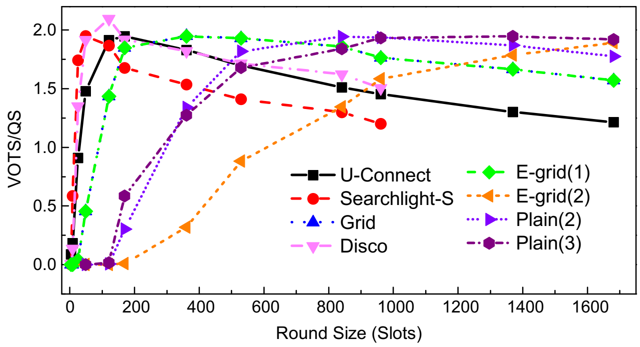

Fig. 13 The VOTS/QS result of clique model with reliable link

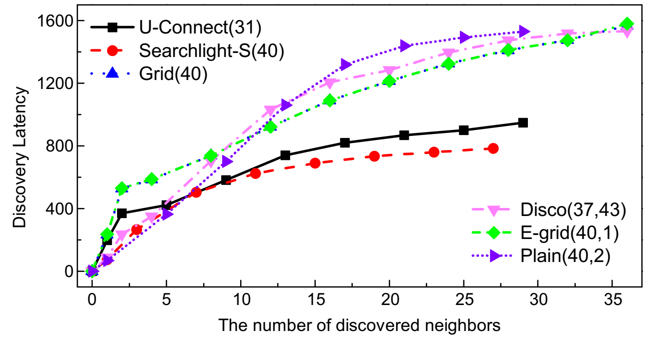

Fig. 15 Clique CDF performance with six different QSs and reliable link

the beacons. Thus, usually the smartphones cannot find all their neighbors. The number of different initialization combinations of QSs is even more tremendous than that of P2P model because there are more smartphones. It is impractical to go through all those permutations. Therefore, we calculate them by randomly selecting the initialization combinations sufficient times (e.g., 10^7 or 10^8) and using the mean of them.

Figure 13 demonstrates the VOTS/QS of QSs in the clique model, where each smartphone has 9 neighbors in its communication range. With the changes of round size, the best VOTS/QS value of each kind of QSs are quite the same. U-Connect is 1.9476, Searchlight-S is 1.9486, Grid is 1.9483, Disco is 1.9482, E-grid(1) is 1.9481, E-grid(2) is 1.9490, Plain(2) is 1.9487 and Plain(3) is 1.9488. The peak of VOTS/QS in U-Connect, Searchlight-S, Grid and Disco QSs come earlier. These QSs are better when round size is small (e.g., 200) and E-grid(k), Plain(k) QSs are better when round size is large (e.g., 800). It implies that these four QSs perform worse than E-grid(k) and Plain(k) QSs with reliable link when round size is very large (e.g., 1600). When round size is 1600, the VOTS/QS of E-grid(2) and Plain(3) are 1.6 and 1.5 times that of Searchlight-S and U-Connect QSs.

Figure 14 shows the VOTS/QS of eight different QSs when the link is unreliable and $P_u = 0.7$. The performances are quite similar to Fig. 13. Compared with Fig. 13, the best value of VOTS/QS of all QSs come earlier. Then the best VOTS/QS value of each QS are still almost the same. With the round size becoming larger, Plain(k) QS outperforms than Disco, Searchlight-S, Grid, and U-Connect QSs and the performance of Plain(k) QS is stable.

Figure 15 shows the CDF in the clique model. As mentioned above, usually the smartphones cannot discover all their neighbors accounts for the interference of concurrent signals. In Fig. 15, each smartphone has 39 neighbors and a time shift to simulate the asynchronous situation. The duty cycle is 5% and the parameters of QSs are the same as Fig. 11. These six QSs all cannot detect 39 neighbors. Grid, Disco and E-grid(k) QSs can discover 37 neighbors, U-Connect and Plain(k) QSs are 30 and Searchlight-S is 27. Though discovery latency of Searchlight-S is quite small, it only can detect 70% neighbors, in contrast, E-grid(k) can find out almost all neighbors. Thus we will prefer to use E-grid(k) QS in a clique situation with reliable link.

As shown in Fig. 16, the CDF performances in a clique model with $P_u = 0.7$ are different from Fig. 15. The settings of QSs are the same as Fig. 15. These six QSs all cannot detect 39 neighbors. Plain(k) QS can discover 17 neighbors, E-grid(k) and Disco are 15, Grid is 14 and Searchlight-S is 13, U-Connect is 12. Though discovery latency of Searchlight-S is quite small, it only can detect 33.3% neighbors, in contrast, Plain(k) QS can find out 43.3% neighbors.

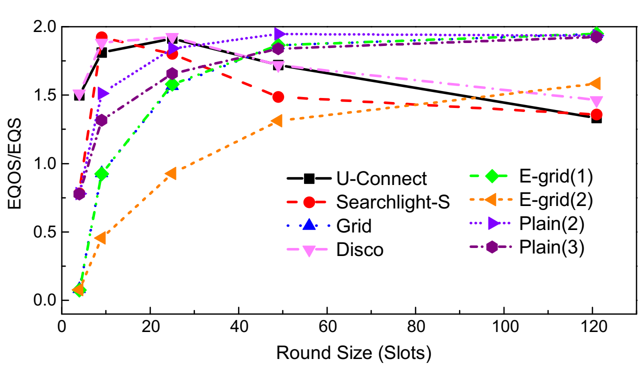

Fig. 14 The VOTS/QS result of clique model with unreliable link

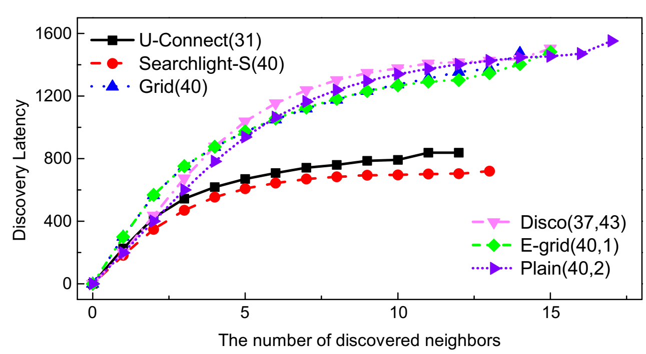

Fig. 16 Clique CDF performance with six different QSs and unreliable link

6 Related work

6.1 Neighbor discovery problem

Neighbor discovery problem is a fundamental problem in wireless sensor networks and wireless mobile networks [3, 24, 29]. It can be applied in many useful applications [27, 30, 31]. A basic problem is asynchronous neighbor discovery problem. Qiu et al. [20] devises a family of Nihao energy-efficient protocols having more beacons at asleep time slots and fewer probes. Chen et al. [3] proposes a protocol focusing on the situation when nodes have heterogeneous antenna configurations. Margolies et al. [17] presents the Power Aware Neighbor Discovery Asynchronously (Panda) protocol in which nodes transform between the sleep, receive, and transmit states. Bracciale et al. [2] designs a different approach to maximizing the total number of discoverable contacts with a battery charge and [22] constructs an ALOHA-like algorithm that is at most a factor $min(\Delta, \ln n)$ worse than optimal. However, in sum, they can't deal with realistic situations, where the links of each smartphone pair are unreliable.

6.2 Deterministic protocols

A deterministic protocol establishes a pattern to schedule the periodical operations of each smartphone. A kind of foundational deterministic protocol is Quorum System and an important QS is presented in [15] called Grid QS. A period in Gird QS has m^2 consecutive time slots, where a smartphone is either awake or asleep. The smartphones transmit and listen to messages during awake time slots. So two neighboring smartphones discover each other when they are both awake. Those m^2 time slots are arranged as an $m \times m$ matrix. Each smartphone selects a quorum from the QS, during which the smartphone stays awake. This QS ensures that two neighboring smartphones will have at least two intersecting awake time slots during each period even in an asynchronous network. However, the quorum size is $2m - 1$ and Luk et al. propose a new QS based on the Grid QS in [14], which arranges the time slots as a right-angled triangle and its quorum size is approximately $\sqrt{2}m$. There is an another new quorum in [14]—Cyclic QS. Cyclic QS is based on the ideas of cyclic block design and cyclic difference sets in combinatorial theory and the optimal solution of it can reach the lower bound of the quorum size—m. Torus QS is presented in [12] relaxing the time slot arrangement condition. Jiang et al. prove that QS satisfying the *rotation closure property* can solve the neighbor discovery problem and propose the e-torus QS in [10], which gives us inspiration.

There exists many important QSs, for instance, Disco, U-connect, and Searchlight. Disco and U-connect QSs are based on the Chinese Remainder Theorem [7]. In Disco protocol, each smartphone chooses a pair of prime numbers (p_1, p_2) and wakes up at multiples of p_1 and p_2. U-connect uses only one prime number p. Each smartphone turns awake only at multiples of p; as well as $\frac{p+1}{2}$ time slots every p^2 time slots. These two prime-based QSs reduce the worst-case discovery latency. Searchlight improves the relationship between the patterns of two smartphones and performs much better.

Except the QS protocols mentioned above, people put forward different deterministic protocols. Meng et al. [19] proposes a code-base protocol. In this protocol, it uses codes to represent the state of a smartphone and constructs optimal codes from a perfect difference set, which gets a better worst-case latency. Chen et al. in [4] designs a non-integer protocol, that is, time is continuous and smartphones may become active or inactive at any point in time, subject to fewer constraints.

6.3 Probabilistic protocols

McGlynn et al. in [18] introduces a family of "Birthday protocols", which use random independent transmissions to discover adjacent smartphones and are the foundation of probabilistic protocols. The time is slotted and each smartphone determines the work mode from transmitting, listening and energy-saving with a specific probability in birthday protocol. Because this protocol is inspired by Birthday Paradox, it gains great performances in median cases. Nevertheless, it doesn't have the upper bound of discovery latency, that is, the worst-case latency will be arbitrarily long.

7 Conclusion

In this paper, we design a novel deterministic QS–E-grid(k) QS and a novel probabilistic QS–Plain(k) QS and propose two algorithms based on these two QSs to solve the asynchronous neighbor discovery problem in wireless mobile networks with unreliable link. We evaluate our algorithms by VOTS/QS and CDF. Then we compare their performances with other protocols mentioned in related work such Searchlight-S, U-Connect, Disco and Grid QSs. In P2P model, Plain(k) and E-grid(k) QSs can reach the best VOTS/QS with reliable or unreliable link. Compared by CDF, Plain(k) QS can perform better than Grid and Disco QSs with reliable link and Grid, Disco and U-Connect QSs with unreliable link. In clique model, Plain(k) and E-grid(k) QSs have higher VOTS/QS when round size is large with reliable or unreliable link. Evaluated by CDF, Plain(k) and E-grid(k) QSs can find out most neighbors with reliable or unreliable link. Both our simulation result and theoretic analysis have shown that these new QSs achieve better performances than existing works.

Acknowledgements This work is under the support of the general program of the National Natural Science Foundation of China (NSFC) under Grants No. 61473109, 61671193, 61572164 and 61772472, the Natural Science Foundation of Zhejiang Province under Grant No. LY17F020020, the Key Research and Development Plan of Zhejiang Province under Grant No. 2018C04012, the Graduate Scientific Research Foundation and the Excellent Dissertation Fostering Foundation of Hangzhou Dianzi University.

References

1. Bakht M, Trower M, Kravets RH (2012) Searchlight: won't you be my neighbor? In: Proceedings of the 18th annual international conference on mobile computing and networking. Istanbul, pp 185–196
2. Bracciale L, Loreti P, Bianchi G (2016) The sleepy bird catches more worms: revisiting energy efficient neighbor discovery. IEEE Trans Mob Comput 15(7):1812–1825
3. Chen L, Li Y, Vasilakos AV (2016) Oblivious neighbor discovery for wireless devices with directional antennas. In: Proceedings of the IEEE international conference on computer communications. San Francisco, pp 1–9
4. Chen S, Russell A, Jin R, Qin Y, Wang B, Vasudevan S (2015) Asynchronous neighbor discovery on duty-cycled mobile devices: integer and non-integer schedules. In: Proceedings of the 16th ACM international symposium on mobile Ad Hoc networking and computing. Hangzhou, pp 47–56
5. Duan X, Zhao C, He S, Cheng P, Zhang J (2017) Distributed algorithms to compute walrasian equilibrium in mobile crowdsensing. IEEE Trans Ind Electron 64(5):4048–4057
6. Dutta P, Culler D (2008) Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In: Proceedings of the 6th ACM conference on embedded network sensor systems. Raleigh, pp 71–84
7. Hardy GH, Wright EM (1979) An introduction to the theory of numbers. Oxford University Press
8. He S, Li X, Chen J, Cheng P, Sun Y, Simplot-Ryl D (2013) Emd: energy-efficient p2p message dissemination in delay-tolerant wireless sensor and actor networks. IEEE J Selected Areas Commun 31(9):75–84
9. Jiang JR (2008) Expected quorum overlap sizes of quorum systems for asynchronous power-saving in mobile ad hoc networks. Comput Netw 52(17):3296–3306
10. Jiang JR, Tseng YC, Hsu CS, Lai TH (2005) Quorum-based asynchronous power-saving protocols for ieee 802.11 ad hoc networks. Mobile Netw Appl 10(1–2):169–181
11. Kandhalu A, Lakshmanan K, Rajkumar RR (2010) U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol. In: Proceedings of the 9th ACM/IEEE international conference on information processing in sensor networks. Stockholm, pp 350–361
12. Lang S, Mao L (1998) A torus quorum protocol for distributed mutual exclusion. In: Proceedings of the 10th international conference on parallel and distributed computing and systems. Las Vegas, pp 635–638
13. Li Z, Zhang J, Shen X, Fan J (2017) Prediction based indoor fire escaping routing with wireless sensor network. Peer-to-Peer Network Appl 10(3):697–707
14. Luk WS, Wong TT (1997) Two new quorum based algorithms for distributed mutual exclusion. In: Proceedings of the 17th International conference on distributed computing systems. Baltimore, pp 100–106
15. Maekawa M (1985) A $\sqrt{n}$ algorithm for mutual exclusion in decentralized systems. ACM Trans Comput Syst 3(2):145–159
16. Malkhi D, Reiter MK, Wool A, Wright RN (2001) Probabilistic quorum systems. Inf Comput 170(2):184–206
17. Margolies R, Grebla G, Chen T, Rubenstein D, Zussman G (2016) Panda: neighbor discovery on a power harvesting budget. IEEE J Selected Areas Commun 34(12):3606–3619
18. McGlynn MJ, Borbash SA (2001) Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks. In: Proceedings of the 2nd ACM international symposium on mobile ad hoc networking & computing. Long Beach, pp 137–145
19. Meng T, Wu F, Chen G (2016) Code-based neighbor discovery protocols in mobile wireless networks. IEEE/ACM Trans Network 24(2):806–819
20. Qiu Y, Li S, Xu X, Li Z (2016) Talk more listen less: Energy-efficient neighbor discovery in wireless sensor networks. In: Proceedings of the IEEE international conference on computer communications. San Francisco, pp 1–9
21. Rahman MR, Tseng L, Nguyen S, Gupta I, Vaidya N (2017) Characterizing and adapting the consistency-latency tradeoff in distributed key-value stores. ACM Trans Autonom Adapt Syst 11(4):20
22. Vasudevan S, Adler M, Goeckel D, Towsley D (2013) Efficient algorithms for neighbor discovery in wireless networks. IEEE/ACM Trans Network 21(1):69–83
23. Wang X, Sun H, Deng T, Huai J (2015) On the tradeoff of availability and consistency for quorum systems in data center networks. Comput Netw 76:191–206
24. Yang Q, He S, Li J, Chen J, Sun Y (2015) Energy-efficient probabilistic area coverage in wireless sensor networks. IEEE Trans Veh Technol 64(1):367–377
25. Zhang H, Zheng WX (2018) Denial-of-service power dispatch against linear quadratic control via a fading channel. IEEE Transactions on Automatic Control. https://doi.org/10.1109/TAC.2018.2789479
26. Zhang H, Qi Y, Zhou H, Zhang J, Sun J (2017a) Testing and defending methods against dos attack in state estimation. Asian Jof Control 19(4):1295–1305
27. Zhang H, Qi Y, Wu J, Fu L, He L (2018) Dos attack energy management against remote state estimation. IEEE Trans Control Netw Syst 5(1):383–394
28. Zhang J, Tang S, Shen X, Dai G, Nayak A (2011) Quorum-based localized scheme for duty cycling in asynchronous sensor networks. In: Proceedings of the IEEE 8th international conference on mobile Adhoc and sensor systems. Valencia, pp 440–449
29. Zhang J, Li Z, Tang S (2016) Value of information aware opportunistic duty cycling in solar harvesting sensor networks. IEEE Trans Indus Inform 12(1):348–360
30. Zhang J, Li Z, Lin X, Jiang F (2017) Composite task selection with heterogeneous crowdsourcing. In: Proceedings of the 14th Annual IEEE international conference on sensing, communication, and networking. San Diego, pp 1–9
31. Zhang Z, Zhang H, He S, Cheng P (2017c) Bilateral privacy-preserving utility maximization protocol in database-driven cognitive radio networks. IEEE Transactions on Dependable and Secure Computing. https://doi.org/10.1109/TDSC.2017.2781248