accepted to appear in IEEE International Conference on Communications (ICC) 2018,
Ad hoc and Sensor Networking (AHSN) Symposium, Kansas City, May, 2018

Charging Skip Optimization with Peer-to-Peer Wireless Energy Sharing in Mobile Networks

Aashish Dhungana, Tomasz Arodz and Eyuphan Bulut
Department of Computer Science, Virginia Commonwealth University
401 West Main St. Richmond, VA 23284, USA
{dhunganaa, tarodz, ebulut}@vcu.edu

***Abstract*—Increasing software capabilities and complicated applications running on smartphones have increased the quality of life for users. However, the battery lives of smartphones have stayed limited due to the respectively slower improvements in battery technology. Users are often required to find a charging port and connect the phone to the port through a cable. A lot of times, this process can be irritating or even infeasible. Through adoption of emerging wireless power transfer technology in these devices, charging process has transformed into a new dimension. Moreover, this has brought the opportunity for wireless energy exchange between mobile devices ubiquitously. In this paper, we investigate the potential of peer-to-peer energy sharing to reduce the burden of traditional cord-based charging process. The devices of users can make use of energy available from other users' devices based on their meeting patterns so that the battery level of their devices could be maintained within acceptable level without the need of charging it through a cable frequently. Our specific goal in this study is to find the maximum number of traditional way of charging times that could be skipped through utilization of available energy in other users in the vicinity with wireless energy sharing. To this end, we use dynamic programming approach to find the optimal skipping patterns for selfish and cooperative energy exchange cases and verify the results with brute force.**

***Index Terms*—Mobile social network, wireless energy exchange, dynamic programming, charging skip.**

I. Introduction

About 5 billion users are carrying a mobile device with a service around the globe [1]. The various uses of these devices and increasing popularity of software applications such as email, Facebook, and maps have made people highly dependent on mobile devices. This intensive use of mobile devices has brought a huge load on battery requirements. The hardware capabilities have significantly improved since the advent of smartphones but the development of powerful batteries have not taken the necessary pace, making the batteries the main bottleneck. As a result, users are required to charge their devices too often.

The process of charging a mobile device has its own challenges in its current form today, as most of the users use cables for charging and they need to find an outlet to plug these cables, which may not be an easy task when the user is outside. This irritating and sometimes infeasible way of charging process has been relieved to some extent through the usage of wireless charging recently. Several phone manufacturers have released various models (including Apple's recently released iPhone X and 8 [2]) with wireless charging capability as a built-in feature. Users can place their devices on a charging pad (which could be embedded in other things such as a desk [3], or cup holder in a car [4]) and start charging their devices without the hassle of cables. However, the charging pad or equipment still needs to be plugged to a power source. Recently, this somewhat limited usage of wireless charging has further been extended with energy transfer between mobile devices [5], [6]. Through bidirectional chargers, mobile devices could exchange energy without the need of being connected to an outlet. Such a peer-to-peer (P2P) energy sharing opportunity brings flexibility to users for finding power ubiquitously and mitigates the risks of facing an emergency situation with depleted battery [7]–[9].

In this paper, we investigate the benefit of P2P energy sharing between mobile devices on reducing the burden of traditional cord-based charging process (simply called wall charging in the rest of the paper). Depending on the meeting schedules with other users, a user can make use of excessive[1] energy available from other users' devices to skip some of the wall chargings while still maintaining the device's charge within an acceptable level. Our goal is to maximize the number of wall chargings that could be skipped through utilization of energy shared by other users in the vicinity. We aim to discover the potential benefit of P2P energy sharing on existing charging habits of users. Hence, we assume that the charging patterns of user devices and as well as their meeting patterns with other users (from which shareable energy amounts could be derived) are given. We exploit dynamic programming approach to find out the optimal skipping patterns for selfish and cooperative cases. In the selfish case, we assume that a user knows the amount of energy that could be received from other users for each charging cycle without giving energy. In the cooperative case, we allow both sharing and receiving of energy between users and study simultaneous optimization of skipping patterns from each user's perspective.

The rest of the paper is structured as follows. We discuss the related work in Section II. In Section III, we define the problem together with an analysis towards its solution. In Section IV, we provide the details of dynamic programming based optimization algorithms. In Section V, we provide and discuss the results for our problem and finally, we conclude

[1]Current charging habits of users show that they charge their devices more often than they need [7], yielding opportunity for energy sharing with others.

the paper and outline future work in Section VI.

II. Related Work

With the recent development in wireless power transfer technologies, a number of studies have been conducted on how to utilize this technology to improve the energy management in mobile networks. Previous work have mainly focused on applying these technologies to prolong the lifetime of wireless sensor networks and mobile ad-hoc networks [10], [11] having low energy requirements.

Recently, a few studies have been done to analyze the impact of P2P energy exchange on the operation of smartphone based mobile networks. In [12], [13], authors exploit P2P wireless energy exchange to balance the energy within a mobile social network and propose various algorithms to be used in sharing protocol. In [14], the impact of P2P energy sharing on network formation and in [15] its benefit on group based charging has been studied. A more generic work can be found in [9], in which authors focus on enhancing the energy usage of wireless networks with wireless energy sharing to minimize the chances of ending up with insufficient energy for their consumption. In all these studies, however, the concept is studied without an integrated analysis of charging habits of individual user devices and meeting patterns between the users that can exchange energy. In [7] and [8] users are first ranked based on their ability to supply energy to one another, then pairs that would help each other the most are assigned to each other using stable matching. While these studies provide an idea on the potential benefit of wireless energy exchange to users, they do not present the optimal benefit that could be reached. In this paper, different than previous work, we define the burden of charging in terms of the number of periods that the devices stay plugged to the outlet (i.e., wall charging) and discuss the minimization of these times exploiting the energy shared by other users without changing the charging habits of any user.

III. Problem Analysis and Formulation

The main goal of this study is to find the maximum number of wall chargings that can be skipped by utilizing the energy available from other users' devices. In other words, we aim to see the survivability of mobile users with the minimum number of wall chargings possible to relieve the users from the burden of wall charging.

For a given charging pattern of a user device, the time between the start of one wall charging and the start of next one represents a charging cycle. The set of all charging and discharging periods for a user could be defined as:

$$\begin{aligned}
\delta_c &= \left\{\delta_c^1, \delta_c^2, \ldots, \delta_c^n\right\} \\
\delta_d &= \left\{\delta_d^1, \delta_d^2, \ldots, \delta_d^n\right\} \text{ where} \\
& \quad \delta_c^i.l_e = \delta_d^1.l_s, \forall i \in \{1 \ldots n\} \text{ and} \\
& \quad \delta_d^i.l_e = \delta_c^{(i+1)}.l_s, \forall i \in \{1 \ldots (n-1)\}
\end{aligned}$$

Here, each (δ_c^i, δ_d^i) represents a charging cycle with one charging and one discharging. The attributes l_s and l_e represent the starting and ending charge levels for each of these periods.

We consider that when a mobile user meets another mobile user, they can exchange energy between each other wirelessly. Moreover, we assume that these meeting periods with energy sharing opportunity correspond to the times when these devices are both discharging. The amount of energy that could be exchanged depends on several factors including transfer speed, efficiency, duration of their meeting and the available capacity in the receiver.

The optimization problem is studied for two different cases; (i) selfish, and (ii) cooperative. While the former looks at the problem from only one user's perspective by considering the available energy that could be received from the others, in the latter, we consider the two way interactions (i.e., receiving and giving of energy) between the users and aim to optimize the problem jointly from the perspective of both users. Next, we discuss the details of the problem within each context.

A. *Selfish case*

In this case, we study the problem from the perspective of a single user that is aware of available energy from other users for each of its charging cycles and aims to skip as many wall chargings as possible. Note that in this case user is not sharing energy with others but receiving from others. Fig.1 shows example charging patterns for two different users for a certain time period. Depending on the applications that are running on the device the discharging rate might vary at different times. Similarly, depending on the equipment used for charging or due to the active usage while charging, the charging of the device could happen at different rates.

The problem here is defined as follows. Given the amount of energy the user could receive from other users during each charging cycle, what is the maximum number of wall charging instances that could have been skipped (completely) without affecting the functionality of the user's device (i.e., without changing the charging habits of the user). It is important to note that, a user may try to skip some of its wall chargings purely by benefiting from the excessive charging in its own charging schedule and without using any energy available from other users.

We formulate the problem using decision points that occur at the beginning of each charging cycle. Decision points divide a given user charging pattern into blocks of time periods known as decision blocks. Each block starts with the starting of charging for a user and ends with the completion of discharging period. In this case, since there is a single user, each decision block corresponds to an individual charging cycle of the user. For user A's charging pattern shown in Fig.1, there are six decision blocks with starting times $D = \{0, 4, 7, 10, 12, 15\}$. Similarly, for user B, there are five decision blocks with starting times $D = \{2, 5, 8, 10, 13\}$. For each decision block D_t, $\forall\ t \in \{1 \ldots |D|\}$, the following has to be maintained:

$$D_{t+1}.l_s - (D_t.l_s + \delta_c[t](1 - X_t) - \delta_d[t] + R[t].X_t) = 0 \quad (1)$$

where, $\delta_c[t]$, $\delta_d[t]$ and $R[t]$ represent the amount of total wall charge, total discharge, and total energy that could be received

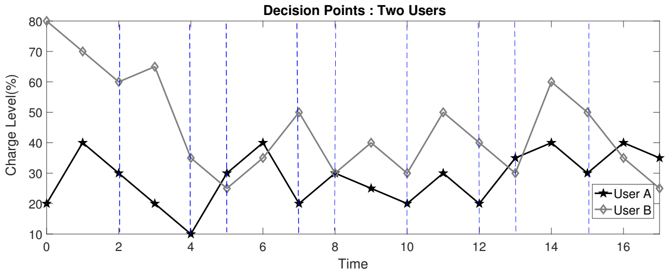

Fig. 1: Charging patterns and decision points of two users.

by A during the t^{th} decision block, and X_t is the skip decision variable $\in$ {0,1}, with 1 meaning skip.

B. Cooperative case

In this case, users are allowed to both send and receive energy between each other. Therefore, the optimal skipping pattern has to be determined considering the amount of energy that will be exchanged between users. The decision points (i.e., start of charging cycles) coming from all users will form decision blocks with partitioned charging cycles of users. Moreover, some decision points might divide a charging period of a user into two or more parts. The set of decision points that come from both users in Fig. 1 is $D = \{0, 2, 4, 5, 7, 8, 10, 12, 13, 15\}$, which is $D_A \cup D_B$. When a decision point causes a split in the charging period of a user, since we base our optimization to the skipping of wall chargings completely (i.e., no partial skipping allowed), the skip decision made for a portion of a wall charging inside a decision block should match with the decision made for the remaining portion of the same wall charging in the next decision points. In order to reach the optimal skipping solution that maintains this, for every such decision point, both results (skipping or not) have to be stored until the split of a charging period with decision points is over and only the optimal one is picked.

In this case, for each decision block D_t, the following equations have to be maintained:

$$D^A_{t+1}.l_s - (D^A_t.l_s + \delta^A_c[t](1-X_t) - \delta^A_d[t] + S^B_t - S^A_t) = 0 \quad (2)$$

$$D^B_{t+1}.l_s - (D^B_t.l_s + \delta^B_c[t](1-X_t) - \delta^B_d[t] + S^A_t - S^B_t) = 0 \quad (3)$$

where, $\delta^A_c[t]$ ($\delta^B_c[t]$) and $\delta^A_d[t]$ ($\delta^B_d[t]$) represent the amount of total wall charge and discharge for A (B), respectively and S^A_t (S^B_t) shows the energy shared (to B) by A (B) during the t^{th} decision block.

IV. Algorithm Development

We use dynamic programming approach to find the optimal skipping pattern in both cases of the problem. At each decision point, the algorithm tries to recursively find the best charging levels that will result in the minimum number of wall chargings for each user. The approach used in the optimization of both cases is slightly different and discussed in the subsequent sections. However, the initialization and solution readout part for both cases are similar. We consider a two dimensional matrix for the first case and a three dimensional matrix for the second case, where the first dimensions represent the decision points and the other dimensions represent the individual charge levels of each user. Note that in the case of multiple users more than two, the dimension of the matrix can be increased accordingly together with additional updates in the algorithms, which will be the subject of our future work.

A. Optimization algorithm for selfish case

There are two important matrices: D matrix and T matrix. D matrix stores the integer value that represents the number of wall chargings required for each charge level by every decision block and T matrix stores the index of the D matrix from which that value is derived. The algorithm takes the list of wall charging amounts (δ_c), the list of discharging amounts (δ_d) and the list of energy available for each decision block (R) as a parameter. $initLevel$ is the initial charging level for the given charging pattern. For example, for A's pattern in Fig. 1, $initLevel$ is 20. $minLevel$ is the minimum acceptable level. Values from $D[0][initLevel]$ to $D[0][0]$ is initialized to 0 because we know that we can achieve each of them without charging from the wall. All other values in D and T matrix are initialized to some higher integer value.

Algorithm 1 SelfishOptimalSkip(δ_c, δ_d, R)

```
for each decision block D_t do
    for each charging level 0 to 100 as l do
        friend = min(100, l + R[t])-δ_d[t]
        wall = min(100, l + δ_c[t]) - δ_d[t]
        Update(t, l, friend, 0)
        Update(t, l, wall, 1)
    end for
end for
```

Algorithm 2 Update(t, curLevel, newLevel, inc)

```
if newLevel ≥ minLevel then
    currentState = D[t][curLevel]
    if currentState + inc ≤ D[t+1][newLevel] then
        D[t+1][newLevel]=currentState+inc
        T[t+1][newLevel] = curLevel
    end if
end if
```

The main body of the algorithm is shown in Algorithm 1. The main principle on which this algorithm works is, for each charge level i.e., from 0 to 100 and at each decision block (D_t), how much energy we can get either from wall or from a friend and to what charge level we can get to utilizing the available energy is found and the path with minimum wall chargings (i.e., maximum skips) is maintained. The corresponding charge level at (t+1)th decision time is updated to the minimum of either the value at t or the current value at (t+1) if we can get to t+1 by receiving energy from a friend and to the minimum of either the (value at t) + 1 or the current value at (t+1) if we can get to (t+1) from wall charging. We apply the same logic recursively for all charging cycles and find the optimal skip sequence. The running time of the algorithm is $O(|D|100)$, while brute force solution has $O(2^{|D|})$ complexity.

Algorithm 3 SolutionReadout(D, T)

index $\leftarrow \arg\min\{D[n-1][i] \; \forall \; i \in [0, 100]\}$
value = D[n-1][index]
index = T[n-1][index]
for i=n-2; i$\geq$0; i$--$ **do**
  if value = D[i][index] **then**
    skip[i] = 1
  else if value - D[i][index] = 1 **then**
    skip[i] = 0
  end if
  value = D[i][index]
  index = T[i] [index]
end for
return skip

The algorithm to readout the solution is presented in Algorithm 3. We start at the last decision block and get the index with the minimum number of charges from D matrix. Each position in D matrix is associated with its previous cell using T matrix. If the value in current index of D matrix has increased (i.e., wall charging used) compared to its previous value, then the skip value for that charging cycle is 0, otherwise it is 1.

B. Optimization algorithm for cooperative case

In this setting, we assume the amount of energy that is available to each user at a point is not certain. The algorithm needs to find the maximum energy that needs to be shared/received for the optimal skip. The energy exchange between users can potentially happen when they actually meet and both user devices are discharging. Hence, the amount of energy that could be shared between these devices will be determined by the meeting and charging patterns of these devices.

In this case, due to the aggregated definition of the decision points from both users, we need to keep track of all possible states when a decision block consists of a partial charging. Two more matrices need to be defined for such cases: D_{skip} and D_{wall}, which are responsible to store the skip or no skip decision for a partial block. However, the final decision is always updated on the D matrix, when the split is over. These matrices are also initialized to some higher integer value. Other initializations are similar to that of selfish case. The details of the dynamic programming based solution are presented in Algorithm 4 and 5.

For every decision block, we check if the wall charging of any user is crossed by the decision points. If we find the crossing, then for the first part of the split, we store the decision for both values of X_t in two different matrices (i.e., D_{skip} for skipping decision, and D_{wall}, for not skipping decision). In our example, the first split is done for A at the decision block starting at time 5. When the algorithm encounters such a decision block, the source matrix (i.e., D

Algorithm 4 CooperativeOptimalSkip (δ_c^A, δ_c^B, δ_d^A, δ_d^B)

merged $\leftarrow$ true
for each decision block as t **do**
  if (merged) **then**
    if A (B) has first half **then**
      compute($A(B)$, $B(A)$, D, D_{skip}, D, D_{wall})
      merged $\leftarrow$ false
    else
      compute($A(B)$, $B(A)$, D, D, D, D)
    end if
  else
    if A (or B) has first half **then**
      compute($A(B)$, $B(A)$, D_{skip}, D_{skip},D_{skip},D_{wall})
      compute($A(B)$, $B(A)$, D_{wall}, D_{Skip},D_{wall},D_{wall})
      merged $\leftarrow$ false
    else if A (B) has second half and !B (!A) has first half **then**
      compute($A(B)$, $B(A)$, D_{skip}, D, D_{wall}, D)
      merged $\leftarrow$ true
    end if
  end if
end for

matrix for the first case), is split into (A=1, B=[0,1]), which is the D_{skip} matrix and (A=0, B=[0,1]), which is the D_{wall} matrix. If the next decision block contains the second part of the charging period, the decision from D_{skip} and D_{wall} is then merged back to the source matrix (i.e., D matrix). The matrices maintained during the runtime of the algorithm for the given example in Fig. 1 is illustrated in Fig. 2.

Algorithm 5 covers all four possible cases and decides the type of the split operation required. Here, T matrix keeps track of all necessary information including the source matrix and energy exchanges in every step for reading out the solution. In case 1 and 2, we track the skipping cases for the user with split charging period, and in case 3 and 4, we track the other cases. Thus, the source matrix is written into D_{skip} in the first two cases and while it is written into D_{wall} in the last two cases. Per our assumption, meeting times correspond to the times when user devices are discharging, thus, the decision for two way energy exchange between users happen only when both users decide to skip wall charging (case 1). The implementation of energy exchanges is shown in lines 9-19 of Algorithm 5. The running time of this algorithm is $O(|D|(100)^2(E))$, where E is the average shareable energy range. Brute force solution again has $O(2^{|D|})$ complexity.

V. Results

We have used the charging patterns for two users shown in Fig. 1. We first run the optimization algorithm for selfish case for each of the nodes A and B separately. In this case, the selfish node (e.g., A) knows the amount of energy that could be shared by the other node for each of its charging cycle and does not share any energy with the other one. As it can be seen in Fig. 1, node B can safely share a total of 25% of its

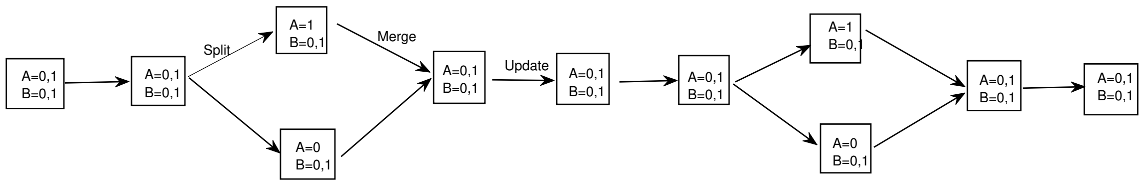

Fig. 2: Matrices maintained during runtime of the optimization algorithm for cooperative case.

Algorithm 5 compute(u_1, u_2, S_1, D_1, S_2, D_2)

for each charging Levels 0 to 100 as l_i^A **do**
  for each charging Levels 0 to 100 as l_i^B **do**
    u_1.skip = $l_i^{u_1}$ - $\delta_d^{u_1}$[t]
    u_1.wall = min(100, $l_i^{u_1}$ + $\delta_c^{u_1}$[t]) - $\delta_d^{u_1}$[t]
    u_2.skip = $l_i^{u_2}$ - $\delta_d^{u_2}$[t]
    u_2.wall = min(100, $l_i^{u_2}$ + $\delta_c^{u_2}$[t]) - $\delta_d^{u_2}$[t]
    Case 1: both users skip
    Update2(t, u_1.skip, u_2.skip, S_1, D_1, 0, 0)
    energyShareable = min($\delta_d^{u_1}$[t], $\delta_d^{u_2}$[t])
    if energyShareable ≥ 0 **then**
      for all energy from 0 to energyShareable as e **do**
        u_2.shared = $l_i^{u_2}$ - e - $\delta_d^{u_2}$[t]
        u_1.received = $l_i^{u_1}$ + e - $\delta_d^{u_1}$[t]
        Update2(t, u_1.received, u_2.shared, S_1, D_1, 0, e)
        u_1.shared = $l_i^{u_1}$ - e - $\delta_d^{u_1}$[t]
        u_2.received = $l_i^{u_2}$ + e - $\delta_d^{u_2}$[t]
        Update2(t, u_1.shared, u_2.received, S_1, D_1, 0, e)
      end for
    end if
    Case 2: only u_1 skips
    Update2(t, u_1.skip ,u_2.wall , S_1, D_1, 1, 0)
    Case 3: both do not skip
    Update2(t, u_1.wall, u_2.wall, S_2, D_2, 2, 0)
    Case 4: only u_2 skips
    Update2(t, u_1.wall, u_2.skip, S_2, D_2, 1, 0)
  end for
end for

Algorithm 6 Update2(t, l_A, l_B, source, dest, inc, energy)

if $l_A \geq$ minLevel and $l_B \geq$ minLevel **then**
  m = $\min\{$source[t][l_A][l_B]+inc, dest[t+1][l_A][l_B]$\}$
  dest[t+1][l_A][l_B] = m
  T[t+1][l_A][l_B] = (l_i^A , l_i^B, source, energy)
end if

energy with node A in any charging periods of A. As node A's charge level will not hit 100% even it gets all of this 25% at its first charging cycle, it does not actually matter how this 25% is split into different charging cycles (assuming that they meet at every charging cycle with sufficiently large duration). Similarly node A can share 10% of its energy with B safely.

In Table I, we show the optimal skip pattern results for selfish case in different scenarios. In scenario 0, we did not let users exchange energy to see how many skips they could have done by their own charging habits only. In scenario A.1 and A.2, we considered the energy transfer from B to A only (i.e., when A is selfish) but with different splits of the maximum energy B can share to different decision blocks. Similarly, in scenario B.1 and B.2, we considered B as the selfish node and split the A's shareable energy to different decision blocks. Note that in these scenarios, decision blocks correspond to the node's charging cycles. A has six decision blocks while B has five decision blocks. In scenario 0, the results show that node A could have skipped 4^{th} and 6^{th} charging blocks, while node B could have skipped its 1^{st} and 4^{th} blocks (skipping 1^{st} and 3^{rd} would also be optimal). This results in a total of 4 skips for both nodes. In other scenarios, when only one node shares energy with the other one and do not skip any wall charging, the other node would increase the skipped charging count to at most 4, but this does not change the total skipped count for both nodes (in scenario A.1, A could skip more chargings compared to A.2 as it is offered B's all shareable amount in earlier charging cycles).

Scenario	Decision Blocks (Charging Cycle)	1	2	3	4	5	6
0	**Energy (B $\leftrightarrow$ A)**	0	0	0	0	0	0
	A's skip sequence	0	0	0	1	0	1
	B's skip sequence	1	0	0	1	0	N/A
A.1	**Energy (B $\rightarrow$ A)**	25	0	0	0	0	0
	A's skip sequence	1	0	1	1	0	1
A.2	**Energy (B $\rightarrow$ A)**	10	0	10	0	5	0
	A's skip sequence	0	0	1	0	1	1
B.1	**Energy (A $\rightarrow$ B)**	10	0	0	0	0	N/A
	B's skip sequence	1	0	1	1	0	N/A
B.2	**Energy (A $\rightarrow$ B)**	0	5	0	5	0	N/A
	B's skip sequence	1	0	1	1	0	N/A

TABLE I: Optimal skipping results for selfish case.

Decision Blocks	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Energy (A $\rightarrow$ B)	0	0	0	0	0	5	0	0	0	4
B's skip sequence	1	1	1	0	1	1	0	1	0	1
Energy (B $\rightarrow$ A)	19	0	0	0	0	0	0	0	0	0
A's skip sequence	0	1	0	0	1	1	0	1	1	1

TABLE II: Skip sequence in decision blocks in cooperative case.

The result for cooperative case is presented in Table II. Out of 10 decision blocks, user A is able to skip 6 of them. However, not all of these are independent decisions as well as some of these skip decisions only consists of discharging

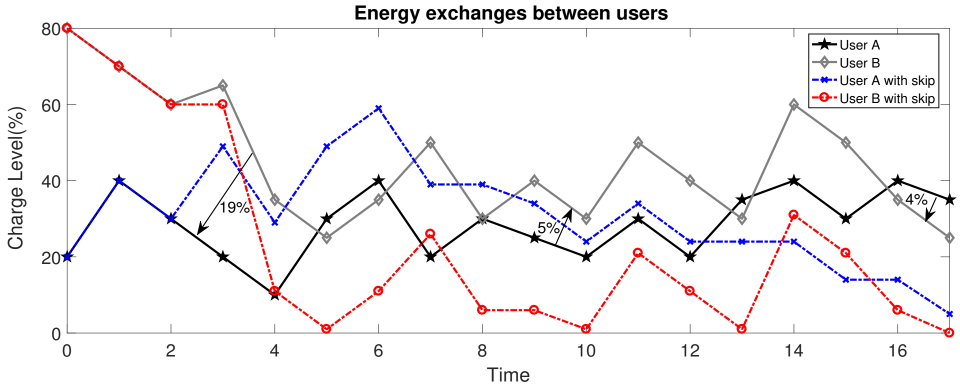

Fig. 3: Charging patterns and skips after cooperative energy exchange.

which is not an actual skipping. Similarly, for user B, 7 of them can be skipped. Note that there are multiple energy exchanges between users in order to get to the optimal point. As the decision blocks do not correspond to the actual individual charging cycles of users, the skipping decisions for each decision block have to be converted to the skipping pattern for charging cycles. From Fig. 3 and Table II, we can deduce the original skip sequence for user A and user B which is shown in Table III. This results in a total of 5 skips for both nodes, showing the advantage of cooperative P2P sharing over selfish case. To achieve that both node A and B share energy between each other and receive energy from each other. Fig. 3 shows the charging patterns after the optimal skips are done[2].

Charging Cycle	**1**	**2**	**3**	**4**	**5**	**6**
A's skip sequence	0	0	1	0	1	1
B's skip sequence	1	0	1	0	0	N/A

TABLE III: Actual skip sequence of wall chargings in cooperative case.

VI. Conclusion

In this paper, motivated by the recent technologies enabling wireless energy sharing between mobile devices, we investigate to what extent the burden of charging process on users could be released. We develop a dynamic programming based optimization model and find out the maximum number of charging times that could be skipped through utilization of excessive energy from other users in the vicinity. We study two cases with selfish and cooperative nodes. The results show that when nodes cooperate they would skip more of their wall chargings. In our future work, we will study cooperation between multiple users and try to apply energy sharing behavior in a more realistic manner by embedding the predictions of charging and meeting patterns in mobile social networks [16], [17].

[2]When a user skips a wall charging, it is assumed that a minimal/zero discharge happened during that duration in this example, however, a discharge could have been applied with an average discharging rate and algorithms could be updated easily.

References

[1] GSMA, "Number of mobile subscribers of worldwide hits 5 billion," 2017. [Online]. Available: https://www.gsma.com/newsroom/press-release/number-mobile-subscribers-worldwide-hits-5-billion/

[2] Apple, "Iphone x," 2017. [Online]. Available: https://www.apple.com

[3] IKEA, "Chargers youll actually want everywhere," 2017. [Online]. Available: http://www.ikea.com/us/en/catalog/categories/departments/wireless_charging/

[4] B. W. Charger, "Zens car wireless charger review," 2016. [Online]. Available: http://bestwirelesscharger.org/zens-car-wireless-charger-review/

[5] P. Worgan, J. Knibbe, M. Fraser, and D. Martinez Plasencia, "Powershake: Power transfer interactions for mobile devices," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 2016, pp. 4734–4745.

[6] T. Perry, "Ces 2013: Share battery power between mobile devices," Online, Jan 2013. [Online]. Available: http://spectrum.ieee.org/tech-talk/consumer-electronics/portable-devices/mobile-devices-share-everything

[7] E. Bulut and B. K. Szymanski, "Mobile energy sharing through power buddies," in *Wireless Communications and Networking Conference (WCNC), 2017 IEEE*. IEEE, 2017, pp. 1–6.

[8] D. Niyato, P. Wang, D. I. Kim, and W. Saad, "Finding the best friend in mobile social energy networks," in *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3240–3245.

[9] D. Niyato, P. Wang, D. I. Kim, W. Saad, and Z. Han, "Mobile energy sharing networks: Performance analysis and optimization," *IEEE Trans. on Vehicular Technology*, vol. 65, no. 5, pp. 3519–3535, 2016.

[10] Y. Peng, Z. Li, W. Zhang, and D. Qiao, "Prolonging sensor network lifetime through wireless charging," in *Real-time systems symposium (RTSS), 2010 IEEE 31st*. IEEE, 2010, pp. 129–139.

[11] L. Xie, Y. Shi, Y. T. Hou, W. Lou, H. D. Sherali, and S. F. Midkiff, "On renewable sensor networks with wireless energy transfer: The multi-node case," in *Proc. SECON*. IEEE, 2012, pp. 10–18.

[12] S. Nikoletseas, T. P. Raptis, and C. Raptopoulos, "Interactive wireless charging for energy balance," in *Wireless Power Transfer Algorithms, Technologies and Applications in Ad Hoc Communication Networks*. Springer, 2016, pp. 585–603.

[13] ——, "Energy balance with peer-to-peer wireless charging," in *Proc. MASS*. IEEE, 2016, pp. 101–108.

[14] A. Madhja, S. Nikoletseas, C. Raptopoulos, and D. Tsolovos, "Energy aware network formation in peer-to-peer wireless power transfer," in *Proc. International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*. ACM, 2016, pp. 43–50.

[15] E. Bulut, M. E. Ahsen, and B. K. Szymanski, "Opportunistic wireless charging for mobile social and sensor networks," in *2014 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2014, pp. 207–212.

[16] E. Bulut and B. K. Szymanski, "Exploiting friendship relations for efficient routing in mobile social networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2254–2265, 2012.

[17] S. C. Geyik, E. Bulut, and B. K. Szymanski, "Grammatical inference for modeling mobility patterns in networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 11, pp. 2119–2131, 2013.