Fast Optimal Bit and Power Allocation Based on the Lagrangian Method for OFDM Systems

Sang-Min LEE†‡, Student Member and Dong-Jo PARK†, Member

SUMMARY This paper examines the bit and power allocation problem for orthogonal frequency division multiplexing systems in which the overall transmission power is minimized by constraining the fixed data rate and bit error rate. To provide the optimal allocation with less computational complexity, we propose new bit and power allocation schemes based on the Lagrangian method. Firstly, we propose an initial search range of the bisection search method to find the optimal Lagrangian multiplier efficiently. The simulation results verify that the proposed initial search range guarantees the optimal solution with less computational complexity. Secondly, a new iterative search method for the optimal Lagrangian multiplier is proposed using Newton’s search method. The simulation results demonstrate that the proposed scheme has significant computational advantages over the conventional algorithms while providing optimal performance.

texts: multi-carrier, OFDM, bit and power allocation, adaptive modulation

1. Introduction

Recent research results have shown that orthogonal frequency division multiplexing (OFDM) is a promising technique for achieving high data rates and combating multipath fading in wireless communication systems [1, 2]. The OFDM techniques are employed in various wireless standards, such as wireless local area network (WLAN), IEEE 802.11a/g and HIPERLAN/2; wireless metropolitan network (WMAN), IEEE 802.16a/d/e; and digital audio and video broadcasting, DAB/DVB, among others. Although high data rates can be achieved, the OFDM systems have a weak point: the subcarrier in deep fading exists in the Rayleigh multipath channel. The performance degradation due to the subcarrier in deep fading is a critical problem in the OFDM systems.

To mitigate the multipath fading problem, many bit and power allocation techniques have been developed [3]–[7]. In order to minimize the performance degradation due to fading and improve the spectral efficiency of multicarrier transmissions, bits and power can be allocated to subcarriers according to their channel qualities, which are assumed to be known by the transmitter. In particular, the subcarriers with large channel gains employ higher order modulations to carry more bits per symbol, while the subcarriers with low channel gains carry one or even zero bits per symbol.

Bit and power allocation problems for multi-carrier transmission schemes have been defined variously according to criteria such as the maximization of the data rate and the minimization of the transmission power or bit error rate (BER). Many algorithms for these bit and power allocation problems have been proposed. Hughes-Hartogs’ algorithm [3] can be applied to minimize the total transmission power with the data rate constraint, or to maximize the data rate with the total transmission power constraint. Hughes-Hartogs’ algorithm provides the optimal allocation by adding one bit at a time to the subchannel, requiring the smallest additional power to increase its data rate. Thus, this algorithm requires intensive computations.

Chow’s algorithm [4] attempts to minimize the total transmission power with less computational complexity. To reduce computational complexity, Chow’s algorithm uses a suboptimal method that relies on rounding to integer rates and the signal-to-noise ratio (SNR) gap approximation [8] to allocate bits to subchannels. Fischer’s algorithm [5] attempts to minimize BER, but the resulting allocation is extremely close to the allocation where the total transmission power is minimized. Fischer’s algorithm shows the improvement of overall SNR with less complexity than Chow’s algorithm. Krongold [6] uses the traditional Lagrange techniques to find the optimal allocation effectively. An optimal Lagrangian multiplier is obtained by an iterative search method. Krongold’s algorithm provides the optimal allocation, and has less computational complexity than other optimal allocation algorithms.

However, in the above methods, there are still too many computational complexities for practical wireless communication systems which require frequent allocations due to time-varying channel characteristics. In this paper, new schemes based on Krongold’s method are proposed to reduce the computational complexity. Firstly, we propose an initial search range of the bisection search method to find the optimal Lagrangian multiplier, which is required to solve the bit and power allocation problem. Secondly, a new iterative search method for the optimal Lagrangian multiplier is proposed using Newton’s search method [9] with the approximated partial derivative of the data rate.

The organization of this paper is as follows. We describe the optimal bit and power allocation algorithm using the Lagrangian method in Sects. 2 and 3. In Sect. 4, we introduce a bisection method in [6] to search for the optimal Lagrangian multiplier. The upper and lower bounds of the optimal Lagrangian multiplier are derived in Sect. 5. In

Notes

† The authors are with the Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology (KAIST), 373-1 Guseong-dong, Yuseong-gu, Daejeon, Republic of Korea.
a) E-mail: lesam@kaist.ac.kr
DOI: 10.1093/ietcom/e89-b.4.1346

Manuscript revised October 26, 2005.
Manuscript received May 17, 2005.

Copyright © 2006 The Institute of Electronics, Information and Communication Engineers
2. Problem Formulation

We consider an adaptive OFDM system in which the transmitter uses combined bit and power allocation algorithms based on the channel information. A modulation mode for each subcarrier is selected corresponding to the number of bits allocated to the subcarrier, and the symbol modulated by the selected mode is then scaled to the allocated power. We define \( c(k) \) and \( p(k) \) as the number of allocated bits and the transmission power level of the \( k \)th subcarrier, respectively.

Assuming that the noise of each subcarrier has the same power, we denote \( f(c) \) as the required received power to satisfy a given BER requirement in a \( c \) bits/symbol modulation scheme. \( f(c) \) can be obtained by using the SNR-BER curves of the corresponding modulation schemes and the noise variance \([10]\). The BER of square M-ary quadrature amplitude modulation (M-QAM) is approximated to

\[
\text{BER} = \frac{2}{\log_2 M} \left( 1 - \frac{1}{\sqrt{M}} \right) \cdot \text{erfc} \left( \sqrt{\frac{3\sigma_n^2}{2(M-1)\sigma_n^2}} \right),
\]

where \( \sigma_n^2 \) and \( \sigma_p^2 \) are the average power of received data symbols and the noise power, respectively. \( \text{erfc}(\cdot) \) is the well-known complementary error function, which is defined by

\[
\text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty \exp(-t^2) dt.
\]

For the square M-QAM, \( c = \log_2 M \), and \( f(c) \) is defined by

\[
f(c) = \frac{2(2^c - 1)}{3} \cdot \sigma_n^2 \cdot \left( \text{erfc}^{-1} \left( \frac{\text{BER}_t \cdot c}{2 - 2^{1-c/2}} \right) \right)^2,
\]

where \( \text{BER}_t \) is target BER.

Assuming that the channel state information is known, the transmission power of the \( k \)th subcarrier can be given by

\[
p(k) = f(c(k))/|H(k)|^2,
\]

where \( H(k) \) is the channel gain of the \( k \)th subcarrier. Thus the combined bit and power allocation algorithm should find the optimal assignment of \( c(k) \) so that the total transmission power is minimized satisfying the transmission rate and BER requirements. The optimization problem can be mathematically formulated as following:

\[
\min_{c(k) \in D} \sum_{k=1}^K f(c(k))/|H(k)|^2
\]

subject to \( \sum_{k=1}^K c(k) = R_T \),

where \( D \) is the set of all possible values for \( c(k) \), \( K \) is the number of subcarriers, and \( R_T \) is a rate requirement. Because of spectral efficiency and ease of implementation, we employ square QAM, which gives \( D = \{0, 2, 4, 6, \ldots\} \).

3. Optimal Bit and Power Allocation

In order to make the optimization problem tractable, (5) can be reformulated as the following unconstrained optimization problem through the Lagrangian multiplier \( \lambda \) if we relax \( D \) to allow to be a nonnegative real number \([6]\):

\[
\min L(c(k), \lambda) = \sum_{k=1}^K \frac{f(c(k))}{|H(k)|^2} - \lambda \left( \sum_{k=1}^K c(k) - R_T \right).
\]

For the optimal bit allocation \( \{c^*(k), \forall k\} \) and the optimal Lagrangian multiplier \( \lambda^* \), the Lagrange cost \( L \) should be minimized, and partial derivatives are equal to zero as

\[
\frac{\partial L}{\partial c(k)}\bigg|_{c(k)} = 0, \forall k \quad \text{and} \quad \frac{\partial L}{\partial \lambda} = 0.
\]

Substituting (6) into these partial derivatives, the following equations can be obtained:

\[
\frac{\partial f(c(k))}{\partial c(k)}\bigg|_{c(k)} = \lambda^*|H(k)|^2, \forall k,
\]

\[
\sum_{k=1}^K c^*(k) = R_T.
\]

From (8), the optimal bit allocation \( \{c^*(k), \forall k\} \) can be uniquely determined because \( f(\cdot) \) is a convex function \([6]\). Here we consider square QAM as available modulation schemes. Therefore \( c^*(k) \) should be selected among even numbers. \( c(k) \) is determined by thresholding \( \lambda^*|H(k)|^2 \) at the levels of \( \{\theta_n, \forall n\} \) as

\[
c(k) = \begin{cases} 2n, & \text{if } \theta_n < \lambda^*|H(k)|^2 \leq \theta_{n+1}, \forall k, \\ 0, & \text{if } \lambda^*|H(k)|^2 < \theta_n, \forall k, \end{cases}
\]

where

\[
\theta_n = (f(2n) - f(2n - 2))/2, \quad \text{for } n = 1, 2, 3, \ldots
\]

The optimal Lagrangian multiplier \( \lambda^* \) provides the optimal bit allocation \( \{c^*(k), \forall k\} \). From (4), the optimal power allocation is given by

\[
p^*(k) = f(c^*(k))/|H(k)|^2.
\]

4. Bisection Method for the Optimal Lagrangian Multiplier

For a given Lagrangian multiplier, the bit and power allocation information, \( c(k) \) and \( p(k) \), can be computed by (9) and (4) respectively. Then the total transmission power and total data rate are obtained by summing as

\[
R = \sum_{k=1}^K c(k), \quad P = \sum_{k=1}^K p(k).
\]
The total data rate $R$ is proportional to the Lagrangian multiplier $\lambda$, and thus $R$ can be adjusted to the rate requirement $R_T$ by searching the optimal Lagrangian multiplier $\lambda'$.

The optimal Lagrangian multiplier $\lambda'$ can be found by using a bisection method [6]. The bisection method uses two previously evaluated Lagrangian multipliers, $\lambda_l$ and $\lambda_h$. Corresponding to $\lambda_l$ and $\lambda_h$, the power and the number of bits allocated to the $k$th subcarrier are denoted by $p_l(k)$, $p_h(k)$, $c_l(k)$ and $c_h(k)$, respectively. By summing $p_l(k)$, $p_h(k)$, $c_l(k)$ and $c_h(k)$ along the subcarrier index $k$, total transmission powers $P_l$ and $P_h$ and total data rates $R_l$ and $R_h$ can be obtained.

The bisectional search method lowers the gap between $\lambda_l$ and $\lambda_h$ by searching the optimal Lagrangian multiplier $\lambda$. The optimal Lagrangian multiplier can be found by computing the following updated Lagrangian multiplier:

$$\lambda_{new} = \frac{P_h - P_l}{R_h - R_l} \quad (13)$$

Then, from (9), the total data rate $R_{new}$ corresponding to $\lambda_{new}$ is evaluated. If $R_{new}$ is greater than the rate requirement $R_T$, we update $\lambda_h$ with $\lambda_{new}$ while keeping $\lambda_l$ the same. The opposite update is done if $R_{new}$ is less than $R_T$. An example of the bisectional search method is shown in Fig. 1. These update procedures are repeated until $R_{new}$ equals to either $R_l$ or $R_h$. Table 1 shows the summary of the bit and power allocation algorithm using the bisectional search method.

5. Bounds of the Optimal Lagrangian Multiplier

The bisection method, presented in the previous section, requires an initial search range, that is, $[\lambda_l, \lambda_h]$. As this initial search range increases, the number of iterations increases. Moreover, for a narrow initial search range, the convergence to the optimal solution can not be guaranteed. To overcome this problem, we will derive the upper and lower bounds of the optimal Lagrangian multiplier and set the initial search range between these bounds. While the initial search range is conventionally constant regardless of time-varying channel characteristics, the proposed initial search range is adjusted instantaneously to the varying channel information at every search. Therefore, our initial search range is narrower than conventional ones, and a fewer number of iterations is required to converge. Furthermore, because the optimal solution always exists in our initial search range, we can guarantee the convergence to the optimal solution.

Before deriving the upper and lower bounds of the optimal Lagrangian multiplier, we will derive the data rate corresponding to new thresholds $[\theta_n(\delta), \forall n]$ which have the constant logarithmic difference $\delta$. The set of new thresholds $[\theta_n(\delta), \forall n]$ is defined as following:

$$\theta_n(\delta) = \theta_1 \cdot (\exp(\delta))^{n-1}, \quad \text{for } n = 1, 2, 3, \cdots. \quad (14)$$

The logarithm of $\hat{\theta}_n(\delta)$ can be expressed as

$$\log \hat{\theta}_n(\delta) = \log \theta_1 + \delta \cdot (n - 1). \quad (15)$$

For a certain $\delta$, $\hat{\theta}_n(\delta)$ is very close to $\theta_n$ because the logarithmic difference of $\theta_n$, $\log \theta_{n+1} - \log \theta_n$, changes very slightly according to $n$ as shown in Table 2.

Let $\hat{c}(k)$ denote the number of bits allocated by using the new thresholds $[\theta_n(\delta), \forall n]$. Then $\hat{c}(k)$ can be obtained by

$$\begin{align*}
\frac{n}{\text{value}} & \quad 1 & \quad 2 & \quad 3 & \quad 4 \\
\text{value} & \quad 0.5803 & \quad 0.5858 & \quad 0.5883 & \quad 0.5899
\end{align*}$$

Table 2

Fig. 1 Bisectional search method.

Table 1 Iterative allocation method [6].
the following equation:

\[ \hat{c}(k) = \begin{cases} 
2n, & \text{if } \hat{\theta}_n(\delta) < \lambda[H(k)]^2 \leq \hat{\theta}_{n+1}(\delta), \\
0, & \text{if } \lambda[H(k)]^2 \leq \hat{\theta}_1(\delta). 
\end{cases} \]  

(16)

By using (15), the two conditions in (16) can be rewritten as

\[ \log \theta_1 + \delta \cdot (n - 1) < \lambda[H(k)]^2 \leq \log \theta_1 + \delta \cdot n, \]

(17)

or

\[ n - 1 < \frac{\lambda[H(k)]^2 - \log \theta_1}{\delta} \leq n, \]

(18)

Using a ceiling function, which is denoted by \( \lceil \cdot \rceil \), the first inequality in (18) becomes

\[ \lceil \frac{\lambda[H(k)]^2 - \log \theta_1}{\delta} \rceil = n. \]

(19)

Then, (16) can be expressed as

\[ \hat{c}(k) = \begin{cases} 
2 \cdot \lceil (\log(\lambda[H(k)]^2) - \log \theta_1)/\delta \rceil, & \text{if } \lambda[H(k)]^2 - \log \theta_1 > 0, \\
0, & \text{if } \lambda[H(k)]^2 - \log \theta_1 \leq 0, 
\end{cases} \]

(20)

and simplified to

\[ \hat{c}(k) = 2 \cdot \max \left( \left\lfloor \frac{\log(\lambda[H(k)]^2) - \log \theta_1}{\delta} \right\rfloor, 0 \right). \]

(21)

5.1 Upper Bound of the Optimal Lagrangian Multiplier

The maximum logarithmic difference of \( \theta_n \) is denoted by \( \delta_u \) as

\[ \delta_u = \max_{n \in \{1, 2, 3, \ldots, n\}} (\log \theta_{n+1} - \log \theta_n), \]

(22)

and let \( \hat{c}_n(k) \) denote the number of bits allocated by the thresholds \( \hat{\theta}_n(\delta) \). \( \hat{\theta}_n(\delta) \) is not greater than \( \theta_n \) for all \( n \) because \( \delta_u \) is the maximum logarithmic difference. \( c(k) \) is not less than \( \hat{c}_u(k) \), that is, \( c(k) \geq \hat{c}_u(k) \), because higher thresholds give smaller number of allocated bits. Then the following inequality can be obtained:

\[ R = \sum_{k=1}^{K} c(k) \geq \sum_{k=1}^{K} \hat{c}_u(k). \]

(23)

By applying (21) and the inequalities of \( \lceil x \rceil \geq x \) and \( \max(x, y) \geq x \) to (23), the following inequality can be obtained:

\[ R \geq \sum_{k=1}^{K} 2 \cdot \max(\log(\lambda[H(k)]^2) - \log \theta_1, 0) \]

\[ \geq \sum_{k=1}^{K} 2 \cdot \frac{\max(\lambda[H(k)]^2 - \log \theta_1)}{\delta_u} \]

(24)

As the optimal Lagrangian multiplier \( \lambda^* \) gives the optimal bit allocation \( c^*(k) \) which satisfies the rate requirement \( R_T \) the following inequality can be obtained:

\[ R_T \geq \sum_{k=1}^{K} 2 \cdot \frac{\log(\lambda[H(k)]^2) - \log \theta_1}{\delta_u}. \]

(25)

From the above equation, the following upper bound of the optimal Lagrangian multiplier is derived:

\[ \lambda^* \leq \frac{\theta_1}{\prod_{k=1}^{K} \frac{\delta_u \cdot R_T}{2K}} \exp \left( \frac{\delta_u \cdot R_T}{2K} \right). \]

(26)

5.2 Lower Bound of the Optimal Lagrangian Multiplier

The minimum logarithmic difference of \( \delta \) is denoted by \( \delta_l \) as

\[ \delta_l = \min_{n \in \{1, 2, 3, \ldots, n\}} (\log \theta_{n+1} - \log \theta_n), \]

(27)

and let \( \hat{c}_l(k) \) denote the number of bits allocated by the thresholds \( \hat{\theta}_l(\delta) \). \( \hat{\theta}_l(\delta) \) is not greater than \( \theta_n \) for all \( n \) because \( \delta_l \) is the minimum logarithmic difference. As lower thresholds give a larger number of allocated bits, \( c(k) \) is not higher than \( \hat{c}_l(k) \), that is, \( c(k) \leq \hat{c}_l(k) \). Then the following inequality can be obtained:

\[ R = \sum_{k=1}^{K} c(k) \leq \sum_{k=1}^{K} \hat{c}_l(k). \]

(28)

By applying (21) and the inequality of \( \lceil x \rceil \leq x + 1 \), the following inequality can be obtained:

\[ R \leq \sum_{k=1}^{K} 2 \cdot \max(\log(\lambda[H(k)]^2) - \log \theta_1, 0) \]

\[ \leq \sum_{k=1}^{K} 2 \cdot \frac{\max(\lambda[H(k)]^2 - \log \theta_1, 0)}{\delta_l} + 2K. \]

(29)

We define \( k_{\max} \) by the index of the subcarrier which has the largest subchannel gain as following:

\[ k_{\max} = \arg \max_{k \in \{1, 2, \ldots, K\}} |H(k)|^2. \]

(30)

By using \( k_{\max} \), (29) becomes

\[ R \leq 2K \cdot \frac{\max(\log(\lambda[H(k_{\max})]^2) - \log \theta_1, 0)}{\delta_l} + 2K. \]

(31)
For the optimal Lagrangian multiplier $\lambda^*$, we can obtain
\[
R_T \leq 2K \cdot \max_{\delta_l} \left( \log(\lambda^*|H(k_{\text{max}})|^2) - \log \lambda_1, 0 \right) + 2K. \tag{32}
\]

When $\lambda^*|H(k_{\text{max}})|^2 < \theta_1$, $c^*(k)$ becomes zero for all $k$ by (9), and thus $R_T = 0$, which means that no transmission is required. The case of $R_T = 0$ will not be considered here because the bit and power allocation is unnecessary. For $R_T > 0$, $\lambda^*|H(k_{\text{max}})|^2$ is not less than $\theta_1$. Then, for $\lambda^*$, (32) becomes
\[
R_T \leq 2K \cdot \frac{\log(\lambda^*|H(k_{\text{max}})|^2)}{\delta_1} - \log \theta_1 + 2K. \tag{33}
\]

From the above equation, the following lower bound of $\lambda^*$ can be obtained:
\[
\lambda^* \geq \frac{\theta_1}{|H(k_{\text{max}})|^2} \exp\left(\frac{R_T}{2K} - 1\right). \tag{34}
\]

Figure 2 shows the optimal Lagrangian multiplier $\lambda^*$ and its lower and upper bounds for Rayleigh fading channels with an exponential power delay profile of 500 ns root mean square (RMS) delay spread. The upper bound relies on all subchannel gains as shown in (26), but the lower bound relies on only the largest subchannel gain as shown in (34). For this reason, the upper bound is tighter than the lower bound.

6. Fast Search for the Optimal Lagrangian Multiplier

The computational load of the bisection method can be reduced by using the previously derived upper and lower bounds as the initial search range, but the computational load is still too much for the practical OFDM systems. In this section, we will improve the search method for the optimal Lagrangian multiplier to achieve lower computational complexity.

The logarithms of the upper and lower bounds in (26) and (34) are linear functions of $R_T$. Furthermore, $R_T$ and $\log \lambda^*$ have approximately linear relationship, as shown in Fig. 2. A fast iterative search scheme for $\lambda^*$ will be proposed based on this linear relationship. Let $R(\lambda)$ denote the number of loaded bits per an OFDM symbol according to $\lambda$. From (24), $R(\lambda)$ has the following lower bound:
\[
R(\lambda) \geq \frac{2K}{\delta_u} \log \lambda + 2 \sum_{k=1}^{K} \log |H(k)|^2 - K \log \theta_1. \tag{35}
\]

The cost function to find the optimal Lagrangian multiplier $\lambda^*$ is defined by
\[
J(\lambda) = (R_T - R(\lambda))^2. \tag{36}
\]

The partial derivative of $R(\lambda)$ with respect to $\log \lambda$ is approximated to the partial derivative of the lower bound of $R(\lambda)$ as
\[
\frac{\partial R(\lambda)}{\partial (\log \lambda)} \approx \frac{\delta_u}{2K} \cdot \frac{\partial^2 R(\lambda)}{\partial (\log \lambda)^2} \approx 0. \tag{37}
\]

Applying Newton’s method [9] to $J(\lambda)$ based on this approximation, we can obtain the following update equation of $\log \lambda$:
\[
\log \lambda^{(n+1)} = \log \lambda^{(n)} + \frac{\delta_u}{2K} \cdot (R_T - R(\lambda^{(n)})). \tag{38}
\]

The above equation can be expressed by
\[
\lambda^{(n+1)} = \lambda^{(n)} \cdot \exp\left(\frac{\delta_u}{2K} \cdot (R_T - R(\lambda^{(n)}))\right). \tag{39}
\]

The initial value of $\lambda^{(0)}$ is set to the upper bound of $\lambda^*$, which is closer to $\lambda^*$ than the lower bound as following:
\[
\lambda^{(0)} = \frac{\theta_1}{\prod_{k=1}^{K} |H(k)|^{2/K}} \exp\left(\frac{\delta_u \cdot R_T}{2K}\right). \tag{40}
\]

<table>
<thead>
<tr>
<th>Table 3</th>
<th>Proposed iterative allocation method.</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Set thresholds $\theta_n$ by $\theta_n = (f(2n) - f(2n - 2))/2$, for $n = 1, 2, 3, \cdots$.</td>
</tr>
<tr>
<td>2.</td>
<td>Set $n = 0$, and set the initial value of $\lambda^{(0)}$ by $\lambda^{(0)} = (\theta_1 / \prod_{k=1}^{K}</td>
</tr>
<tr>
<td>3.</td>
<td>Determine $c^{(0)}(k)$ and $R(\lambda^{(0)})$ by $c^{(0)}(k) = \begin{cases} 2n, &amp; \text{if } \theta_n &lt; \lambda^{(0)}</td>
</tr>
<tr>
<td>4.</td>
<td>$n = n + 1$, and update $\lambda^{(n)}$ by $\lambda^{(n)} = \lambda^{(n-1)} \cdot \exp\left(\frac{\delta_u}{2K} \cdot (R_T - R(\lambda^{(n)}))\right)$.</td>
</tr>
<tr>
<td>5.</td>
<td>Determine $c^{(n)}(k)$ and $R(\lambda^{(n)})$ by (T3) $R(\lambda^{(n)}) = \sum_{k=1}^{K} c^{(n)}(k)$.</td>
</tr>
<tr>
<td>6.</td>
<td>If $R(\lambda^{(n)}) &gt; R_T$, then go to Step 4. Elseif $n &gt; 1$ and $(R(\lambda^{(n)}) - R_T) \cdot (R(\lambda^{(n-1)}) - R_T) &lt; 0$, then use a bisectional method over $[\lambda^{(0)}, \lambda^{(n-1)}]$.</td>
</tr>
<tr>
<td>7.</td>
<td>Compute $p(k)$ by $p(k) = f(c^{(n)}(k))/</td>
</tr>
<tr>
<td>8.</td>
<td>Assign $c^{(n)}(k)$ and $p(k)$ to the $k$th subcarrier.</td>
</tr>
</tbody>
</table>
The convergence of the update Eq. (39) is not guaranteed due to the approximation in (37). However, \( \lambda^{(n)} \) does not diverge, and is bounded as
\[
\lambda^* \exp \left( -\frac{\delta_j}{2K} (R\lambda^{(0)} - R_T) \right) \leq \lambda^{(n)} \leq \lambda^{(0)}. \tag{41}
\]
The proof of (41) is shown in Appendix. \( \lambda^{(0)} \) converges to \( \lambda^* \), or oscillates between the bounds of (41).

To guarantee the convergence to the optimal value, the update Eq. (39) is combined with a bisection method. \( \lambda^{(n)} \) quickly approaches to the optimal value by the update Eq. (39). Because \( \lambda^{(0)} \) is set to the upper bound of \( \lambda^* \), \( \lambda^{(n)} \) becomes less than \( \lambda^* \) immediately after crossing \( \lambda^* \). To avoid the oscillations of \( \lambda^{(n)} \), when \( \lambda^{(n)} \) becomes less than \( \lambda^* \), the bisection method starts searching \( \lambda^* \) over the range between \( \lambda^{(n-1)} \) and \( \lambda^{(n)} \). This two stage method is described in Table 3.

7. Simulation Results

In this section, the simulation results are provided to illustrate the performance of the proposed bit and power allocation methods. Simulations were performed in an uncoded OFDM system with 256 subcarriers over a 10 MHz bandwidth. We considered the Rayleigh fading channels with an exponential power delay profile of 500 ns RMS delay spread. The target BER is set to \( 10^{-4} \). Square signal constellations (4-QAM, 16-QAM, 64-QAM, and 256-QAM) were used to carry two, four, six, or eight bits per symbol. The following allocation methods were simulated.

- Proposed: New iterative allocation method, as described in Table 3.
- m-Krongold: Krongold’s method with the proposed initial search range.
- Krongold-1: Krongold’s method with the initial search range \([1, 1000]\).
- Krongold-2: Krongold’s method with the initial search range \([3, 300]\).
- Chow: Chow’s method.

Figure 3 shows the transmit SNR which is the total transmission power divided by noise power. The target data rate varies from 256 bits to 1,280 bits per OFDM symbol. ‘Proposed’ has exactly the same transmit SNR as ‘Krongold-1’ and ‘Hughes-Hartogs,’ which provide the optimal allocation. This result verifies that ‘Proposed’ provides the optimal allocation. Since ‘Chow’ is a suboptimal allocation method, its transmit SNR is a little larger.

Figure 4 depicts the logarithm of the Lagrangian multipliers according to iteration stages, and Fig. 5 shows the average number of iterations required to converge to an optimal solution. The target data rate was set to 512 bits per OFDM symbol for both Figs. 4 and 5. In Fig. 5, the numbers of iterations were averaged over simulations with 10,000 different channel instances. Krongold’s method was simulated with two initial search ranges, \([1, 1000]\) and \([3, 300]\), which are denoted by ‘Krongold-1’ and ‘Krongold-2,’ respectively. For 10,000 randomly generated channel instances, ‘Krongold-1’ converged to an optimal solution, but ‘Krongold-2’ failed to converge with the probability of 0.065. Though ‘Krongold-1’ converges to an optimal so-
lution in the simulations, it does not guarantee the optimal solution. To guarantee the optimal solution, it should be searched within the proposed initial search range, which is denoted by ‘m-Krongold.’

In order to guarantee convergence, Krongold’s method requires a very large initial search range including the optimal solutions for all channel instances. But ‘m-Krongold’ uses a narrow initial search range between the upper and lower bounds adjusted to each channel instance. In Fig. 5, ‘m-Krongold’ provides an optimal allocation and converges faster than ‘Krongold-1’ and ‘Krongold-2.’ Also Fig. 5 shows that ‘Proposed’ converges to an optimal solution and reduces the number of iterations to half that of Krongold’s method.

Figure 6 depicts the computational load to solve the bit and power allocation problem. The algorithms were coded in MATLAB, the platform being used was an Intel Pentium 4 2.4 GHz personal computer, and the computation time was measured in seconds. ‘Hughes-Hartogs’ requires a very intensive computational load. Although ‘Chow’ provides a suboptimal solution, it requires a larger computational load than ‘Krongold-1,’ which provides the optimal allocation. As ‘m-Krongold’ and ‘Proposed’ have less number of iterations than ‘Krongold-1,’ their CPU time are less than ‘Krongold-1.’ Particularly, the CPU time of ‘Proposed’ is less than half that of ‘Krongold-1.’

8. Conclusions

In this paper, the bit and power allocation problem for OFDM systems was solved using the well-known Lagrangian method. First, the upper and lower bounds of the optimal Lagrangian multiplier were derived. By searching for an optimal value within these bounds, an optimal solution was guaranteed and computational complexity was reduced. Second, we proposed a new fast iterative allocation method based on Newton’s method. The simulation results showed that the proposed method provides an optimal allocation and reduces the number of iterations to half that of Krongold’s method.

References


Appendix: Derivation of Inequality (41)

A.1 Derivation of the inequality, \( \lambda^{(n)} \leq \lambda^{(0)} \)

The update Eq. (39) for \( \lambda^{(n)} \) can be rewritten as

\[
\lambda^{(n)} = \lambda^{(n-1)} \cdot \exp\left(\frac{\delta_{u}}{2K} (RT - R(\lambda^{(n-1)}))\right). \tag{A-1}
\]

From the lower bound (35) of \( R(\lambda) \), we can obtain

\[
\exp\left(-\frac{\delta_{u}}{2K} R(\lambda^{(n-1)}(0))\right) \leq \frac{\theta_{1}}{\lambda^{(n-1)}(0)} \prod_{k=1}^{K} |H(k)|^{2}\lambda^{(n-1)}. \tag{A-2}
\]

(A-1) and (A-2) give the following result:

\[
\lambda^{(n)} \leq \lambda^{(0)} \cdot \exp\left(\frac{\delta_{u} R_{T}}{2K}\right). \tag{A-3}
\]

A.2 Derivation of the inequality, \( \lambda^{(n)} \geq \lambda^{*} \exp\left(\frac{\delta_{u}}{2K} (RT - R(\lambda^{(0)}))\right) \)

The method of mathematical induction is used to prove that the following inequality holds for all \( n \):

\[
\lambda^{(n)} \geq \lambda^{*} \exp\left(\frac{\delta_{u}}{2K} (RT - R(\lambda^{(0)}))\right). \tag{A-4}
\]

We will prove the following two statements for the mathe-
matical induction.

- Inequality (A-4) holds for \( n = 0 \).
- If Inequality (A-4) holds for \( n = m-1 \), Inequality (A-4) holds for \( n = m \).

\( \lambda^{(0)} \) was set to the upper bound of \( \lambda^* \) in (40) as

\[
\lambda^{(0)} \geq \lambda^*. \tag{A-5}
\]

Based on (9) and (12), it can be easily shown that \( R(\lambda) \) is the increasing function of \( \lambda \). Thus we can obtain the following inequality:

\[
R(\lambda^{(0)}) \geq R(\lambda^*) = R_T. \tag{A-6}
\]

From (A-5) and (A-6), the lower bound of \( \lambda^{(0)} \) can be derived as

\[
\lambda^{(0)} \geq \lambda^* \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(0)})) \right). \tag{A-7}
\]

Thus the first statement for mathematical induction was proved.

To prove the second statement for the mathematical induction, let us assume that Inequality (A-4) holds for \( n = m-1 \). According to the range of \( \lambda^{(m-1)} \), we consider two cases, and prove that Inequality (A-4) holds for \( n = m \) at each case.

1) \( \lambda^{(m-1)} \geq \lambda^* \): The lower bound of \( \lambda^{(m)} \) is derived by using (39) and (A-3) as

\[
\lambda^{(m)} = \lambda^{(m-1)} \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(m-1)})) \right) \\
\geq \lambda^* \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(m-1)})) \right) \tag{A-8} \\
\geq \lambda^* \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(0)})) \right).
\]

2) \( \lambda^{(m-1)} < \lambda^* \): As \( R(\lambda) \) is the increasing function of \( \lambda \), \( R_T - R(\lambda^{(m-1)}) \leq 0 \). We can obtain the following inequality from (39):

\[
\lambda^{(m)} \geq \lambda^{(m-1)} \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(m-1)})) \right) \\
\geq \lambda^{(m-1)}. \tag{A-9}
\]

By using (A-4) for \( n = m - 1 \), the lower bound of \( \lambda^{(m)} \) is derived as

\[
\lambda^{(m)} \geq \lambda^* \exp \left( \frac{\delta_u}{2K}(R_T - R(\lambda^{(0)})) \right). \tag{A-10}
\]

Therefore Inequality (A-4) holds for \( n = m \) regardless of the range of \( \lambda^{(m-1)} \).