Subcarrier Allocation in OFDMA Systems: Beyond water-filling

Somsak Kittipiyakul and Tara Javidi
Department of Electrical Engineering
University of Washington, Seattle, 98195-2500, USA
{somsak, tjavidi}@ee.washington.edu

Abstract—This paper considers the issue of optimal subcarrier allocation in OFDMA. When including time-varying packet arrivals and channels, we show, via a counter example, that water-filling-based subcarrier allocation policies, contrary to conventional wisdom, fail to provide rate-stability for an otherwise stabilizable OFDMA system. We briefly discuss the long-run throughput optimality in OFDM as demonstrated in [13]. In particular under a simple channel model, we consider a non-idling policy which balances the queues and outperforms water-filling policies by achieving the minimum average delay. In this paper, we provide simple computation algorithms for implementing such policies and provide simulations for a numerical comparative study.

I. INTRODUCTION

The problem of optimal real-time subcarrier allocation in OFDMA systems has been recently studied (e.g. [1]-[9]). Most of the papers in the literature ([1]-[3], [8]-[9]) provide solutions to the multi-user water-filling problem at each decision epoch (in this paper, we refer to this as “instantaneous throughput maximization”). We instead focus on a long-term average delay performance. We argue that maximizing instantaneous throughput (water-filling) is too myopic unless it is accompanied by a queue-averse allocation. Through the counter example in Figure 1, we show how a policy that maximizes the instantaneous throughput causes rate-instability (unbounded queue buildups) in an otherwise stabilizable system.

By adaptively assigning subcarriers to the best users, water-filling achieves the instantaneous maximum throughput by taking advantage of channel diversity among users in different locations. Since it is unlikely that selective fading affects users on similar frequencies, there is an overall gain in selectively assigning subcarriers to users. This is known as multiuser diversity gain. When each user is assumed to have infinite bits in the buffer, water-filling is throughput optimal in both short- and long-term. However, when considering finite time-varying packet arrivals and time-varying channels, water-filling is not long-term throughput optimal.

From dynamic programming [13], we know that instantaneous throughput maximization does not guarantee long-term throughput optimality in a stochastic environment. In general, an optimal long-term policy must trade off between two competing goals: on one end is the desire to get maximum throughput now and on the other end the desire to get the maximum throughput in the future. The second goal requires positioning the system to enjoy the highest multiuser diversity gain in the future. This can be generally achieved by avoiding the situations where some queues are empty while others are heavily backlogged. In other words, the optimal long-term policy must take advantage of the knowledge of the statistics of the channel and arrival processes. This knowledge is used to find the current action that optimizes both the expected future outcome and the current throughput.

In this paper and [13], we focus on a simple special case of the arrival and channel processes. We assume that the packet arrival process and the channel connectivity process have periodic structures with period of four timeslots. (a) Packet arrivals for users 1 and 2 (b) Channel connectivity profile (white square means ON) (c) MTLB policy π∗ (d) Maximal Instantaneous Throughput policy π (e) Queue occupancies over time: queue lengths for users U1 and U2 under π, (b1∗, b2∗), and under π∗, (b1∗, b2∗). Under policy π, queue of user U2 (b2) is building up over time.

Fig. 1. A counter-example showing that water-filling causes unbounded queue buildups. We assume that the packet arrival process and the channel connectivity process have periodic structures with period of four timeslots. (a) Packet arrivals for users 1 and 2 (b) Channel connectivity profile (white square means ON) (c) MTLB policy π∗ (d) Maximal Instantaneous Throughput policy π (e) Queue occupancies over time: queue lengths for users U1 and U2 under π, (b1∗, b2∗), and under π∗, (b1∗, b2∗). Under policy π, queue of user U2 (b2) is building up over time.
one modulation and coding in the system. With the above assumptions, a subcarrier then can be considered as a server that is either connected or disconnected from users as shown in Figure 2. In [13], we proved that a maximum throughput (non-idling) load balancing (MTLB) policy minimizes the average total backlog over any arbitrary horizon time T. This policy not only provides optimal long-term throughput (rate-stability) for all admissible traffic but also minimizes average delay. The contribution of this paper is to extend the result in [13] on the construction of MTLB policy and to show the performance comparisons with simulation under a symmetric on-off channel model. However, for the readers’ convenience we briefly provide problem statement and the result here. Detailed description is provided in [13] and [14].

Here we would like to emphasize that the load balancing in this work is shown to be essential in providing optimal throughput. This requirement is absolutely independent of fairness considerations. In other words, our proposed MTLB policy is not motivated by concerns regarding fairness, instead it demonstrates the following fundamental point: under realistic arrival patterns, load balancing is an essential element of any throughput optimal policies.

II. Problem Formulation

In this paper, we consider a single-hop OFDMA system composed of one cell or cluster with one base station. There are N users and K sub-carriers. The sub-carriers are time-slotted. We assumed a centralized controller at the base that allocates subcarriers to users using perfect knowledge of channel state information and queue lengths. Channel state information is defined to be whether a subcarrier can be used to transmit a packet successfully or not. Under such assumptions, our problem is mapped to the following multi-server queuing problem (as a generalization of the problems studied in [10]).

Problem (P)

Consider a discrete-time model of N queues \( Q_1, \ldots, Q_N \) served by K servers \( K > N \). At each time, each server can serve one packet from one queue; we allow for one queue to be served by multiple servers. At each time, a queue \( n \) is either available to be served by a server \( k \) (connected or ON) or it is not available (disconnected or OFF).

At each time, the connectivities of all queue/server pairs are known for that time. We allow for arrivals at each queue at each time and arrivals at a given time are assumed to occur after server allocations at that time. The statistics of arrival and connectivity processes are assumed to be symmetric across queues and independent across time. We wish to determine a server (subcarrier) allocation policy \( \pi \) that minimizes the average backlog over (finite or infinite) horizon T:

\[
J^* = \frac{1}{T} E \left\{ \sum_{t=0}^{T} \sum_{n=1}^{N} b_n(t) \right\}
\]

where \( b_n(t) \) is the backlog of \( Q_n \).

Definition 1: Considering ordering function \( ord : \mathbb{R}^N \to \mathbb{R}^N \) to be such that for all \( x \in \mathbb{R}^N \), \( y = ord(x) \) has the ordered elements of \( x \) in descending order i.e. \( y_i \geq y_{i+m}, m > 0 \). Let \( \tilde{x} = ord(x), \tilde{y} = ord(y) \), and \( m = \min \{ \{ i : \tilde{x}_i \neq \tilde{y}_i \} \}, N \). Then we say that \( x \leq_{LQO} y \) (read \( x \) is smaller than \( y \) in term of Longest Queue Ordering) or in another word \( x \) is more balanced than \( y \) if and only if \( \tilde{x}_m \leq \tilde{y}_m \). We say \( x \) is strictly more balanced than \( y \), \( x <_{LQO} y \) if and only if \( x \leq_{LQO} y \) but \( y \not<_{LQO} x \).

Definition 2: Given the queue length vector \( b = \{ b_n \} \) and channel connectivity matrix \( C = \{ c_{kn} \} \) at the beginning of time slot \( t \), the MTLB policy chooses a subcarrier allocation matrix \( \omega^*(t) = \{ \omega_{kn}^* \} \) such that

\[
(\text{C1}) \text{ Maximum Throughput: } \omega^*(t) = \arg \max_{\omega \in \mathcal{W}(b, C)} \sum_{n=1}^{N} \sum_{k=1}^{K} \omega_{kn}
\]

where \( \omega \in \mathcal{W}(b, C) \) if

- \( \text{(C1.a)} \omega_{nk} \leq c_{kn} \);
- \( \text{(C1.b)} \sum_{k=1}^{K} \omega_{kn} \leq 1, \forall k = 1, \ldots, K \); and
- \( \text{(C1.c)} \sum_{n=1}^{N} \omega_{kn} \leq b_n, \forall n = 1, \ldots, N \).

(C2) Load Balancing: Let \( L \) be the maximum throughput achieved in (2) and \( \mathcal{W}_L(b, C) \) contains all possible assignments achieving throughput \( L \). \( \omega^*(t) \in \mathcal{W}_L(b, C) \) produces the most balanced \( b - \omega^* \) queue configuration.

We have shown in [13] the following main theorem.

Theorem 1: Consider Problem (P), MTLB policy is optimal for any initial state \( (b, C) \).

Note that condition (C1) is a water-filling condition. Hence, Theorem 1 implies that water-filling criteria is not sufficient to guarantee long-term throughput optimality unless it is complemented by a load-balancing criteria (condition (C2)).

III. Computation of MTLB

Since a subcarrier can serve only one packet, the MTLB policy basically finds the matching that matches one-to-one each subcarrier to a packet in the most balanced manner. Using concepts from graph matching literature ([11],[12]), we here give an algorithm to compute MTLB. Specifically,
if we convert the original graph of queues and servers (Fig. 2) into the following equivalent bipartite graph with weights along the edges, we want to show that the maximum weight matching on the equivalent bipartite graph gives MTLB assignment.

**Equivalent Bipartite Construction**

1. Associated with each queue \( n \), construct \( m_n = \min(b_n, \sum_{k=1}^{K} c_{kn}) \) nodes labeled as \( a_{n1}, a_{n2}, \ldots, a_{nm_n} \).
2. Let \( U = \{a_{11}, a_{12}, \ldots, a_{1m_1}, a_{21}, \ldots, a_{Nm_N}\} \) be the set of all such nodes.
3. Let \( V = \{v_k\}_{k=1}^{K} \) be the set of servers.
4. Let \( E = \{(a_{nm}, v_k) : c_{kn} = 1\} \) be the set of edges representing connectivities.
5. Let \( \psi : E \mapsto \mathbb{Z}^+ \), \( \psi(a_{nm}, v_k) = b_n - m + 1 \) be the positive integer weight of each edge in \( E \).

To prove this, we restate the following definitions and lemma from [14] used on the original graph in Figure 2:

**Definition 3:** The ordered sequence \( S(\omega, u_1, u_k) \) is said to be a balancing path from queue \( u_1 \) to queue \( u_k \) corresponding to the allocation \( \omega(b, C) \) if \( S(\omega, u_1, u_k) = (u_1, v_1, u_2, \ldots, u_{k-1}, v_k, u_k) \) is such that:
   a) For all \( l = 1, \ldots, k \), queues \( u_l \) and subcarriers \( v_l \) are distinct.
   b) Both queues \( u_l \) and \( u_{l+1} \) have connectivity to subcarrier \( v_l \), i.e. \( c_{vl, ul} = c_{vl-1, ul-1} = 1 \); and
   c) Queue \( u_l \) is served by subcarrier \( v_{l-1} \), i.e. \( \omega_{vl-1, ul} = 1 \).
   d) \( b_{u_l} - \omega_{v_l} \geq b_{u_k} - \omega_{v_k} + 2 \).

**Definition 4:** We say that \( \omega^b \) is the balancing of the allocation \( \omega \) along a balancing path \( S(\omega, u_1, u_k) = (u_1, v_1, u_2, \ldots, u_{k-1}, v_k, u_k) \) if each queue \( u_l \) is reasigned to be served by subcarrier \( v_l \), i.e. \( \omega_{vl, ul} = 1 \) and \( \omega_{vl-1, ul} = 0 \).

**Lemma 1:** An allocation \( \omega \in \mathcal{W}_k (b, C) \) which satisfies Condition (C1) also satisfies the Load-Balancing Condition (C2) if and only if it has no balancing path (Proof: see [14]).

Now, we have the following:

**Lemma 2:** Maximum weight matching (MWM) on the equivalent bipartite graph achieves MTLB assignment.

**Proof:** The maximum throughput condition (C1) is achieved by the fact that the weights are positive integers. We prove the load balancing condition (C2) by contradiction. Suppose the maximum weight matching \( M \) in the equivalent bipartite graph results in the allocation \( \omega \in \mathcal{W}_k \) that achieves maximum throughput but does not produce the most balanced queues. Then, from Lemma 1, we know that there exists a balancing path from queue \( u_j \) to \( u_i \) such that \( b_j - \omega_j \geq b_i - \omega_i + 2 \). Thus, in the equivalent bipartite graph, the node \( a_{i\omega_i} \) with weight \( b_i - \omega_i + 1 \) is matched while the node \( a_{j\omega_j+1} \) with weight \( b_j - \omega_j \) is not matched. We can construct another matching \( M' \) whose total weight is greater than that of \( M \) by reassigning subcarriers along the balancing path according to the balancing of the allocation \( \omega \). Thus, node \( a_{j\omega_j+1} \) is matched instead of node \( a_{i\omega_i} \) and the total weight is now increased by \( b_j - \omega_j - (b_i - \omega_i + 1) \geq 1 \). A contradiction to the maximum weight matching \( M \).

An example of MTLB assignment based on maximum weight matching is shown in Figure 3. The worst case running time of our MTLB based on the maximum weight matching algorithm is \( O(N \cdot K^3) \) ([12]).

### IV. SIMULATION RESULTS

We consider OFDMA system in a single cell with one base station, 64 users and 128 subcarriers. We assume 25-tap multipath channels with exponential multipath intensity profile. We compare the performance of MTLB policy and water-filling policy by simulation with on-off channel and independent homogeneous users. The pre-assigned transmit powers, varied to compensate for different distances of the users, are selected such that on average about 6.4 subcarriers are considered ‘on’ for each user. Two scenarios of packet arrivals are considered: Poisson and bursty. In case of bursty traffic, we consider heavy tail distribution where we pick an exaggerated distribution to illustrate the impact of arrival bursts.

We compare the performance of MTLB policy and water-filling policy on the average total queue length. Under an on-off channel model, the assignment based on water-filling policy is not unique. Thus, we consider three variations of water-filling: 1) WF-FIX, which allocates subcarriers to users in a fixed priority order; 2) WF-REV, which reverses the priority order every timeslot; and 3) WF-PERM, which assigns the subcarriers randomly (random priority).

Figure 4 shows the comparison of the average queue length at each timeslot for Poisson arrivals at the rate near the stable region (less than two packets/user/timeslot). The growing average queue length and instability of WF-FIX can be explained by the fact that the high priority queues.
MTLB stabilizes the queues while WF-REV and WF-PERM cannot. Figure 6 shows the average delay over the last 100 timeslots as a function of arrival rates. The average delays grow exponentially as the arrival rates closer to the edges of the stable arrival regions. Although MTLB outperforms the proposed WF-based policies, WF-PERM shows potential to rate stability. This suggests a promising future research direction.

**V. CONCLUSION AND FUTURE RESEARCH**

In this paper, we showed through a counter example that although water-filling based subcarrier allocation policies maximize the instantaneous throughput, they in general fail to provide rate-stability for an otherwise stabilizable OFDMA system. We used this example to argue that water-filling is too myopic and ignores variable state (queue length) information. We focused on a symmetric on-off channel model and constructed the MTLB policy that achieved the instantaneous maximum throughput as well as balancing the queue lengths. The MTLB policy was proved in [13] to achieve the minimum average delay and hence the long-run throughput optimality. By simulations, the MTLB policy was demonstrated to outperform water-filling-based policies.

For practical implementation with a large number of users and subcarriers, we hope to use randomized algorithm to reduce the running time and simplify the MTLB calculation [16]. In addition, the implementation of MTLB requires the existence of a centralized controller. This could be problematic for a system with large number of users and subcarriers. We hope to construct a decentralized allocation algorithm in our future research.

Future work will extend to a more realistic channel models where there are multiple modulation and coding schemes available (a subcarrier can now support multiple rates i.e. $0 \leq c_{kn} \leq c_{kn}^{max}$ is an integer representing the number of packets that can be transmitted). Figure 7 shows an example to illustrate that under more realistic channel models it may not be possible to satisfy both MT condition (C1) and LB condition (C2). The example shows that it is optimal to sacrifice instantaneous throughput in order to take the state of the system into a better (more balanced) state in anticipation of loss of connectivities in the future i.e. to better take advantage of the future multiuser diversity. This conceptually means that under more realistic channel models maximizing throughput is neither sufficient nor necessary. When optimality condition is relaxed to be throughput optimality, a Max-Weight policy is shown in [15] to be optimal by providing stability. The Max-Weight policy is shown to provide throughput optimality whenever the arrival process satisfies the strong law of large number and is admissible. However, this policy does not guarantee...

---

**Fig. 4.** Average queue length at each timeslot for MTLB, WF-FIX, WF-REV and WF-PERM algorithms. Assume Poisson distribution for the arrival packets with mean arrival rate $\lambda = 1.9$ packets/queue/timeslot.

**Fig. 5.** Average queue length at each timeslot for MTLB, WF-FIX, WF-REV and WF-PERM policies. Assume bursty arrival packets with mean $\mu = 1.9$ packets/queue/timeslot i.e. probability of 0 or 1 arriving packets is 0.493 while the probability for 100 arriving packets is 0.013.

**Fig. 6.** Average total queue length (averaging over the last 100 slots) for 8000-timeslot simulation for MTLB, WF-FIX, WF-REV and WF-PERM algorithms. Assume bursty packet arrivals with the mean arrival rate varying from 1.6 to 2.2 packets/queue/timeslot.
delay optimality. This is because the packet in shorter queues or lower-rate queues are delayed and thus packet delays might not be the appropriate objective. We intend to consider a utility-based optimization (see [17]) while considering the dynamics of arrivals and connectivity. Maximizing utility function also has a further benefit that it is able to automatically balance resource efficiency and fairness. We also hope to adapt similar max-weight scheduling methods for 3G CDMA High-Data-Rate (HDR) to OFDMA subcarrier allocation problem. These methods are modified largest weighted delay first (M-LWDF) [18] and exponential rule [19].

As seen above, when a system is stabilizable, MTLB provides an optimal average delay by fully taking advantage of multiuser diversity. In heavy load situations, where no policy can stabilize the system, MTLB, Max-Weight, M-LWDF as well as other load-balancing algorithms favor users with the highest rate (longest queue backlogs). This results in unfairness. Furthermore, when combined with a distributed implementation, this gives rise to incentive issues. In particular, the users will have incentive to exaggerate their constraint in broadband OFDMA networks, "IEEE Wireless Comm. and Networking", v. 2, pp. 1037-1042, March 2003.


Fig. 7. Example of multiple values of packet transmission capacity showing that water-filling causes unbounded queue buildups. We assume that the packet arrival process and the channel connectivity process have periodic structures with period of six timeslots. (a) Packet arrivals for Users 1 and 2 (b) Multi-level channel connectivity profile (c) Stabilizing policy π∗ per timeslot (d) Modified largest weighted delay first (M-LWDF) (e) Queue occupancies over time: queue lengths for users 1 and 2 under π, (b1, b2), and under π∗, (b∗1, b∗2). Under policy π, queue of user 2 (b2) is building up over time.

ACKNOWLEDGMENT
This work was supported in part by the National Science Foundation ADVANCE Cooperative Agreement No. SBE-0123552, and in part by the UW RRF Grant 65-2139.

REFERENCES


