Evaluation of Inference Algorithms for Distributed Channel Allocation in Wireless Networks

Roza Chatzigeorgiou, Panos Alevizos and Aggelos Bletsas
School of Electrical and Computer Engineering, Technical University of Crete, Chania 73100, Greece
{rchatzigeorgiou, palevizos}@isc.tuc.gr, aggelos@telecom.tuc.gr

***Abstract*—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. Belief Propagation/message passing (inference) algorithms have been proposed for constraint satisfaction problems (CSP), since they are inherently amenable to distributed implementation. This work compares two message passing algorithms for time and frequency allocation, satisfying signal-to-interference-and-noise-ratio, half-duplex-radio operation and routing constraints. 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 (i.e., constraint-satisfying) 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. Future work will focus on next generation 5/6G wireless networks.**

***Index Terms*—Resource Allocation, Constraint Satisfaction, Message Passing, Wireless Networks.**

I. Introduction

In a real-world scenario, time and frequency allocation in wireless sensor networks (WSN) is a very challenging task, due to the limited availability of resources [1]. Existing work in WSN resource allocation includes centralized [1]–[3], as well as distributed protocols [4]–[8]. In distributed protocols, the communication is limited only to neighboring nodes, whereas in centralized protocols, each node requires knowledge of the global network topology. Especially for large-scale WSN, the use of centralized protocols is discouraged, due to the high computational cost and delay times.

In this paper, we jointly solve the problem of WSN time and frequency allocation by utilizing Belief Propagation (BP); the latter is an algorithm inherently amenable to distributed implementation, due to its message passing nature. The problem formulation follows our previous work [9], [10], inspired by [11], where the joint frequency and time allocation problem is encoded into a factor graph. However, in this work, we test a different inference procedure, based on perturbed BP [12] and compare it with the existing, state-of-the-art distributed algorithm in [10].

The rest of this paper is organized as follows: Section II offers the problem formulation and the basic assumptions. Section III describes the factor graph model that encodes the resource allocation problem, while sections IV-V provide a description of the algorithms utilized for the underlying constraint satisfaction problem (CSP). Section VI offers a comparison on algorithm performance and finally, Section VII concludes this work.

II. System Model

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 $k \in \mathcal{K} \triangleq \{1, 2, \ldots, K\}$. 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:

$$SINR_{i \to j} = \frac{P_i |h_{i,j}|^2}{\sigma_j^2 + P_{i'} |h_{i',j}|^2} < \theta, \tag{1}$$

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.

III. 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 specified frequency and time slot. The set of all factor nodes is given by [10]:

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

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. We seek to satisfy all constraints, which means that we want to find a random variable assignment for which the output of all factor functions is one. For this purpose, the Belief Propagation (BP) algorithm is utilized. In BP, the neighboring nodes of a graph iteratively send messages to each other to compute the marginals of the joint distribution, encoded by the graph. 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. A detailed scheme for a factor graph of a simple WSN can be found in [10]. The BP messages exchanged in such graph are given by:

$$m_{J\to v}^{(n)}(x_v) = C_{J\to v}^{(n)}(\bar{\mathbf{q}})\times \sum_{\mathbf{x}_y : y\in\partial(J)\setminus v} \left\{ \mathbf{g}\left(x_v; \mathbf{x}_y\right) \prod_{y\in\partial(J)\setminus v} m_{y\to J}^{(n-1)}(x_y) \right\}, \quad (3)$$

$$m_{v\to J}^{(n)}(x_v) = C_{v\to J}^{(n)}(\bar{\mathbf{q}})\times P_v(x_v) \prod_{I\in\partial(v)\setminus J} m_{I\to v}^{(n)}(x_v), \quad (4)$$

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 $C_{v\to J}^{(n)}(\mathbf{q})$ and $C_{J\to v}^{(n)}(\mathbf{q})$ 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. Their value depends on a subset of the priors $\mathbf{q} \triangleq \begin{bmatrix} q_1 & q_2 & \cdots & q_{(N-1)MK} \end{bmatrix} \in [0,1]^{(N-1)MK}$ as well as the current iteration. The BP messages are initialized in the following manner:

$$m_{v\to J}^{(n=0)}(x_v = 0) = 1 - m_{v\to J}^{(n=0)}(x_v = 1) = q_v, \quad (5)$$

$$m_{J\to v}^{(n=0)}(x_v = 0) = m_{J\to v}^{(n=0)}(x_v = 1) = 0, \quad (6)$$

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:

$$\hat{\mu}_v^{(n)}(x_v) = P_v(x_v) \prod_{I\in\partial(x_v)} m_{I\to v}^{(n)}(x_v). \quad (7)$$

Finally, given Eq. (7), we compute the random variable decisions by performing hard decision on the estimated marginals:

$$\mathbf{x}^* = \hat{\boldsymbol{\mu}}(\mathbf{x} = 0) < \hat{\boldsymbol{\mu}}(\mathbf{x} = 1). \quad (8)$$

A random variable assignment $\mathbf{x}^*$ is satisfiable, if given this assignment, all constraints (factors) are satisfied. 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. (7) are not always accurate. Therefore, we seek for methods to guarantee BP convergence and correctness.

IV. Restarting method

A method called restarting [10] 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 1, terminates after a given number of iterations, denoted by T_{max}.

In this context, we should mention that this method has strong connections to already existing, control theory techniques for system stabilization to a (specific) fixed point [13], [14]. The re-initialization of the priors can be seen as an intentionally added perturbation to the discrete-time system that describes the BP messages [15], as in [14]. 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 [13].

V. Perturbed Belief Propagation

The second method, called Perturbed Belief Propagation [12], smoothly interpolates two well-known inference procedures; it starts as BP and ends as a Gibbs sampler:

$$m_{i\to I} \leftarrow \gamma \cdot m_{i\to I} + (1-\gamma)\cdot \delta(x_i - \hat{x}_i), \quad (9)$$

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

$$\hat{x}_i \sim \hat{\mu}_i(x_i). \quad (10)$$

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), may provide a valid solution. [1]

[1] Other variations of perturbed BP were also examined (e.g., incorporating local re-initialization/restarting of messages or message contradiction check as in [12]), but failed to provide better algorithm performance.

Algorithm 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. (3);
    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. (4);
    calculate $\hat{\mu}_i(x_i)$ using Eq. (7);
  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 $\mathbf{x}^*$;

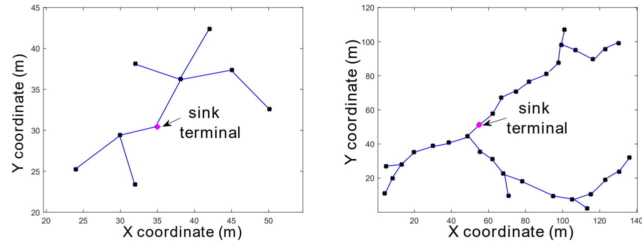
Algorithm 2: Perturbed BP method

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. (3);
  end
  for *each variable* x_i **do**
    calculate $\mathbf{m}_{i\to I}^{(t)}, \forall I \in \partial(i)$ using Eq. (4);
    calculate $\hat{\mu}_i^{(t)}(x_i)$ using Eq. (7);
    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: $\mathbf{x}^*$;

VI. Simulation Results

For the simulation results, we will use two different networks, whose routing connectivity is depicted in Fig. 1. 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,



(a) 9-terminal WSN (b) 35-terminal WSN

Fig. 1: Connectivities for the wireless sensor networks used in simulations

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 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. 2-3 and Fig. 4 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. 4, the restarting method offers lower outage probability for a given T_{max} value, when compared to perturbed BP. A comparison of the results in Figs. 2-3 also shows that for larger θ (i.e. larger number of interference constraints), perturbed BP provides better performance results. This is a very interesting result that requires further investigation.

VII. Conclusion

In this work, we jointly solve the time and frequency allocation problem for wireless networks by utilizing perturbed BP. We then show that the algorithm in prior art outperforms perturbed BP by providing faster convergence to a satisfiable solution for the imposed Constraint Satisfaction Problem.

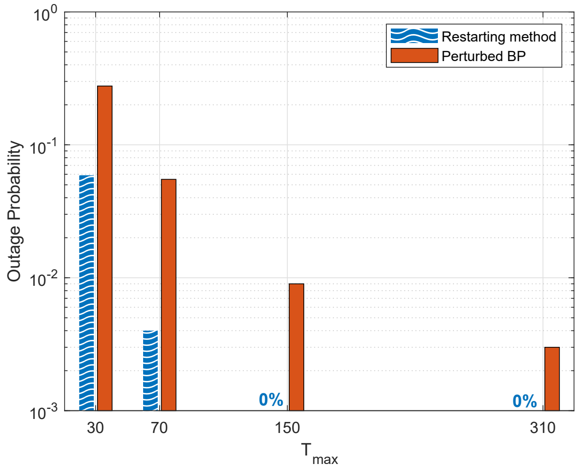


Fig. 2: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 9-terminal network ($N = 9$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 3$ dB.

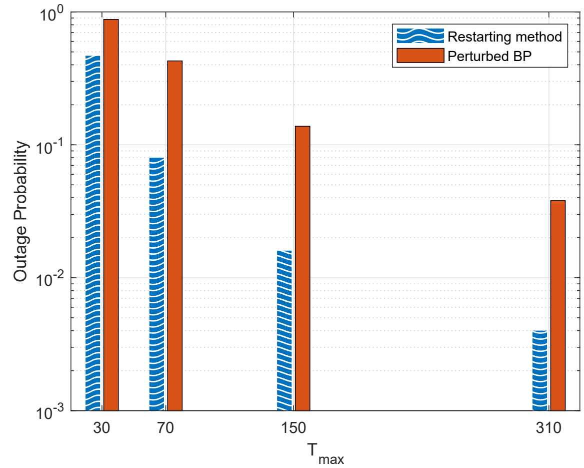


Fig. 4: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 35-terminal network ($N = 35$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 8$ dB.

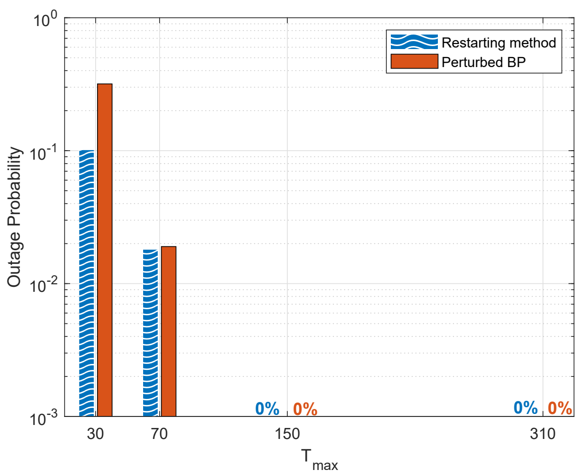


Fig. 3: Outage probability of convergence to a valid solution VS maximum number of BP iterations, T_{max}, for the 9-terminal network ($N = 9$, $M = 4$, $K = 2$). The SINR threshold is set to $\theta = 9$ dB.

Future work will focus on the extension of both algorithms for heterogeneous and 5/6G wireless networks.

ACKNOWLEDGEMENT

The research work was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "First Call for H.F.R.I. Research Projects to support Faculty members and Researchers and the Procurement of High-cost research equipment" (Project #: 2846).

REFERENCES

[1] Y. Wu, J. A. Stankovic, T. He, 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, April 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*, Macau, China, 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*, 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 *Proc. IEEE 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 Transactions on Automatic 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, April 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. 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.

[10] 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

[11] J.-C. Chen, Y.-C. Wang, and J.-T. Chen, "A novel broadcast scheduling strategy using factor graphs and the sum-product algorithm," *IEEE Trans. Wireless Commun.*, vol. 5, no. 6, pp. 1241–1249, Jun. 2006.

[12] S. Ravanbakhsh and R. Greiner, "Perturbed message passing for constraint satisfaction problems," *Journal of Machine Learning Research*, vol. 16, no. 1, pp. 1249–1274, Jan. 2015.

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

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

[15] C. Knoll, "Understanding the behavior of belief propagation," Ph.D. dissertation, Graz University of Technology, 2019.