On the Optimal Scheduling of Uplink Resources in OFDMA-Based Wireless Networks

Patrick Hosein

Huawei Technologies Co., Ltd., 10180 Telesis Court, Suite 365, San Diego, CA 92121, USA.
e-mail: phosein@huawei.com

Abstract: Orthogonal Frequency Division Multiplexing (OFDM) is used for both 802.11 and 802.16 wireless network standards and is being evaluated for Fourth Generation (4G) networks such as those being proposed in the 3GPP and 3GPP2 wireless standards bodies. In both uplink and downlink directions, resources can be allocated in three dimensions, frequency, time and power. The uplink resource allocation problem is more complicated due to the limited transmission power of the subscriber stations. In this paper we consider the uplink resource allocation problem for the case of 802.16 networks and derive the optimal solution taking into account constraints that were not considered in previous work.

1. Introduction

The recently defined 802.16 standards use OFDM for the physical layer transmissions because it has been shown to be robust against frequency selective fading. These networks will provide users with high data rates with a much wider coverage than WiFi networks (802.11). They will also support QoS services and 802.16e will provide support for limited mobility. In OFDM networks, resources are scheduled in the frequency, time and power domains. In the downlink, the rate achieved by a user increases with the number of sub-channels (frequency domain) assigned, the number of time slots (time domain) assigned and the fraction of base station (BS) power allocated. The same is true for the uplink except that there is a limit on the total transmission power of each subscriber station (SS) and this needs to be taken into account. Both uplink and downlink resource allocation is performed by the BS. The resource (slot) assignments are broadcast to all users in the downlink within each frame. Each SS decodes this information to determine which, if any, data slots have been allocated to it in the downlink frame. It can then go ahead and retrieve its information. It can also tell which uplink slots have been reserved for its use and so it can transmit its information in those slots in the subsequent uplink frame.

In this paper we focus on the uplink scheduling problem. Although several papers have been published on the downlink scheduling problem (e.g., [1, 2, 3, 4, 5]), fewer papers have been published on the uplink scheduler (e.g., [6, 7]). In both of these uplink scheduling papers, the time dimension is ignored, users are allocated to sub-carriers and the objective is to maximize the total user throughput (i.e., maximize the rate sum capacity). For each user, the reverse link path gain is assumed to be different for each subcarrier. Under these assumptions they show that subcarriers should be sequentially assigned to the user with the largest rate increment and that water-filling should then be used to make subcarrier power assignments.

Our paper makes some practical assumptions as follows. We assume that subchannels, made up of multiple subcarriers, are assigned to users in discrete time slots. Each subchannel/time-slot pair will be referred to as simply a slot. Therefore the number of available slots is the product of the number of subchannels and the number of time-slots. We assume that each slot can be assigned to at most one user. Secondly, we assume each user is assigned a utility function of their throughput and the objective is to maximize the total utility over all users. Note that throughput maximization is a special case in which the utility is a linear function of the user’s average throughput. However, we are more interested in more sophisticated utility functions such as the logarithm function which provides proportionally fair user throughputs and utility functions that can be used for QoS traffic.

We assume that the subcarriers of a subchannel are chosen randomly from the set of sub-carriers allocated for data (i.e., the distributed permutation option in the 802.16 standard). In light of this, we assume that for each user the reverse link path gain as well as the reverse link inter-cell interference is the same for all subchannels assigned to that user. This implies that the received Signal to Interference and Noise ratio (SINR) for a given subchannel transmission power is the same for all subchannels. If this is the case then, given the subchannel assignments, the optimal power allocation strategy is simply to spread the total transmission power evenly over all assigned subchannels. Hence we are only interested in the number of subchannels assigned to each SS.

One other constraint we take into account is the fact that users are allocated resources along the time domain first. This constraint will be made more clear later but we will also show that it is a natural property of the optimal solution. Finally we take into account finite user queues. This becomes important when scheduling time sensitive traffic which typically is conveyed by small packets generated at periodic intervals. One such application is Voice over Internet Protocol (VoIP).

Note that we have taken into account all important constraints for the distributed permutation option of the 802.16 standard. In the next section we formulate the optimization problem and then derive the necessary and sufficient conditions for optimality. We then present an algorithm for producing an optimal solution followed by an example of pseudo-code. Note that since we prove optimality for the proposed algorithm then, under the stated assumptions no other algorithm can perform better hence we do not include simulation results.
2. Formulation of the Scheduling Problem

We assume \( N \) users and \( M \) reverse link subchannels with \( T \) time slots per frame (Figure 1). Each SS has a maximum transmission power of \( P \). Since we assume that each sub-channel consists of a set of sub-carriers which are randomly chosen across the bandwidth then we can reasonably assume that the path loss, fading, inter-cell interference and background noise is the same for all sub-channels. Therefore the Signal to Interference Noise ratio will be the same for transmissions on any of these sub-channels. We denote the SINR of the signal received at the BS by an amount \( g \). Hence if a transmission power of \( p \) is used to transmit a packet over a sub-channel then the SINR of the received signal is \( pg \) (independent of the sub-channel used). So note that we take into account all factors that affect the transmission (including inter-cell interference).

The average throughput of a user at the start of the frame being scheduled is denoted by \( r \). We assign a throughput dependent utility function to each user. This function, which may be different for different SSs, represents the utility to that user of the corresponding throughput achieved. It is denoted by \( U(r) \). Note that utility functions that are dependent on other metrics (such as queue size, delay, etc.) are also covered by the approach described in this paper. The size of the data queue of a user before transmission of the concerned frame is denoted by \( q \).

We assume that the reverse link SINR per unit of transmission power, \( g \), is reported after each frame. Given this, together with the transmission power allocated to the subchannel (which will depend on the number of subchannels assigned to the user during that time-slot), we can then compute the maximum payload for that transmission. This mapping depends on several factors such as the modulation and coding scheme. However, in [9] it is shown that the shape of this mapping function between SINR and rate closely follows the Shannon capacity function but with some effective bandwidth \( B \). We also assume that the overhead percentage is constant for all modulation and coding schemes and include this factor in \( B \) so that the Shannon function provides the data payload rate. In addition, the factor \( B \) takes into account the logarithm factor since we will be using natural logarithms instead of base 2 logarithms. We will denote the Shannon rate, normalized to the effective bandwidth \( B \), by \( d \). Therefore if \( z \) sub-channels are assigned to a particular user in a particular time slot then \( P \) power is allocated to each sub-channel and hence the achievable (normalized) rate using the effective Shannon rate is given by

\[
    d(z) = z \ln \left( 1 + \frac{P g}{z} \right). 
\]

(1)

Note that a similar approach is taken in [10]. In their approach they discount the SINR of the received signal by an amount \( \beta \) that depends on the required bit error rate (BER) using the following formula

\[
    \beta = \frac{1.5}{-\ln(5BER)}. 
\]

This takes into account the efficiency of the modulation and coding scheme used. In this case the normalized rate is given by \( d(z) = z \ln(1 + \beta P g/z) \). The methodology presented in our paper can also be used with this model by normalizing the received SINR values \( g \) with the constant \( \beta \) instead of using an effective bandwidth approach.

Let \( x_i(t) \) denote the number of subchannels that are assigned to user \( i \) in time slot \( t \). Note that, since the reverse link gain is subchannel independent then, any \( x_i(t) \) subchannels can be used for the user in that time slot as long as we maintain \( \sum_{i=1}^{N} x_i(t) \leq M \) for each time slot \( t \). Therefore the actual allocation of subchannels in that time slot can be made after finding the optimal value of \( x_i(t) \). Note that \( x_i(t) \) must be an integer since at most one user is served per subchannel in each slot.

Another 802.16 constraint that must be taken into account is the fact that user allocations are made contiguously first in the time dimension and then the subchannel dimension as shown in Figure 1. This restriction implies that for each SS, \( i \), we must have \( |x_i(t_a) - x_i(t_b)| \leq 1 \) for any pair of time-slots \( t_a \) and \( t_b \). However one can show that the optimal solution will have this property as follows. Note that the second derivative of \( d(z) \) is given by

\[
    d''(z) = \frac{-1}{z} \left( \frac{P g}{P g + z} \right)^2 < 0 \quad \text{for} \quad z > 0. 
\]

This implies that \( d(z) \) is a concave function. Consider some optimal allocation and suppose for this allocation that \( x_i(t_a) - x_i(t+b) > 1 \) for two time slots \( t_a \) and \( t_b \). If one slot was moved from sub-channel \( t_a \) to sub-channel \( t_b \) then, because of the concavity property of \( d(z) \), the decrease in rate in sub-channel \( t_a \) is less than the increase in rate in sub-channel \( t_b \) and hence there is an overall increase in throughput which contradicts our assumption of optimality. Therefore the number of slots in any two sub-channels must differ by at most one.

This constraint can be used to further simplify the problem as follows. Let \( x_i = \sum_{t=1}^{T} x_i(t) \) denote the the total number of slots allocated to user \( i \). The above
property implies that in \( x_i - T \left\lfloor \frac{x_i}{T} \right\rfloor \) time-slots \( \left\lfloor \frac{x_i}{T} \right\rfloor \) sub-channels are allocated to the user while in the remaining time-slots \( \left\lfloor \frac{x_i}{T} \right\rfloor \) sub-channels are used. This information is sufficient for us to determine the total rate achieved by the user which is needed for computing the user’s throughput and hence the user’s utility. Therefore, we can instead use \( x_i \) as the decision variable and once this is computed we can then determine the number of sub-channels per time slot and the allocation of these sub-channels within each time-slot. The normalized rate achieved by user \( i \) if provided with \( x_i \) slots is therefore given by

\[
d_i(x_i) = \left(x_i - T \left\lfloor \frac{x_i}{T} \right\rfloor \right) \left( \left\lfloor \frac{x_i}{T} \right\rfloor + 1 \right) \ln \left( 1 + \frac{P g_i}{\left\lfloor \frac{x_i}{T} \right\rfloor + 1} \right) + \left(T - x_i + T \left\lfloor \frac{x_i}{T} \right\rfloor \right) \left\lfloor \frac{x_i}{T} \right\rfloor \ln \left( 1 + \frac{P g_i}{\left\lfloor \frac{x_i}{T} \right\rfloor} \right).
\]

(2)

Let \( r_i \) denote the user’s average throughput before transmission of the frame. This throughput is updated at the end of the transmission based on \( x_i \) as follows:

\[
\tilde{r}_i(x_i) = \alpha r_i + (1 - \alpha) d_i(x_i) \quad 0 < \alpha < 1,
\]

(3)

where \( d_i(x_i) \) is determined by 2 and \( \alpha \) is the filter constant that is typically chosen close to unity for stable performance. For a given allocation \( x_i \) the utility of that user after frame transmission is then given by \( U_i(\tilde{r}_i(x_i)) \).

Note that we can use 3 and 2 to express the utility of each SS as a function of \( x_i \). We can then maximize the sum of the utilities of all SSs. Consider the utility function \( U(r) = \ln(r) \) which provides proportionally fair user throughputs. In Figure 2 we plot \( U(\tilde{r}_i(x_i)) = U(\tilde{r}_i(0)) \) for a typical set of parameter values, which we call the baseline problem. Note that the expression is only valid for integer values of \( x_i \) but we perform linear interpolation to better illustrate the function. This expression shows the change in utility as more slots are assigned to the user. In this case we used \( M = 17 \) and \( T = 6 \) for a total of 102 slots. Note that all of these are typically not available for data traffic so in practice the scheduler will have to skip over a subset of these slots. We also plot the case of (a) the baseline problem but with half the initial throughput, \( r_i \), and (b) the case of the baseline problem but with double the reverse link SINR \( g_i \). Note that the utility function increases with increasing path gain (radio conditions) as well as with decreasing average throughput. Also note that it is a strictly concave function.

The final constraint that we consider is the fact that the amount of data queued for each user is finite. Therefore we need to ensure that a user is only allocated transmission resources when data is available to fill the corresponding packet. For a given allocation \( x_i \), the achievable rate is \( B d_i(x_i) \) kbps. If we denote the total service duration by \( \tau \) then the queue constraint becomes \( B d_i(x_i) \tau \leq q_i \). For simplicity we assume that the amount of data in the queue can always fit exactly into an integer number of allocated slots. Let us summarize the notation that has so far been introduced and then formulate the optimization problem.

\[
N = \text{number of SSs}
\]

\[
M = \text{number of reverse link sub-channels}
\]

\[
T = \text{number of time-slots}
\]

\[
P = \text{maximum transmission power of each SS}
\]

\[
B = \text{effective Shannon bandwidth per subchannel}
\]

\[
g_i = \text{reverse link SINR of } i \text{ per unit transmission power}
\]

\[
r_i = \text{average throughput of } i \text{ before frame transmission}
\]

\[
\tilde{r}_i = \text{average throughput of } i \text{ after frame transmission}
\]

\[
U_i = \text{utility function of } i \text{ as a function of throughput}
\]

\[
q_i = \text{queue size of user } i \text{ before frame transmission}
\]

\[
x_i = \text{total number of slots allocated to user } i
\]

\[
\tilde{x} = [x_1, \ldots, x_N]
\]

\[
\tau = \text{duration of an uplink frame}
\]

The optimization problem can be stated as follows:

\[
\max_{\tilde{x} \in \{0,1,\ldots,M T\}^N} F(\tilde{x}) \equiv \sum_{i=1}^{N} U_i(\tilde{r}_i(x_i))
\]

(4)

subject to \( \sum_{i=1}^{N} x_i \leq MT \) and \( d_i(x_i) \leq \frac{q_i}{B \tau} \)

where \( \tilde{r}_i(x_i) \) is given by 3 and \( d_i(x_i) \) is given by 2.

3. Optimality Conditions

Note that 4 is an integer programming problem. We relax the integer constraint and solve the resulting problem. We then illustrate that the resulting relaxed problem has an integer solution and hence this solution is also optimal for the original problem. We assume that \( U(r) \) is a concave, increasing function over \( r \geq 0 \). We now show that \( U(x) \) is also concave and increasing over \( x \geq 0 \).

First consider the function \( d(x) \) given by 2 in the range \( 0 \leq x \leq MT \). Suppose that \( KT \) is such that \( K T < x < (K + 1)T \) for any integer \( K \geq 0 \). In this case we have

\[
d(x) = (x - KT)(K + 1) \ln \left( 1 + \frac{Pg}{K + 1} \right)
\]

\[
+ (T - x + KT)K \ln \left( 1 + \frac{Pg}{K} \right)
\]

(5)
For this range of \( x \) we compute the derivative of \( d \) with respect to \( x \) as,

\[
d'(x) = (K + 1) \ln \left(1 + \frac{Pg}{K + 1}\right) - K \ln \left(1 + \frac{Pg}{K}\right).
\]

Therefore the derivative is constant throughout the range \( KT < x < (K + 1)T \) which means that \( d(x) \) is a piecewise linear function with breakpoints wherever \( x \) is a multiple of \( T \). Next consider the function

\[
g(x) = \frac{x}{T} \ln \left(1 + \frac{Pg}{x/T}\right).
\]

Note that \( g(KT) = d(KT) \) for non-negative integers \( K \). Taking the first and second derivatives of \( g(x) \) we get

\[
g'(x) = \frac{1}{T} \ln \left(1 + \frac{PgT}{x}\right) - \frac{Pg}{x + PgT} > 0
\]

and

\[
g''(x) = -\frac{(Pg)^2 T}{x(x + PgT)^2} < 0
\]

Therefore \( g(x) \) is a concave, increasing function of \( x \). Observe that \( d(x) \) consists of piecewise linear segments that touch \( g(x) \) at multiples of \( T \). Hence it can easily be shown that \( d(x) \) is also increasing and concave. Let us now turn to the function \( U(x) \). Taking the derivative with respect to \( x \) we get

\[
\frac{dU}{dx} = \frac{dU}{dr} \frac{dr}{dx} = \frac{dU}{dr} (1 - \alpha)d'(x)
\]

Using the fact that both \( U(r) \) and \( d(x) \) are increasing functions (and hence have positive gradients) then the product on the right is positive and hence \( U(x) \) is also an increasing function of \( x \). The second derivative is given by

\[
\frac{d^2U}{dx^2} = \frac{d^2U}{dr^2} (1 - \alpha)^2 (d'(x))^2 + \frac{dU}{dr} (1 - \alpha) d''(x)
\]

Using the fact that \( U(r) \) and \( d(x) \) are both strictly concave so that their second derivatives are negative and also the fact that \( U(r) \) is an increasing function of \( r \) (and hence its derivative is positive) then the second derivative of \( U(x) \) is negative and hence it is also concave. Therefore \( U(x) \) is a piecewise linear, increasing, concave function. Note that this property holds for any concave, strictly increasing utility function as for example the case plotted in Figure 2.

Note that the derivative of \( U(x) \) is discontinuous at \( x = KT \) for non-negative integers \( K \) (see Figure 3). Hence let \( U'_+(x) \) denote the right sub-gradient and \( U'_-(x) \) denote the left sub-gradient (i.e., all sub-gradients supported at this point lie between these two values). These gradients will be equal except when \( x \) is a multiple of \( T \). At \( x = KT \) with \( K > 0 \) the left and right sub-gradients are given by

\[
U'_+(x) = (K + 1) \ln \left(1 + \frac{Pg}{K + 1}\right) - K \ln \left(1 + \frac{Pg}{K}\right)
\]

\[
U'_-(x) = K \ln \left(1 + \frac{Pg}{K}\right) - (K - 1) \ln \left(1 + \frac{Pg}{K - 1}\right)
\]

Therefore we have shown that Problem 4 is the maximization of a concave function (i.e., the sum of the concave utility functions) over a compact set. It therefore has a unique solution and we can use Lagrangian Multiplier methods to determine the optimal point. If we denote the optimal point by \( \bar{x}^* \) then necessary and sufficient conditions for optimality are

\[
\gamma \in [U'_-(x_i^*),U'_+(x_i^*)] \quad \text{iff} \quad 0 < B\tau d_i(x_i^*) < q_i, \tag{6}
\]

\[
\gamma < U'_-(x_i^*) \quad \text{iff} \quad B\tau d_i(x_i^*) = q_i, \tag{7}
\]

\[
\gamma > U'_+(x_i^*) \quad \text{iff} \quad x_i^* = 0, \tag{8}
\]

for some \( \gamma \geq 0 \). If the derivative of the utility function was continuous then \( \gamma \) is simply the gradient satisfying the Kuhn-Tucker conditions. In this case since the derivative is not continuous then we need to introduce sub-gradients but \( \gamma \) can be interpreted in a similar manner. Next we use these optimality conditions to design an algorithm for determining the optimal allocation.

4. Algorithm Description

We show how the solution for the case of \( k \) subchannels can be used to determine the optimal solution for \( k + 1 \) subchannels. Hence the optimal solution for \( M \) subchannels can be determined iteratively. Note that if \( M = 0 \) then the optimal allocation is \( x^*_i = 0 \) for all users \( i \) and this allocation satisfies 8 since, if we assume \( q_i > 0 \) for all \( i \), we can choose \( \gamma = \infty \).

Let us now assume that the optimal allocation, \( x_i^* \), for \( k \) subchannels is known and we show how the optimal allocation for \( k + 1 \) subchannels can be determined. Let \( \gamma(k) \) denote the corresponding \( \gamma \) for the case of \( k \) subchannels. We choose \( \gamma(k + 1) \) as follows:

\[
\gamma(k + 1) = \max_i \{U'_i(x_i^*)\} \quad \text{forall} \quad \text{s.t.} \quad q_i > B\tau d_i(x_i^*).
\]

and denote the SS for which this condition holds by \( i^* \). We assign the new subchannel to user \( i^* \). If this user has sufficient data to fill all slots in that subchannel then we are done. Note that the optimality conditions for this new allocation are still satisfied. If on the other hand the user does not have sufficient data to fill all time slots for that subchannel then as many time slots are assigned as needed to serve all data for that user. Again the optimality conditions are maintained for all users but in this case...
case $\gamma$ must be chosen smaller than $\gamma(k + 1)$ but large enough so that it exceeds the second largest of the set \{\(U_i(x_i)\)\}. We can pick the user with the next largest gradient and repeat until the subchannel is completely used or all users run out of data. In either case we find that the optimality conditions are maintained.

We illustrate this solution with a simple example. We consider the case of $N = 4, M = 8, T = 4$. In Figure 4 we plot, for each SS, the utility $U(x)$ as a function of the number of slots allocated to the SS. Recall that this is a piecewise linear function. The optimal allocations $x_1, x_2, x_3, x_4$ are indicated on the x axis. $x_1 = 0$ because, at optimality, the gradient of $U_1(x)$ at the origin is smaller than $\gamma$ (the gradient of the line segments touching each of the utility curves). The queue size of each SS is listed on the top horizontal line. This is the number of slots that are required to serve all data in the user's buffer. We find that, although $U_2(x)$ is a rapidly growing function, the optimal allocation is $x_2 = 4$ because that is the number of slots needed to serve its data. Note that $\gamma$ is less than the maximum sub-gradient of $U_2(x)$ at $x = 4$. The optimal allocation of the other SSs are $x_3 = 12$ and $x_4 = 16$. Note that in both of these cases the segment of gradient $\gamma$ is a sub-gradient at the optimal points. The total number of slots allocated is therefore $4 + 12 + 16 = 32$. Since $T = 4$ this implies that 8 sub-channels are allocated in each time slot as it should be since $M = 8$. Pseudo-code for proportional fair rate allocations is provided in Figure 5.

5. Conclusions and Future Work

In this paper we presented a mathematical model for the allocation of reverse link resources in a OFDMA network. We derived the optimality conditions for this problem and then used these to develop an algorithm for determining the optimal allocation. The resulting algorithm is simple and can easily be implemented. We showed that the algorithm was optimal under the assumptions made and hence any other algorithm is sub-optimal (hence simulation comparisons were not provided).

Define $d_i(z) = z \log (1 + \frac{p_j z}{T})$

$S \equiv \emptyset, j = 0$

for $(i = 1 : N)$ {
$x_i = 0$
$\Delta_i = d_i(1)$
if $((q_i > 0) \& \& (\Delta_i > d_{\text{min}}))$ \{ $S = S \cup \{i\}$ \}
}

while ($j < MT$) {
\{ $S \neq \emptyset$ \}
$i^\ast = \arg \max_{i \in S} \{ \frac{\Delta_i}{\gamma} \}$
$n = \frac{x_{i^\ast}}{\gamma}$
$\theta = \min \{ T, MT - j, \left( \frac{1}{\Delta_{i^\ast}} \cdot \frac{\theta}{\theta - Td_{i^\ast}(n)} \right) \}$
$x_{i^\ast} = x_{i^\ast} + \theta$
$j = j + \theta$
if $((d_{i^\ast}(n) + 2 < d_{\text{min}}) || (\theta < T))$ \{ $S = S \setminus \{i^\ast\}$ \}
else \{ $\Delta_{i^\ast} = d_{i^\ast}(n + 2) - d_{i^\ast}(n + 1)$ \}
\} \{ $j = MT$ \}

Figure 4: Example of Optimal Allocation

Figure 5: Optimal Allocation Algorithm Code

REFERENCES


