Probabilistic Asynchronous Inference in Wireless Networks with Spectral Clustering

Emmanouil Kariotakis
School of ECE & Telecommunication Systems Institute
Technical University of Crete
Chania, Greece
ekariotakis@tuc.gr

Aggelos Bletsas
School of ECE & Telecommunication Systems Institute
Technical University of Crete
Chania, Greece
aggelos@telecom.tuc.gr

***Abstract*—This work builds upon the concept of asynchronous, in-network processing with probabilistic graphical model (PGM)-based inference algorithms. Specifically, this work studies the problem of *efficient* mapping of the PGM to wireless sensor network (WSN) terminals, exchanging messages in a (probabilistic) asynchronous manner. The PGM considered stems from Gaussian belief propagation (GBP), which is versatile and powerful, able to describe many known algorithms. It is shown that node clustering methods based on spectral clustering outperform autonomous clustering and k-means, in most cases, in terms of convergence rate, for given probabilities of WSN terminals being active. Interestingly, it is also found that fast convergence rate can be achieved, which is *independent* of the mapping method applied. Finally, optimization of WSN communication energy consumption is also addressed with spectral clustering.**

***Index Terms*—GBP, PGM, WSN, Spectral Clustering, Asynchronous Inference**

I. Introduction

Probabilistic graphical models (PGM) have the power to express a variety of powerful algorithms including (but not limited to) Expectation-Maximization, Viterbi and Kalman filtering [1], [2]. Of particular importance is the Gaussian Belief Propagation (GBP) algorithm, which can solve problems relevant to clustering, minimum mean square error (MMSE) estimation and systems of linear equations [3], [4].

Inference algorithms, such as sum-product (also known as belief propagation) or max-product offer inference by exchanging messages on carefully crafted graphs that encode (in)dependencies of the variables of interest, modeled as random variables [5], [6]. A special case of such graph is the (bipartite) factor graph of Fig. 1, with edges only between factors (rectangles in Fig. 1) and variables (circles in Fig. 1) and the inherent probability distribution encoded by a product of factors [7], [8].

Such inherent message passing in PGMs is amenable to distributed processing; work in [9] showed that inference algorithms - facilitated by PGMs - can be distributed in wireless sensor networks (WSN), ambiently powered and specific *in-network* inference examples were realized in embedded WSNs. The key concept was *asynchronous* message passing, allowing for WSN terminals to re-use existing messages, especially when lack of sufficient energy (or any other reason) prohibited communication in parts of the wireless network.

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) and "Support for the Internationalization of Higher Education, School of Electrical and Computer Engineering, Technical University of Crete" (MIS5150766), under the call for proposals "Support of the Internationalization of Higher Education - Technical University of Crete" (EDULLL 153). The project is co-financed by Greece and the European Union (European Social Fund- ESF) through the Operational Programme "Human Resources Development, Education and Lifelong Learning 2014-2020."

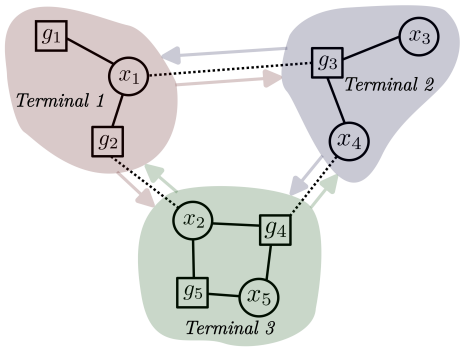


Fig. 1: PGM mapping to WSN terminals example.

This work builds upon the concept of asynchronous, in-network inference and studies the problem of *efficient* mapping of the PGM to WSN termninals; the goal is to accelerate convergence, while minimizing energy communication consumption among WSN terminals, operating in a (probabilistic) asynchronous manner.

II. GBP & Affine Model Preliminaries

A. GBP under High-Order Factorization

Gaussian Belief Propagation (GBP) [3] is a special case of the sum-product algorithm where the distributions are Gaussian. The sum-product algorithm is a message passing algorithm that operates in undirected graphical models or factor graphs [1]. Given a probability distribution function, sum-product computes -either exactly or approximately- various marginal distribution functions using message passing between the graph's nodes and following a few simple computational rules [7], [8].

Let $\mathbf{x} \in \mathbb{R}^n$ be a random Gaussian vector. Its probability density function (pdf) in information form follows:

$$p(\mathbf{x}) \propto \exp\left\{-\frac{1}{2}\mathbf{x}^\top \mathbf{J}\mathbf{x} + \mathbf{h}^\top \mathbf{x}\right\}, \tag{1}$$

where $\mathbf{J}$ and $\mathbf{h}$ are the precision matrix and potential vector, respectively. A general factorization of the aforementioned matrix and vector is considered in [10], and as shown, GBP under high-order factorization utilizes factor-to-variable messages $m^{(l)}_{g_j \to x_i}(x_i)$ at each iteration, which are proved to be valid Gaussian pdfs, with $m^{(l)}_{g_j \to x_i}(x_i) = \mathcal{N}\left(x_i; \mu^{(l)}_{g_j \to x_i}, 1/\nu^{(l)}_{g_j \to x_i}\right)$. In addition, the algorithm outcome, i.e., the beliefs of the random variables, are valid Gaussian probability density functions.[1] With proper initialization, precision messages $\nu^{(l)}_{g_j \to x_i}$ can be guaranteed to converge; $\boldsymbol{\mu}^{(l)}$ stacks all mean messages $\mu^{(l)}_{g_j \to x_i}$ at iteration (l), and GBP can be reduced to the *synchronous* affine fixed point problem:

$$\boldsymbol{\mu}^{(l)} = \mathbf{A}\boldsymbol{\mu}^{(l-1)} + \mathbf{c}, \tag{2}$$

where $\mathbf{A}$ and $\mathbf{c}$ depend on the converged precision messages.[2]

B. Solving Systems of Linear Equations

Assume that the WSN is designed to solve linear equations:

$$\mathbf{M}\,\mathbf{x} = \mathbf{s}, \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{M} \in \mathbb{R}^{m \times n}$ $(m \geq n)$ a full rank matrix and $\mathbf{s} \in \mathbb{R}^m$ (m is the number of factors and n the number of variables), for which the least squares solution is [11]

$$\mathbf{x} = \left(\mathbf{M}^\top \mathbf{M}\right)^{-1} \mathbf{M}^\top \mathbf{s}. \tag{4}$$

GBP can be utilized for the following Gaussian distribution:

$$p(\mathbf{x}) \propto \exp\left\{-\frac{1}{2}\mathbf{x}^\top \mathbf{M}^\top \mathbf{M}\mathbf{x} + \mathbf{s}^\top \mathbf{M}\mathbf{x}\right\}, \tag{5}$$

where inference of the mean value will yield the desired problem solution [3]. Using matrix $\mathbf{M}$, the corresponding PGM can be created by connecting variable and factor nodes for each non-zero element of $\mathbf{M}$.

C. Message Passing Probabilities of GBP in WSNs

Messages from factor node g_j to variable node x_i depend on the messages that were sent to the neighboring variable nodes of the factor node g_j, except x_i. Thus, $\mathcal{C}_{i,j}$ is defined as the set that contains the WSN terminals that g_j, x_i and neighboring variable nodes of g_j belong to. Also, p_c is the probability of a WSN terminal c being active. Thus, the probability of message $m_{g_j \to x_i}$ being updated properly follows:

$$p^{\text{update}}_{m_{g_j \to x_i}} = \prod_{c \in \mathcal{C}_{i,j}} p_c. \tag{6}$$

Vector $\tilde{\mathbf{p}}$ contains elements $p^{\text{update}}_{m_{g_j \to x_i}}$ for all $(i,j) \in \mathcal{E}$ (the set of PGM edges), arranged first on i and then on j, i.e., contains one probability element for each edge of the PGM graph.

[1] Analytical expressions for all quantities can be found in [10].

[2] Similar model is possible for parallel update of precisions and means.

D. (A)synchronous Affine Fixed Point (AFP) Problem

Let real vectors $\mathbf{x}^{(0)},\ \mathbf{b} \in \mathbb{R}^n$ and real square matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$; the following *synchronous* recursion is defined:

$$\mathbf{x}^{(l)} = \mathbf{A}\mathbf{x}^{(l-1)} + \mathbf{b}, \quad l = 1, 2, \ldots \tag{7}$$

and $\mathbf{b} \neq \mathbf{0}$. The solution of this problem, i.e., the *fixed point* is denoted by $\mathbf{x}^* \triangleq \lim_{l \to \infty} \mathbf{x}^{(l-1)} = \lim_{l \to \infty} \mathbf{x}^{(l)}$.

Using notation from seminal work in [9], [10], [12], at each iteration (l) the following diagonal matrix $\boldsymbol{\Psi}^{(l)}$ is defined, with x_k denoting the k-th element of vector $\mathbf{x}$, $\forall k \in \{1, 2, \ldots, n\}$:

$$\boldsymbol{\Psi}^{(l)}(k,k) = \begin{cases} 1, & \text{if } x_k \text{ is updated at iteration } (l), \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

where $\boldsymbol{\Psi}^{(l)}$ are independent across l. In that way, Eq. (7) becomes *probabilistic asynchronous*, as follows:

$$\begin{aligned} \mathbf{x}^{(l)} &= \boldsymbol{\Psi}^{(l)}\left(\mathbf{A}\mathbf{x}^{(l-1)} + \mathbf{b}\right) + \left(\mathbf{I} - \boldsymbol{\Psi}^{(l)}\right)\mathbf{x}^{(l-1)} \\ &= \left(\boldsymbol{\Psi}^{(l)}\mathbf{A} + \mathbf{I} - \boldsymbol{\Psi}^{(l)}\right)\mathbf{x}^{(l-1)} + \boldsymbol{\Psi}^{(l)}\mathbf{b}. \end{aligned} \tag{9}$$

where $\mathbb{E}[\boldsymbol{\Psi}^{(l)}] \triangleq \mathbf{P} = \text{diag}\{\tilde{\mathbf{p}}\}$, with $\tilde{\mathbf{p}}$ as in Sec. II-C. Notice that for $\mathbf{P} = \mathbf{I}$, Eq. (9) simplifies to the synchronous case of Eq. (7).

Synchronous AFP Problem: A necessary and sufficient condition for Eq. (7) to converge is $\rho(\mathbf{A}) < 1$ [11], [13].

Asynchronous AFP Problem: A necessary and sufficient condition for convergence in the mean sense, i.e., convergence of $\lim_{l \to \infty} \mathbb{E}\left[\mathbf{x}^{(l)}\right]$ is $\rho(\bar{\mathbf{A}}) < 1$, where $\bar{\mathbf{A}} = \mathbf{P}(\mathbf{A} - \mathbf{I}) + \mathbf{I}$ [10], [14], [15]. In addition, given that $\rho(\bar{\mathbf{A}}) < 1$, a necessary and sufficient condition for convergence in the mean square sense, i.e., covariance matrix of the error vector approaches the all-zero matrix for $l \to \infty$ is $\rho(\mathbf{S}) < 1$, where $\mathbf{S} = \bar{\mathbf{A}} \otimes \bar{\mathbf{A}} + ((\mathbf{I} - \mathbf{P})) \otimes \mathbf{P})\,\mathbf{J}\,((\mathbf{A} - \mathbf{I}) \otimes (\mathbf{A} - \mathbf{I}))$ [14], [15], with $\mathbf{J} = \text{diag}(\text{vec}(\mathbf{I})) \in \mathbb{R}^{n^2 \times n^2}$, where $\otimes$ denotes the Kronecker product.

III. Mapping PGM-to-WSN Methods

A. Clustering Algorithms

Mapping PGMs to WSN terminals requires first PGM clustering. Thus, three different clustering algorithms are utilized, namely k-means [16], spectral clustering [17]–[19] and autonomous clustering [20].

Given an undirected and weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with vertex set $\mathcal{V} = \{v_1, \ldots, v_n\}$, we can define cut [17] of $\mathcal{A} \subseteq \mathcal{V}$ to be:

$$cut(\mathcal{A}) = \sum_{i \in \mathcal{A}, j \in \bar{\mathcal{A}}} w_{ij}, \tag{10}$$

where $\bar{\mathcal{A}}$ is the complement of $\mathcal{A}$ in $\mathcal{V}$ and w_{ij} represents the weight between vertices v_i and v_j. Now, we can define $RatioCut$ [17] to be:

$$RatioCut(\mathcal{A}_1, \ldots, \mathcal{A}_k) = \sum_{i=1}^{k} \frac{cut(\mathcal{A}_i)}{|\mathcal{A}_i|}, \tag{11}$$

where $\mathcal{A}_1, \ldots, \mathcal{A}_k$ is a partition of $\mathcal{V}$ and $|\mathcal{A}_i|$ is the number of elements in $\mathcal{A}_i$. In the case that all weights are equal to 1, $RatioCut$ is the sum of normalized number of edges connecting each cluster with all the others. Its advantage over simpler cut, is that its minimization, not only gives a clustering with a small number of edges connecting each cluster with the others, but also a balanced number of elements per cluster.

Spectral clustering is an algorithm that clusters a graph into k clusters, by minimizing $RatioCut$. The solution of this minimization problem is given by choosing a matrix $\mathbf{H}$ that contains the first k eigenvectors of graph Laplacian $\mathbf{L}$ as columns. This is a real valued matrix and thus, it needs conversion to a discrete partition. The standard way is to use the k-means algorithm on the rows of $\mathbf{H}$. More detailed descriptions of the algorithm can be found in [17]–[19].

Clustering can be performed *autonomously*, by the WSN network itself, for the special case of $k = 2$ clusters/WSN terminals [20]. Such approach utilizes the AFP problem of Eq. (7), by setting $\mathbf{A} = h(\mathbf{L})$ and $\mathbf{b} = \mathbf{0}$ (i.e., relaxing the non-zero constraint):

$$\mathbf{x}^{(l)} = h(\mathbf{L})\mathbf{x}^{(l-1)}, \tag{12}$$

where $\mathbf{L}$ is the graph Laplacian matrix and $h(\mathbf{L})$ is an r^{th}-order polynomial of $\mathbf{L}$, i.e., $h(\mathbf{L}) = \sum_{n=0}^{r} h_n \mathbf{L}^n$. It is proved in [20] that for $r = 2$ and using only *algebraic connectivity*, i.e., second smallest eigenvalue λ_2 of $\mathbf{L}$, and largest degree $d_{\max}$ of the graph, a polynomial sufficient to satisfy $h(\lambda_2) = 1$, $|h(\lambda_i)| < 1 \ \forall i \neq 2$, can be created. Then, the probabilistic asynchronous updates of Eq. (12) can converge to (Fiedler) vector $\mathbf{v_2}$, i.e., the eigenvector of $\mathbf{L}$ corresponding to eigenvalue λ_2. Hence, clustering of nodes within $k = 2$ clusters can be derived from $\mathbf{y} = \text{sign}(\mathbf{v_2})$.

B. Node Clustering Approach

In the *node clustering* approach, the PGM nodes are clustered first and then its edges. Node clustering is performed utilizing *k-means*, *spectral clustering* and *autonomous clustering*, with different types of inputs. Input of k-means is the distance matrix $\mathbf{D}$ of the graph, with $D_{i,j}$ being the length of the *shortest path* between nodes i and j. Using that input and squared Euclidean distance metric, the nodes of the PGM are being clustered. Spectral clustering and autonomous clustering take as input the adjacency matrix of the factor graph (PGM), treating factor and variable nodes in the same way.

Then, the edges that connect two nodes of the same cluster are also clustered to that cluster and the edges that connect nodes of different clusters stay unclustered. Those unclustered edges connect different clusters and convey the required messages among distinct WSN terminals.

Assessment of clustering results requires first the definition of the comparison metric. A suitable comparison metric will become more clear in Sec. IV, but for now, "better" clustering is considered the one that offers less connecting edges between different clusters; such quality metric targets at minimization of the communication load among different WSN terminals. Figs. 2 (for $k = 2$ WSN terminals) and 3 (for $k = 3$ or $k = 4$ WSN terminals) show that spectral clustering, autonomous clustering and k-means perform better than random clustering. Moreover, spectral clustering performs slightly better than the others.

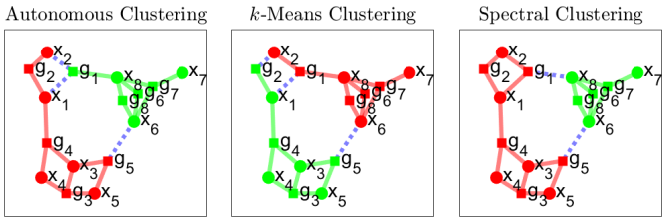


Fig. 2: Clustering results, for $k = 2$ WSN terminals. Different colors represent different clusters and dotted lines represent connecting (among different clusters) edges.

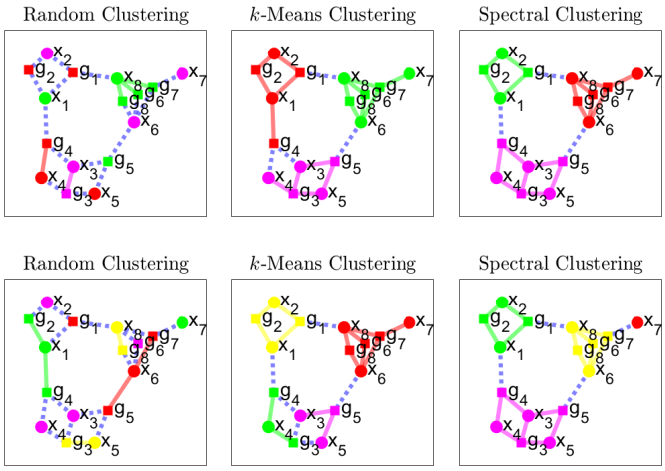


Fig. 3: Clustering results, for $k = 3$ (up) and $k = 4$ (down) WSN terminals. Different colors represent different clusters and dotted lines represent connecting (among different clusters) edges.

IV. WSN Energy Optimization

The following optimization problem is defined:

$$\begin{array}{ll} \min & \text{WSN Energy Consumption} \\ \text{s.t.} & \rho(\bar{\mathbf{A}}) < 1 \quad \text{and} \quad \rho(\mathbf{S}) < 1. \end{array} \tag{13}$$

It is assumed that each WSN terminal consists of a Silab's Thunderboard Sense 2 embedded radio & 32-bit computing module, for which it is true that computation cost $\ll$ wireless communication cost [21]. Thus, an equivalent problem is that of minimization of *communication* energy consumption, which is proportional to the number of edges that connect the WSN terminals, which in turn is optimized by minimizing the $RatioCut$, as described in Sec. III-A. Hence, the problem is equivalent as follows:

$$\begin{array}{ll} \min & RatioCut \\ \text{s.t.} & \rho(\bar{\mathbf{A}}) < 1 \quad \text{and} \quad \rho(\mathbf{S}) < 1 \end{array}. \tag{14}$$

The minimization of $RatioCut$ is a problem that can be solved using spectral clustering (Sec. III-A). Hence, the constraints are removed and the relaxed problem is solved. Then it is checked if the constraints are fulfilled. It is noted that clustering affects matrix $\mathbf{P}$ and therefore, matrices $\bar{\mathbf{A}}$ and $\mathbf{S}$.

V. Numerical Results

This section presents experimental results for GBP solving a linear system of equations $\mathbf{Mx} = \mathbf{s}$, in an asynchronous manner. In all experiments $\mathbf{s} = \mathbf{1}$ and

$$\mathbf{M} = \begin{bmatrix} -1.76 & -0.16 & 0 & 0 & 0 & 0 & 0 & 1.51 \\ 0.73 & -0.54 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1.13 & 0.66 & 1.31 & 0 & 0 & 0 \\ 0.86 & 0 & 1.79 & -0.54 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1.37 & 0 & -0.99 & 0.01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & -0.44 \\ 0 & 0 & 0 & 0 & 0 & -1.10 & -1.90 & -0.79 \\ 0 & 0 & 0 & 0 & 0 & 0.01 & 0 & -0.84 \end{bmatrix}.$$

In particular, estimation of the expected value $\mathbb{E}[\|\mathbf{e}^{(l)}\|_2]$ per iteration is plotted, using 20 independent experiments. At iteration (l) error $\mathbf{e}$ is defined as

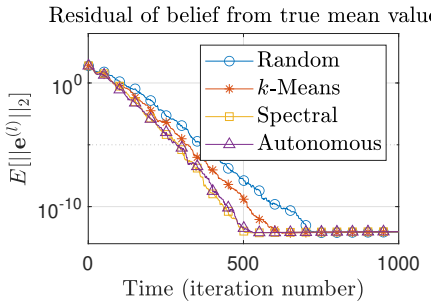
$$\mathbf{e}^{(l)} \triangleq \boldsymbol{\epsilon}^{(l)} - \hat{\mathbf{x}}, \tag{15}$$

where $\hat{\mathbf{x}} \triangleq (\mathbf{M}^\top\mathbf{M})^{-1}\mathbf{M}^\top\mathbf{s}$ is the least squares solution of the system and $\boldsymbol{\epsilon}^{(l)}$ is the vector of all belief means (Sec. II).

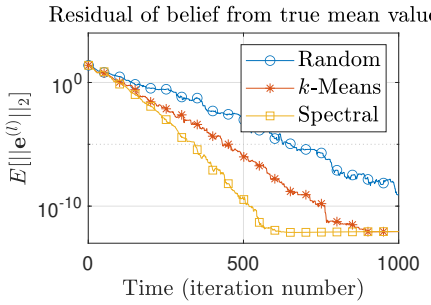
Fig. 4 presents results of GBP convergence in a WSN where all terminals have probability of being active equal to $p_c = 0.5$, as a function of number of clusters (WSN terminals) k. It is shown that the choice of the clustering algorithm *heavily* affects the convergence rate of the GBP algorithm. For instance, notice that when k-means, spectral clustering or autonomous clustering is applied instead of random clustering then faster convergence is achieved. In most cases spectral clustering outperforms all the other methods.

Fig. 5 offers spectral radius results as a function of WSN probability (to be in active mode) p_c and number of clusters k. Selecting a value of $p_c = 0.8$, which is close to the value that minimizes $\rho(\bar{\mathbf{A}})$ offers the convergence results of Fig. 6. Interestingly, faster convergence than before is achieved that is *independent* of the mapping method applied, i.e., all clustering methods offer roughly the same convergence rate.

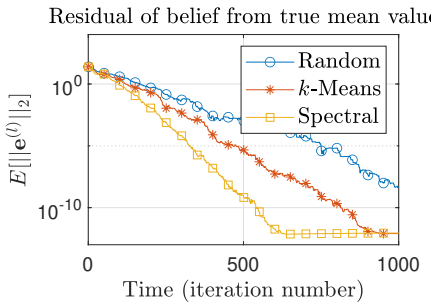
Fig. 7 offers the WSN energy consumption (detailed in Sec. IV) during distributed solution of $\mathbf{Mx} = \mathbf{s}$, with $p_c = 0.8$ for all WSN terminals; specifically, 20 experiments were run, for WSNs terminals that have probabilities of being active nearly equal to the value that minimizes spectral radius (Fig. 5), until a threshold of 10^{-5} is reached. It is shown that in most cases, as the number of clusters increases, the energy consumption also increases. Spectral clustering energy consumption is almost equal between $k = 3$ and $k = 4$. Moreover, spectral clustering performs better than the others, and autonomous clustering has a performance close to both k-means and spectral clustering. Careful mapping of PGMs to WSN terminals can reduce total WSN energy consumption by a factor (approximately) between $5 - 7$, compared to random mapping. Similar results have been observed for other matrices $\mathbf{M}$ [22].



(a) $k = 2$

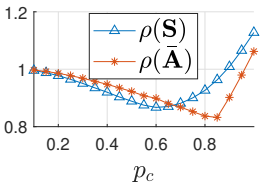


(b) $k = 3$

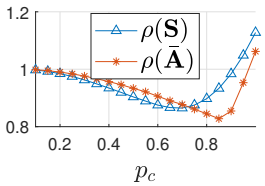


(c) $k = 4$

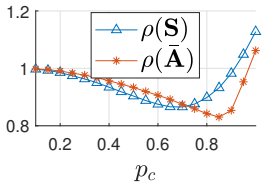
Fig. 4: Convergence of asynchronous GBP for WSN terminals with $p_c = 0.5$ for all terminals.



(a) $k = 2$

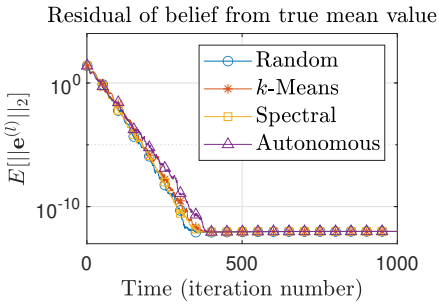


(b) $k = 3$

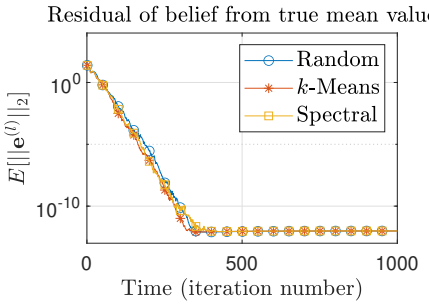


(c) $k = 4$

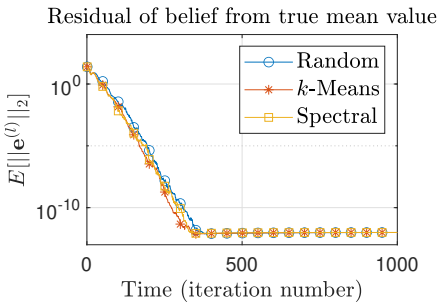
Fig. 5: Experimental computation of $\rho(\bar{\mathbf{A}})$ and $\rho(\mathbf{S})$ vs p_c. All WSN terminals with equal probability p_c being active.



(a) $k = 2$



(b) $k = 3$



(c) $k = 4$

Fig. 6: Convergence of asynchronous GBP for WSN terminals with $p_c = 0.8$ for all terminals.

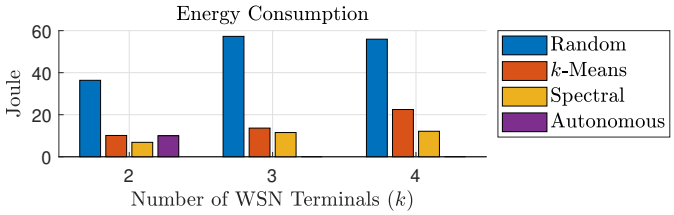


Fig. 7: WSN energy consumption for distributed GBP.

VI. Conclusion

The goal of this work was to *efficiently* map a PGM to a WSN, so that the later can solve inference problems with probabilistic asynchronous message passing, cast through GBP. This work showed that spectral clustering can minimize energy consumption of WSN, while convergence rate can be independent of the mapping method with a carefully selected set of probabilities p_c. The application that was presented was that of solving linear systems of equations. However, GBP can solve a variety of problems [3], including distributed minimum mean squared error (MMSE) estimator and Kalman filtering. This work is perhaps a concrete step towards (probabilistic) asynchronous inference over practical wireless networks.

References

[1] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning*, The MIT Press, 2009.

[2] "6.438 algorithms for inference," Fall 2014, Massachusetts Institute of Technology: MIT OpenCouseWare, https://ocw.mit.edu/. License: Creative Commons BY-NC-SA.

[3] D. Bickson, *Gaussian Belief Propagation: Theory and Application*, Ph.D. thesis, Hebrew University of Jerusalem, Oct. 2008, Advisors: Prof. Danny Dolev and Prof. Dahlia Malkhi.

[4] J. Du, S. Ma, Y.-C. Wu, S. Kar, and J. M. F. Moura, "Convergence analysis of distributed inference with vector-valued gaussian belief propagation.," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6302–6339, Jan. 2017.

[5] J. S. Yedidia, "Message-passing algorithms for inference and optimization," *Journal of Statistical Physics*, vol. 145, no. 4, pp. 860–890, Nov. 2011.

[6] R. G. Gallager, *Low-Density Parity-Check Codes*, Ph.D. thesis, Massachusetts Institute of Technology, 1963.

[7] F. R. Kschischang, B. J. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inf. Theor.*, vol. 47, pp. 498–519, Feb. 2001.

[8] B. J. Frey, F. R. Kschischang, H.-A. Loeliger, and Niclas Wiberg, "Factor graphs and algorithms," in *Proceedings of the Annual Allerton Conference on Communication Control and Computing*. Citeseer, 1997, vol. 35, pp. 666–680.

[9] V. Papageorgiou, A. Nichoritis, P. Vasilakopoulos, G. Vougioukas, and A. Bletsas, "Towards ambiently powered inference on wireless sensor networks: Asynchrony is the key!," in *17th ACM Int. Conf. on Distrib. Comput. in Sensor Systems (DCOSS)*, Cyprus, July 2021, pp. 458–465.

[10] B. Li and Y.-C. Wu, "Convergence analysis of gaussian belief propagation under high-order factorization and asynchronous scheduling," *IEEE Trans. Signal Processing*, vol. 67, no. 11, pp. 2884–2897, June 2019.

[11] C. D. Meyer, *Matrix analysis and applied linear algebra*, vol. 71, Siam, 2000.

[12] V. Papageorgiou, A. Nichoritis, P. Vasilakopoulos, G. Vougioukas, and A. Bletsas, "Towards ambiently powered internet-of-things-that-think with asynchronous principles," Tech. Rep., School of Electrical and Computer Engineering, Technical University of Crete, 2022.

[13] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice-Hall, Inc., USA, 1989.

[14] O. Teke and P. P Vaidyanathan, "Random node-asynchronous graph computations: Novel opportunities for discrete-time state-space recursions," *IEEE Signal Processing Mag.*, vol. 37, no. 6, pp. 64–73, Nov. 2020.

[15] O. Teke and P. P Vaidyanathan, "Randomized asynchronous recursions with a sinusoidal input," in *Proc. Asilomar Conf. on Signals, Systems and Computers*, Nov. 2019, pp. 1491–1495.

[16] J. MacQueen, "Classification and analysis of multivariate observations," in *5th Berkeley Symp. Math. Statist. Probability*, 1967, pp. 281–297.

[17] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, Aug. 2007.

[18] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[19] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems*, vol. 14, 2001.

[20] O. Teke and P. P Vaidyanathan, "Random node-asynchronous updates on graphs," *IEEE Trans. Signal Processing*, vol. 67, no. 11, pp. 2794–2809, June 2019.

[21] Silicon Laboratories Inc., *EFR32xG12 Wireless Gecko Reference Manual*, 400 West Cesar Chavez, Austin, TX 78701, USA, 2022 [Online].

[22] E. Kariotakis, "Clustering of inference algorithms in communication networks," Diploma thesis, School of Electrical and Computer Engineering, Technical University of Crete, Sept. 2022, Supervisor: Prof. A. Bletsas.