TECHNICAL UNIVERSITY OF CRETE
ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT
TELECOMMUNICATIONS DIVISION

Inference-Based Distributed User Association and Resource Allocation in Wireless Networks

by

Roza Chatzigeorgiou

A DISSERTATION SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE MASTER OF SCIENCE OF

ELECTRICAL AND COMPUTER ENGINEERING

October 2022

THESIS COMMITTEE

Professor Aggelos Bletsas, *Thesis Supervisor*
Professor Thrasyvoulos Spyropoulos
Associate Professor Vasileios Samoladas

Abstract

This work distributively solves the problems of user association and resource allocation in wireless networks. Distributed methods dispense the computational burden across the terminals/base stations of a network and are usually preferred over centralized methods, in which computing is done at a central location. First, the belief propagation (BP) algorithm is utilized for joint time and frequency allocation in wireless sensor networks; BP is inherently distributed, due to its message passing nature. In this work, the convergence of BP for the resource allocation problem is re-visited from a control-theory approach; performance evaluation is also provided for two methods, namely restarting and perturbed BP. Next, the problem of energy efficient user association in the downlink of heterogeneous networks is solved in a distributed, inference-based manner. The problem is first expressed as a linear program (LP). Max product BP is a distributed, message passing algorithm which can be utilized to obtain the solution of the LP, under certain conditions. Considering the loopy nature of the established graphical model, it is important that convergence and correctness of the max product algorithm is guaranteed. Therefore, the problem is relaxed, fixing the number of users per base station and the number of active base stations; then it is proved that max product BP converges with arbitrary initialization to the solution of the LP, in a finite number of iterations. Complexity is also reduced and expressed as a function of the number of active base stations and serving users. Finally, numerical results compare the performance of the proposed distributed algorithm with state-of-the-art. The results show that the proposed approach offers higher geometric mean of energy efficiency compared to previous work, without sacrificing spectral efficiency.
Thesis Supervisor: Professor Aggelos Bletsas

Acknowledgements

First of all I would like to thank my family for their unconditional love and endless support.

I could not have undertaken this journey without my supervisor Prof. Aggelos Bletsas, who generously provided me knowledge and expertise.

I would also like to thank my friends for the wonderful and unforgettable memories we shared all these years.

Special thanks to Iosif Vardakis for all the love, support and hiking adventures!

To my family.

Table of Contents

List of Figures

Chapter 1

Introduction

Belief propagation (BP) is an inference algorithm which utilizes a distributed protocol, due to its message-passing nature. Distributed protocols allow only the communication between neighboring nodes; this dispenses the computational burden across all terminals (or base stations) of the network. On the other hand, in centralized protocols, a central node collects information from each link of the network, and then solves the problem. This increases considerably the computational cost and introduces latency. In this work, two challenging and fundamental problems in Wireless Networks are solved distributively, using the BP algorithm. Considering the loopy nature of the established graphs, the convergence and correctness of BP for solving these problems is also thoroughly studied.

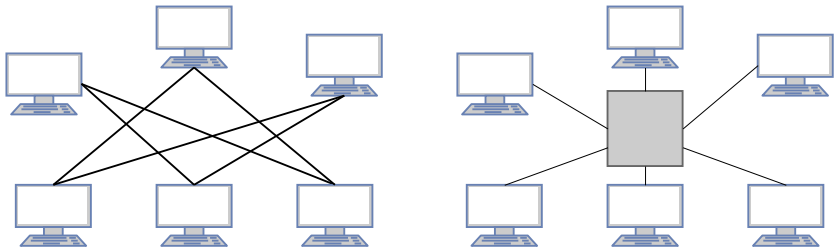

Figure 1.1: Schemes for distributed (a) and centralized (b) computing.

1.1 Thesis Contribution

The contribution of this thesis for the problem of time and frequency allocation in wireless sensor networks (WSN) is summarized in the following bullets:

- The convergence of BP for this problem is re-visited from a control-theory perspective.
- Perturbed BP is utilized as a distributed solver.
- Computer simulations compare the performance of two distributed solvers, namely Perturbed BP and restarting method.

Finally, for the problem of energy efficient user association in the downlink of Wireless Heterogeneous Networks, the contribution of this thesis is outlined by:

- Energy efficient user association is relaxed and expressed as a Linear Program, which is then solved distributively with max-product BP over base stations.
- BP convergence and correctness guarantees are offered.
- Computation complexity is analyzed and reduced, with careful precomputations.
- Numerical results compare the performance of the proposed algorithm to state-of-the-art.

1.2 Thesis Outline

The thesis is organized as follows: in Chapter 2, the convergence of BP for the problem of inference-based joint time and frequency allocation in wireless sensor networks is re-visited from a control theory perspective. Additionally, the performance of two inference-based algorithms, namely perturbed belief propagation and restarting method, is evaluated. Chapter 3 describes the proposed algorithm for energy efficient user association in the downlink of heterogeneous networks. Chapter 4 offers numerical results for the algorithm of Chapter 3 and compares its performance with state-of-the-art. Thesis is concluded at Chapter 5, in which ideas for future work are also provided.

Chapter 2

Time and Frequency Allocation in Wireless Sensor Networks

Resource allocation in wireless networks, i.e., assigning time and frequency slots over specific terminals under spatio-temporal constraints, is a fundamental and challenging problem, due to the limited availability of resources [1]. Existing work in Wireless Sensor Network (WSN) resource allocation includes centralized [1; 2; 3], as well as distributed protocols [4; 5; 6; 7; 8]. The problem of resource allocation in WSN can be formulated as a Constraint Satisfaction Problem (CSP). Belief Propagation/message passing (inference) algorithms have been proposed for CSP, since they are inherently amenable to distributed implementation. In this chapter, the problem of resource allocation is first revisited from a control theory perspective. First, BP is described as a dynamical system and then, the feasibility of analyzing its stability is discussed. However, it was found that such approach results in a prohibitively high computational cost.

The performance of two heuristic methods for the convergence of BP to a valid (i.e., constraint-satisfying) fixed point is also compared. The first method periodically checks whether the constraints are satisfied locally and restarts specific messages, when the local constraints (encoded in corresponding factors) are not satisfied. The second method stochastically perturbs Belief Propagation, using Gibbs sampling. The methods are evaluated, based on how often they fail to converge to a valid allocation, coined as *outage probability*. Numerical results demonstrate that, as the maximum number of iterations increase, both methods decrease the outage probability. However, the restarting method offers faster convergence to a valid CSP solution.

2.1 Problem Formulation

Assume M time slots and K orthogonal frequency channels are available for a WSN, which consists of N half-duplex radio terminals. Each terminal transmits one packet at a given time slot $m \in \mathcal{M} \triangleq \{1, 2, \ldots, M\}$, on a specific frequency channel $k \in$

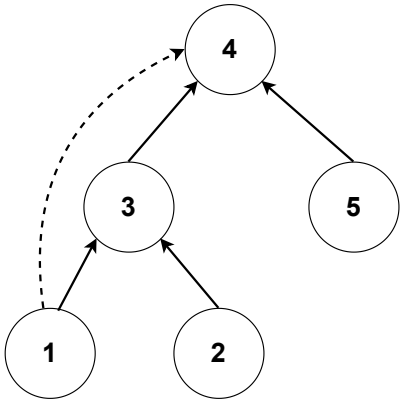

Figure 2.1: Scheme showing the routing (solid lines) and interference (dotted lines) links of a network of $N = 5$ sensors.

$\mathcal{K} \triangleq \{1, 2, \ldots, K\}$. Let us also define the set of all terminals by $\mathcal{N} \triangleq \{1, \ldots, N\}$ and $\mathcal{N}_{\setminus s}$ the set of all terminals, excluding the sink. The latter is the destination of all packets and operates only in receiver mode. A sensor i' will be an actual interferer of the link (i, j) between two sensors, if the following condition holds:

$$\gamma_{i \to j} = \frac{P_i |h_{i,j}|^2}{\sigma_j^2 + P_{i'} |h_{i',j}|^2} < \theta, \tag{2.1}$$

where P_i is the power of transmitter i, $h_{i,j}$ is the instantaneous channel gain coefficient between transmitter i and receiver j incorporating both large and small scale fading, σ_j^2 is the thermal noise power at receiver j, and θ is a threshold parameter that depends on the sensitivity of each receiver.

2.2 Factor Graph Model

A factor graph is a bipartite graph with vertices that are either variable or factor nodes. Let us define the random binary variables as $s_{i,m}^{(k)}$, where $i \in \mathcal{N}_{\setminus s}$, $m \in \mathcal{M}$ and $k \in \mathcal{K}$. The binary variables $s_{i,m}^{(k)}$ are called scheduling variables, meaning that if $s_{i,m}^{(k)} = 1$, the terminal i is scheduled to transmit a packet at time slot m, on frequency channel k. Likewise, when $s_{i,m}^{(k)} = 0$, no transmission occurs for the terminal i on the

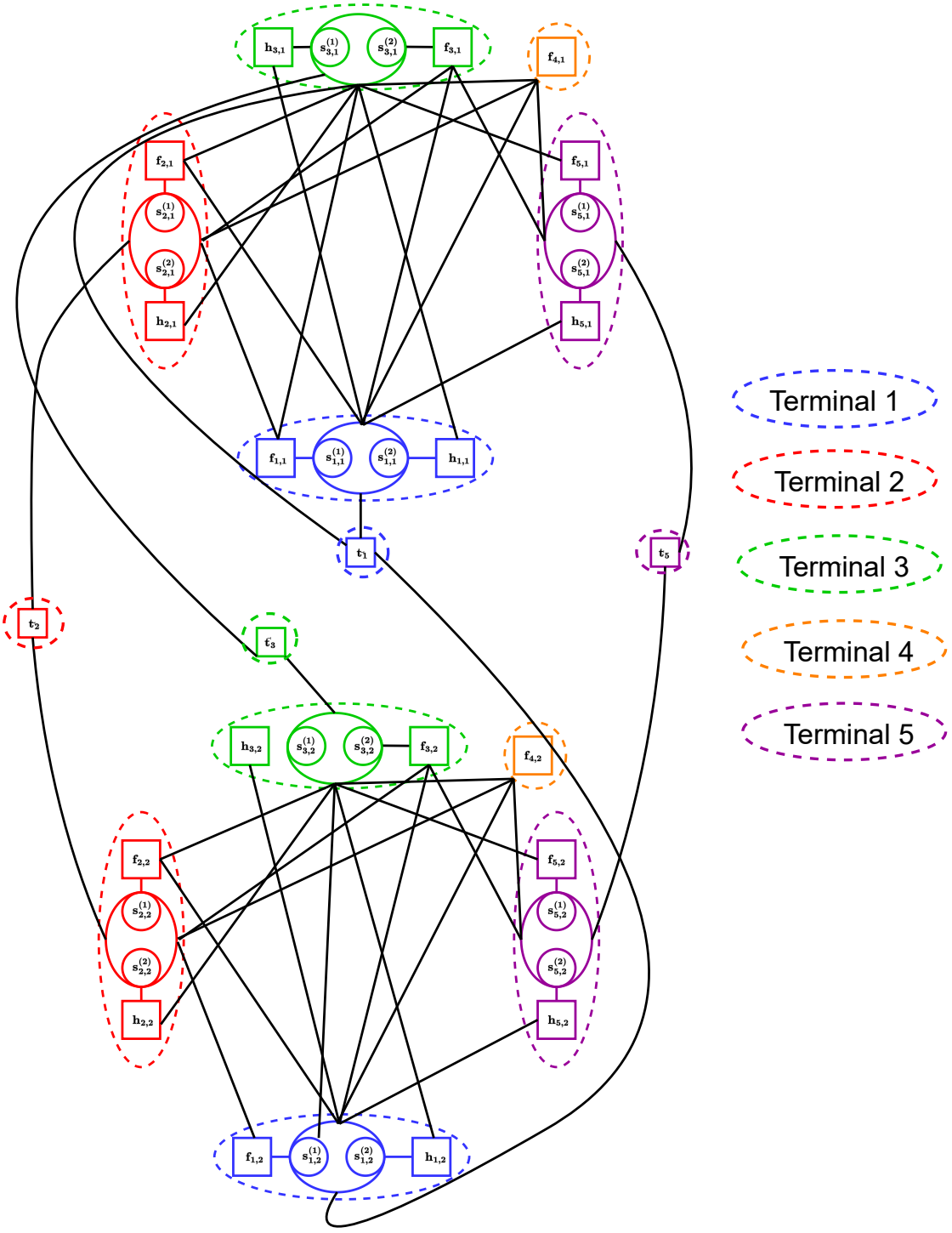

Figure 2.2: A factor graph for the Wireless Sensor Network of $N = 5$ terminals as depicted in Fig. 2.1. The selected number of time and frequency slots is $K = M = 2$.

specified frequency and time slot. The set of all factor nodes is given by [9]:

$$\{\mathbf{g}_J\}_{J=1}^{(N-1)(2M+1)+M} = \Big\{ \{\mathbf{f}_{i,m}\}_{(i,m)\in N\times M}\,, \{\mathbf{h}_{i,m}\}_{(i,m)\in N_{\setminus s}\times M}\,, \{\mathbf{t}_i\}_{i\in N_{\setminus s}} \Big\}. \tag{2.2}$$

The factors denoted by $\mathbf{f}_{i,m}$ impose the routing constraints (so that information reaches the sink), where $i \in \mathcal{N}$ and $m \in \mathcal{M}$. The interference constraints are imposed by the set of factors $\mathbf{h}_{i,m}$, where $i \in \mathcal{N}_{\setminus s}$ and $m \in \mathcal{M}$. Lastly, factors $\mathbf{t}_i$ enforce the transmission constraints, where $i \in \mathcal{N}_{\setminus s}$. Thus, the total number of factors is equal to $NM + (N-1)M + (N-1) = (N-1)(2M+1) + M$. Each factor node is a binary function, which outputs one if the corresponding constraint is satisfied and zero otherwise. A factor graph scheme for the small wireless sensor network of Fig. 2.1 is depicted in Fig. 2.2.

2.3 Belief Propagation

Graphical models compactly represent a joint distribution as a product of factors. For the factor graph model of Section 2.2, the corresponding joint probability is calculated by:

$$Pr(\mathbf{s}) \propto \Big(\prod_{i\in\mathcal{N}\setminus s} t_i(\mathbf{s}_{t_i}) \Big) \cdot \Big(\prod_{m\in\mathcal{M}} \Big(\prod_{i\in\mathcal{N}\setminus s} h_{im}(\mathbf{s}_{h_{im}}) \Big) \Big) \cdot \Big(\prod_{i\in\mathcal{N}} f_{i,m}(\mathbf{s}_{f_{im}}) \Big), \tag{2.3}$$

which should be equal to 1. The variables $\mathbf{s}_{t_i}, \mathbf{s}_{h_{im}}, \mathbf{s}_{f_{im}}$ denote the neighbors of t_i, h_{im}, f_{im} factor, respectively. To obtain a variable assignment that is satisfiable (i.e., constraint-satisfying), the Belief Propagation algorithm is utilized. BP is an iterative inference procedure in which the neighboring nodes of a graph send messages to each other for computing the marginals of the underlying joint distribution. In a factor graph, BP messages are grouped into two categories: variable-to-factor and factor-to variable. Let us first define a vector denoted by $\mathbf{x}$, which consists of all $x_v \equiv s_{i,m}^{(k)}$ variables, where $v \in \{1, \ldots, (N-1)MK\}$, since the sink terminal cannot transmit

on any slot. The BP messages exchanged in such graph are given by:

$$m_{J\to v}^{(n)}(x_v) = C_{J\to v}^{(n)} \times \sum_{\mathbf{x}_y : y\in\partial(J)\setminus v} \Big\{ \mathbf{g}\left(x_v; \mathbf{x}_y\right) \prod_{y\in\partial(J)\setminus v} m_{y\to J}^{(n-1)}(x_y) \Big\}, \tag{2.4}$$

$$m_{v\to J}^{(n)}(x_v) = C_{v\to J}^{(n)} \times P_v(x_v) \prod_{I\in\partial(v)\setminus J} m_{I\to v}^{(n)}(x_v), \tag{2.5}$$

which describe the factor-to-variable messages and variable-to-factor messages at iteration n, respectively. The notation $\partial(\cdot)$ indicates the set of neighbors for the factor or variable inside the parenthesis, while the quantity $P_v(x_v)$ refers to the prior probability of the variable x_v, where $P_v(x_v = 0) = q_v$ and $P_v(x_v = 1) = 1 - q_v$. The normalization factors $C_{v\to J}^{(n)}$ and $C_{J\to v}^{(n)}$ guarantee that $m_{v\to J}^{(n)}(x_v = 0) + m_{v\to J}^{(n)}(x_v = 1) = 1$ and $m_{J\to v}^{(n)}(x_v = 0) + m_{J\to v}^{(n)}(x_v = 1) = 1$, respectively. The BP messages are initialized in the following manner:

$$m_{v\to J}^{(n=0)}(x_v = 0) = 1 - m_{v\to J}^{(n=0)}(x_v = 1) = q_v, \tag{2.6}$$

$$m_{J\to v}^{(n=0)}(x_v = 0) = m_{J\to v}^{(n=0)}(x_v = 1) = 0, \tag{2.7}$$

with $q_v \sim \mathcal{U}[0, 1]$, where the notation $\mathcal{U}[0, 1]$ denotes the continuous uniform distribution over the (closed) interval $[0, 1]$. The estimated marginals for a variable x_v at iteration n, are given by:

$$\hat{\mu}_v^{(n)}(x_v) = P_v(x_v) \prod_{I\in\partial(x_v)} m_{I\to v}^{(n)}(x_v). \tag{2.8}$$

Finally, given Eq. (2.8), we compute the random variable decisions by performing hard decision on the estimated marginals:

$$\mathbf{x}^* = \hat{\boldsymbol{\mu}}(\mathbf{x} = 0) < \hat{\boldsymbol{\mu}}(\mathbf{x} = 1). \tag{2.9}$$

Due to the loopy nature of the factor graphs that encode the specific WSN resource allocation problem, the convergence and correctness of the BP messages is not guaranteed and thus, the estimated marginals in Eq. (2.8) are not always accurate. Therefore, we seek for methods to guarantee BP convergence and correctness.

2.4 Belief Propagation: A Control Theory Approach

2.4.1 Discrete Time Map

A discrete time map is defined by an n-dimensional state vector $\mathbf{x} \in \mathbb{R}^n$ and a function $\mathcal{F} : \mathbb{R}^n \to \mathbb{R}^n$. Note that $\mathcal{F}$ is a composition of n functions $f_i : \mathbb{R}^n \to \mathbb{R}$. We will denote the value of the state vector at time n by $\mathbf{x}^n$, so that

$$\mathbf{x}^{n+1} = \mathcal{F}(\mathbf{x}^n). \tag{2.10}$$

A discrete time map is further said to be non-linear if some (or all) functions $f_i(x_i)$ are nonlinear. For creating a discrete time map for BP, the messages of the current iteration are expressed as a function of the previous iteration. For exposition purposes, the variable to factor messages are denoted by the q_{Jv}, while the variable to factor are denoted by r_{Jv}. For the same purpose, the normalization factors $C_{vJ}^{(n)}$ and $C_{Jv}^{(n)}$ are denoted by $\alpha_{vJ}^{(n)}$ and $\beta_{vJ}^{(n)}$, respectively. Therefore, the messages of Eqs.(2.4)-(2.5) are re-expressed by:

$$\begin{aligned} q_{Jv}^{(n)}(x_v) = \alpha_{Jv}^{(n)} \times \\ \sum_{y \in \partial(J) \setminus v} \Big\{ \mathbf{g}\left(x_v; \mathbf{x}_y\right) \prod_{y \in \partial(J) \setminus v} r_{Jy}^{(n-1)}(x_y) \Big\}, \end{aligned} \tag{2.11}$$

$$\begin{aligned} r_{Jv}^{(n)}(x_v) = \beta_{Jv}^{(n)} \times \\ P_v(x_v) \prod_{I \in \partial(v) \setminus J} q_{Iv}^{(n)}(x_v). \end{aligned} \tag{2.12}$$

Substituting Eq. (2.11) into Eq. (2.12), gives:

$$\begin{aligned} r_{Jv}^{(n)}(x_v) = \beta_{Jv}^{(n)} P_v(x_v) \times \Bigg\{ \prod_{I \in \partial(v) \setminus J} \alpha_{Iv}^{(n)} \times \\ \sum_{y \in \partial(I) \setminus v} \Big\{ \mathbf{g}\left(x_v; \mathbf{x}_y\right) \prod_{y \in \partial(I) \setminus v} r_{Iy}^{(n-1)}(x_y) \Big\} \Bigg\}. \end{aligned} \tag{2.13}$$

Additionally, for each edge $(v, J) \in E$ of the graph, the following equations must be satisfied:

$$q_{Jv}^{(n)}(x_v = 0) + q_{Jv}^{(n)}(x_v = 1) = 1, \tag{2.14}$$

$$r_{Jv}^{(n)}(x_v = 0) + r_{Jv}^{(n)}(x_v = 1) = 1. \tag{2.15}$$

Therefore, the mapping induced by BP for a single edge $(v, J) \in E$ is described by the following four equations:

$$\begin{aligned} r_{Jv}^{(n)}(x_v = 0) = \beta_{Jv}^{(n)} P_v(x_v = 0) \times \Bigg\{ \prod_{I \in \partial(v) \setminus J} \alpha_{Iv}^{(n)} \times \\ \sum_{y \in \partial(I) \setminus v} \Big\{ \mathbf{g}\left(x_v = 0; \mathbf{x}_y\right) \prod_{y \in \partial(I) \setminus v} r_{Iy}^{(n-1)}(x_y) \Big\} \Bigg\}, \end{aligned} \tag{2.16}$$

$$\begin{aligned} r_{Jv}^{(n)}(x_v = 1) = \beta_{Jv}^{(n)} P_v(x_v = 1) \times \Bigg\{ \prod_{I \in \partial(v) \setminus J} \alpha_{Iv}^{(n)} \times \\ \sum_{y \in \partial(I) \setminus v} \Big\{ \mathbf{g}\left(x_v = 1; \mathbf{x}_y\right) \prod_{y \in \partial(I) \setminus v} r_{Iy}^{(n-1)}(x_y) \Big\} \Bigg\}, \end{aligned} \tag{2.17}$$

$$\begin{aligned} \alpha_{Jv}^{(n)} \sum_{y \in \partial(J) \setminus v} \mathbf{g}\left(x_v = 0; \mathbf{x}_y\right) \prod_{y \in \partial(J) \setminus v} r_{Jy}^{(n-1)}(x_y) + \\ \alpha_{Jv}^{(n)} \sum_{y \in \partial(J) \setminus v} \mathbf{g}\left(x_v = 1; \mathbf{x}_y\right) \prod_{y \in \partial(J) \setminus v} r_{Jy}^{(n-1)}(x_y) = 1, \end{aligned} \tag{2.18}$$

$$r_{Jv}^{(n)}(x_v = 0) + r_{Jv}^{(n)}(x_v = 1) = 1, \tag{2.19}$$

where Eq. (2.18) is calculated from substituting Eq. (2.11) into Eq. (2.14). The set of equations in (2.16)-(2.19) defined for all $(v, J) \in E$, describes the (non-linear) mapping induced by BP, which is denoted by:

$$\mathbf{r}^{(n)} = \mathcal{BP}(\mathbf{r}^{(n-1)}, \boldsymbol{\alpha}^{(n)}, \boldsymbol{\beta}^{(n)}). \tag{2.20}$$

2.4.2 Fixed Points

The fixed points of a dynamical system are crucial for the understanding of how the latter responds and evolves over time, given initial conditions. A fixed point $\mathbf{x}^*$ of a mapping $\mathcal{F}$ is a state vector that maps to itself, so that:

$$\mathbf{x}^* = \mathcal{F}(\mathbf{x}^*). \tag{2.21}$$

Therefore, for the mapping induced by BP, the fixed points satisfy the following equation:

$$\mathbf{r}^* = \mathcal{BP}(\mathbf{r}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*). \tag{2.22}$$

A discrete time system does not necessarily converge to a fixed point of the corresponding discrete time map; such fixed points are called *unstable*. On the other hand, the system is guaranteed to converge[1] to the *stable* fixed points, which satisfy the following condition:

$$\rho\left(\mathbf{J}(\mathcal{F}(\mathbf{x}^*))\right) < 1, \tag{2.23}$$

where $\rho\left(\mathbf{J}(\mathcal{F}(\mathbf{x}^*))\right)$ is the spectral radius of the Jacobian matrix $\mathbf{J}$ of the discrete time map $\mathcal{F}(\cdot)$, calculated at the fixed point $\mathbf{x}^*$.

2.4.3 Solving the Fixed Point Equations

The fixed points of the BP discrete time map are given by solving the equation system that is induced by:

$$\mathbf{r}^* = \mathcal{BP}(\mathbf{r}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \Longleftrightarrow \mathbf{r}^* - \mathcal{BP}(\mathbf{r}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = 0. \tag{2.24}$$

The corresponding system of fixed point equations consists of $4|E|$ equations, as defined in Eqs. (2.16)-(2.19) $\forall(v, J) \in E$. The PhD thesis in [10] studied extensively the stability of BP; it was found that the solver in the hom4-ps3 toolbox [11] is the most appropriate solver for the fixed points equations of BP. The complexity of this solver lies on the number of solution sets (i.e. root count). Tighter bounds on the number of solution sets can be provided if the corresponding equation system is sparse. The challenge for the stability analysis of BP lies in solving the fixed point equations of the Belief Propagation system; it was found that the computational cost of solving the equations of this specific problem formulation is prohibitively high. The computation of the Jacobian matrix and its spectral radius is also non-trivial as the size of the factor graph grows. *Therefore, for this problem formulation, the stability analysis of BP is not feasible due to the high underlying computation cost.*

2.5 Belief Propagation: A Heuristic Approach

It is important to highlight that stability (i.e. convergence) does not imply correctness; Belief Propagation may converge to a fixed point that does not satisfy all

[1] On the condition of appropriate initialization.

problem constraints. Considering the prohibitively high computational cost of the stability analysis of Section 2.4, resorting to heuristic methods may be the only solution for guarantying BP's convergence to a fixed point that satisfies the underlying CSP.

2.5.1 Restarting Method

A method called restarting [9] solves the WSN CSP with a low outage probability (i.e., probability of convergence to a non-valid solution), when compared to standard Belief Propagation. The restarting method includes a periodic check of the factor functions output, given the random variable decisions. More specifically, every N_{interm} iterations, the priors of the random variables connected to unsatisfied factors are (randomly) re-initialized, therefore resulting in restarting the messages that are a function of these variables. For convergence acceleration, damping is performed on the BP messages. The restarting method, as described in Algorithm 2.5.1, terminates after a given number of iterations, denoted by T_{max}.

Connections to Control Theory

In this context, we should mention that the Restarting method has strong connections to already existing, control theory techniques for system stabilization to a (specific) fixed point [12; 13]. The re-initialization of the priors can be seen as an intentionally added perturbation to the discrete-time system that describes the BP messages [10], as in [13]. Alternatively, the periodic satisfiability check can be seen as a test for the occurrence of the event that triggers the re-adjustment of the input (i.e., prior probabilities) of the same system, as in [12].

2.5.2 Perturbed Belief Propagation

The second method, called Perturbed Belief Propagation [14], originally established in [15], smoothly interpolates two well-known inference procedures; it starts as BP and ends as a Gibbs sampler:

$$m_{i\to I} \leftarrow \gamma \cdot m_{i\to I} + (1-\gamma)\cdot \delta(x_i - \hat{x}_i), \tag{2.25}$$

where $\delta(\cdot)$ is the Dirac function and $\gamma \in [0,1]$. The Gibbs sampler updates each sample $\hat{x}_i$ by sampling from:

$$\hat{x}_i \sim \hat{\mu}_i(x_i). \tag{2.26}$$

Algorithm 2.5.1: Restarting method

Data: CSP factor graph, number of maximum iterations T_{max}, N_{interm}, damping factor α
Result: variable assignment $\mathbf{x}^*$

initialize the messages and priors;
for $t = 1$ *to* T_{max} **do**
  for *each variable* x_i **do**
    calculate $\mathbf{m}_{I\to i}^{(t)}, \forall i \in \partial(I)$ using Eq. (2.4);
    damping: $\mathbf{m}_{I\to i}^{(t)} = \alpha \cdot \mathbf{m}_{I\to i}^{(t-1)} + (1-\alpha) \cdot \mathbf{m}_{I\to i}^{(t)}$;
  end
  for *each variable* x_i **do**
    calculate $\mathbf{m}_{i\to I}^{(t)}, \forall I \in \partial(i)$ using Eq. (2.5);
    calculate $\hat{\mu}_i(x_i)$ using Eq. (2.8);
  end
  if mod $(t, N_{interm}) == 0$ **then**
    if *CSP unsatisfied* **then**
      for *each unsatisfied factor* J **do**
        for *each* $x_y \in \partial(J)$ **do**
          restart $\mathbf{m}_{y\to I}^{(t+1)}, \forall I \in \partial(y)$
        end
      end
    end
  end
end
return $\mathbf{x}^*$;

The parameter γ smoothly increases from 0 to 1, stochastically biasing the BP messages towards the current estimate of marginals. Since the procedure is inherently stochastic, if the CSP is satisfiable, re-application of perturbed BP to the problem (i.e., as described in Algorithm 2.5.2), may provide a valid solution[2].

Algorithm 2.5.2: Perturbed BP method

Data: CSP factor graph, number of iterations T
Result: variable assignment $\mathbf{x}^*$

initialize the messages and priors;
$\gamma \leftarrow 0$;
for $t = 1$ *to* T **do**
  for *each variable* x_i **do**
    calculate $\mathbf{m}_{I \to i}^{(t)}, \forall i \in \partial(I)$ using Eq. (2.4);
  end
  for *each variable* x_i **do**
    calculate $\mathbf{m}_{i \to I}^{(t)}, \forall I \in \partial(i)$ using Eq. (2.5);
    calculate $\hat{\mu}_i^{(t)}(x_i)$ using Eq. (2.8);
    sample $\hat{x}_i \sim \hat{\mu}_i^{(t)}(x_i)$;
    $\mathbf{m}_{i \to I}^{(t)} \leftarrow \gamma \cdot \mathbf{m}_{i \to I}^{(t)} + (1 - \gamma) \cdot \delta(x_i, \hat{x}_i)$;
  end
  $\gamma \leftarrow \gamma + \frac{1}{T-1}$;
end
return: $\mathbf{x}^*$;

2.6 Simulation Results

For the simulation results, we will use two different networks, whose routing connectivity is depicted in Figs. 2.3-2.4. The number of available frequency channels/slots is set to $K = 2$ and the number of available time slots is set to $M = 4$. For the 9-terminal WSN, the SINR threshold θ is set to 3 and 9 dB, while for the 35-terminal WSN it is set to 8 dB. Regarding the restarting method, the N_{interm} parameter is set to 5 and 8, for the small and large network topology, respectively, while the damping factor is set to $\alpha = 0.3$. For perturbed BP, the number of iterations is set to $T = 10$ at the starting attempt, which was increased by a factor of 2 in case of failure (i.e., unsatisfying assignment $\mathbf{x}^*$). This is repeated until the cumulative number of iterations

[2]Other variations of perturbed BP were also examined (e.g., incorporating local re-initialization/restarting of messages or message contradiction check as in [15]), but failed to provide better algorithm performance.

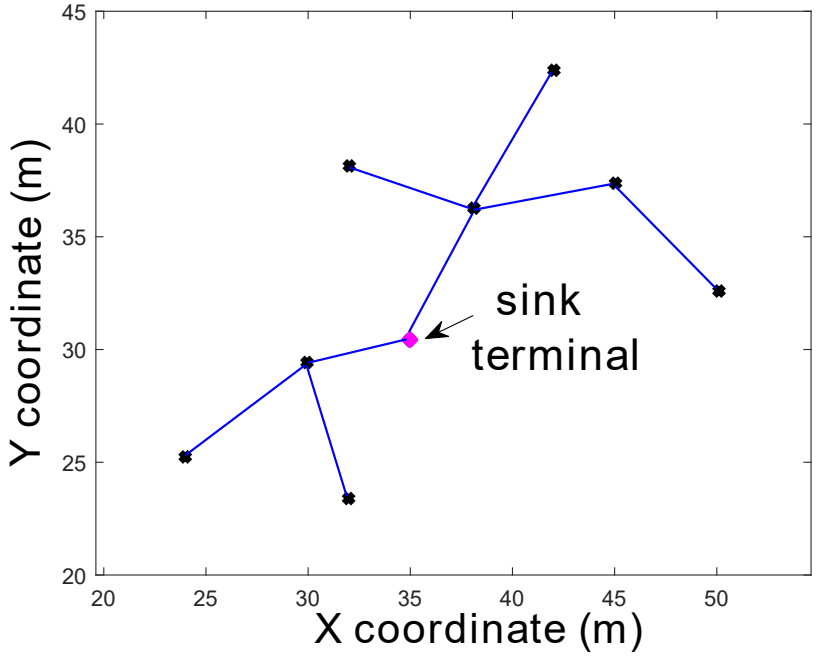

Figure 2.3: Connectivity for a wireless sensor network of $N = 9$ sensors.

exceeds a predetermined value, denoted by T_{max}, or until a satisfiable assignment $\mathbf{x}^*$ is found. We evaluate the performance of the restarting and perturbed methods by calculating the outage probability, i.e., the probability of a method providing a non-valid solution for a given maximum number of BP iterations. The maximum number of iterations is set to $T_{max} = \{30, 70, 150, 310\}$ so that Perturbed BP completes at most $2, 3, 4$ and 5 attempts, respectively. The results in Fig. 2.5-2.6 and Fig. 2.7 were obtained by averaging over 10^3 and 500 Monte Carlo experiments, respectively.

These results demonstrate that, as the maximum number of iterations increases, both methods decrease the probability of outage. Additionally, both methods provide higher outage probability for the 35-terminal network, when compared to the 9-terminal network results. This can be explained by the higher problem complexity of the 35-terminal resource allocation problem, caused by large number of terminals and limited number of resources. Regarding the performance comparison of the two methods, it can be seen that perturbed BP is outperformed by the restarting method, regardless of the network size. As it can be seen clearly in Fig. 2.7, the restarting method offers lower outage probability for a given T_{max} value, when compared to Perturbed BP.

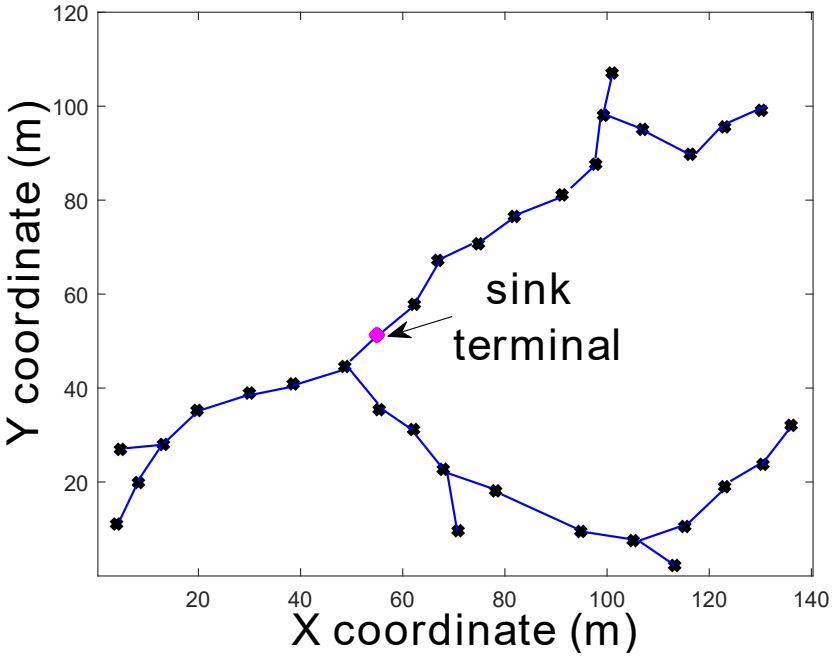

Figure 2.4: Connectivity for a wireless sensor network of $N = 35$ sensors.

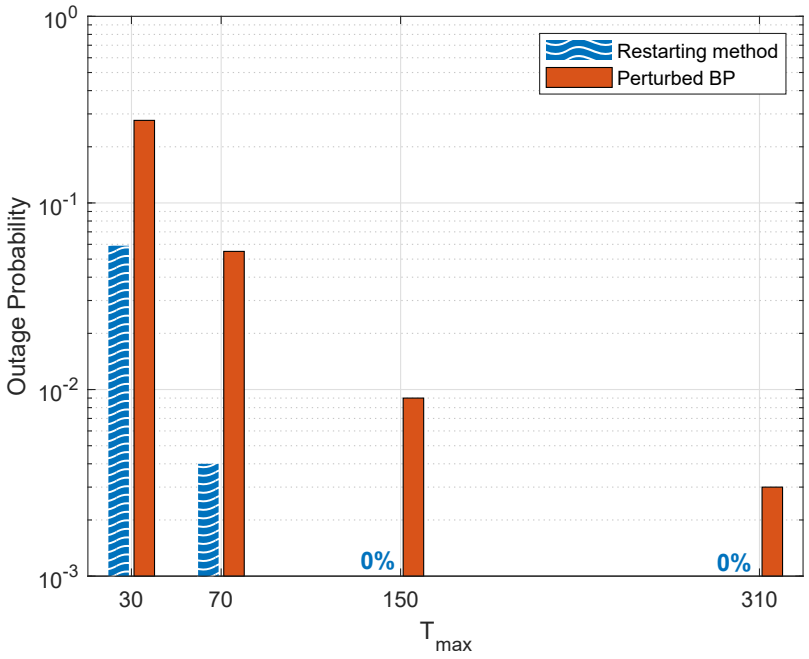

Figure 2.5: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 9-terminal network ($N = 9$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 3$ dB.

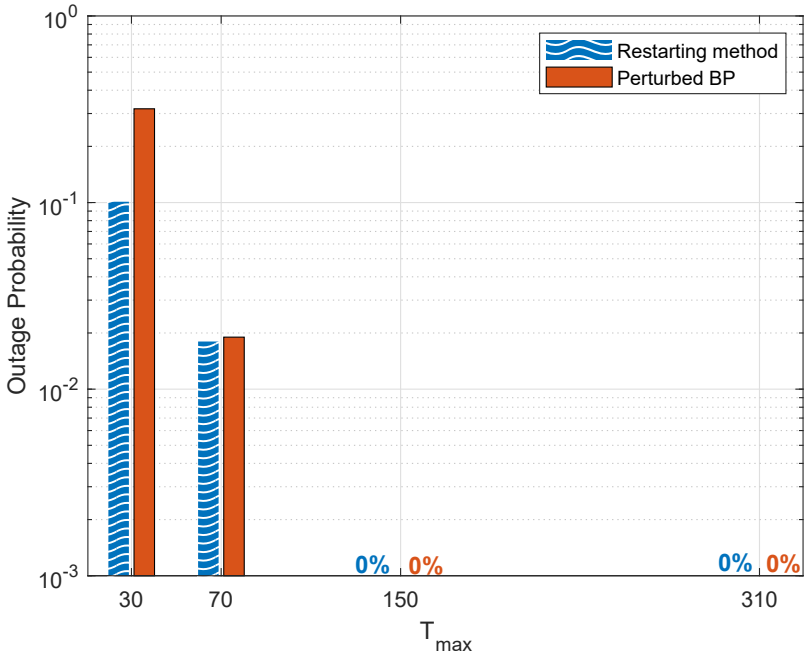

Figure 2.6: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 9-terminal network ($N = 9$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 9$ dB.

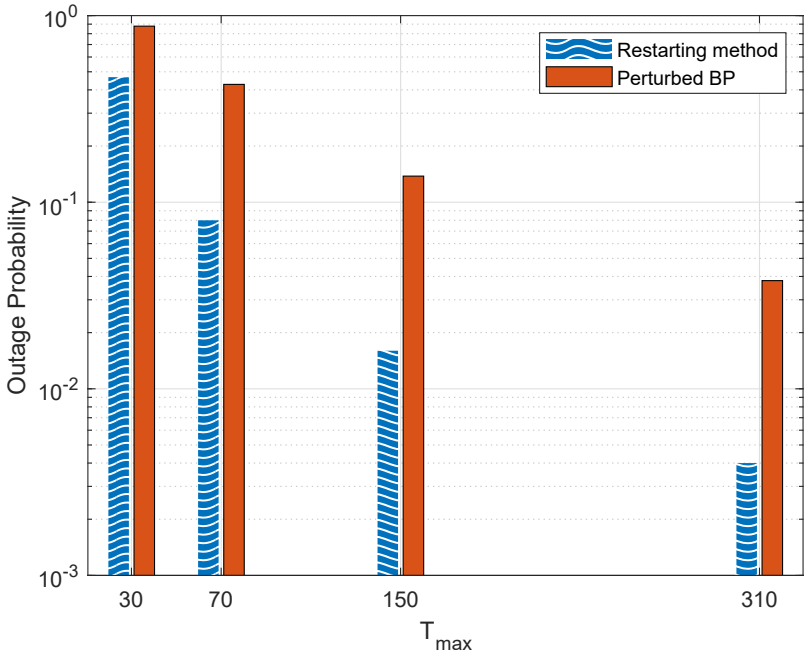

Figure 2.7: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 35-terminal network ($N = 35$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 8$ dB.

Chapter 3

Energy Efficient User Association in Wireless Heterogeneous Networks

The rapid growth of data traffic, driven by a new generation of wireless devices contributes to global energy consumption. Thus, recent research has focused on the optimization of the energy efficiency (EE) towards environmental and economic sustainability. In next generation heterogeneous networks, small cells are overlaid with macro cells; user association (to micro or macro base stations) algorithms optimize the EE of a cellular network (e.g., [16] and references therein).

Offering distributed solutions to wireless communication problems has been mainly attacked with game theory techniques (e.g., work in [16]). In the past years, it has been realized that inference techniques, such as belief propagation (BP), based on message passing over probabilistic graphical models, can offer (distributed or centralized) solutions to certain classes of challenging optimization problems [17].

The main difficulty is to guarantee convergence to a valid solution, when the employed graphical model is loopy. For example, BP can solve combinatorial optimization problems, including maximum weight matching [18]. Sufficient conditions for the convergence and correctness of BP for general linear program (LP) formulations were recently provided [19]. BP is presumably a better choice than centralized LP solvers, considering that no central authority is needed and solution can be offered through structured message passing. Energy efficient user association with inference techniques has been also offered in [20]. However, no theoretical guarantees for convergence were offered, despite the loopy nature of the employed graphical model.

In this chapter, the problem of Energy efficient user association is relaxed and expressed as a Linear Program, which is then solved distributively with max-product BP over BSs. BP convergence and correctness guarantees are offered and the computation complexity is analyzed and reduced, with careful precomputations. Finally, the performance of the proposed algorithm is compared with state-of-the-art.

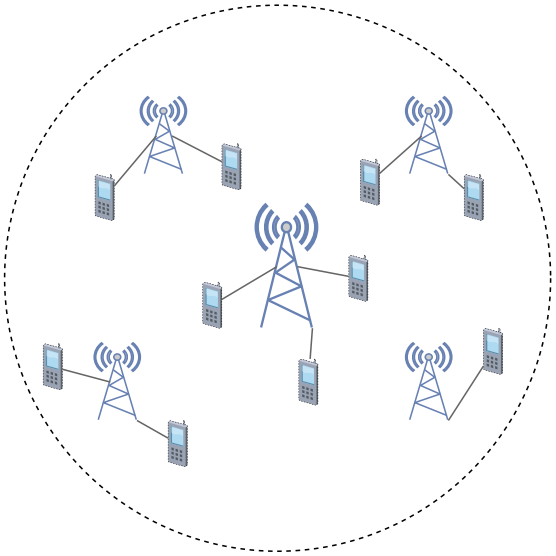

Figure 3.1: Scheme of a small, two-tier Heterogeneous Network. A MBS (Macro Base Station) is located in the center of the cell, while a number of mBS (micro Base Stations) are arranged within the cell. User Equipments (UEs) within the Macrocell coverage area associate to a BS.

3.1 System Model

A two-tier heterogeneous network is considered, consisting of $B-1$ micro base stations (mBS) and one macro base station (MBS), all indexed by set $I = \{1, \ldots, B\}$. Users are located inside the coverage area of the macro-cell and they are denoted by set $A = \{1, \ldots, N\}$. Given spectrum sharing in practice, the achievable rate for the downlink between user $\alpha \in A$ and BS $i \in I$ follows:

$$R_{i\alpha} = \frac{1}{m_i} \log_2 \left(1 + \frac{u_i \, h_{i,a} \, P_i}{\sigma^2 + \sum_{j \in I \setminus i} u_j \, h_{j,a} \, P_j} \right), \tag{3.1}$$

where m_i denotes the number of users served by base station (BS) $i \in I$; average noise and transmit power are denoted by σ^2, P_i, respectively; each binary variable u_i for $i \in I$, indicates whether BS i is active, i.e., transmitting ($u_i = 1$) or in sleep mode, i.e., not transmitting ($u_i = 0$). The channel between user and BS is modeled according to Rayleigh distribution, as follows:

$$h_{i,\alpha} = |g_{i,\alpha}|^2 \, \frac{\lambda_c}{4\pi d_0^2} \left(\frac{d_0}{D_{i,\alpha}} \right)^{\beta}, \tag{3.2}$$

where $g_{i,\alpha}$ denotes the small-scale fading coefficient and follows a circularly symmetric complex Gaussian with unit variance, i.e., $g_{i,\alpha} \sim \mathcal{CN}(0,1)$, β is the path-loss exponent, distance between BS i and user equipment (UE) α is denoted by $D_{i,\alpha}$, reference distance d_0 is set to 1m and λ_c denotes the carrier wavelength.

Energy efficiency (EE) of a heterogeneous network is measured in bits/Hz/Joule and is given as follows [20]:

$$\eta = \frac{\text{Throughput [bits/second/Hz]}}{\text{Power Expenditure [Joule/second]}} \tag{3.3}$$

$$= \sum_{i \in I} \frac{\sum_{\alpha \in A} R_{i\alpha} \cdot x_{i\alpha}}{P_{all}(v, u_1)}, \tag{3.4}$$

where variable $x_{i\alpha}$ denotes association ($x_{i\alpha} = 0$) or no association ($x_{i\alpha} = 1$) of user α to BS i; total power expenditure of the network is modeled as follows:

$$P_{all}(v, u_1) = (v-1) \cdot (P_T^m + P_{op}^m) + u_1 \cdot (P_T^M + P_{op}^M), \tag{3.5}$$

where $v = \sum_{i=1}^{B} u_i$ indicates the total number of active (transmitting) base stations, while u_1 denotes the state (active or not) of the macro BS. The transmission and operation power for the mBS are denoted by P_T^m and P_{op}^m, respectively, while for the MBS, the corresponding quantities are given by P_T^M and P_{op}^M.

It is noted that energy efficiency, as formulated in Eq. (3.4) [20], is a *nonlinear* function of the user association variables $\{x_{i\alpha}\}$, due to binary variables $\{u_i\}$, denominator parameter v, as well as term $1/m_i$ at the downlink rates per active BS.

3.2 Problem Relaxation and Re-Formulation

Assume that binary variables $\{u_i\}$ and parameters $\{m_i\}$ are set (with a specific methodology, offered below). In that way, $v = \sum_{i=1}^{B} u_i$ and $P_{all}(v, u_1)$ are also set; let $\mathbf{x}$ the user association vector and $\mathbf{w}$ the corresponding weight vector, given as follows:

$$\mathbf{x} = [x_{11} \dots x_{1N} \dots x_{B1} \dots x_{BN}]^T \in \{0,1\}^{BN}, \tag{3.6}$$

$$\mathbf{w} = [w_{11} \dots w_{1N} \dots w_{B1} \dots w_{BN}]^T \in \mathbb{R}_{0+}^{BN}. \tag{3.7}$$

Given $\{u_i\}$, $\{m_i\}$, it is easy to see that maximization of energy efficiency η in Eq. (3.4) can be relaxed to a linear programming (LP) problem, described as follows:

$$\begin{aligned} \mathbf{x}^* \quad &= \arg\max_{\mathbf{x}} \mathbf{w}^T \cdot \mathbf{x} = \arg\max_{\mathbf{x}} \sum_{i\in I} \sum_{\alpha\in A} w_{i\alpha} \cdot x_{i\alpha} \\ \text{s.t.} \quad &\text{C1: } \sum_{\alpha\in A} x_{i\alpha} = m_i, \ \forall\ i \in I, \\ &\text{C2: } \sum_{i\in I} x_{i\alpha} = 1, \ \forall\ \alpha \in A, \end{aligned} \tag{3.8}$$

where C1 sets the traffic load of each BS $i \in I$ to m_i users, while C2 states that each user $\alpha \in A$ is served by exactly one BS. The weights $w_{i\alpha}$ for each (i,α) are given by:

$$w_{i\alpha} = \begin{cases} R_{i\alpha} \cdot P_{all}(v, u_1)^{-1} & \text{if } m_i > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{3.9}$$

The weight parameters $u_i, v, m_i\ \forall i \in I$ are set considering a sleep/on-off technique for the BS power modes and a uniformly distributed traffic load, described below:

3.2.1 Sleep mode/on-off Technique

The energy efficiency of the LP solution $\mathbf{x}^*$ is strongly connected to the BS power states u_i and corresponding number of active BS v, which are given as input to the LP, due to the weights definition in Eq. (3.9). A number of $k = B - v$ BSs are put into sleeping (non-transmitting) mode. The selection of inactive BS is based only on channel state criteria. Specifically, the set $\mathcal{I}$ of active BSs is obtained as follows:

$$\mathcal{I} = I - \left\{ \mathbf{argmin}_k \left(\begin{bmatrix} \gamma_1 & \cdots & \gamma_B \end{bmatrix}^T \right) \right\}, \tag{3.10}$$

$$\gamma_i = \sum_{\alpha\in A} \gamma_{i\alpha} = \sum_{\alpha\in A} \left(\frac{h_{i,a} P_i}{\sigma^2 + \sum_{j\in I\setminus i} h_{j,a} P_j} \right) \ \forall i \in I, \tag{3.11}$$

where $\mathbf{argmin}_k(\cdot)$ returns the set of indices corresponding to the k lowest values of the input vector. Distributed calculation of the maximum in a set is feasible and examples have been reported in the literature [21]. The BS power states u_i are then calculated by:

$$u_i = \begin{cases} 1 & \text{if } i \in \mathcal{I}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.12}$$

3.2.2 BS Traffic Load

The traffic load is uniformly distributed to the transmitting/active BS. Therefore, the number of serving users $m_i \; \forall i \in I$ is given by:

$$m_i = \begin{cases} N \div v & \text{if } u_i = 1, \\ 0 & \text{otherwise,} \end{cases} \tag{3.13}$$

where $\div$ denotes the integer division. The remaining network traffic is given by the modulo of N/v and is also re-distributed to the active BSs, by performing the operation $m_i = m_i + 1$ for each BS, until $\sum_{i \in I} m_i = N$ i.e., all users are served.

3.2.3 Variable Elimination

The association variables associated with inactive BSs always remain zero, i.e., $x_{i\alpha} = 0, i \notin \mathcal{I}, \forall\, \alpha$. Thus, discarding such variables will have no impact on the optimization process. The user association variables are then re-defined as follows:

$$\mathbf{x} = \Big\{ \mathbf{x}_{i,\alpha} \in \{0,1\}^{Nv} \text{ for } i \in \mathcal{I}, \alpha \in A \Big\}. \tag{3.14}$$

Likewise, the weights are given by:

$$\mathbf{w} = \Big\{ \mathbf{w}_{i,\alpha} \in \mathbb{R}_{+}^{Nv} \text{ for } i \in \mathcal{I}, \alpha \in A \Big\}. \tag{3.15}$$

Thus, the LP solution must satisfy the following:

$$\mathbf{x}^* \in \Big\{ \mathbf{x}_{i,\alpha} \in \{0,1\}^{Nv} \text{ for } i \in \mathcal{I}, \alpha \in A \Big\}. \tag{3.16}$$

3.3 Graphical Model Formulation

The goal of this work is to offer a distributed solution (across the BSs) of the relaxed problem in Eq. (3.8). Towards this goal, the graphical model (GM) formulation for the LP of Eq. (3.8) is offered subsequently. The binary association variables $\mathbf{x} = \{x_{i\alpha}\}^{Nv}$ are input to specific factor nodes that check the validity of the LP constraints in Eq. (3.8) and return 1 if the constraints are satisfied and 0 otherwise. Given two constraints in Eq. (3.8), two kinds of factor nodes are constructed, denoted by F_α and H_i, $\forall\; i \in \mathcal{I}$, $\forall\; \alpha \in A$. These factors are called non-variable factors and their

domain is given by:

$$\mathbf{dom}F_{\alpha} = \Big\{\{x_{i'\alpha} \in \mathbb{B}\} : i' \in \mathcal{I}\Big\} = \mathbb{B}^{v}, \tag{3.17}$$

$$\mathbf{dom}H_{i} = \Big\{\{x_{i\alpha'} \in \mathbb{B}\} : \alpha' \in A\Big\} = \mathbb{B}^{N}, \tag{3.18}$$

where $\mathbb{B}^N$ is a set containing all binary vectors of length N. The corresponding factor functions are given below:

$$F_{\alpha}(\mathbf{x}_{*\alpha}) = \begin{cases} 1, & \text{if } \sum\limits_{i\in I} x_{i\alpha} = 1, \\ 0, & \text{otherwise,} \end{cases} \tag{3.19}$$

$$H_{i}(\mathbf{x}_{i*}) = \begin{cases} 1, & \text{if } \sum\limits_{\alpha\in A} x_{i\alpha} = m_{i}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.20}$$

Vectors $\mathbf{x}_{*\alpha}$ and $\mathbf{x}_{i*}$ denote all variables associated with user α and active BS i, respectively. The variable factor function $\psi_{i\alpha}$ for each (i, α) is given by:

$$\psi_{i\alpha}(x_{i\alpha}) = \exp\{w_{i\alpha}\, x_{i\alpha}\}, \tag{3.21}$$

where the weights $w_{i\alpha}$ are defined in Section 3.2. By eliminating the variables and factors associated with inactive BS, the size of the factor graph is reduced. As an example, the factor graph for a heterogeneous network of $B = 3$ BSs - out of which $v = 2$ are active - is depicted in Fig. 3.2, for a number of $N = 4$ users.

The joint probability distribution function, encoded by the factor graph that describes the LP of Eq. (3.8), is given below:

$$\begin{aligned} \Pr(\mathbf{x}) \propto & \Bigg\{\Big\{\prod_{i\in\mathcal{I}}\prod_{\alpha\in A}\psi_{i\alpha}(x_{i\alpha})\Big\}\times \\ & \Big\{\prod_{i\in\mathcal{I}} H_{i}(\mathbf{x}_{i*})\Big\} \times \Big\{\prod_{\alpha\in A} F_{\alpha}(\mathbf{x}_{*\alpha})\Big\}. \end{aligned} \tag{3.22}$$

The maximum posterior probability (MAP) assignment for the probability distribution in Eq. (3.22) is identical to the optimal solution $\mathbf{x}^*$:

$$\mathbf{x}^{*} = \arg\max_{\mathbf{x}} \Pr(\mathbf{x}). \tag{3.23}$$

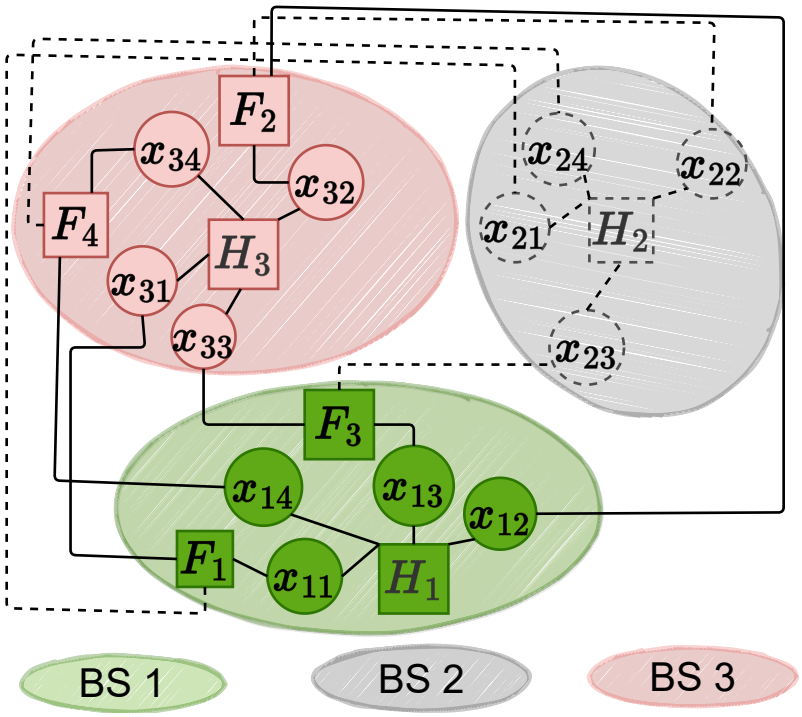

Figure 3.2: Factor graph model for a heterogeneous network with $B = 3$ BS and $N = 4$ users. Here, the BS indexed by $i = 2$ is switched off (i.e. $v = 2$), thus factors and variables of the specific BS are eliminated, which is depicted with dotted lines.

3.4 Proposed Loopy Max Product Algorithm

The MAP assignment of Eq. (3.23) can be distributively computed with the max-product algorithm on the corresponding GM (factor graph). A more convenient notation for the random variables in $\mathbf{x}$ (converted to $\mathbf{z}$) and weights in $\mathbf{w}$, is adopted:

$$\mathbf{z} = [z_1, \ldots, z_{N \cdot v}]^T \in \{0, 1\}^{Nv}, \tag{3.24}$$

$$\mathbf{w}' = [w'_1, \ldots, w'_{N \cdot v}]^T \in \mathbb{R}_+^{Nv}, \tag{3.25}$$

accordingly. The variable factor for each variable z_n, $n \in \{1, \ldots, N \cdot v\}$, is given by:

$$\psi_n(z_n) = \exp\{w'_n \, z_n\}. \tag{3.26}$$

The factor functions are also re-labeled as:

$$\mathbf{g}_{J=1}^{N+v} = \{\{H_i\}_{i \in \mathcal{I}}, \{F_\alpha\}_{\alpha=1}^{N}\}. \tag{3.27}$$

Denote by $\mathcal{N}, \mathcal{J}$ the set of association variables and factors, respectively. The neighborhood in the (bipartite) factor graph[1] of association variable indexed by $n \in \{1, \ldots, v \cdot N\}$ and factor indexed by $J \in \{1, \ldots, v + N\}$ is defined as follows:

$$\mathcal{J}_n \triangleq \{J \in \mathcal{J}\text{: variable } z_n \text{ is adjacent to factor } g_J\}, \tag{3.28}$$

$$\mathcal{N}_J \triangleq \{n \in \mathcal{N}\text{: factor } g_J \text{ is adjacent to variable } z_n\}. \tag{3.29}$$

Given these definitions, the max-product messages exchanged in the factor graph, are given below [19; 22]:

$$\mu_{J\to n}^{(t+1)}(z_n = c) \propto \max_{\mathbf{z}_{\mathcal{N}_J}: z_n = c} \mathbf{g}_J(\mathbf{z}_{\mathcal{N}_J}) \times \prod_{y \in \mathcal{N}_J \setminus n} \mu_{y\to J}^{(t)}(z_y), \tag{3.30}$$

$$\mu_{n\to J}^{(t+1)}(z_n = c) \propto \psi_n(z_n) \prod_{f \in \mathcal{J}_n \setminus J} \mu_{f\to n}^{(t)}(z_n), \tag{3.31}$$

where $\mathbf{z}_{\mathcal{N}_J}$ denotes the set of neighboring association variables of factor J and notation $a \in \mathcal{B} \setminus b$ means that a assumes all possible values of set $\mathcal{B}$ excluding b. The estimated marginals at iteration t are given by:

$$b_n^{(t)}(z_n) = \psi_n(z_n) \prod_{f \in \mathcal{J}_n} \mu_{f\to n}^{(t)}(z_n). \tag{3.32}$$

The output at iteration t is obtained by performing hard decision-making on the estimated marginals:

$$\hat{z}_n^{(t)} = 1 \text{ if } b_n^{(t)}(z_n = 1) \geq b_n^{(t)}(z_n = 0) \text{ else } \hat{z}_n^{(t)} = 0, \tag{3.33}$$

$\forall\ n \in \{1, \ldots, N \cdot v\}$; the estimates $\{\hat{z}_n^{(t)}\}$ are stacked in vector $\mathbf{s}^{(t)}$, which is the estimate of the MAP solution $\mathbf{x}^*$. The message passing procedure, has converged at iteration t if there is no time step $t' > t$ such that $\mathbf{s}^{(t)} \neq \mathbf{s}^{(t')}$.

Let by w_{max} the maximum weight of the LP weights $\{w_{i\alpha}\}$ in Eq. (3.9). Additionally, $\rho > 0$ is a positive constant given by:

$$\rho \triangleq \inf_{\mathbf{x} \in \mathcal{P} \setminus \mathbf{x}^*} \frac{\mathbf{w}^T \cdot \mathbf{x} - \mathbf{w}^T \cdot \mathbf{x}^*}{\|\mathbf{x} - \mathbf{x}^*\|_1} > 0, \tag{3.34}$$

where $\mathcal{P}$ is the polytope of all LP solutions. Given these definitions, the following theorem offers convergence guarantees to the LP-optimal solution:

[1]Factor graphs are bipartite, i.e., variables connect to factors only and vice versa.

Theorem 3.4.1. *The inference algorithm of this work converges to the LP solution in at most* $\tau = \left(\frac{w_{max}}{\rho} + 1\right)(N \cdot v)^{2.5}$ *iterations.*

Proof. Work in [19] establishes sufficient conditions for max-product convergence to the solution of a LP. In the following lines, we re-write these conditions and prove their satisfiability for the GM and LP formulations in Sections 3.2 and 3.3, respectively.

- ***Condition 1***: *The solution of the LP is integral and unique.*
  Uniqueness of solution can be guaranteed by adding sufficiently small independent random noise $n_{i\alpha} \sim \mathcal{N}(0, 10^{-2} \cdot w_{i\alpha})$ to the weights $w_{i\alpha}$ [18; 19]. The solution of the LP from Section 3.2 is by definition integral, i.e., $\mathbf{x}^* \in \{0,1\}^{Nv}$.

- ***Condition 2***: *Each variable of the graphical model is connected to at most two non-variable factors.*
  Each variable $x_{i\alpha}$ of the GM described by Eq. (3.22) is connected to exactly two non-variable factors (i.e., H_i and F_α). This is also clearly shown in Fig. 3.2.

- ***Condition 3***: *From [19, Lemma 3.3], all non-variable factor functions should be expressed by:*

$$g_J(\mathbf{z}_{\mathcal{N}_J}) = \begin{cases} 1, & \text{if } L_J \leq \mathbf{c}^T \cdot \mathbf{z}_{\mathcal{N}_J} \leq U_J \\ 0, & \text{otherwise.} \end{cases}, \tag{3.35}$$

  where $\mathbf{c} \in \{-1,1\}^{|\mathcal{N}_J|}$ *and* $U_J, L_J \in \mathbb{R}$.
  Let by $\mathbf{c}_{J_1} = \mathbf{1}_N$ and $J_1 \in \{1, \ldots, v\}$ the indices for $g_{J_1} \in \{H_i\}_{i \in \mathcal{I}}$ factors. The corresponding factor functions are given by Eq. (3.20) and can be expressed as:

$$g_{J_1}(\mathbf{z}_{\mathcal{N}_{J_1}}) = \begin{cases} 1, & \text{if } \mathbf{c}_{J_1}^T \cdot \mathbf{z}_{\mathcal{N}_{J_1}} = m_{J_1}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.36}$$

  Let by $\mathbf{c}_{J_2} = \mathbf{1}_v$. The functions for factors $\mathbf{g}_{J_2}$, where $J_2 \in \{v+1, \ldots, v+N\}$, are described by Eq. (3.19) and can be expressed as:

$$g_{J_2}(\mathbf{z}_{\mathcal{N}_{J_2}}) = \begin{cases} 1, & \text{if } \mathbf{c}_{J_2}^T \cdot \mathbf{z}_{\mathcal{N}_{J_2}} = 1, \\ 0, & \text{otherwise.} \end{cases} \tag{3.37}$$

  Finally, given the constants $L_{J_1} = U_{J_1} = m_{J_1}$ and $L_{J_2} = U_{J_2} = 1$, the factor

functions in Eqs. (3.36)-(3.37) can be expressed as in Eq. (3.35):

$$g_{J_1}(\mathbf{z}_{\mathcal{N}_{J_1}}) = \begin{cases} 1, & \text{if } L_{J_1} \leq \mathbf{c}_{J_1}^T \cdot \mathbf{z}_{\mathcal{N}_{J_1}} \leq U_{J_1}, \\ 0, & \text{otherwise,} \end{cases} \tag{3.38}$$

$$g_{J_2}(\mathbf{z}_{\mathcal{N}_{J_2}}) = \begin{cases} 1, & \text{if } L_{J_2} \leq \mathbf{c}_{J_2}^T \cdot \mathbf{z}_{\mathcal{N}_{J_2}} \leq U_{J_2}, \\ 0, & \text{otherwise.} \end{cases} \tag{3.39}$$

Therefore, max-product on the GM that encodes Eq. (3.22) is guaranteed to converge with arbitrary message initialization to the solution $\mathbf{x}^*$ of the LP in Section 3.2, for $\tau \leq \left(\frac{w_{max}}{\rho} + 1\right) K$ iterations. From [19, Lemma 3.4], K is a constant no greater than $|\mathbf{z}|^{2.5}$, where $|\mathbf{z}| = N \cdot v$ is the number of variables. □

3.5 Computational Complexity

Max-product complexity is determined by the computational complexity of the factor to variable messages in Eq. (3.30), which is exponential on the node degree of the corresponding factor J. Each factor $\{F_\alpha\}_{\alpha=1}^N$ neighbors to v variables and thus, the complexity when computing a message associated with such factor follows [23]:

$$\mathcal{O}(v \cdot 2^v). \tag{3.40}$$

Likewise, for factors $\{H_i\}_{i\in\mathcal{I}}$, the node degree is N and thus, the computation complexity for a message associated with such factor is given by:

$$\mathcal{O}(N \cdot 2^N). \tag{3.41}$$

Thus, the computational cost of max-product per iteration for a single message, assuming that all messages are computed in parallel, is given by:

$$\mathcal{O}(v \cdot 2^v + N \cdot 2^N). \tag{3.42}$$

The computational cost of max-product can be reduced if only the valid variable configurations (i.e., $\mathbf{z}_{\mathcal{N}_J} : \mathbf{g}_J(\mathbf{z}_{\mathcal{N}_J}) = 1$) are considered for Eq. (3.30). Valid configurations give positive message values, while non-valid configurations (i.e., $\mathbf{z}_{\mathcal{N}_J} : \mathbf{g}_J(\mathbf{z}_{\mathcal{N}_J}) = 0$) always give zero. Therefore, discarding non-valid configurations has no effect on the

maximization operation in Eq. (3.30). Let by:

$$J_1 \in \{1, \ldots, v\},\ J_2 \in \{v+1, \ldots, v+N\}, \tag{3.43}$$

the indices for $g_{J_1} \in \{H_i\}_{i\in\mathcal{I}}$ and $g_{J_2} \in \{F_\alpha\}_{\alpha=1}^{N}$ factors, respectively. The valid configurations for messages associated with $\mathbf{g}_{J_1}$ factors are then given by:

$$\begin{aligned}\mathcal{Z}_{(J_1,n)}(z) \triangleq \Big\{ &\mathbf{z}_{\mathcal{N}_{J_1}\setminus n} \in \mathbb{B}^{N-1} : \\ &\mathbf{g}_{J_1}(\mathbf{z}_{\mathcal{N}_{J_1}\setminus n}, z_n = z) = 1 \Big\},\end{aligned} \tag{3.44}$$

while for $\mathbf{g}_{J_2}$ factors, the valid configurations are given by:

$$\begin{aligned}\mathcal{Z}_{(J_2,n)}(z) \triangleq \Big\{ &\mathbf{z}_{\mathcal{N}_{J_2}\setminus n} \in \mathbb{B}^{v-1} : \\ &\mathbf{g}_{J_2}(\mathbf{z}_{\mathcal{N}_{J_2}\setminus n}, z_n = z) = 1 \Big\}.\end{aligned} \tag{3.45}$$

From the definition of factors $\mathbf{g}_{J_2} \triangleq \{\mathbf{F}_\alpha\}_{\alpha=1}^{N}$ in Eq. (3.19), each configuration in $\mathcal{Z}_{(J_2,n)}(z=0)$ contains only one digit that is equal to 1; thus, all valid configurations are rows of the identity matrix of size $v-1$. On the other hand, for $\mathcal{Z}_{(J_2,n)}(z=1)$, only a single configuration of zeros is valid. Thus, the number of valid configurations for each factor J_2 is given by:

$$|\mathcal{Z}_{(J_2,n)}(z=0)| + |\mathcal{Z}^1_{(J_2,n)}(z=1)| = v - 1 + 1 = v. \tag{3.46}$$

From the definition of factors $\mathbf{g}_{J_1} \triangleq \{\mathbf{H}_i\}_{i\in\mathcal{I}}$ in Eq. (3.20), the valid configurations of each $\mathcal{Z}_{(J_1,n)}(z=0)$ and $\mathcal{Z}_{(J_1,n)}(z=1)$, are given by all different combinations of $N-1$ binary digits containing exactly m_{J_1} and $m_{J_1}-1$ ones, respectively. Thus, the number of these configurations is described by the binomial coefficient:

$$|\mathcal{Z}_{(J_1,n)}(z=0)| = \binom{N-1}{m_{J_1}}, \tag{3.47}$$

$$|\mathcal{Z}_{(J_1,n)}(z=1)| = \binom{N-1}{m_{J_1}-1}, \tag{3.48}$$

with $0 < m_{J_1} \leq N-1$ for all $J_1 \in \mathcal{I}$, due to the uniform base station traffic load. For each edge (J, n), the set of valid assignments for Eq. (3.30) can be pre-computed and stored efficiently.[2] Performing the maximization over valid configurations (instead

[2]e.g., by utilizing standard MATLAB functions such as **nchoosek**$(\cdot)$, **bin2dec**$(\cdot)$.

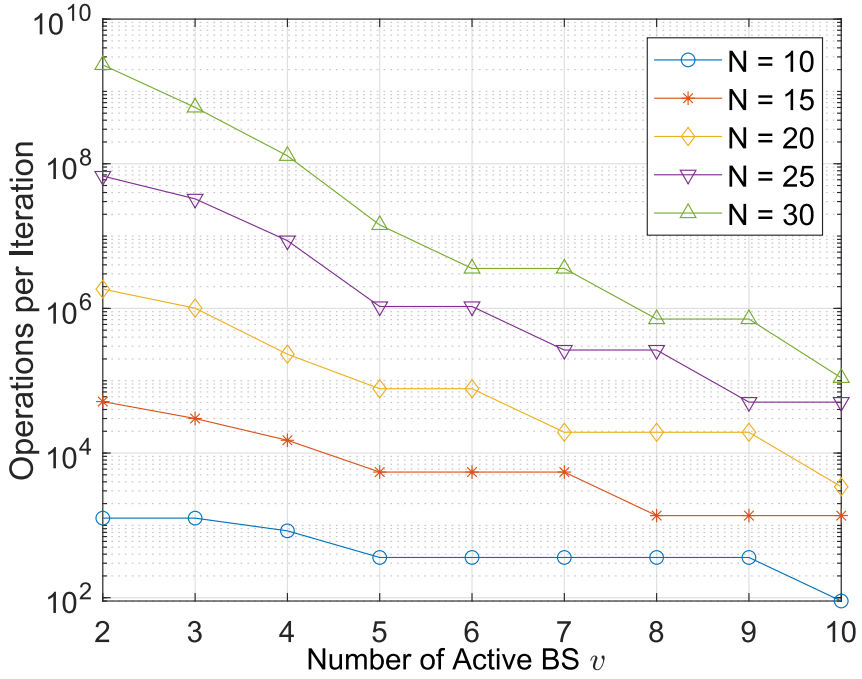

Figure 3.3: Computational complexity of a single iteration of the proposed algorithm as a function of the number of active BS v, for a different number of users N.

of all possible configuration) for all factors, the computational complexity of max-product for a message per iteration, assuming parallel computation of all messages, is reduced as follows:

$$\mathcal{O}\left(v^2 + N \cdot \max_{i \in \mathcal{I}} \binom{N-1}{m_i} + N \cdot \max_{i \in \mathcal{I}} \binom{N-1}{m_i - 1}\right), \tag{3.49}$$

which for $N \geq 10$ and $2 \leq v \leq 10$ (and $\{m_i\}$'s set as in this work) is bounded by:

$$\mathcal{O}\left(N \cdot \max_{i \in \mathcal{I}} \binom{N-1}{m_i}\right). \tag{3.50}$$

Fig. 3.3 plots the reduced computational complexity in Eq. (3.50) of a single max-product iteration, as a function of v and N. As expected, the computational complexity increases with N, for given v; this is the price to pay for the *distributed* solution of the association problem by a network of BSs. For fixed N, complexity is a non-increasing function of v.

Chapter 4

Numerical Results

In this chapter, the energy efficiency (EE) of a two-tier heterogeneous network with $B = 10$ BSs is compared to state-of-the-art distributed algorithms, with simulation parameters offered in Table 4.1. Fig. 4.1 depicts the user association output of the proposed approach for $v = 8$ active BSs and $N = 20$ users at random positions; the simulated macro-cell has a diameter of $R = 200$m and users are randomly arranged inside the cell, while the MBS is located in the center of the cell. Simulation results are obtained by calculating the EE of different methods for 200 Monte Carlo experiments. In each experiment, the position of users randomly changes, as well as the small scale fading. For the proposed approach, the number of active BS v is randomly set for each experiment. The performance of the proposed approach is then compared to state-of-the-art algorithms, denoted by IE and SC. The IE algorithm [24] utilizes a distributed protocol for the sub-gradient method with complexity $\mathcal{O}(BN)$. On the other hand, SC [25] is a low-complexity, matrix-based EE algorithm. The SC algorithm chooses the BS that has the greatest sum of the maximum achievable data rate. Then, the chosen BS is turned on, and all UEs having connectivity with the chosen BS are associated with it. This is repeated until all UEs are associated. The EE geometric mean is shown in Fig. 4.2; the proposed method outperforms IE and SC for $N = \{10, 15, 20\}$. Figs. 4.3-4.5 clearly show that for a number of Monte Carlo

Table 4.1: Simulation parameters.

Description	**Value**
Micro BS transmit power P_T^m	37.99 dBm
Macro BS transmit power P_T^M	21.14 dBm
Macro BS operating power P_{op}^M	56.62 W
Micro BS operating power P_{op}^m	6.80 W
Thermal noise power σ^2	-101 dBm
Bandwidth W	20 MHz
Carrier frequency F_c	1.9 GHz
Path loss exponent β	4
Maximum number of users at MBS max(m_1)	18
Maximum number of users at mBS max(m_i)	9

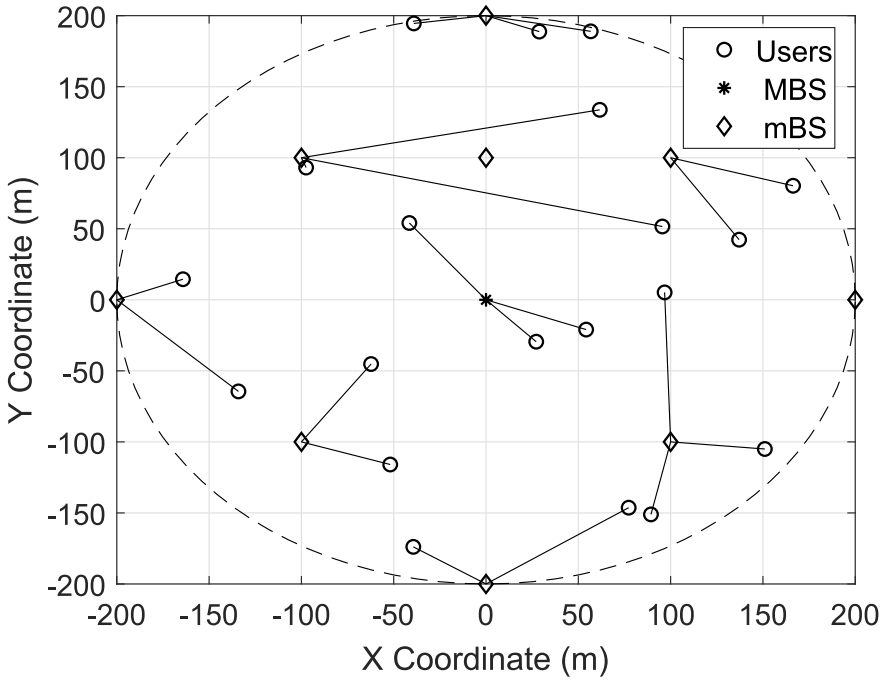

Figure 4.1: Scheme depicting a result of the proposed user association algorithm for $N = 20$ users, $B = 10$ available BS and $v = 8$ active BS.

experiments, the proposed approach converges very slowly to the optimal solution $\mathbf{x}^*$ and therefore, the median value of iterations is calculated. In Fig. 4.6, the median of Max Product iterations is shown for $N = \{10, 15, 20\}$. Finally, Fig. 4.6 shows the median of max-product iterations vs N.

The EE empirical cumulative distribution (ECDF) function for $N = \{10, 15, 20\}$ are demonstrated in Figs. 4.7-4.9. Since the rightmost curve has always the best performance, these results show that in general, the proposed approach offers higher EE than the IE method. That is, only for low EEs (i.e., in Fig. 4.8 for $\eta < 0.07$ bits/Hz/Joule and in Fig. 4.9 for $\eta < 0.1$ bits/Hz/Joule), the IE method outperforms the proposed approach. It is also observed that for all N, there is a region of EEs in which SC outperforms the other two methods. Figs. 4.10-4.12 however show the spectral efficiency ECDF and demonstrate that SC provides low throughput. Figs. 4.10-4.12 also clearly show that IE outperforms the proposed method in terms of spectral efficiency, for all N. This is probably due to the fair throughput allocation the specific method promotes.

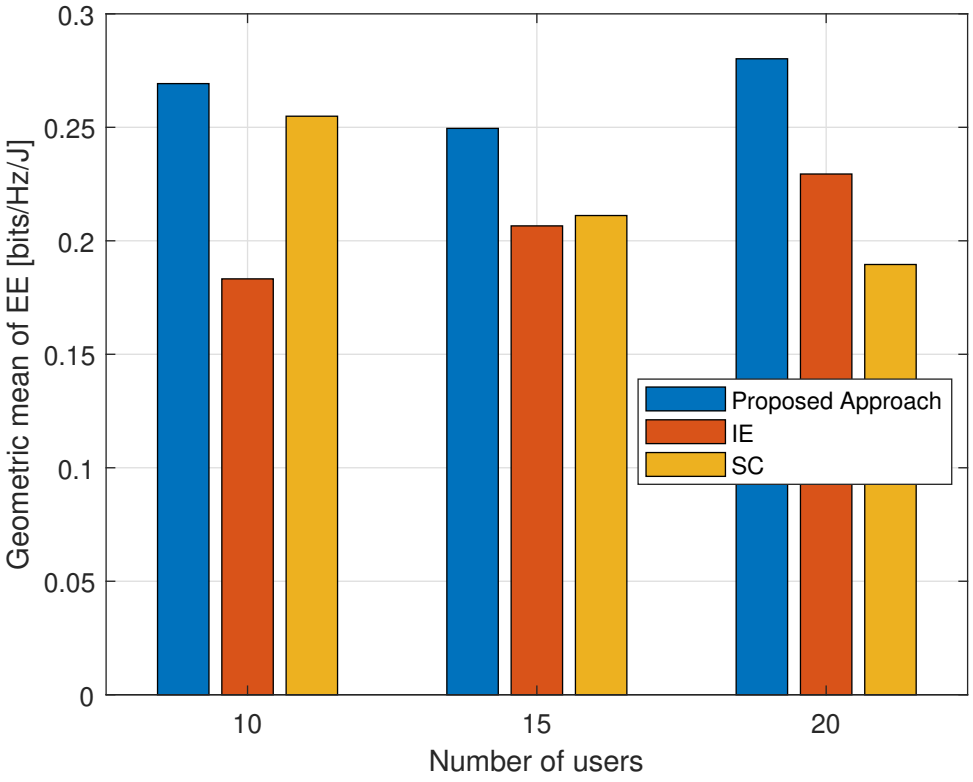

Figure 4.2: Average Energy Efficiency for $B = 10$ BS and $N = \{10, 15, 20\}$ users. The results for each N are obtained by calculating the geometric mean of 200 Monte Carlo experiments.

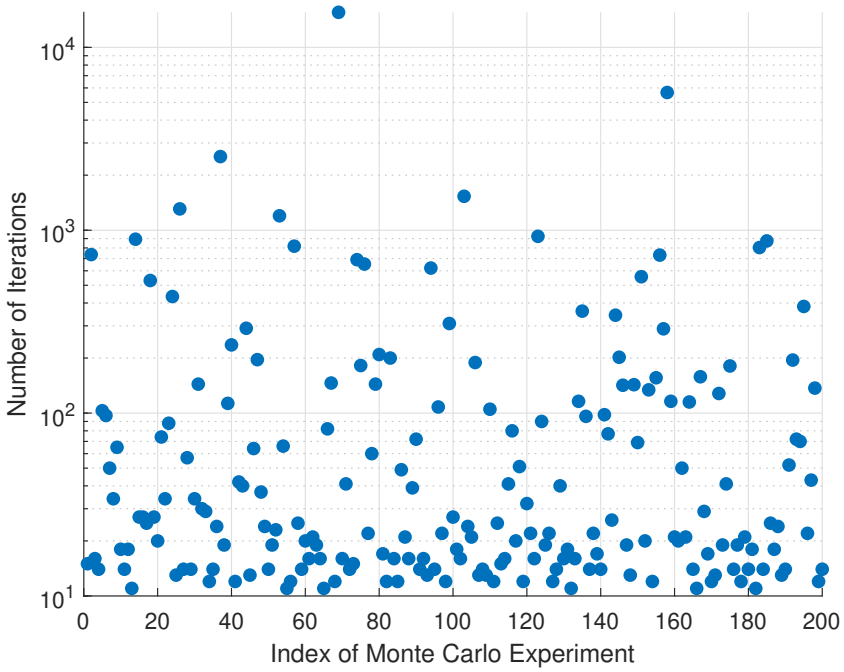

Figure 4.3: The number of iterations in each Monte Carlo experiment for the convergence of Max Product for $N = 10$ users and $B = 10$ BS. The figures show that the convergence of the Max Product algorithm for some Monte Carlo experiments is too slow.

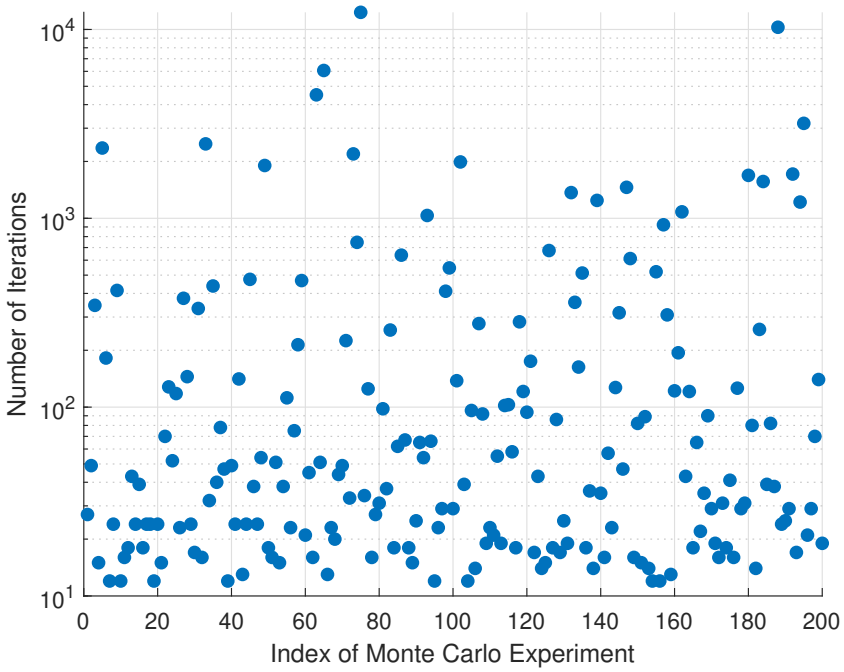

Figure 4.4: The number of iterations in each Monte Carlo experiment for the convergence of Max Product for $N = 15$ users and $B = 10$ BS. The figures show that the convergence of the Max Product algorithm for some Monte Carlo experiments is too slow.

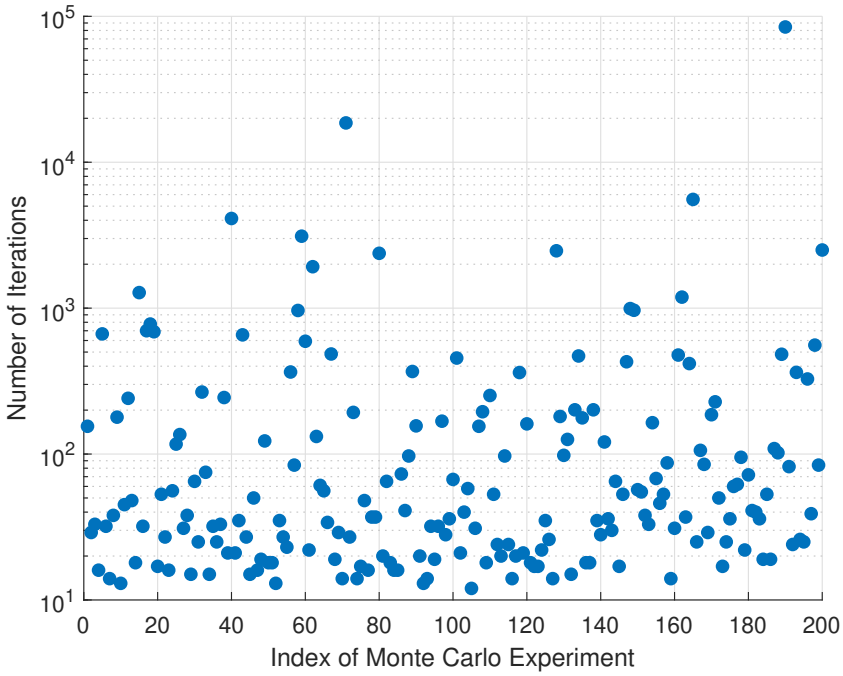

Figure 4.5: The number of iterations in each Monte Carlo experiment for the convergence of Max Product for $N = 20$ users and $B = 10$ BS. The figures show that the convergence of the Max Product algorithm for some Monte Carlo experiments is too slow.

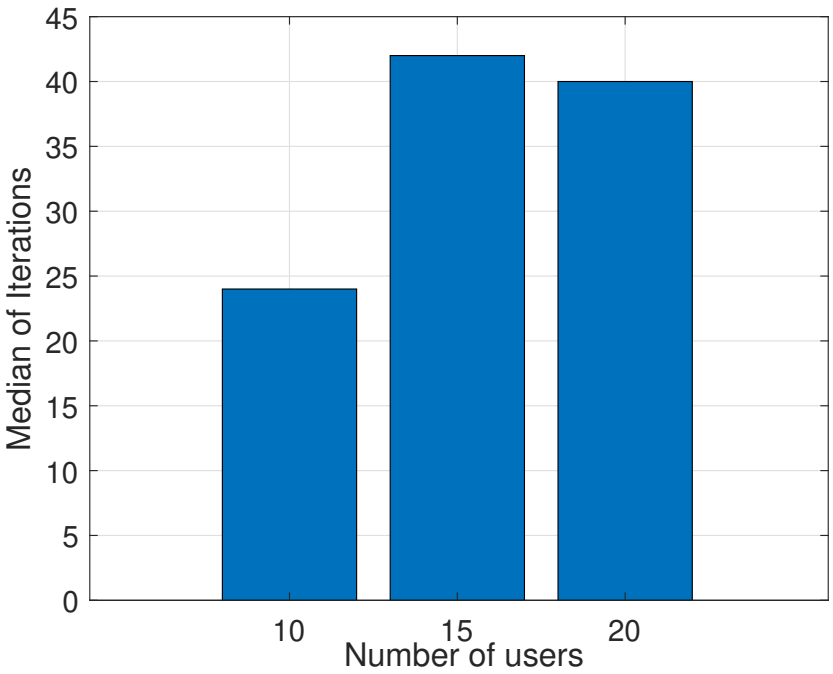

Figure 4.6: The median number of iterations for Max Product to converge to the optimal solution for $N = \{10, 15, 20\}$ users and $B = 10$ available BS.

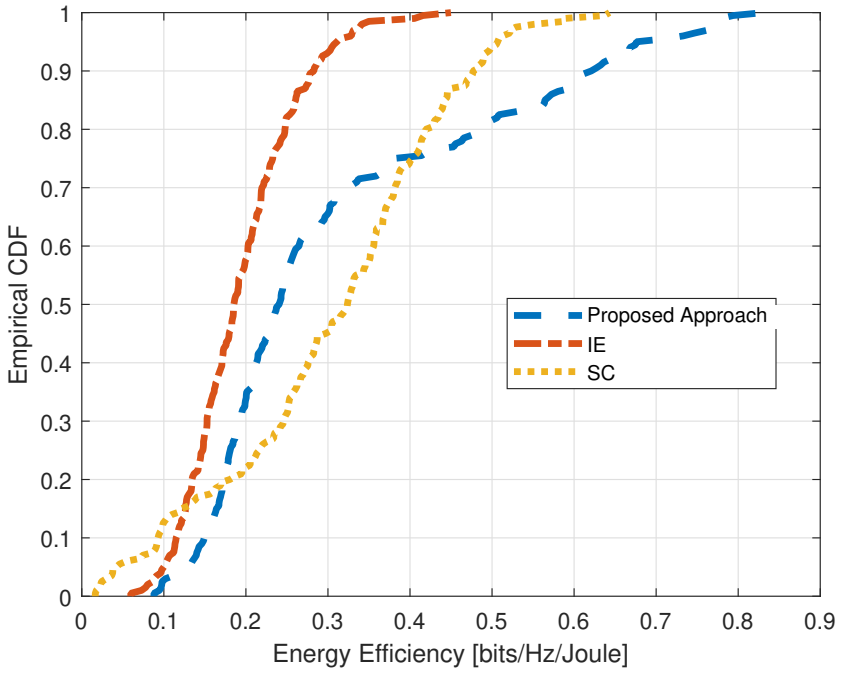

Figure 4.7: Empirical CDF of network Energy Efficiency for $B = 10$ and $N = 10$.

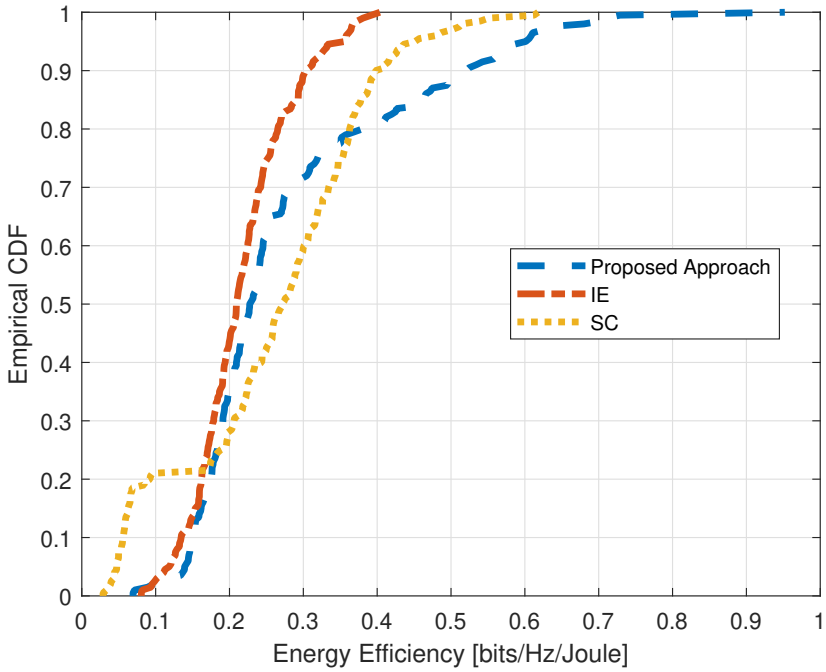

Figure 4.8: Empirical CDF of network Energy Efficiency for $B = 10$ and $N = 15$.

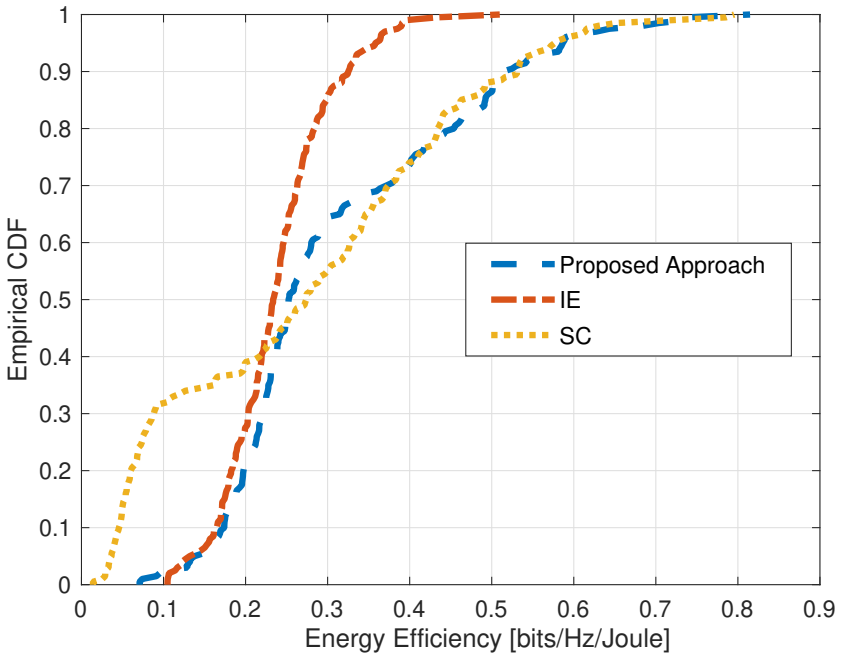

Figure 4.9: Empirical CDF of network Energy Efficiency for $B = 10$, and $N = 20$.

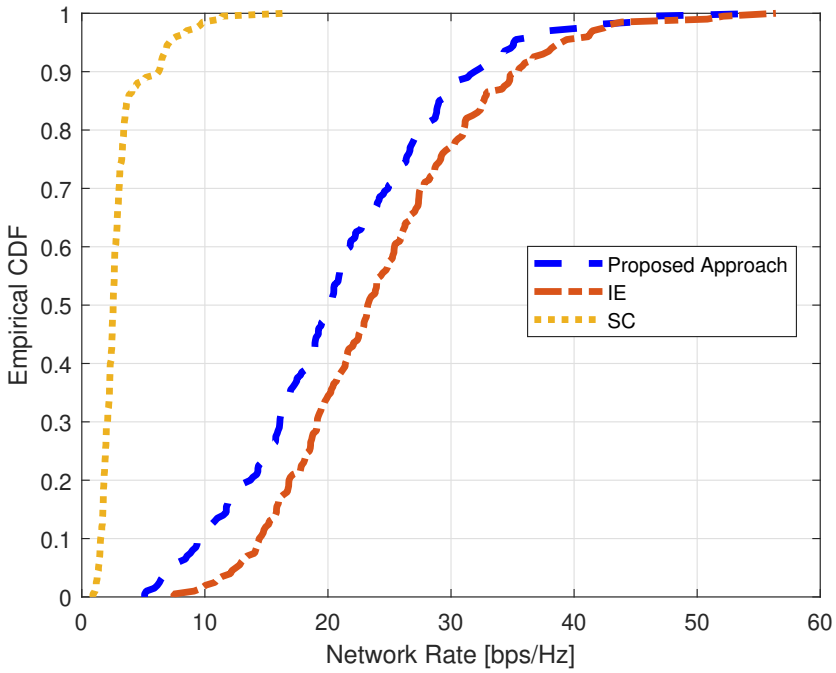

Figure 4.10: Empirical CDF of Network Rate for $B = 10$ and $N = 10$.

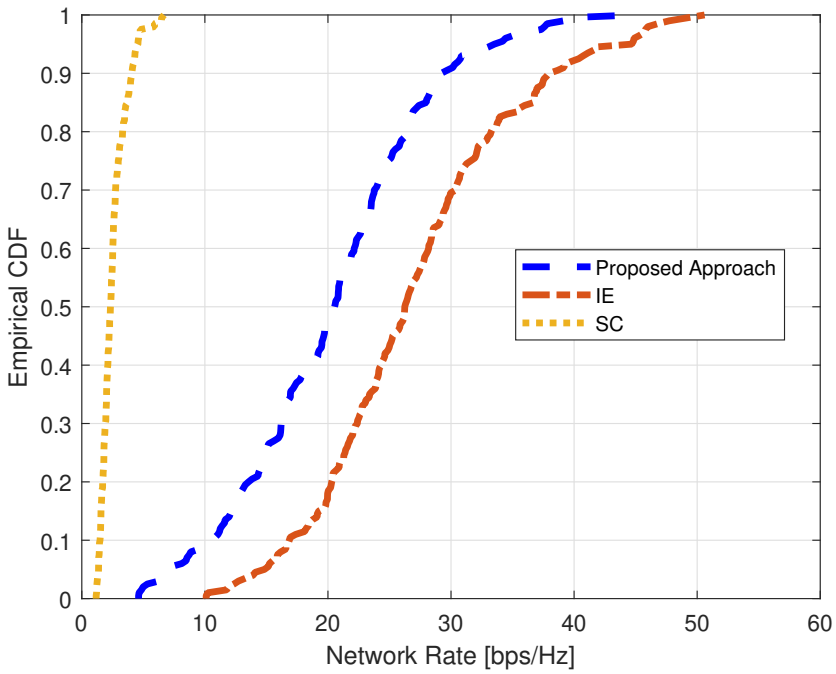

Figure 4.11: Empirical CDF of Network Rate for $B = 10$ and $N = 15$.

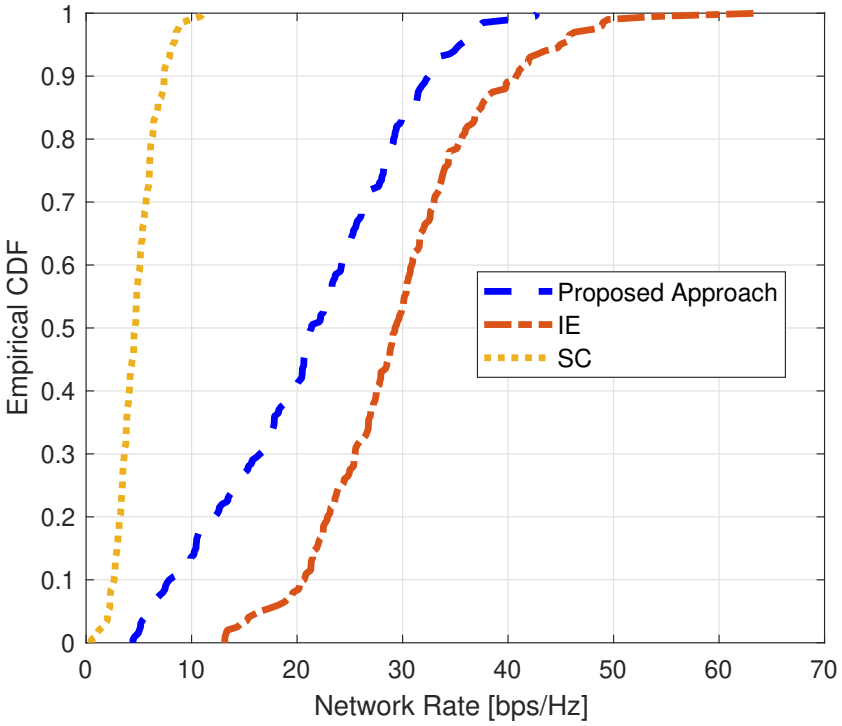

Figure 4.12: Empirical CDF of Network Rate for $B = 10$ and $N = 20$.

Chapter 5

Conclusion

5.1 Conclusions

This work focuses on distributed, inference-based solvers for the problems of resource allocation and user association in wireless networks. For the resource allocation problem, the belief propagation (BP) algorithm is utilized. First, the convergence of Belief Propagation is re-visited from a control theory perspective; however, the resulting complexity is too high. Next, two BP-based methods were evaluated; perturbed BP and a method called restarting. The simulation results indicate that the restarting method converges faster to a valid solution.

Finally, a relaxed formulation is established for the problem of energy efficient user association in the downlink of heterogeneous networks. A distributed solver based on max-product BP is provided; correctness and convergence guarantees are also offered. The results for a small heterogeneous network indicate that the proposed approach offers higher geometric mean of EE than state-of-the-art, while not sacrificing spectral efficiency. The price for distributed - across the BSs - user association, based on message passing (inference) is the increased computation per BS.

5.2 Future Work

As future work, the performance of the proposed user association algorithm should be evaluated for a higher number of users; different network architectures could also be tested. As a final note, NS-3 simulations could be provided.

Bibliography

[1] Y. Wu, J. A. Stankovic, T. He, J. Lu, and S. Lin, "Realistic and efficient multi-channel communications in wireless sensor networks," in *Proc. IEEE Int. Conf. on Computer Communications (Infocom)*, Phoenix, ZA, USA, May 2008, pp. 1193–1201.

[2] A. Ghosh, O. D. Incel, V. S. A. Kumar, and B. Krishnamachari, "Multi-channel scheduling algorithms for fast aggregated convergecast in sensor networks," in *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MSN)*, Macau, China, Oct. 2009, pp. 363–372.

[3] X. Wang, X. Wang, X. Fu, G. Xing, and N. Jha, "Flow-based real-time communication in multi-channel wireless sensor networks," in *Wireless Sensor Networks, 6th European Conference (EWSN)*, Cork, Ireland, Feb. 2009.

[4] A. Saifullah, Y. Xu, C. Lu, and Y. Chen, "Distributed channel allocation protocols for wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 9, pp. 2264–2274, Sep. 2014.

[5] K. R. Chowdhury, P. Chanda, D. P. Agrawal, and Q. A. Zeng, "DCA - a distributed channel allocation scheme for wireless sensor networks," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Berlin, Germany, Sep. 2005, pp. 1–11.

[6] J. Chen, Q. Yu, P. Cheng, Y. Sun, Y. Fan, and X. Shen, "Game theoretical approach for channel allocation in wireless sensor and actuator networks," *IEEE Trans. Autom. Control*, vol. 56, no. 10, pp. 2332–2344, Oct. 2011.

[7] B. Han, V. S. A. Kumar, M. V. Marathe, S. Parthasarathy, and A. Srinivasan, "Distributed strategies for channel allocation and scheduling in software-defined radio networks," in *Proc. IEEE Int. Conf. on Computer Communications (Infocom)*, Rio de Janeiro, Brazil, Apr. 2009, pp. 1521–1529.

[8] P. N. Alevizos and A. Bletsas, "Inference-based resource allocation for multi-cell backscatter sensor networks," in *Proc. IEEE Int. Conf. Communications*, Shanghai, China, May 2019.

[9] P. N. Alevizos, E. A. Vlachos, and A. Bletsas, "Inference-based distributed channel allocation in wireless sensor networks," 2017. [Online]. Available: http://arxiv.org/abs/1703.06652

[10] C. Knoll, "Understanding the behavior of belief propagation," Ph.D. dissertation, Graz University of Technology, Graz, Austria, Nov. 2019, advisor: F. Pernkopf.

[11] Tien-Yien Li,Tsung-Lin Lee and Nick Ovenhouse, "Hom4ps3 polyhedral homotopy continuation solver." [Online]. Available: http://www.hom4ps3.org

[12] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. Autom. Control*, vol. 52, no. 9, pp. 1680 – 1685, Sep. 2007.

[13] J.-Y. Choi, M. Krstic, K. Ariyur, and J. Lee, "Extremum seeking control for discrete-time systems," *IEEE Trans. Autom. Control*, vol. 47, no. 2, pp. 318–323, Feb. 2002.

[14] R. Chatzigeorgiou, P. Alevizos, and A. Bletsas, "Evaluation of inference algorithms for distributed channel allocation in wireless networks," in *11th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, Bremen, Germany, Jun. 2022, pp. 1–4.

[15] R. G. Siamak Ravanbakhsh, "Perturbed message passing for constraint satisfaction problems," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1249–1274, May 2015.

[16] N. Sapountzis, T. Spyropoulos, N. Nikaein, and U. Salim, "User Association in Hetnets: Impact of Traffic Differentiation and Backhaul Limitations," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3396–3410, Sep. 2017.

[17] P. N. Alevizos, E. Vlachos, and A. Bletsas, "Factor graph-based distributed frequency allocation in wireless sensor networks," in *Proc. IEEE Global Commun. Conf. (Globecom)*, Austin, TX, 2014, pp. 3395–3400.

[18] M. Bayati, D. Shah, and M. Sharma, "Max-product for Maximum Weight Matching: Convergence, Correctness, and LP duality," *IEEE Trans. Inf. Theory*, vol. 54, no. 3, pp. 1241–1251, Mar. 2008.

[19] S. Park and J. Shin, "Convergence and Correctness of Max-Product Belief Propagation for Linear Programming," *SIAM Journal on Discrete Mathematics*, vol. 31, pp. 2228–2246, Sep. 2017.

[20] S. H. Lee, M. Kim, H. Shin, and I. Lee, "Belief Propagation for Energy Efficiency Maximization in Wireless Heterogeneous Networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 56–68, Sep. 2021.

[21] A. Bletsas, "Intelligent antenna sharing in cooperative diversity wireless networks," Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA, Sep. 2005.

[22] S. Ahn, S. Park, M. Chertkov, and J. Shin, "Minimum Weight Perfect Matching via Blossom Belief Propagation," Sep. 2015. [Online]. Available: https://arxiv.org/abs/1509.06849

[23] D. Shah, *Sum-product on factor tree graphs, MAP elimination algorithm*, ser. Lecture notes on Algorithms for Inference. Massachusetts Institute of Technology, Department of Electrical Engineering and Computer Science, 2014.

[24] D. Liu, L. Wang, Y. Chen, T. Zhang, K. K. Chai, and M. Elkashlan, "Distributed Energy Efficient Fair User Association in Massive MIMO Enabled HetNets," *IEEE Commun. Lett.*, vol. 19, no. 10, pp. 1770–1773, Jul. 2015.

[25] G. Lee and H. Kim, "Green Small Cell Operation of Ultra-Dense Networks Using Device Assistance," *Energies*, vol. 9, no. 12, pp. 1–19, Dec. 2016.