# PERFORMANCE OF DISTRIBUTED UTILITY-BASED POWER CONTROL FOR WIRELESS AD HOC NETWORKS

Jianwei Huang, Randall Berry, Michael L. Honig
Department of Electrical and Computer Engineering
Northwestern University
2145 Sheridan Road, Evanston, IL, 60208
{jianweih,rberry,mh}@ece.northwestern.edu

## ABSTRACT

*We study the performance of a distributed and asynchronous power control scheme for a spread spectrum wireless ad hoc network. The users exchange prices that reflect their loss in utility due to interference. The prices are then used to determine optimal (utility maximizing) power levels for each user. We present simulation results illustrating the convergence of the algorithm, and the effect of limited information. With logarithmic utilities, the pricing algorithm exhibits rapid convergence to the unique optimal power allocation. We then study the effect of limiting the amount of information users can exchange. Results are presented, which show performance (average utility per user) assuming each transmitter can decode interference prices only from receivers within a specified radius. The performance is shown to degrade gracefully as the radius decreases. We also compare the performance of the pricing algorithm with a Request to Send/Clear to Send (RTS/CTS) protocol. Numerical results show that in a dense network the pricing algorithm can offer large improvements in total efficiency (i.e., when utility corresponds to information rate). The effect of coarse rate control on performance is also examined.*

## I. INTRODUCTION

We consider power control in a spread spectrum (SS) peer-to-peer network where all users spread their power over a single frequency band. The transmission rate for each user depends on the received signal-to-interference plus noise ratio (SINR). Due to the mutual interference in the spread spectrum environment, users generate *negative externalities* [1], i.e., each user's transmission has a direct negative impact on other users' achievable rates. Our objective is to coordinate user power levels to optimize overall performance, measured in terms of total network utility.

We study protocols in which the users exchange price signals that indicate the "cost" of received interference. Namely, we consider the *asynchronous distributed pricing (ADP)* algorithm, in which each user announces a price, which is the marginal decrease in utility with respect to a marginal increase in received interference. The prices are then used to determine an optimal power level for each user. This pricing scheme can internalize the negative externalities among users, and in certain cases can induce a centralized optimal solution in a distributed way. This can be interpreted as a type of *Pigovian Tax* [1], which, in economics, is a tax imposed by an agency (e.g., the government) to penalize user behaviors that generate negative externalities. Pigovian taxation and variations have been presented for congestion pricing in communication networks (e.g., [2]–[5]). The power control scheme presented here discovers the optimal prices (taxes) distributively and asynchronously, instead of in a static and centralized way (as in [2]–[4]). In previous work [6], we have shown that the ADP algorithm converges globally to the socially optimal (utility maximizing) solution for a general class of utility functions.

In this paper, we study the performance of the ADP algorithm through simulations. We first show that with logarithmic utilities, the ADP algorithm converges rapidly to the globally optimal power allocation (i.e., much faster than the gradient algorithm for power control proposed in [7]). We then study the effect of limiting the amount of information nodes can exchange. Specifically, we assume that each transmitter can decode prices only from receivers within a specified radius. There is, then, no explicit coordination between transmitters and receivers separated by more than this radius. (A radius of zero corresponds to uncoordinated power control, i.e., all transmitters transmit with the maximum power.) Numerical results are presented, which show performance (total utility) as a function of this radius, user density, and bandwidth. The performance of the ADP algorithm is observed to degrade gracefully with decreasing radius.

We also compare the performance of the ADP algorithm

This work was supported by the Northwestern-Motorola Center for Communications, ARO under grant DAAD190310119, and NSF CAREER award CCR-0238382.

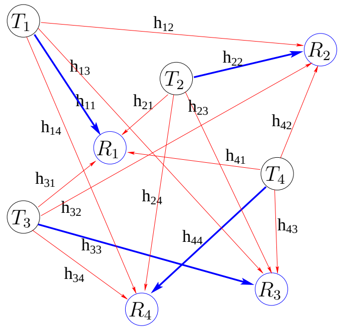

Fig. 1. An example wireless network with four users (pairs of nodes) ($T_m$ and $R_m$ denote the transmitter and receiver of "user" $m$, respectively).

with the Request to Send/Clear to Send (RTS/CTS) random access protocol in the 802.11 standard. Numerical results show that in a dense network the ADP algorithm can offer large improvements in total efficiency (i.e., when utility corresponds to information rate). The effect of rate control on performance is also examined. Namely, our results show that if the rate can be adjusted to match the received SINR, then RTS/CTS random access offers only a modest improvement relative to uncoordinated power control. This improvement increases significantly when the allowable rates are quantized.

In the next section, we describe the system model and the ADP algorithm. In Sect. III, we briefly summarize our previous results on the convergence of the ADP algorithm [6]. Numerical results are given in Sect. IV, and conclusions are given in Sect. V.

## II. Asynchronous Distributed Pricing (ADP) Algorithm

We consider a snap-shot of an ad hoc network with a set $\mathcal{M} = \{1, ..., M\}$ of distinct node pairs. As shown in Fig. 1, each pair consists of one dedicated transmitter and one dedicated receiver. We do not consider multihop channels, although these peer-to-peer connections could represent a particular schedule of transmissions determined by an underlying routing and MAC protocol. We use the terms "pair" and "user" interchangeably in the following. We assume that each user $m$ transmits an SS signal spread over the total bandwidth of $B$ Hz. Over the time-period of interest, the channel gains for each pair are fixed. The channel gain between user $m$'s transmitter and user $j$'s receiver is denoted by $h_{mj}$. Note that in general $h_{mj} \neq h_{jm}$, since the latter represents the gain between user $j$'s transmitter and user $m$'s receiver.

Each user $m$'s quality of service is characterized by a utility function $u_m(\gamma_m)$, which is an increasing and strictly concave function of the received SINR,

$$\gamma_m(\boldsymbol{p}) = \frac{p_m h_{mm}}{n_0 + \frac{1}{B}\sum_{j\neq m} p_j h_{jm}},$$

where $n_0$ is the background noise power and $\boldsymbol{p} = (p_m)_{m=1}^{M}$ is a vector of the users' transmission powers. We can also write $\boldsymbol{p} = (p_m; p_{-m})$, where $p_{-m} = (p_j)_{j\neq m}^{M}$ contains all users' transmission powers except user $m$'s. The users' utility functions are coupled due to mutual interference. As an example, the *logarithmic utility function* $u_m(\gamma_m) = \theta_m \log(\gamma_m)$, where $\theta_m$ is a user dependent priority parameter.[1]

The problem we consider is to specify $\boldsymbol{p}$ to maximize the utility summed over all users, where each user $m$ must also satisfy a transmission power constraint, $p_m \in \mathcal{P}_m = \left[P_m^{\min}, P_m^{\max}\right]$, i.e.,

$$\max_{\{\boldsymbol{p}: p_m \in \mathcal{P}_m, \forall m\}} \sum_{m=1}^{M} u_m(\gamma_m(\boldsymbol{p})). \tag{P1}$$

A special case is $P_m^{\min} = 0$, i.e., the user may choose not to transmit. For certain utilities, e.g., $\theta_m \log(\gamma_m)$, all assigned powers must be strictly positive, since as $p_m \to 0$, the utility approaches $-\infty$.[2]

As a baseline distributed approach, consider the case where the users do not exchange any information and simply choose transmission powers to maximize their individual utilities. Since each user's payoff $u_m(\gamma_m(p_m, p_{-m}))$ is strictly increasing with $p_m$ for fixed $p_{-m}$, and there is no penalty for high transmission power as long as $p_m \in \mathcal{P}_m$, each user would choose to transmit at the maximum power, i.e., $p_m = P_m^{\max}$. This solution leads to the maximum interference among users and can be far from the solution to Problem P1.

To improve the total network utility, users need to adjust their transmission powers in an *interference-aware* fashion. We achieve this through the following pricing scheme. Each user $j \in \mathcal{M}$ announces an *interference price* $\pi_j$ to all other users,

$$\pi_j(p_j, p_{-j}) = -\frac{\partial u_j(\gamma_j(p_j, p_{-j}))}{\partial I_j(p_{-j})},$$

where $I_j(p_{-j}) = \sum_{k\neq j} p_k h_{kj}$ is the total interference received by user $j$ (before bandwidth scaling). Here, $\pi_j(p_j, p_{-j})$ is always nonnegative and is user $j$'s marginal

[1] In the high SINR regime, the logarithmic utility approximates the Shannon capacity $\log(1+\gamma_i)$ weighted by $\theta_i$. At low SINRs, a user's rate is approximately linear in SINR, and so this utility is proportional to the logarithm of the rate. Logarithmic utility also captures fairness constraints by ensuring that no user has a very small SINR.

[2] In Sect. III, we require $P_m^{\min} > 0$ for global convergence of the ADP algorithm. In that case, $P_m^{\min}$ can be chosen arbitrarily small so that this restriction has little effect.

decrease given a marginal increase in total interference. Assuming fixed $p_{-m}$, and given the prices announced by other users $\pi_{-m} = (\pi_j)_{j \neq m}^{M}$, each user $m \in \mathcal{M}$ chooses transmit power $p_m$ to maximize the surplus

$$s_m\left(p_m; p_{-m}, \pi_{-m}\right) = u_m\left(\gamma_m\left(p_m, p_{-m}\right)\right) - p_m \sum_{j \neq m} \pi_j h_{mj}.$$

User $m$ therefore maximizes its utility minus its payment to other users for the interference it generates.

In the ADP algorithm, each user announces a single price and all users set their transmission powers based on the received prices. Prices and powers are asynchronously updated. For $m \in \mathcal{M}$, let $\mathcal{T}_{m,p}$ and $\mathcal{T}_{m,\pi}$ be two unbounded sets of positive time instances at which user $m$ updates its power and price, respectively. The algorithm is specified as the following (where $t^-$ denotes the time immediately before $t$):

**ADP Algorithm**:

(1) INITIALIZATION: Each user $m \in \mathcal{M}$ chooses some power $p_m(0) \in \mathcal{P}_m$ and price $\pi_m(0) \geq 0$.

(2) POWER UPDATE: At each $t \in \mathcal{T}_{m,p}$, user $m$ updates its power according to

$$p_m(t) = W_m\left(p_{-m}(t^-), \pi_{-m}(t^-)\right),$$

where

$$\begin{aligned} & W_m(p_{-m}, \pi_{-m}) \\ & = \arg\max_{\hat{p}_m \in \mathcal{P}_m} s_m\left(\hat{p}_m; p_{-m}, \pi_{-m}\right) \\ & = \left[\frac{p_m}{\gamma_m\left(\boldsymbol{p}\right)} g_m\left(\frac{p_m}{\gamma_m(\boldsymbol{p})}\left(\sum_{j \neq m} \pi_j h_{mj}\right)\right)\right]_{P_m^{\min}}^{P_m^{\max}}, \end{aligned}$$

where $[x]_a^b = \max\left\{a, \min\left\{x, b\right\}\right\}$, $p_m/\gamma_m\left(\boldsymbol{p}\right)$ is independent of $p_m$, and

$$g_m\left(x\right) = \begin{cases} \infty, & 0 \leq x \leq u_m'\left(\infty\right), \\ \left(u_m'\right)^{-1}\left(x\right), & u_m'\left(\infty\right) < x < u_m'\left(0\right), \\ 0, & u_m'\left(0\right) \leq x. \end{cases}$$

(3) PRICE UPDATE: At each $t \in \mathcal{T}_{m,\pi}$, user $m$ updates its price according to

$$\pi_m(t) = C_m\left(\boldsymbol{p}(t^-)\right),$$

where

$$C_m(\boldsymbol{p}) = -\frac{\partial u_m\left(\gamma_m\left(\boldsymbol{p}\right)\right)}{\partial I_m\left(p_{-m}\right)} = \frac{\partial u_m(\gamma_m(\boldsymbol{p}))}{\partial \gamma_m(\boldsymbol{p})} \frac{\left(\gamma_m(\boldsymbol{p})\right)^2}{B p_m h_{mm}}.$$

Note that in addition to being asynchronous across users, each user also need not update its power and price at the same time. To implement the power and price updates, each user $m$ only needs to know its own utility $u_m$, the current SINR $\gamma_m$, channel gain $h_{mm}$, *adjacent* channel gains $h_{mj}$ $(j \neq m)$, and prices $\pi_j$ $(j \neq m)$. The SINR $\gamma_m$ and channel gain $h_{mm}$ can be measured at the receiver and fed back to the transmitter. Measuring the adjacent channel gains $h_{mj}$ can be accomplished by having each receiver periodically broadcast a beacon; assuming reciprocity, the transmitters can then measure these channel gains. The price information could also be periodically broadcast through this beacon. Since each user announces only a single price, the number of prices scales linearly with the size of the network.

## III. PROPERTIES AND CONVERGENCE OF THE ADP ALGORITHM

Here we briefly summarize our previous results on the convergence of the ADP algorithm [6]. Denote the fixed points set of the ADP algorithm by

$$\mathcal{F}^{ADP} \equiv \left\{(\boldsymbol{p}, \boldsymbol{\pi}) \,|\, (\boldsymbol{p}, \boldsymbol{\pi}) = \left(\boldsymbol{W}\left(\boldsymbol{p}, \boldsymbol{\pi}\right), \boldsymbol{C}(\boldsymbol{p})\right)\right\},$$

where $\boldsymbol{W}(\boldsymbol{p}, \boldsymbol{\pi}) = (W_m(p_{-m}, \pi_{-m}))_{m=1}^M$ and $\boldsymbol{C}(\boldsymbol{p}) = (C_m(\boldsymbol{p}))_{m=1}^M$. In [6] it is shown that $\mathcal{F}^{ADP}$ corresponds to the solutions of the KKT conditions of Problem P1. Although $u_m\left(\gamma_m\right)$ is strictly concave in $\gamma_m$, the objective in Problem P1 may not be concave in $\boldsymbol{p}$. Thus in general, $\mathcal{F}^{ADP}$ may contain multiple points including local optima or saddle points. However, if there is only one solution to the KKT conditions, then it must be the global maximum and the ADP algorithm would reach that point if it converges.

Let $\gamma_m^{\min} = \min\{\gamma_m(\boldsymbol{p}) : p_m \in \mathcal{P}_m\}$ and $\gamma_m^{\max} = \max\{\gamma_m(\boldsymbol{p}) : p_m \in \mathcal{P}_m\}$ for all $m \in \mathcal{M}$. Also define $G_m\left(\gamma_m\right) = -\gamma_m u_m''(\gamma_m)/u_m'(\gamma_m)$. We have the following result.

*Proposition 1:* If for all $m \in \mathcal{M}$:

$(i)$ $P_m^{\min} > 0$, and

$(ii)$ $G_m\left(\gamma_m\right) \in [a, b]$ for all $\gamma_m \in [\gamma_m^{\min}, \gamma_m^{\max}]$, where $[a, b]$ is a strict subset of $[1, 2]$;

then Problem P1 has a unique optimal solution, to which the ADP algorithm globally converges.

The term $G_m\left(\gamma_m\right)$ is called the *coefficient of relative risk aversion* in economics [1] and measures the relative concaveness of $u_m\left(\gamma_m\right)$. Some utility functions which satisfy condition $(ii)$ in Proposition 1 include $\theta_m \log(\gamma_m)$, $\theta_m \gamma_m^{\alpha}/\alpha$ (with $\alpha \in [1-b, 1-a]$), and $1 - \exp\left(-\theta_m \gamma_m\right)$ (with $a/\gamma_m^{\min} \leq \theta_m \leq b/\gamma_m^{\max}$). The utility function $\theta_m \log\left(1 + \gamma_m\right)$ does *not* satisfy condition $(ii)$. In that case, the ADP algorithm can converge to different fixed points depending on the initialization.

## IV. NUMERICAL RESULTS

We now provide some simulation results to illustrate the performance of the ADP algorithm. We simulate a network contained in a 10m×10m square area. Transmitters are randomly placed in this area according to a uniform distribution, and the corresponding receiver is randomly placed within a 6m×6m square centered around the transmitter.

### A. Comparison with Gradient Updates

First we compare the convergence of the ADP algorithm with the gradient method proposed in [7], where prices are updated in the same way as in the ADP algorithm, but powers are updated according to

$$p_m(t) = \Big[p_m(t^-) + \kappa\left(W_m\left(p_{-m}(t^-), \pi_{-m}(t^-)\right) - p_m(t^-)\right)\Big]_{P_m^{\min}}^{P_m^{\max}}.$$

where the constant step-size $\kappa$ has to be small enough to guarantee convergence. All users have the same logarithmic utility function $u_m = \log(\gamma_m)$. The channel gains $h_{mj} = d_{mj}^{-4}$, $P_m^{\max}/n_0 = 40$ dB, and spreading factor $B = 128$. Figure 2 shows the convergence of the powers and prices for each user under both algorithms for a network with $M = 10$ users. Users start from random power and price initializations and update their power and prices synchronously (i.e., time sets $\mathcal{T}_{m,p} = \mathcal{T}_{m,\pi} = \mathcal{T}$ for all $m$). The step-size $\kappa = 0.01$, which is the largest step-size for which the gradient algorithm consistently converges. Both algorithms converge to the socially optimal power allocation, but the ADP algorithm converges much faster. The ADP algorithm essentially uses an "adaptive step-size", i.e., users adapt the power in "larger" step-sizes when they are far away from the optimal solution, and use smaller steps when close to the optimal.

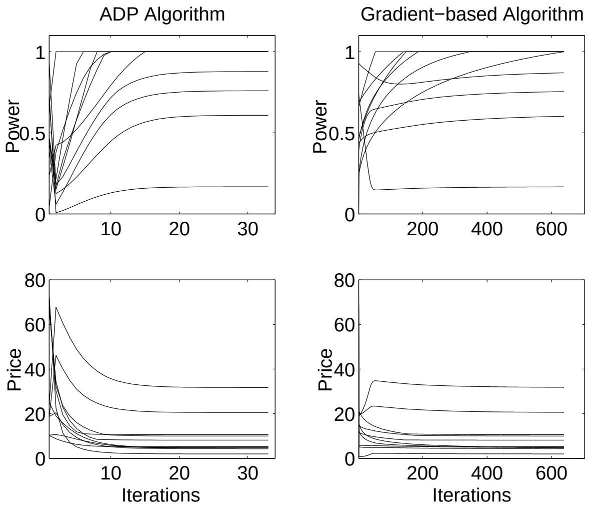

Fig. 2. Convergence of the prices and powers for the ADP algorithm (left) and a gradient algorithm (right) in a network with 10 users and logarithmic utility functions. Each curve corresponds to the power or price for one user with a random initialization.

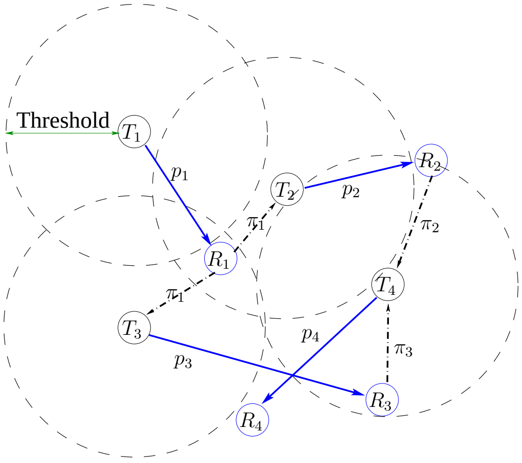

Fig. 3. An example of a wireless network with limited information exchange.

### B. Effects of Limited Information Exchange

In practice, users may be able to decode price messages only from neighboring users, and may not account for prices from users farther away. Figure 3 illustrates this situation in a network with four users. Each user $m$ can decode pricing information only from other users whose receivers are within a *threshold* distance of the transmitter $m$ (i.e., the radius of the corresponding circle). The dash-dotted arrows represent the prices that can be decoded by the corresponding users. For example, user $4$ can decode prices $\pi_2$ and $\pi_3$, whereas user $2$ can only decode price $\pi_1$.

Figure 4 shows average utility per user for the ADP algorithm versus user density with various threshold values. Each user has the same logarithmic utility function $u_m = \log(\gamma_m)$. The channel gains $h_{mj} = d_{mj}^{-4}$, $P_m^{\max}/n_0 = 40$ dB, and spreading gain $B = 5$. Each data point is averaged over $100$ random topology realizations. Due to the small spreading gain and high user density, most users obtain a low SINR, which leads to negative utility. The full information ADP algorithm, which accounts for all prices in the network, achieves the socially optimal solution. Since the total network area is $10$ meters by $10$ meters, the same performance can be achieved by letting the *threshold* equal to $10$ meters. The performance of the limited information ADP algorithm degrades gracefully with a decreasing threshold, e.g., the performance is still very close to optimal even with a threshold of $1$ meter. When the threshold decreases to zero, each user transmits at maximum power since no pricing information is taken into account. This leads to a much lower utility compared with the full information ADP algorithm.

In addition to the logarithmic utility function which captures fairness constraints, we are interested to see how

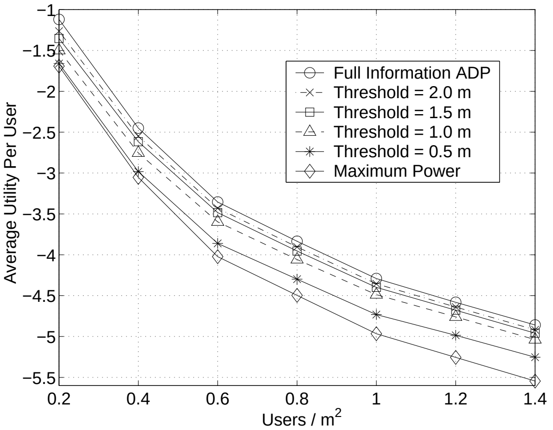

Fig. 4. Performance of the ADP algorithm with limited pricing information vs. user density (Users have logarithmic utility).

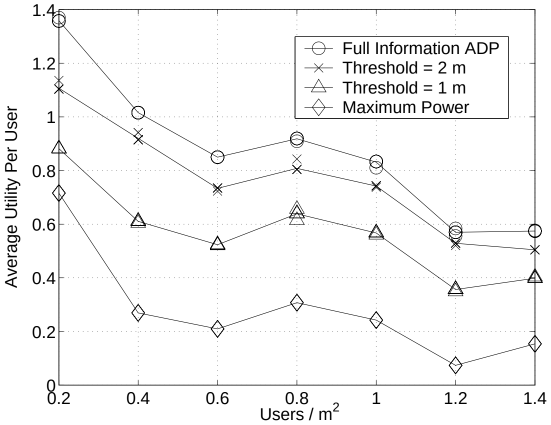

Fig. 5. Performance of the ADP algorithm with limited pricing information vs user density (Users have rate utility functions and random initial powers and prices with fixed topologies).

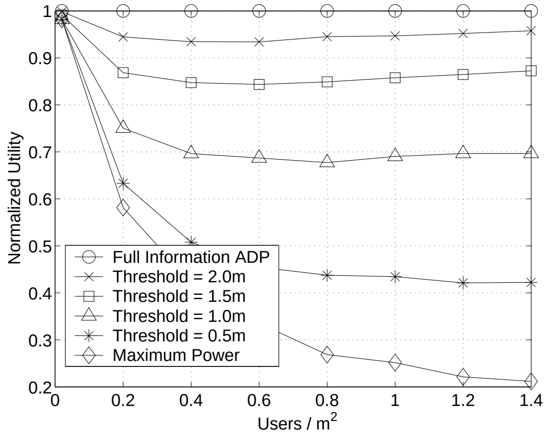

Fig. 6. Performance of the ADP algorithm with limited pricing information vs. user density averaged over network topologies.

the ADP algorithm perform in terms of network throughput maximization. For this purpose, we let each user have the same *rate* utility function $u_m = \log(1+\gamma_m)$, i.e., we assume that the users can perfectly adapt their modulation/coding schemes to reach the Shannon capacity. In this case, the ADP algorithm is not guaranteed to converge to the globally optimal solution, i.e., the algorithm may converge to different fixed points depending on the initialization, or may not converge at all. In the latter case, we stop the algorithm after $100$ synchronous power and price updates. Figure 5 shows the performance of the ADP algorithm versus user density. The parameters are the same as Figure 4. For each user density, we randomly generate one network topology, and run the algorithm with $10$ different random power and price initializations (for the same topology). Each point corresponds to the average utility per user of a particular realization. The figure shows that although in some cases different initializations lead to different fixed points, the corresponding utilities are typically very close. (The fluctuation in utility with user density is due to the change in network topology.)

Figure 6 shows average performance of the ADP algorithm versus user densities with rate utility functions. Here we plot normalized utility, i.e., each point represents the average utility per user normalized by the achievable utility using the full information ADP algorithm, averaged over $100$ random topology realizations. The parameters are the same as Figure 4. The ADP algorithm with only a $2$ meters threshold achieves a normalized utility as high as $95\%$, and the performance degradation with decreasing thresholds is quite graceful. The normalized utility decreases with increasing user density when the threshold is less than or equal to $0.5$ meter, due to the increasing number of interfering users farther away than the threshold. On the other hand, the normalized utility stays the same, or even increases slightly with increasing user density when the threshold is larger than $1$ meter. This is due to the fact that the threshold is large enough to capture most of the strong interfering users, so that the out-of-zone interfering users become less important.

Figure 7 shows the normalized utility of the ADP algorithm versus bandwidth (spreading gain) with rate utility functions. The user density is fixed at $1.4$ users/m$^2$. All other parameters are the same as in Figure 6. It is not surprising that increasing the bandwidth decreases the mutual interference, and increases the achievable network utility.

Figure 8 shows the average utility per user of the ADP algorithm versus path loss exponent $r$ with rate utility functions. The channel gains satisfy $h_{mj} = d_{mj}^{-r}$, and the user density is fixed at $1$ user/m$^2$. All other parameters are the same as in Figure 6. With uncoordinated maximum

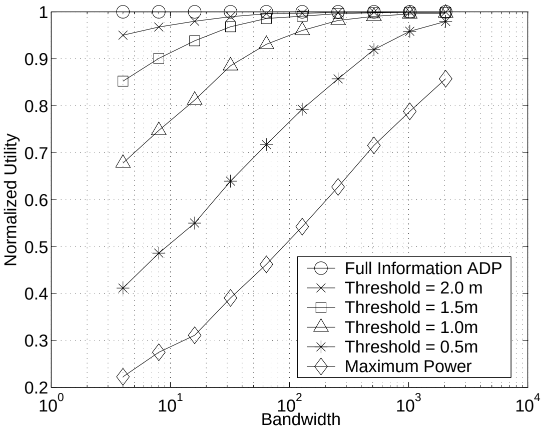

Fig. 7. Performance of the ADP algorithm with limited pricing information vs. bandwidth (spreading gain).

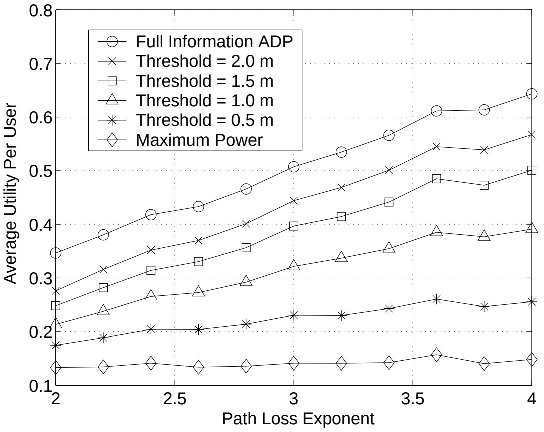

Fig. 8. Performance of the ADP algorithm with limited information vs. path loss exponent.

power transmission, the utility stays roughly unchanged for different values of $r$. This is because at each user's receiver, both the useful signal and the interference decrease at the same rate with increasing $r$, so that the SINR stays constant[3]. However, for the full information ADP algorithm, power control is performed to take advantage of the increasing $r$ (thereby decreasing interference), which leads to a higher utility. The performance gain decreases as the threshold becomes smaller.

### *C. Comparison with* $802.11$ *RTS/CTS MAC Protocol*

Figure 9 compares the performance of the ADP algorithm with the $802.11$ Request to Send/Clear to Send (RTS/CTS) MAC protocol with rate utility functions. To simulate the $802.11$ protocol, we determine the random locations of transmitter-receiver pairs sequentially, from user $1$ to user $M$. Both user $m$'s transmitter and receiver are *active* if its transmitter (respectively, receiver) is more than $3$ meters away from an active receiver (respectively, transmitter) for users $1$ to $m-1$. Otherwise, its transmitter is silent. Only active transmitters (receivers) can transmit (receive) data. To make a fair comparison, we plot the normalized utility of both the full information ADP algorithm and the limited information ADP algorithm (threshold = $3$ meters). We also plot the normalized utility where all users transmit at maximum powers. The system parameters are the same as in Figure 6. Results are shown with both perfect rate adaptation (denoted by $\times$) assuming an optimal coding scheme that achieves the $\log(1+\gamma_m)$ utility, and quantized rate adaptation (denoted by $\bigcirc$) in which the rate is chosen from the set $\{0, 5, 10, 15, 20\}$ bits/Hz. All results are normalized with respect to the achievable utility of the full information ADP with perfect rate adaptation.

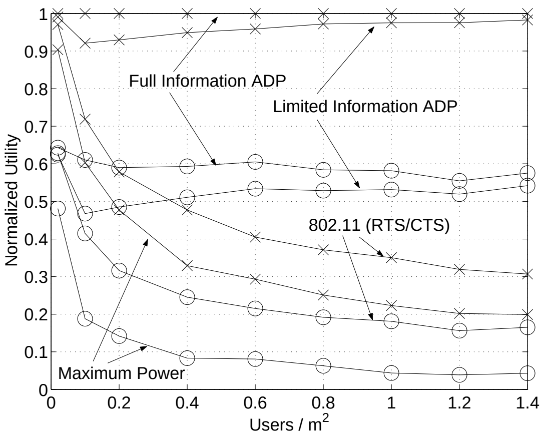

Fig. 9. Performance comparison of the ADP algorithm with the $802.11$ RTS/CTS protocol, and uncoordinated maximum transmission powers ($\times$: perfect rate adaption; $\bigcirc$: quantized rate adaption).

As the density of users increases, the ADP algorithm achieves much higher utility than the RTS/CTS protocol (as much as a factor of three with a user density of $1.4$ users/m$^2$). The performance gap is approximately the same with quantized rates. The $802.11$ protocol achieves a utility up to $1.5$ times of that achieved by maximum power transmission with perfect rate adaptation. The ratio increases to a factor of four when the rates are quantized. This is due to the fact that maximum power transmission leads to strong interference, and very small SINRs for many users, who would get zero utility with quantized rates.

## V. Conclusions

We have evaluated the performance of a distributed power control algorithm for spread spectrum ad hoc wireless net-

[3] With maximum power transmission, the background noise is small compared with the interference generated.

works. According to the algorithm, users announce prices to reflect their sensitivities to the current interference levels and then adjust their powers to maximize the individual surpluses. The algorithm can be implemented asynchronously, and requires only limited knowledge of channel gains to neighboring users. Numerical results show that the algorithm converges rapidly to the socially optimal solution, and has a graceful performance degradation when the information exchange among users is limited. We have also observed a significant performance improvement relative to the $802.11$ RTS/CTS protocol, both with and without perfect rate control. Throughout the paper we have focused on a static setting, where the communicating pairs and the channel conditions are fixed. An interesting future direction is to consider dynamic environments.

## REFERENCES

[1] A. Mas-Colell, M. D. Whinston, and J. R. Green, *Microeconomic Theory*. Oxford University Press, 1995.

[2] J. K. MacKie-Mason and H. R. Varian, “Pricing congestible network resources,” *IEEE Journal on Selected Areas of Communications*, vol. 13, no. 7, pp. 1141–1148, Sept. 1995.

[3] ——, *Public Access to the Internet*. The MIT Press, 1995, ch. Pricing the Internet, pp. 269–314.

[4] D. Acemoglu and A. E. Ozdaglar, “Competition and efficiency in congested markets,” working paper, 2005.

[5] A. Gupta, D. O. Stahl, and A. B. Whinston, “A stochastic equilibrium model of internet pricing,” *Journal of Economic Dynamics and Control*, vol. 21, pp. 697–722, 1997.

[6] J. Huang, R. Berry, and M. L. Honig, “A game theoretic analysis of distributed power control for spread spectrum ad hoc networks,” to apprear in the proceedings of *IEEE ISIT 2005*, journal version submitted to *IEEE Journal on Selected Areas in Communications*.

[7] M. Chiang, “Balancing transport and physical layers in wireless multihop networks: Jointly optimal congestion control and power control,” *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 1, pp. 104– 116, Jan 2005.