JOURNAL OF KOREA MULTIMEDIA SOCIETY, VOL. 9, NO. 12, DECEMBER 2006(pp. 1636-1648)

A New Approach to Solve the Rate Control Problem in Wired-cum-Wireless Networks

Le Cong Loi†, Won-Joo Hwang††

ABSTRACT

In this paper, we propose a new optimization approach to the rate control problem in a wired-cum-wireless network. A primal-dual interior-point (PDIP) algorithm is used to find the solution of the rate optimization problem. We show a comparison between the dual-based (DB) algorithm and PDIP algorithm for solving the rate control problem in the wired-cum-wireless network. The PDIP algorithm performs much better than the DB algorithm. The PDIP can be considered as an attractive method to solve the rate control problem in network. We also present a numerical example and simulation to illustrate our conclusions.

Keywords: Wired-cum-wireless networks, Rate control problems, Optimization problems, Algorithms, Convex programming, Non-convex programming, Convex functions, Concave functions

1. INTRODUCTION

In recent years, a rate control problem in wired networks has been extensively studied[1-4]. It has been proved that, since the feasible rate region can be represented by a set of simple, separable, convex constraints in the wired networks, the globally fair rate is attainable via distributed approaches based on convex programming. Besides, the rate optimization for single-hop flows in wireless networks has also been widely considered. In[5], Tassiulas and Sarkar have proposed a centralized algorithm to attain max-min fair rate in certain ad hoc networks. On the other hand, Nadagopal et al. [6] and Ozugur et al.[7] have presented decentralized algorithms that try to achieve some fair rate allocations. Recently, Wang, Kar and Low[8] have considered the rate control problem in a so-called wired-cum-wireless network where a session may run across both wired and wireless links. In a typical wired-cum-wireless network, mobile hosts (MHs), such as laptop computers, peripherals and storage devices can roam in the wireless network, called basic service sets (BSSs), which are attached at the periphery of a wired backbone (infrastructure). The wired infrastructure can be an IEEE 802 style Ethernet LAN or some other IP based networks. An example of a typical wired-cum-wireless network is shown in Fig. 1.

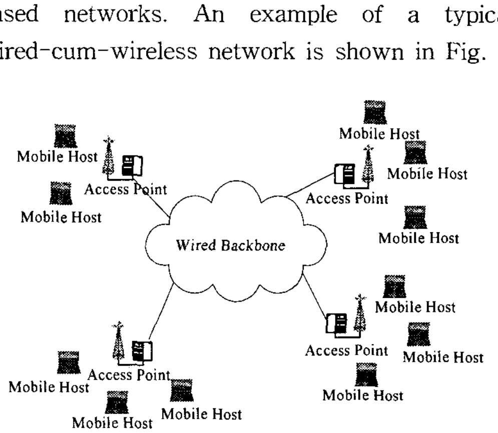


Fig. 1. A wired-cum-wireless network.

※ Corresponding Author : Won-Joo Hwang, Address : (621-749) 607 Obang-dong, Gimhae, Gyungnam, Korea, TEL : +82-55-320-3847, FAX : +82-322-6275, E-mail : ichwang@inje.ac.kr
Receipt date : Jan. 15, 2004, Approval date : Jan. 15, 2004
† Department of Information and Communications Engineering Inje University
(E-mail : loilc@vnu.edu.vn)
†† Department of Information and Communications Engineering Inje University
※ This work was supported by 2006 INJE University Post-Master Research Grant.

The wired and wireless networks are inter-connected via access points (APs), which are actually fixed based stations that provide interfaces between the wired and wireless parts of the network, and control each BSS. The MHs can roam from one BSS to another. A MH within a BSS can only access the infrastructure through its AP, and it is assumed that in each BSS all the MHs are within the broadcast region of that particular AP.

In[8], the rate control problem in the wired-cum-wireless network has been introduced. It is different to the rate optimization in the wired networks which is a convex programming problem [2,3], that is, the former is not a convex programming problem. This makes the rate control problem in the wired-cum-wireless network even more difficult to solve. To overcome this difficulty, authors of[8] have proposed a DB distributed algorithm to iteratively solve the problem of rate optimization in the wired-cum-wireless network. Note that, in this network model, one could find not only the best rate for each session but also the transmission rates on the wireless links that support the session rates. In fact, the DB algorithm solves iteratively the rate control problem in the wired-cum-wireless network by outer iterations and inner iterations. The inner iterations work at the transport layer to solve the rate control problem when the wireless links therein have fixed capacities and the link transmission rates are given by the gradient projection method. The problem is essentially the same as the end-to-end proportional fair rate control problem in the wired network[2]. After carrying out the inner iterations, the DB algorithm will work at the link layer to update the transmission rate of the wireless links using also the gradient projection algorithm for solving another optimization problem. Noting that, the convergence rate of the gradient projection algorithm is only linear.

The PDIP method is known to be an omnipresent, conspicuous feature of the constrained optimization landscape today[9]. Moreover, under the suitable assumption, the PDIP method converges at a quadratic rate[10]. Boyd and Vandenberghe [11] have applied a barrier method to solve the rate control problem in the wired network. On the other hand, the PDIP method is often more efficient than the barrier method, especially when high accuracy is required, since they can exhibit better than linear convergence.

Motivated by an impressive computational performance of the PDIP algorithm, we solve the rate control problem in the wired-cum-wireless network using the PDIP algorithm. Although we can solve a non-convex problem using the PDIP method, in this paper, we first transform the rate control problem of the wired-cum-wireless network which is a non-convex problem into an equivalent problem that is a convex programming problem, and then, we describe the implementation of the PDIP algorithm to solve the equivalent problem. An interesting feature of Wang, Kar and Low's approach is that its computation is simple, whereas in our approach the computational time is low. See a more detailed comparison in the end of Section 4.

The present paper is structured as follows. In Section 2, we review the rate optimization problem in the wired-cum-wireless network and propose an equivalent convex problem. The description of the DB distributed algorithm is discussed in Section 3. In Section 4, the PDIP algorithm is described in detail, and in Section 5, the simulation results show that the PDIP converges significantly faster than the DB algorithm of[8]. Section 6 concludes the paper with remarks on future research topics of interest.

2. OPTIMIZATION PROBLEM

We consider a general wired-cum-wireless network that consists of a set M of all MHs, a set W of CSMA/CA based BSSs, a set N of fixed nodes in the wired backbone and a set L of unidirectional links which connect the fixed nodes in the wired backbone. Similarly in[8], we assume that each MH belongs to one and only one BSS, and each BSS has one and only one AP. Each node has a single transceiver and a node can not transmit and receive simultaneously, and can not receive more than one frame at a time.

For each BSS $w, w \in W$, we let N_w be the set of nodes in that particular BSS w. Noting that BSS w has only one AP, hence we denote $A(s)$ as the AP associated with MH s, i.e., $A(s) = A_w$ if $s \in N_w$, where A_w is an AP of the BSS w. In BSS w, a link exists between two nodes if and only if they can receive each other's signals. A directed edge (s,t) represents an active communication pair for $s,t \in N_w$, and E_w is the set of directed edges in BSS w. For any node $s \in N_w$, let $D_s = \{t : (s,t) \in E_w\}$ and $J_s = \{t : (t,s) \in E_w\}$ be the set of neighbors to which s is sending traffic and the set of neighbors from which s is receiving traffic, respectively. We further assume that the scheduling point process for each used link $(s,t) \in E_w$ is Poisson and is independent with other processes in the network. In addition, the frame lengths are assumed to be exponentially distributed. The transmission rate for a wireless link (s,t) is denoted as $\rho_{s,t}$, which is the average Poisson transmission attempts made during an average frame transmission time. On the other hand, the propagation delay is assumed to be zero in the network model. The lengths of Request to Send (RTS) and Clear to Send (CTS) are also assumed to be very small, their transmission time can be ignored, and acknowledgments are obtained instantaneously.

By end-to-end sessions within a BSS are not allowed according to the assumption, an immediate result is that all links in the BSS are between its MHs and the AP, and therefore no two links in a BSS can be scheduled at the same time. Let $\rho = (\rho_{s,t} : (s,t) \in E_w, w \in W) \in \Re_+^{|M|}$ be the vector of transmission rates for all wireless links in the wired-cum-wireless network, where $|M|$ denotes its cardinality. It has recently been shown[12] that the attainable throughput on link (s,t) in BSS w, in which either s or t must be the AP A_w, can be expressed as

$$c_{s,t}(\rho) = \frac{\rho_{s,t}}{1 + \sum_{k \in D_{A_w}} \rho_{A_w,k} + \sum_{k \in J_{A_w}} \rho_{k,A_w}}, (s,t) \in E_w. \tag{1}$$

Here, noting that the terms $\sum_{k \in D_{A_w}} \rho_{A_w,k}$ and $\sum_{k \in J_{A_w}} \rho_{k,A_w}$ are the sums of transmission rates on all downlinks and uplinks respectively in BSS w.

The wired backbone of the network connects the set of APs using the set L of unidirectional wired links whose capacity is denoted by $c_l\ (l \in L)$, here c_l is fixed for all $l \in L$. AP A_w connects AP A_v through a path $L(A_w, A_v)$, where the path $L(A_w, A_v)$ is the set of links that are used for the communication from AP A_w to AP A_v. For each link $l \in L$, let

$$S(l) = \{(A_w, A_v) : w, v \in W \text{ and } l \in L(A_w, A_v)\}$$

be the set of communication pairs consisting of APs that use link l. The wired-cum-wireless network is shared by the set S of end-to-end sessions. A session $u \in S$ can be usually expressed as (i,j), meaning that the origin of the session (i,j) is MH i in BSS w and the sink is MH j in BSS v, where $w, v \in W$. Let y_{ij} be the session rate for the session $(i,j) \in S$. As mentioned before, for simplicity of exposition, we assume that BSS w and BSS v denote different BSSs. However, those results can be easily extended to the scenarios where end-to-end sessions within a BSS are allowed. It is worth noting that each session in this network model runs across both wired links which have fixed link capacities and wireless links whose capacities depend on the transmission rates of MHs in that particular BSS (see Eq. (1)). Also, note that APs generate no traffic. They only forward the traffic between the wireless and the wired parts of the network.

We have now specified the following rate control problem in wired-cum-wireless network[8]:

$$\begin{aligned}
\text{maximize} \quad & \sum_{(i,j)\in S} \log(y_{ij}) \\
\text{subject to} \quad & y_{ij} \le c_{i,A(i)}(\rho), \quad \forall (i,j)\in S, \\
& y_{ij} \le c_{A(j),j}(\rho), \quad \forall (i,j)\in S, \\
& \sum_{(A(i),A(j))\in S(l)} y_{ij} \le c_l, \quad \forall l \in L, \\
& y, \rho \ge 0,
\end{aligned} \tag{2}$$

where the optimization variables are both session rates $y=(y_{ij}:(i,j)\in S)\in\Re_{+}^{|S|}$ and transmission rates $\rho=(\rho_{s,t}\,|\,(s,t)\in E_w, w\in W)\in\Re_{+}^{|M|}$. Note that link capacities $c_l, l\in L$ are assumed to be fixed parameters.

Since the sessions originate from one wireless network and end at another, they will travel two and only two wireless links, one is from the origin to the access point, and the other is from the access point in the destined wireless network to the sink. The first and the second sets of constraints say that the rate of the end-to-end sessions do not exceed the capacities of the two wireless links that are traveled, while $c_{i,j}(\rho)$ is given in Eq. (1). Next, the sessions will across through a set of links in the wired backbone, hence the third sets of constraints states that, the aggregate session rates at any wired link l can not exceed the capacity of that link.

In contrast to the wired backbone which has fixed capacities for each link, the attainable throughput on a link of a CSMA/CA based BSS has changeable value. Note that, under certain resource allocation scheme, the wireless link rates are computed and the session rates are adjusted so that the aggregate utility is maximized. However, since the wireless link capacities are not fixed, it is possible that, through resource reallocation, capacities of those bottleneck links in the wireless network can be increased and the aggregate utility can be further increased. Thus, the problem (2) involves not only the rate control for each end-to-end session, but also the best resource allocation schemes to support the session rates.

Note that the functions $c_{s,t}(\rho)$ are not convex or concave functions of the transmission rates ρ, and hence the problem (2) is not a convex programming. Now we will propose another optimization problem which is equivalent to the problem (2) and is a convex programming problem. In fact, this problem has been introduced in[8]. However, authors of[8] did not solve the equivalent convex programming problem. They only used it to prove the convergence of the DB algorithm. In our approach, we will solve the equivalent convex programming problem by the PDIP algorithm. Let z_{ij}, $r_{s,t}$ and d_l be the logarithmic values of the session rate y_{ij}, transmission rate $\rho_{s,t}$, and wired link capacity c_l, respectively, i.e., $z_{ij}=\log(y_{ij})$, $r_{s,t}=\log(\rho_{s,t})$, and $d_l=\log(c_l)$. Since both sides of the first three constraints in the problem (2) are positive, and since the logarithmic function is strictly increasing, taking logarithmic values on both sides of the first three constraints, the end-to-end proportionally fair rate control problem in the wired-cum-wireless network, as given (2), can be rewritten as

$$\text{minimize} \quad -\sum_{(i,j)\in S} z_{ij}$$

subject to

$$z_{ij} + \log\Big(1 + \sum_{k\in D_{A(i)}} e^{r_{A(i),k}} + \sum_{k\in J_{A(i)}} e^{r_{k,A(i)}}\Big) - r_{i,A(i)} \le 0,$$

$$z_{ij} + \log(1 + \sum_{k \in D_{A(j)}} e^{r_{A(j),k}} + \sum_{k \in J_{A(j)}} e^{r_{k,A(j)}}) - r_{A(j),j} \le 0, \quad \forall (i,j) \in S,$$

$$\log(\sum_{(A(i),A(j)) \in S(l)} e^{z_{ij}}) - d_l \le 0, \forall l \in L, \tag{3}$$

where $z = (z_{ij} : (i,j) \in S) \in \Re^{|S|}$ and $r = (r_{s,t} \mid (s,t) \in E_w,\ w \in W) \in \Re^{|M|}$ are optimization variables. According to Chiang[13], for $a \in \Re$ and $x_i \in \Re$, $i = 1,\ldots,n$, the function $\log(a + \sum_{i=1}^{n} e^{x_i})$ is a convex function for $(x_1,\ldots,x_n)$. It implies that the constraints in the problem (3) constitute to convex set. Thus, we obtain that the equivalent problem (3) is a convex programming problem. In Section 4, instead of solving the non-convex problem (2), we will solve the equivalent convex problem (3) by applying the PDIP algorithm.

3. THE DUAL-BASED DISTRIBUTED ALGORITHM

We now review the DB algorithm[8] to solve the problem (2) iteratively. The non-convex programming problem (2) may be solved through approach as follows. Instead of solving (2) directly, Wang, Kar, and Low[8] have considered the parameterized version of an end-to-end proportionally fair rate optimization problem when the attainable throughputs on wireless link $c_{s,t}(\rho)$ are parameterized as $x_{s,t}$ for any $(s,t) \in E_w$, and $w \in W$,

$$\begin{aligned} \text{maximize} \quad & \sum_{(i,j) \in S} \log(y_{ij}) \\ \text{subject to} \quad & y_{ij} \le x_{i,A(i)}, \quad \forall (i,j) \in S, \\ & y_{ij} \le x_{A(j),j}, \quad \forall (i,j) \in S, \\ & \sum_{(A(i),A(j)) \in S(l)} y_{ij} \le c_l, \quad \forall l \in L, \\ & y \ge 0. \end{aligned} \tag{4}$$

Clearly, the key difference between the problem (2) and the problem (4) is that, the wireless link capacities in (2), whose values are not fixed and depend on the transmission rates of wireless links, are parameterized in (4). Thus, the optimum value in (4) is a function on x, where x is the vector of capacities of all wireless links which the sessions travel through, that means $x = (x_{i,A(i)}, x_{A(j),j} : (i,j) \in S) \in \Re_+^{2|S|}$, and we denote $\hat{U}(x)$ as the optimum value in (4) when x is parameterized, i.e.,

$$\hat{U}(x) = \max \{ \sum_{(i,j) \in S} \log(y_{ij}) \mid y_{ij} \le x_{i,A(i)}, \ y_{ij} \le x_{A(j),j}, \sum_{(A(i),A(j)) \in S(l)} y_{ij} \le c_l, y \ge 0 \}.$$

From the vector of capacities of the wireless links in (2) in turn is a function on the link transmission rates, we can define function $\tilde{U}(\rho) = \hat{U}(c(\rho))$, where

$$c(\rho) = (c_{i,A(i)}(\rho), c_{A(j),j}(\rho) : (i,j) \in S) \in \Re_+^{2|S|}.$$

Hence, problem (2) can be rewritten as follows

$$\begin{aligned} & \text{maximize} \quad \tilde{U}(\rho) \\ & \text{subject to} \quad \rho_{s,t} \ge 0, \ \forall (s,t) \in E_w, w \in W. \end{aligned} \tag{5}$$

Now, we will present the DB distributed algorithm[8]. Firstly, when the vector of transmission rates $\rho^{(n)}$ is given, the convex programming problem (4) will be solved using a synchronous gradient projection algorithm in[2]. The session rates and the Lagrange multipliers of the dual problem of (4) when $x = c(\rho^{(n)})$ are computed by the following formulae

$$y_{ij}^{(n,k)} = \frac{1}{\lambda_{i,A(i)}^{(k)} + \lambda_{A(j),j}^{(k)} + \sum_{l \in L(A(i),A(j))} \gamma_l^{(k)}}, \quad \forall (i,j) \in S, \tag{6}$$

$$\lambda_{i,A(i)}^{(k+1)} = \left[\lambda_{i,A(i)}^{(k)} + \beta\left(y_{ij}^{(n,k)} - c_{i,A(i)}\left(\rho^{(n)}\right)\right)\right]^+, \quad \forall (i,j) \in S, \tag{7}$$

$$\lambda_{A(j),j}^{(k+1)} = \left[\lambda_{A(j),j}^{(k)} + \beta\left(y_{ij}^{(n,k)} - c_{A(j),j}\left(\rho^{(n)}\right)\right)\right]^{+}, \quad \forall (i,j) \in S, \tag{8}$$

$$\gamma_l^{(k+1)} = \left[\gamma_l^{(k)} + \beta\left(\sum_{(A(i),A(j))\in S(l)} y_{ij}^{(n,k)} - c_l\right)\right]^{+}, \quad l \in L, \tag{9}$$

where $\beta > 0$ is a step size, $[z]^{+} = \max\{z,0\}$.

After using the synchronous gradient projection algorithm for solving the problem (4) and its dual problem, we apply a gradient projection algorithm[14] to solve iteratively the problem (5) as follows:

$$\rho_{s,t}^{(n+1)} = \left[\rho_{s,t}^{(n)} + \delta \sum_{v\in W} \sum_{(k,m)\in E_v} \lambda_{k,m}^{*(n)} \frac{\partial c_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right)\right]^{+}, \tag{10}$$

where $\delta > 0$ is a step size, for any $(s,t) \in E_w$, $w \in W$ and $(k,m) \in E_v$, $v \in W$ then $\frac{\partial c_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right)$ is computed with the following formula

$$\frac{\partial c_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right) = -\frac{\rho_{k,m}^{(n)}}{\left(1 + \sum_{u\in D_{A_w}} \rho_{A_w,u}^{(n)} + \sum_{u\in J_{A_w}} \rho_{u,A_w}^{(n)}\right)^2}$$

if $v = w$ and $(k,m) \neq (s,t)$,

$$\frac{\partial_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right) = \frac{1 + \sum_{u\in D_{A_w}} \rho_{A_w,u}^{(n)} + \sum_{u\in J_{A_w}} \rho_{u,A_w}^{(n)} - \rho_{s,t}^{(n)}}{\left(1 + \sum_{u\in D_{A_w}} \rho_{A_w,u}^{(n)} + \sum_{u\in J_{A_w}} \rho_{u,A_w}^{(n)}\right)^2}$$

if $(k,m) = s,t)$,

$\frac{\partial c_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right) = 0$, otherwise,

and $\lambda_{s,t}^{*(n)}$ is the optimum solution of the dual problem of (4) when $x = c\left(\rho^{(n)}\right)$ which are obtained for solving the problem (4) and its dual problem iteratively using the formulae Eq. (6)-Eq. (9). We summarize.

ALGORITHM 1. **DUAL-BASED**[8]

Choose vectors

$$\rho^{(0)} = \left(\rho_{s,t}^{(0)} > 0 : (s,t) \in E_w, w \in W\right) \in \Re^{|M|},$$

$$\lambda^{(0)} = \left(\lambda_{i,A(i)}^{(0)} > 0, \lambda_{A(j),j}^{(0)} > 0 : (i,j) \in S\right) \in \Re^{2|S|},$$

$$\gamma^{(0)} = \left(\gamma_l^{(0)} > 0 : l \in L\right) \in \Re^{|L|},$$

and take $\varepsilon > 0$.

Outer loop

For $n = 0,1,\ldots$ until $\left\|\rho^{(n+1)} - \rho^{(n)}\right\| \le \varepsilon$ do

Compute $y_{ij}^{(n,0)} = \frac{1}{\lambda_{i,A(i)}^{(0)} + \lambda_{A(j),j}^{(0)} + \sum_{l\in L(A(i),A(j))} \gamma_l^{(0)}}.$

Inner loop

For $k = 0,1,\ldots$ until $\left\|y^{(k+1)} - y^{(k)}\right\| \le \varepsilon$ do

$$\lambda_{i,A(i)}^{(k+1)} = \left[\lambda_{i,A(i)}^{(k)} + \beta\left(y_{ij}^{(n,k)} - c_{i,A(i)}\left(\rho^{(n)}\right)\right)\right]^{+},$$

$$\lambda_{A(j),j}^{(k+1)} = \left[\lambda_{A(j),j}^{(k)} + \beta\left(y_{ij}^{(n,k)} - c_{A(j),j}\left(\rho^{(n)}\right)\right)\right]^{+},$$

$$\gamma_l^{(k+1)} = \left[\gamma_l^{(k)} + \beta\left(\sum_{(A(i),A(j))\in S(l)} y_{ij}^{(n,k)} - c_l\right)\right]^{+},$$

$$y_{ij}^{(n,k+1)} = \frac{1}{\lambda_{i,A(i)}^{(k+1)} + \lambda_{A(j),j}^{(k+1)} + \sum_{l\in L(A(i),A(j))} \gamma_l^{(k+1)}}$$

End for.

End inner loop

Set $\lambda_{i,A(i)}^{(0)} = \lambda_{i,A(i)}^{(k+1)}, \lambda_{A(j),j}^{(0)} = \lambda_{A(j),j}^{(k+1)}, \gamma_l^{(0)} = \gamma_l^{(k+1)}.$

Compute

$$\rho_{s,t}^{(n+1)} = \left[\rho_{s,t}^{(n)} + \delta \sum_{v\in W} \sum_{(k,m)\in E_v} \lambda_{k,m}^{(0)} \frac{\partial c_{k,m}}{\partial \rho_{s,t}}\left(\rho^{(n)}\right)\right]^{+}$$

End for.

End outer loop

Note that the authors of[8] have proved that the solutions, which are generated by the above algorithm, converge to the globally optimum point of the non-convex problem (2).

4. THE PRIMAL-DUAL INTERIOR-POINT ALGORITHM

As shown in the end of Section 2, the problems (2) and (3) are equivalent, furthermore the problem (2) is not a convex problem but the problem (3) is a convex problem. Thus, in this section, instead of solving the rate control problem (2), we will solve the equivalent problem (3) using the PDIP method[11].

Obviously, in the convex problem (3), the objective function $f(z,r)$, constraint functions $g_{ij}^{(1)}(z,r)$, $g_{ij}^{(2)}(z,r)$ $((i,j)\in S)$, and $h_l(z,r)$ $(l\in L)$ are convex and twice continuously differentiable, where

$$f(z,r):=-\sum_{(i,j)\in S} z_{ij},$$

$$g_{ij}^{(1)}(z,r):=z_{ij}+\log\left(1+\sum_{k\in D_{A(i)}} e^{r_{A(i),k}}+\sum_{k\in J_{A(i)}} e^{r_{k,A(i)}}\right)-r_{i,A(i)},$$

$$g_{ij}^{(2)}(z,r):=z_{ij}+\log\left(1+\sum_{k\in D_{A(j)}} e^{r_{A(j),k}}+\sum_{k\in J_{A(j)}} e^{r_{k,A(j)}}\right)-r_{A(j),j},$$

and

$$h_l(z,r):=\log\left(\sum_{(A(i),A(j))\in S(l)} e^{z_{ij}}\right)-d_l.$$

It leads to the necessary assumptions of the PDIP method[11] for (3) to hold. The gradient of $f(z,r)$ is denoted as $\nabla f(z,r)$, and $\nabla^2 f(z,r)$ denotes the Hessian matrix of second partial derivatives of $f(z,r)$. The gradient and Hessian of $g_{ij}^{(1)}(z,r)$, $g_{ij}^{(2)}(z,r)$ $((i,j)\in S)$, and $h_l(z,r)$ $(l\in L)$ are denoted as $\nabla g_{ij}^{(1)}(z,r)$, $\nabla g_{ij}^{(2)}(z,r)$ $((i,j)\in S)$, and $\nabla h_l(z,r)$ $(l\in L)$ and $\nabla^2 g_{ij}^{(1)}(z,r)$, $\nabla^2 g_{ij}^{(2)}(z,r)$ $((i,j)\in S)$, and $\nabla^2 h_l(z,r)$ $(l\in L)$. The $(2|S|+|L|)\times(|S|+|M|)$ Jacobian matrix $J(z,r)$ of first derivatives of $g_{ij}^{(1)}(z,r)$, $g_{ij}^{(2)}(z,r)$ $((i,j)\in S)$, and $h_l(z,r)$ $(l\in L)$ has rows

$$\{\nabla g_{ij}^{(1)}(z,r)^T, \nabla g_{ij}^{(2)}(z,r)^T((i,j)\in S), \nabla h_l(z,r)^T(l\in L)\}.$$

Finally, e denotes the vector of all ones whose dimension is determined by the context and put

$$\begin{aligned} c(z,r) &= (c_1(z,r),\ldots,c_{2|S|+|L|}(z,r)) \\ &:= (g_{ij}^{(1)}(z,r), g_{ij}^{(2)}(z,r), h_l(z,r): \\ &\qquad (i,j)\in S, l\in L)\in \Re^{2|S|+|L|}, \end{aligned}$$

$$H(z,r,\lambda):=\nabla^2 f(z,r)+\sum_{i=1}^{2|S|+|L|}\lambda_i\nabla^2 c_i(z,r).$$

We now present the PDIP algorithm for solving iteratively the convex problem (3).

ALGORITHM 2. **PRIMAL-DUAL INTERIOR-POINT**[11]

Choose vectors

$\lambda\in\Re^{2|S|+|L|}, x=(z^T,r^T)\in\Re^{|S|+|M|}$ satisfy

$\lambda>0$, $c(x)<0$.

Take $\varepsilon>0$ and $\mu>1$.

Repeat

Compute $\bar{\eta}=-c(x)^T\lambda$, $t=\mu(2|S|+|L|)/\bar{\eta}$.

Solve a linear system to obtain a primal-dual search direction $(\Delta x,\Delta\lambda)$

$$\begin{pmatrix} H(x,\lambda) & J(x)^T \\ -\mathrm{diag}(\lambda)J(x) & -\mathrm{diag}(c(x)) \end{pmatrix}\begin{pmatrix}\Delta x\\ \Delta\lambda\end{pmatrix}=-\begin{pmatrix} r_{\mathrm{dual}}(x,\lambda)\\ r_{\mathrm{cent}}(x,\lambda)\end{pmatrix} \quad (11)$$

where $r_{\mathrm{dual}}(x,\lambda):=\nabla f(x)+J(x)^T\lambda$ and

$r_{\mathrm{cent}}(x,\lambda):=-\mathrm{diag}(\lambda)c(x)-(1/t)e$.

Line search and update

Determine

$$s^{\max} = \min\left\{1, \min\left\{-\frac{\lambda_i}{\Delta\lambda_i} : \Delta\lambda_i < 0\right\}\right\}.$$

Compute $s = 0.99 s^{\max}$.

Repeat

$$x^+ = x + s\Delta x, \quad s = \beta s$$

Until $c(x^+) < 0$.

Repeat

$$x^+ = x + s\Delta x, \; \lambda^+ = \lambda + s\Delta\lambda, \quad s = \beta s$$

Until $\|r_t(x^+, \lambda^+)\| \le (1 - \alpha s)\|r_t(x, \lambda)\|$, where

$$r_t(x, \lambda) := \begin{pmatrix} r_{\text{dual}}(x, \lambda) \\ r_{\text{cent}}(x, \lambda) \end{pmatrix}.$$

Compute error= $s\|\Delta x\| \; (= \|x^+ - x\|)$.

Set $x = x + s\Delta x, \;\; \lambda = \lambda + s\Delta\lambda$

Until $\bar{\eta} \le \varepsilon, \|r_{\text{dual}}(x, \lambda)\| \le \varepsilon$, and error$\le \varepsilon$.

Note that, values of the parameter μ in the PDIP algorithm on the order of 10 appear to work well. The line search in the PDIP method is a standard backtracking line search, based on the norm of the residual, and modified to ensure that $\lambda > 0$ and $c(x) < 0$. The backtracking parameters α and β, are typically chosen in the range 0.01 to 0.1, and in the range 0.3 to 0.8, respectively (ref.[11]).

Our first goal in the PDIP algorithm is to solve the linear system Eq. (11) on each loop. We denote the coefficient matrix of Eq. (11) by $H_{\text{pd}}(x, \lambda)$, i.e.,

$$H_{\text{pd}}(x, \lambda) := \begin{pmatrix} H(x, \lambda) & J(x)^T \\ -\operatorname{diag}(\lambda) J(x) & -\operatorname{diag}(c(x)) \end{pmatrix}.$$

It is known that under certain assumptions, the matrices $H_{\text{pd}}(x, \lambda)$ are always nonsingular[10]. Furthermore, if the problem (3) is sparse, which means that the objective and every constraint function each depend on a modest number of variables, then the gradient and Hessian matrices of the objective and constraint functions are all sparse. It follows that the matrix $H_{\text{pd}}(x, \lambda)$ is then likely to be sparse, so a sparse matrix method can be used to solve Eq. (11). As in[9], there are some ways to solve the linear system Eq. (11). Firstly, the linear system Eq. (11) can be solved directly by using LU factorization method. The second approach is to use block elimination to obtain smaller "condensed" system. Since $c(x) < 0$, the (2, 2) block of Eq. (11) may be eliminated to give the following $(|S|+|M|) \times (|S|+|M|)$ system for Δx:

$$H_c(x, \lambda)\Delta x = -\nabla f(x) + \frac{1}{t}\left(\sum_{k=1}^{2|S|+|L|} \frac{1}{c_k(x)} \nabla c_k(x)\right), \qquad (12)$$

where a condensed matrix $H_c(x, \lambda)$ is given as

$$H_c(x, \lambda) := H(x, \lambda) + J(x)^T D(x, \lambda)^{-1} J(x),$$

with $D(x, \lambda) := \operatorname{diag}(\lambda)^{-1} \operatorname{diag}(-c(x))$.

The second component of variable $\Delta\lambda$ in the linear system Eq. (11) is computed as

$$\Delta\lambda = \operatorname{diag}(c(x))^{-1}\left(r_{\text{cent}}(x, \lambda) - \operatorname{diag}(\lambda) J(x) \Delta x\right).$$

As observed, the condensed matrix $H_c(x, \lambda)$ is positive definite. Thus, the condensed system Eq. (12) can be solved by direct methods, such as Cholesky factorization. It is shown[9] that both the matrices $H_{\text{pd}}(x, \lambda)$ and $H_c(x, \lambda)$ become increasingly ill-conditioned in a highly structured way as the iteration converges. Although this ill-conditioning is usually harmless[15], we still obtain warning of ill-conditioning from Matlab when the computation of our simulation results in Section 5 is carried out in Matlab.

In the end of this section, to avoid the ill-conditioning when solving Eq. (12) we are concerned with alternative, iterative methods for solving Eq. (12). Gould[16] has proposed a stabilized conjugate-gradient (SCG) method for solving iteratively Eq. (12). We will use this algorithm to solve the condensed systems as presented by Eq. (12).

The simulation example is presented in Section 5. For now, we present the SCG method for Eq. (12). For simplicity of exposition, we denote $H(x,\lambda)$, $J(x)$, $D(x,\lambda)$, and

$$-\nabla f(x)+\frac{1}{t}\left(\sum_{k=1}^{2|S|+|L|}\frac{1}{c_k(x)}\nabla c_k(x)\right)$$

as H, J, D, and b respectively when we present the SCG method for solving Eq. (12). For the SCG method presented below, we denote I as an identity matrix whose dimension is determined by the context.

ALGORITHM 3. **STABILIZED CONJUGATE GRADIENT**[16]

Take $\varepsilon_r>0,\varepsilon_a>0,\Delta x=0,w=0,z=0,$ and let $v=-b.$

Solve a linear system

$$\begin{pmatrix} I & J^T \\ J & -D\end{pmatrix}\begin{pmatrix} r \\ u\end{pmatrix}=\begin{pmatrix} v \\ w\end{pmatrix}.$$

Update

$$\begin{pmatrix} v \\ w \\ z\end{pmatrix}\leftarrow\begin{pmatrix} v-J^Tu \\ w+Du \\ z+u\end{pmatrix}.$$

Solve the linear system

$$\begin{pmatrix} I & J^T \\ J & -D\end{pmatrix}\begin{pmatrix} r \\ u\end{pmatrix}=\begin{pmatrix} v \\ w\end{pmatrix}.$$

Set $s=z+u,p=-r,q=-s,$ and $\sigma=r^Tv+s^Tw,\sigma_0=\sigma.$

Repeat

Compute $\delta=\sigma/(p^THp+q^TDq).$

Update

$$\begin{pmatrix} \Delta x \\ z \\ v \\ w\end{pmatrix}\leftarrow\begin{pmatrix} \Delta x \\ z \\ v \\ w\end{pmatrix}+\delta\begin{pmatrix} p \\ q \\ Hp \\ Dq\end{pmatrix}.$$

Solve the linear system

$$\begin{pmatrix} I & J^T \\ J & -D\end{pmatrix}\begin{pmatrix} r \\ u\end{pmatrix}=\begin{pmatrix} v \\ w\end{pmatrix}.$$

Update

$$\begin{pmatrix} v \\ w \\ z\end{pmatrix}\leftarrow\begin{pmatrix} v-J^Tu \\ w+Du \\ z+u\end{pmatrix}.$$

Solve the system

$$\begin{pmatrix} I & J^T \\ J & -D\end{pmatrix}\begin{pmatrix} r \\ u\end{pmatrix}=\begin{pmatrix} v \\ w\end{pmatrix}.$$

Update $s=z+u.$

Compute $\sigma_{\text{new}}=r^Tv+s^Tw,$

$\bar{\sigma}=\sigma_{\text{new}}/\sigma,\quad \sigma=\sigma_{\text{new}}.$

Update

$$\begin{pmatrix} p \\ q\end{pmatrix}\leftarrow-\begin{pmatrix} r \\ s\end{pmatrix}+\bar{\sigma}\begin{pmatrix} p \\ q\end{pmatrix}$$

Until $\sigma\le\max\{\varepsilon_r\sigma_0,\varepsilon_a\}.$

We remark that the SCG algorithm also requires a solving a linear system

$$\begin{pmatrix} I & J^T \\ J & -D\end{pmatrix}\begin{pmatrix} r \\ u\end{pmatrix}=\begin{pmatrix} v \\ w\end{pmatrix}\qquad(13)$$

in each iteration. Observing that the linear system Eq. (13) is near with one Eq. (11), however the linear system is well-conditioned[16], they can be solved direct by LU factorization, i.e., the SCG method is an efficient algorithm for ill-conditioned linear system from the PDIP method.

It is well known that the computation of DB algorithm is simple. It only requires computing first derivatives of the constraint functions in the optimization problem (2), while the PDIP method requires computing both the first and second derivatives of the objective and constraint functions.

Using the gradient projection method, DB algorithm solves the problem (2) iteratively, where the convergence rate is only Q-linear[17], hence, we can conclude that the convergence rate of the DB algorithm is at the most Q-linear.

The purpose of this paper is to propose another efficient algorithm in solving the non-convex programming problem (2). It is shown[10], [9] that the PDIP method converges at a Q-quadratic rate or Q-superlinear rate. Moreover, the PDIP method is an omnipresent and conspicuous feature of the constrained optimization landscape today. Therefore, solving the rate control problem (2) in the wired-cum-wireless network by the PDIP algorithm is very attractive. It is not too surprising that the total CPU-time of the PDIP algorithm is significantly smaller than one of the DB algorithm when those algorithms are used to solve a simulation example in Section 5.

5. NUMERICAL EXAMPLE

We now present a numerical example, which is taken in[8], to illustrate the performance of the PDIP algorithm in providing proportional fairness amongst the end-to-end flows in the wired-cum-wireless network. A median-sized network is considered, which is composed of 4 APs, 8 MHs, 4 wired links and 8 wireless links. The network is configured as follows

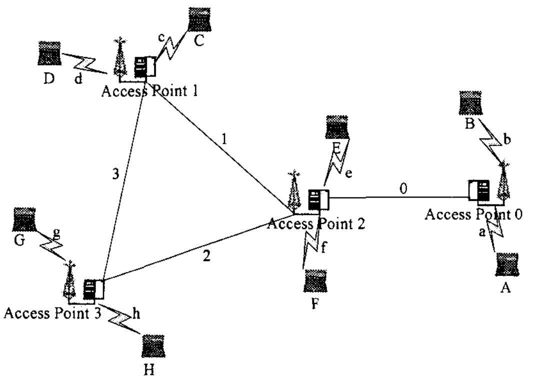


Fig. 2 The network configuration.

In this network, there are 4 APs, which are denoted as 0, 1, 2 and 3. The wired part of the network connects the APs through the wired links, denoted as 0, 1, 2 and 3. The capacities of the wired links are 0.5, 0.2, 0.6 and 0.8 respectively. In each BSS, there are 2 MHs, who connect to the network through the AP. We denote the 8 MHs as A, B, C, D, E, F, G, H, and the wireless links a, b, c, d, e, f, g and h respectively. The network has 4 end-to-end sessions, labeled as f_0, f_1, f_2 and f_3, and they are set up as in Table 1.

For the DB algorithm we used a small step sizes $\beta = \delta = 0.15$, which are usually chosen for solving the rate control problem of the network using gradient projection method[2,3], and the stop threshold ε is 10^{-8}. Here, in the PDIP algorithm, the linear systems Eq. (11) are solved using the Algorithm 3 (SCG algorithm) with $\varepsilon_r = 10^{-12}$ and $\varepsilon_a = 10^{-16}$. We start the PDIP algorithm at initial vectors (logarithm of session and transmission rate) $z = -2e \in \Re^{|S|}$, $r = -e \in \Re^{|M|}$, and take $\lambda_i = -1/c_i(x)$ ($i = 1,\ldots,2|S| + |L|$). Other parameter values that we used for the PDIP method are $\varepsilon = 10^{-8}$, $\mu = 10$, $\beta = 0.5$, and $\alpha = 0.01$.

Our experiment was done in Matlab 7.0 on a Pentium 4 CPU 3.00GHz, 1.00 GB of RAM running Windows XP. Fig. 3 and Fig. 4 show the convergence behavior of the DB and PDIP algorithm for the session rates in wired links, respectively.

Table 1. The source, sink, and path of the flows

Session	Source node	Sink node	Links on the path
f_0	E	A	e, 0, a
f_1	B	G	b, 0, 2, g
f_2	C	F	c, 3, 2, f
f_3	H	D	h, 2, 1, d

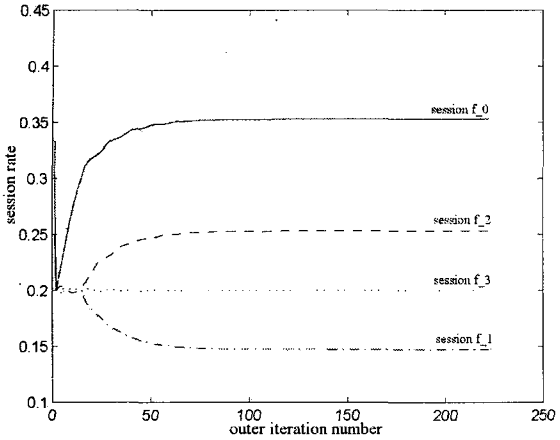


Fig. 3. The convergence of the session rates in DB algorithm.

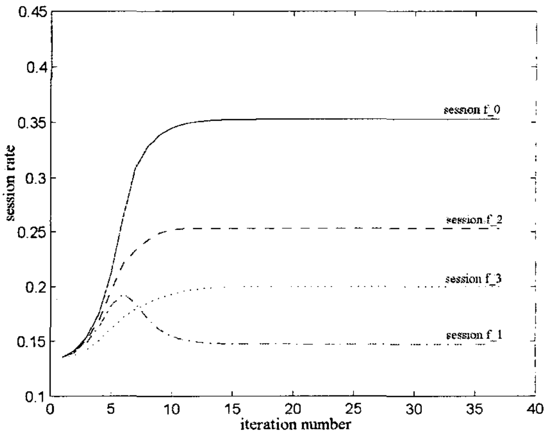


Fig. 4. The convergence of the session rates in PDIP algorithm.

Noting that, this simulation example has a unique optimum solution for session rates in wired links, but one has multiple optimum solution for transmission rates in wireless links. We will not present a unique solvability or not unique solvability of the rate control problem (2) for transmission rates in wireless links in this paper. In fact, we are now investigating the unique solvability of the rate control problem in wired-cum-wireless network. In[8], the authors have not cared for the transmission rates in wireless links in this simulation example. To illustrate a performance of the proposed PDIP algorithm, we compare a globally optimum session rate solutions solved by AMPL[8] with a solutions given by the DB and PDIP algorithm. We also plot a rate of convergence of the DB and PDIP algorithm, that means we plot the expression $\log_{10}(\|y^{(k)} - y^*\|)$, in Fig. 5, here the exact optimum solution is taken by AMPL, i.e.,

$$y^* = (0.352753, 0.147247, 0.252753, 0.200000),$$

and $y^{(k)}$ is an k iterative session rate solution vector of the DB and PDIP method.

Now we give in Table 2 an overview of the exact optimum solution, an approximate optimum solution by DB algorithm, approximate optimum solution by PDIP algorithm for the session rates, and a required CPU-time of DB and PDIP algorithm.

Noting that since we have not known that how the authors of[8] have taken the parameters β and δ for the DB algorithm, then our numerical result does not completely coincide with a numerical result in[8] for the DB algorithm. The Fig. 5 indicates that PDIP significantly enhances convergence speed. Therefore, we can conclude that PDIP algorithm may be considered as a competitive algo-

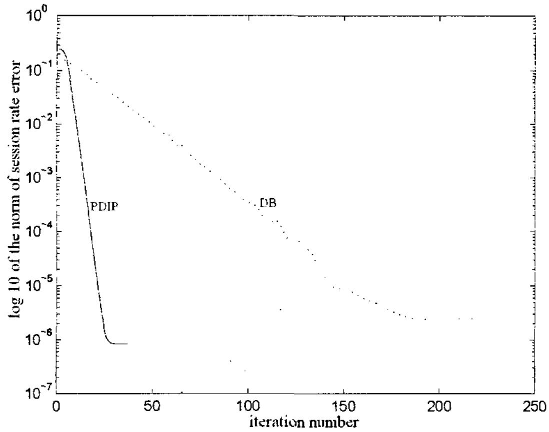


Fig. 5. The rate of convergence in DB and PDIP algorithm.

Table 2. The rate of convergence in DB and PDIP algorithm

Variables	Exact solution by AMPL	Solution by DB	Solution by PDIP
y_0	0.352753	0.35275176	0.35275252
y_1	0.147247	0.14724803	0.14724748
y_2	0.252753	0.25275454	0.25275252
y_3	0.200000	0.19999891	0.20000000
iteration number		221	36
CPU-time		0.703 seconds	0.312 seconds

rithm to solve the rate control problems in wired-cum-wireless network.

6. CONCLUSIONS

The PDIP algorithm is used in this paper for solving the rate control problem in the wired-cum-wireless network. It is well known that the formulated problem appears to be non-convex, i.e., it is a difficult optimization problem, and Wang, Kar and Low have proposed the DB algorithm to solve this problem. In this paper we showed that the PDIP algorithm is a competitive algorithm with the DB algorithm to solve the rate control solution problem in the wired-cum-wireless network. Simulation results are provided to support our conclusions. In our future work, we plan to investigate a unique solvability of the rate control problem for transmission rates in wireless links in the wired-cum-wireless network, which depends on the topology of the wireless network.

7. REFERENCES

[1] F. Kelly, A. Maulloo, and D. Tan, "Rate Control for Communication Networks: Shadow Prices, Proportional Fairness and Stability," *Journal of Operations Research Society*, Vol. 49, No. 3, pp. 237-252, 1998.

[2] S. Low and D. E. Lapsley, "Optimization Flow Control, I: Basic Algorithm and Convergence," *IEEE/ACM Transactions on Networking*, Vol. 7, No. 6, pp. 861-884, 1999.

[3] S. Athuraliya and S. Low, "Optimization Flow Control with Newton-Like Algorithm," *Telecommunication Systems*, Vol. 15, No. 3-4, pp. 345-358, 2000.

[4] K. Kar, S. Sarkar, and L. Tassiulas, "A Simple Rate Control Algorithm for Maximizing Total User Utility," *in Proceedings of Infocom 2001*, Vol. 1, pp. 133-141, 2001.

[5] L. Tassiulas and S. Sarkar, "Maxmin Fair Scheduling in Wireless Networks," *in Proceedings of Infocom 2002*, vol. 2, pp. 763-772, 2002.

[6] T. Nadagopal, T.-E. Kim, X. Gao, and V. Bharghavan, "Achieving MAC Layer Fairness in Wireless Packet Networks," *in Proceeding ACM/IEEE MobiCom '00*, pp. 87-98, 2000.

[7] T. Ozugur, M. Naghshineh, P. Kermani, C. M. Olsen, and B. Rezvani, "Balanced Media Access Methods for Wireless Networks," in *Proceeding ACM/IEEE MobiCom '98*, pp. 21-32, 1998.

[8] X. Wang, K. Kar, and S. H. Low, "Cross-Layer Rate Optimization in Wired-cum-Wireless Networks," *Technical Report*, 2005, http://www.ecse. rpi.edu/%7Ekoushik/TR-wired-cum-wireless.pdf

[9] A. Forsgren, P. E. Gill, and M. H. Wrighi, "Interior Methods for Nonlinear Optimization," *SIAM Review*, Vol. 44, No. 4, pp. 525-597, 2002.

[10] A. S. E. Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang, "On the Formulation and Theory of the Newton Interior-Point Method for Nonlinear Programming," *Journal of Optimization Theory and Applications*, Vol. 89, No. 3, pp. 507-541, 1996.

[11] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[12] X. Wang and K. Kar, "Throughput Modelling and Fairness Issues in CSMA/CA Based Ad-Hoc Networks," *in Proceedings of Infocom 2005*, Vol. 1, pp. 23-34, 2005.

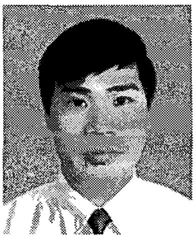
[13] M. Chiang, "To Layer or Not To Layer: Balancing Transport and Physical Layers in Wireless Multihop Networks," *in Proceedings of Infocom 2004*, Vol. 4, pp. 2525-2536, 2004.

[14] D. P. Bertsekas, *Nonlinear Programming*, Second Edition, Athena Scientific, Belmont, Massachusetts, 1999.

[15] M. H. Wright, "Ill-Conditioning and Computational Error in Interior Methods for Nonlinear Programming," *SIAM Journal on Optimization*, Vol. 9, No. 1, pp. 84-111, 1998.

[16] N. I. M. Gould, "Iterative Methods for Ill-Conditioned Linear Systems from Optimization," *in Nonlinear Optimization and Related Topics*, G. Di Pillo and F. Giannessi, editors, Kluwer Publishing, 1999, pp. 123-142.

[17] D. P. Bertseks and J. N. Tsitsiklis, *Partial Solutions Manual Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, Belmont, Massachusetts, 1997.



Le Cong Loi

He received the B.S. degree in mathematics from Hanoi University of Education, Hanoi, Vietnam, in 1994. He received the M.S. and Ph.D. degrees in computational mathematics from Vietnam National University, Hanoi, Vietnam, in 2000 and 2004, respectively. Since 2000, he has been lecturer at Vietnam National University, Hanoi, Vietnam. Now he is a postdoc at Department of Information and Communications Engineering, Inje University,, Republic of Korea. His research interests include difference equations, numerical methods for differential equations and partial differential equations, optimal control and optimization in communication networks.

Won-Joo Hwang

He received the Ph.D Degree from Osaka University Japan in 2002. He received his bachelor's degree and M.S. degree in Computer Engineering from Pusan National University, Pusan, Republic of Korea, in 1998 and 2000. Since September 2002, he has been a assistance professor at Inje University, Gyeongnam, Republic of Korea. His research interests are in sensor networks and ubiquitous sensor networks.