Distributed Association Control and Relaying in MillimeterWave Wireless Access Networks

Yuzhe Xu, *Member, IEEE,* George Athanasiou, *Member, IEEE,*
Carlo Fischione, *Member, IEEE,* and Leandros Tassiulas, *Fellow, IEEE,*

***Abstract*—In millimeterWave wireless networks the rapidly varying wireless channels demand fast and dynamic resource allocation mechanisms. This challenge is hereby addressed by a distributed approach that optimally solves the fundamental resource allocation problem of joint client association and relaying. The problem is posed as a multi-assignment optimization. Distributed auction based algorithms are proposed, where the clients and relays act asynchronously. It is shown that the algorithms always converge to a solution that maximizes the total network throughput within a desired bound. Both theoretical and numerical results evince numerous useful properties in comparison to standard approaches and the potential applications to forthcoming millimeterWave wireless access networks.**

A. Problem Formulation

Consider a mmW wireless network with M clients (mobile devices) and K APs or small base stations, where each client can be associated to one of the available APs. Each client may skip association with APs and establish a connection via one of the available $N \leq M$ relays, where a relay is a client that in addition to its transmissions help another client with an AP. An example of this network is shown in Fig. 1.

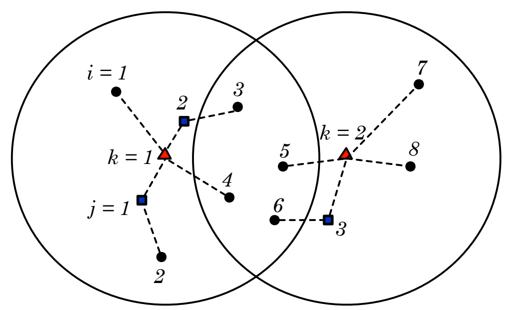

Fig. 1. Example network with 8 clients, 3 relays, and 2 APs.

When client i is associated to AP k, we denote client i's throughput benefit as $a_{(i,k)}$, whereas when client i is associated to AP k by an intermediate relay client j, we denote the client i's throughput benefit as $a_{(i,j,k)}$. It follows that the link rate is bounded by the lowest rate when using a relay. To describe the client-AP or client-relay-AP association, we introduce the binary decision variables $x_{(i,k)} = 1$ if $(i,k) \in \mathcal{S}$ and $x_{(i,k)} = 0$ otherwise, for all $(i,k) \in \mathcal{A}$. Moreover, $x_{(i,j,k)} = 1$ if $(i,j,k) \in \mathcal{S}$ and $x_{(i,j,k)} = 0$ otherwise, for all $(i,j,k) \in \mathcal{A}$. Naturally, the total throughput benefit of the clients in the network is given by

$$u = \sum_{(i,k)\in\mathcal{A}} a_{(i,k)} x_{(i,k)} + \sum_{(i,j,k)\in\mathcal{A}} a_{(i,j,k)} x_{(i,j,k)}, \quad (1)$$

Our goal is to find the optimal assignment $\mathcal{S}^*$ that maximizes u. This resource allocation problem can be formulated into a multi-dimensional assignment problem as

$$\max_{x_{(i,k)},\, x_{(i,j,k)}} \quad u \quad (2a)$$

$$\text{s.t.} \sum_{(i,k)\in\mathcal{A}} x_{(i,k)} + \sum_{(i,j,k)\in\mathcal{A}} x_{(i,j,k)} = 1, \ \forall i\,, \quad (2b)$$

$$\sum_{(i,j,k)\in\mathcal{A}} x_{(i,j,k)} \leq 1, \ \forall j\,, \quad (2c)$$

$$x_{(i,j,k)}, x_{(i,k)} = \{0,1\}, \forall i,j,k\,, \quad (2d)$$

where the known parameters are $a_{(i,k)}$ and $a_{(i,j,k)}$. Constraint (2b) assures that client i is associated to one AP or connected to one relay. Constraint (2c) ensures that relay j can assist one client at most.

Y. Xu, G. Athanasiou and C. Fischione are with the Automatic Control Lab, Electrical Engineering School and Access Linnaeus Center, KTH Royal Institute of Technology, Stockholm, Sweden. E-mail: {yuzhe, georgioa, carlofi}@kth.se

L. Tassiulas is with Computer and Communication Engineering Department, University of Thessaly, Volos, Greece. E-mail: leandros@uth.gr

This work was supported by the Swedish Research Council and the EU project Hydrobionets.

Constraint (2d) indicates that the decision variables are binary. Note also that it is important to keep the load balance for APs (in terms of the number of connections) over the time. This introduces a time dimension and the consecutive solutions of problem (2) must also satisfy the following constraint: for all k

$$\mathbf{E}\left[\sum_{(i,k)\in\mathcal{A}} x_{(i,k)} + \sum_{(i,j,k)\in\mathcal{A}} x_{(i,j,k)}\right] = \frac{M}{K}, \qquad (3)$$

where the expectation is taken with respect to the random distribution of clients and relays over the time. Problem (2) is a special case of multi-assignment problems that in general have no closed form solutions and are NP-complete. Moreover, the problem is combinatorial, and may have multiple optimal solutions. The computational cost for directly solving problem (2) by searching is very high, since the number of possible combinations is $\mathcal{O}(MN!\,K^M)$ when $M > N$.

B. Distributed Auction Algorithms

The distributed solution method is based on the application of Algorithm 1. In the solution algorithms, the vector $P_i \in \mathbb{R}^N$ denotes the prices vector for the relays (stored in client i), p_j denotes the price of a relay node j (stored in relay j), and k_i^* represents AP k_i^* for client i. In what follows we present the basic steps to establish the distributed algorithms.

Proposition 0.1: Consider M clients, N relay nodes, and K APs. The distributed auction algorithms given in Algorithm 1 terminate within a finite number of iterations bounded by $MN^2\lceil\Delta/\epsilon\rceil$, where $\Delta = \max_{(i,q)\in\mathcal{A}^*}\beta_{(i,q)} - \min_{(i,q)\in\mathcal{A}^*}\beta_{(i,q)}$.

Proposition 0.2: Let ϵ be a desired positive constant. The final assignment obtained by Algorithm 1 is within $M\epsilon$ of the optimal assignment benefit of problem (2).

C. Numerical Examples

We have implemented the distributed auction algorithms. Fig. 2 shows the objective value of problem (2), obtained by Algorithms 1 after termination (AUCTION), in comparison to random association policy (RAND), RSSI-based policy (RSSI), and the optimal policy (OPTM).

Algorithm 1 Distributed Auction Algorithms

for Client i
Initialize $q_i = k_i^*, P_i = \mathbf{0}$
while true do
  if receive **no** and new price p_{q_i} from q_i **then**
    procedure RC(**no**, p_{q_i}) ▷ Released Connection
      Disconnect to relay q_i
      Connect to object k_i^*
      $[P_i]_{q_i} \to p_{q_i}$, and $q_i \to k_i^*$
    end procedure
  end if

  procedure DAC(q_i, P_i) ▷ Distributed Auction for Client
    if $q_i = k_i^* \neq \arg\max_{q\in\mathcal{Q}(i)}\left\{\beta_{(i,q)} - [P_i]_q\right\}$ **then**
      $q_i' \to \arg\max_{q\in\mathcal{Q}(i)}\left\{\beta_{(i,q)} - [P_i]_q\right\}$,
      $u_i \to \max_{q\in\mathcal{Q}(i)}\left\{\beta_{(i,q)} - [P_i]_q\right\}$,
      $\omega_i \to \max_{q\in\mathcal{Q}(i),q\neq q_i}\left\{\beta_{(i,q)} - [P_i]_q\right\}$,
      $b_{iq_i} \to p_{q_i} + u_i - \omega_i + \epsilon$
      Send request with b_{iq_i} to relay q_i', and wait response

      Receive respond, (**yes** or **no**) and p_{q_i}' ▷ After receiving response
      if respond contains **yes then**
        Connect to object q_i', and $q_i \to q_i'$
      end if
      $p_{q_i} \to p_{q_i}'$
    end if
  end procedure
end while

for Relay j
Initialize the client $i_j = \varnothing$, and price $p_j = 0$
if receive request from clients i and b_i **then**
  procedure R2CR(i, b_i) ▷ Respond to Connection Request
    if $b_i - p_j \geq \epsilon$ **then**
      Send **yes** and p_j, to client i_q
      Send **no** and p_j, to client i_j
      Connect to client i, and $i_j \to i$, $p_j \to b_i$
    else
      Send **no** and p_j, to client i
    end if
  end procedure
end if

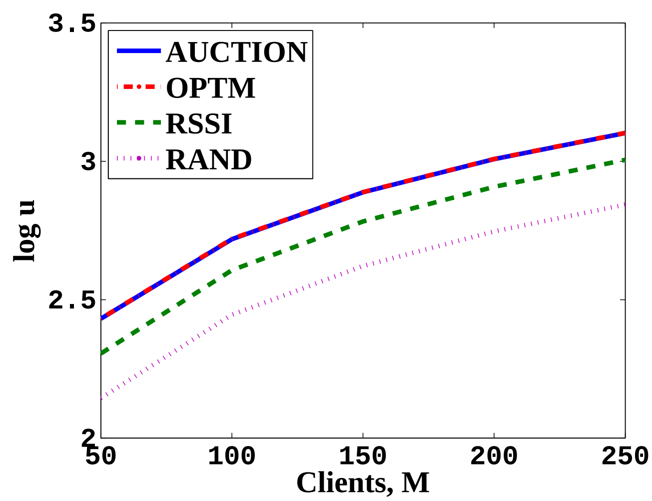

Fig. 2. Objective values of AUCTION, OPTM, RAND, and RSSI. $\log u$ vs. number of clients with 10 APs and 25 relays.