Sorted Order-K Voronoi Diagrams for Model-Independent Source Localization in Wireless Sensor Networks

Dimitris Ampeliotis, *Student Member, IEEE*, and Kostas Berberidis, *Senior Member, IEEE*

Abstract—Localization of an isotropic source using energy measurements from randomly deployed sensors is considered. In particular, an optimization problem that does not require knowledge of the underlaying energy decay model is proposed, and a condition under which the optimal solution can be computed is given. This condition employs a new geometric construct introduced here, called the sorted order-K Voronoi diagram. We give centralized and distributed algorithms for source localization in this setting. Finally, analytical results and simulations are used to verify the performance of the developed algorithms.

Index Terms—Distributed algorithms, optimization methods, source localization, Voronoi diagrams, wireless sensor networks.

I. INTRODUCTION

WIRELESS sensor networks have recently received great attention because they hold the potential to change many aspects of our economy and life. Among many applications, ranging from environmental monitoring to manufacturing, source localization and tracking has been widely viewed as a canonical problem of wireless sensor networks. Furthermore, it constitutes an easily perceived application that can be used as a vehicle to study more involved information processing and organization problems [1]. On the other hand, the design, implementation and operation of a sensor network requires the synergy of many disciplines, including signal processing, networking and distributed algorithms. Moreover, sensor networks must operate using minimum resources: typical sensor nodes are battery powered and have limited processing ability. These constraints impose new challenges in algorithm development, and imply that power efficient, distributed and cooperative techniques should be employed.

Most of the source localization methods that have appeared in the literature can be classified into two broad categories. The algorithms of the first category utilize time difference of arrival (TDOA) measurements, whereas the algorithms of the second category use direction of arrival (DOA) measurements. DOA estimates are particularly useful for locating sources emitting

narrowband signals [2], while TDOA measurements offer the increased capability of localizing sources emitting broadband signals [3]. However, the methods of both categories impose two major requirements that render them inappropriate to be used in wireless sensor networks, i.e., (a) the analog signals at the outputs of the spatially distributed sensors should be sampled in a synchronized fashion, and (b) the sampling rate should be high enough so as to capture the features of interest. These requirements, in turn, imply that accurate distributed synchronization methods should be implemented so as to keep the remote sensors synchronized and that high frequency electronics as well as increased bandwidth are needed to communicate the acquired measurements.

Recently [4], a new approach to source localization was proposed, that utilizes received signal strength (RSS) measurements. In particular, the spatially distributed sensors measure the power of the source signal that arrives at their location. Then, assuming that the source is isotropic, in the sense that it radiates uniformly in all directions, and the emitted signal decays radially according to an energy-decay model, each sensor is able to extract some information about its distance to the source of interest. Finally, the required location of the source is derived by proper fusion of the information extracted at a number of *active* sensor nodes. Note that, a sensor node is characterized as active if its measurement is greater than a predetermined threshold. In [4], in order to avoid the ambiguities that arise due to the unknown transmit power of the source, it was proposed to compute ratios of measurements taken at pairs of active sensors. In [5], maximum likelihood multiple-source localization based on RSS measurements was considered. In [6], the problem of source localization was formulated as a coverage problem and estimates of the necessary sensor density which can guarantee a localization error bound, were derived. In [7], a distributed "incremental subgradient" algorithm was proposed to yield iteratively the source location estimate. More recently, a distributed localization algorithm enjoying good convergence properties was proposed in [8], based on the method of projections onto convex sets (POCS). In [9], a nonlinear cost function for localization was proposed and it was proved that its gradient descent minimization is globally converging. In [10], as well as in our work in [11], least squares based localization algorithms were considered. However, all the aforementioned approaches require knowledge of the energy decay model and/or the transmit power of the source of interest.

In [12], the case of unavailable information about the energy decay model and the transmit power of the source (i.e.,

model-independent case) was considered. The location of the source was derived by properly averaging the locations of active sensor nodes. Another model-independent localization method, that can also be viewed as a special case of the aforementioned estimator, is to detect the sensor node with the strongest energy measurement and set the location estimate equal to its location [4], assuming that this node is the closest one to the source, the so-called closest point of approach (CPA). In our previous work [13], a POCS based algorithm for model-independent localization was derived. In particular, a method for estimating the distances between the sensor nodes and the source of interest was developed, based on the assumption of uniform deployment of the sensors over the field being monitored. It is important to make clear that the method proposed here does not require such an assumption.

In this paper, following a novel approach, we generalize the notion of the CPA estimator which uses only one node, into an estimator that uses a number of nodes. Thus, while in the CPA estimator the source may lie close to the CPA node (in fact, in the Voronoi cell of the CPA node), in our approach the source is restricted to lie in a convex set that we call the *sorted order-K Voronoi cell* of the nodes that participate in the estimation. As this convex set is smaller than the respective one given by the CPA estimator, better performance is obtained. In the following, in Section II a formulation of the localization problem is given. In Section III, we introduce the sorted order-K Voronoi diagram and discuss some of its properties. In Section IV, an optimization problem for model-independent localization is given, and a condition for computing the optimal solution is provided. Also, centralized and distributed algorithms for localization are derived. In Section V, a performance analysis of the proposed method is given. In Section VI, simulation results are presented, and the work is concluded in Section VII.

II. PROBLEM FORMULATION

Let us consider that N sensor nodes have been deployed, at arbitrary locations, over a territory of interest. Let us also consider that a source is present in the same area, and that it emits a signal (acoustic, seismic, electromagnetic, etc.) whose attenuated energy can be measured by the nodes using suitable sensors. Denote the 2×1 location vector of the nth sensor node as $\mathbf{r}_n$ and the unknown location vector of the source as $\mathbf{r}$. We assume that each sensor n obtains an energy measurement y_n according to the model

$$y_n = \alpha \, g(\|\mathbf{r}_n - \mathbf{r}\|) + w_n \quad n = 1, 2, \ldots, N \qquad (1)$$

where $\alpha > 0$ denotes the energy of the emitted signal as measured at some nominal distance from $\mathbf{r}$, function $g : \mathbb{R}^+ \to \mathbb{R}^+$ is monotone decreasing, $\|\cdot\|$ denotes the Euclidean distance and w_n denote zero mean white noise terms. The set of nodes that detect the presence of the source is defined by the nodes with measurement above a suitably selected threshold T as

$$\mathcal{A} = \{l_1, l_2, \ldots, l_L\} = \{n : y_n \geq T\}. \qquad (2)$$

A node whose index is in $\mathcal{A}$ is called *active node*. The localization problem in such a context is a parameter estimation problem

which can be stated as: Given the RSS measurements y_l of all active nodes $l \in \mathcal{A}$, and the location vectors $\mathbf{r}_n$, of all nodes, provide an estimate $\hat{\mathbf{r}}$ of the location vector of the source, taking into account any *known information* about the energy decay model. In this paper, the only such information about the energy decay model that we assume is summarized in that the measurements y_n follow (1), where $g(\cdot)$ is an unknown monotone decreasing function.

Recently, it has been made clear that the solution to the localization problem can be computed very efficiently provided that the distances between the sensor nodes and the source of interest have been previously estimated. In particular, [8] and [9] provide globally converging algorithms for obtaining an estimate of the location of the source in such a case.

However, it should be stressed that [8] and [9] presume that the required distance quantities, i.e., $\|\mathbf{r}_n - \mathbf{r}\|$, are somehow available. Commonly, the estimation of these distances is based on proper assumptions with respect to the form of function $g(\cdot)$. In many applications, function $g(\cdot)$ takes the usual form

$$g(x) = \frac{1}{x^\beta} \qquad (3)$$

where β denotes the energy decay exponent, with typical values in the range $\beta \in [2, 4]$. From (1) and (3), it can be seen that if the parameters α and β are known, then each node can estimate its distance from the source using

$$\|\mathbf{r}_n - \mathbf{r}\| \approx \left(\frac{\alpha}{y_n}\right)^{1/\beta}. \qquad (4)$$

However, in practice, the aforementioned parameters may not be known. Furthermore, it is also possible that function $g(\cdot)$ does not take the form in (3). In this paper, we propose algorithms that deal with such cases and make no assumptions regarding the form of the actual energy decay function.

At this point it is important to make clear that the model of (1), although quite general, does not take into account the cases of sources that radiate nonuniformly (i.e., anisotropic sources) or propagation environments with an energy decay exponent that is different along different directions. Also, it is unable to model strong reverberations of the signal emitted from the source due to the presence of obstacles in the environment. Finally, many nonline-of-sight channels cannot be modeled accurately. In such cases, localization methods that are based on location fingerprinting [14] can be used, however, these methods require a so-called offline initial phase during which the characteristics of the source/environment must be accurately measured.

III. A GENERALIZATION OF VORONOI DIAGRAMS

In order to better explain the derivation of the proposed localization method, we begin our study by introducing a generalization of the Voronoi diagram that we call the *sorted order-K Voronoi diagram* . This diagram is a geometric construct which turns out to be particularly useful for the problem at hand.

A. Voronoi Diagrams and Generalizations

Voronoi diagrams constitute an important mathematical tool with applications in a wide variety of fields inside and outside computer science [15]. Given a set of points
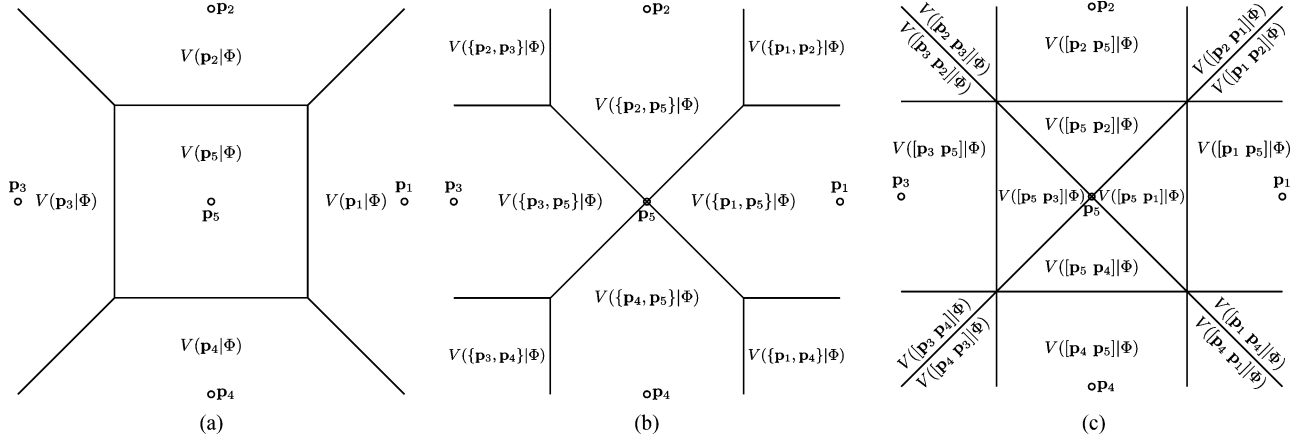
Fig. 1. (a) Voronoi diagram for some example points Φ. (b) Order-2 Voronoi diagram for Φ. (c) Sorted order-2 Voronoi diagram for Φ.

$\Phi = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}$ in $\mathbb{R}^2$, which are also termed as particles, the Voronoi cell of a point $\mathbf{p}_i \in \Phi$ is a convex set that consists of all points in $\mathbb{R}^2$ for which $\mathbf{p}_i$ is the closest particle. Setting $\Phi_i = \Phi \backslash \{\mathbf{p}_i\}$, the Voronoi cell of $\mathbf{p}_i$ is defined by

$$V(\mathbf{p}_i|\Phi) = \{\mathbf{p} \in \mathbb{R}^2 : \|\mathbf{p}_i - \mathbf{p}\| \le \|\mathbf{p}_j - \mathbf{p}\|, \forall \mathbf{p}_j \in \Phi_i\}. \quad (5)$$

The aggregate of these cells is the Voronoi (or Dirichlet) tessellation of the plane $\mathbb{R}^2$. The set of all Voronoi cells defines the Voronoi diagram of Φ, and we denote it as $V(\Phi)$. Also, point $\mathbf{p}_i$ is called the nucleus of the cell $V(\mathbf{p}_i|\Phi)$.

An existing generalization of the Voronoi cell, so-called *order-K Voronoi cell* , can be defined if we consider a set of nuclei from Φ, rather than a single one. The properties and construction of the order-K Voronoi cells have been well studied [16]–[18]. Each such cell, is the locus of points on the plane with the same K-nearest neighbors from Φ. In particular, if we define the locus of points closer to point $\mathbf{p}_i$ than point $\mathbf{p}_j$ as the half-plane

$$h(\mathbf{p}_i, \mathbf{p}_j) = \{\mathbf{p} \in \mathbb{R}^2 : \|\mathbf{p}_i - \mathbf{p}\| \le \|\mathbf{p}_j - \mathbf{p}\|\} \quad (6)$$

then, given a set of points $H \subseteq \Phi$ with $|H| = K$, the order-K Voronoi cell of H is defined as

$$V(H|\Phi) = \bigcap_{\mathbf{p}_i \in H, \mathbf{p}_j \in \Phi \backslash H} h(\mathbf{p}_i, \mathbf{p}_j). \quad (7)$$

Similarly to the Voronoi diagram defined in the previous, the set of all nonempty order-K Voronoi cells, is the order-K Voronoi diagram of the points in Φ.

In this paper, we introduce another generalization of the Voronoi cell, by defining the *Sorted order-K Voronoi cell* that is of interest to us.

Definition 1: Let $\Phi = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N\}$ be a set of points in $\mathbb{R}^2$, and consider a vector of points $\mathbf{v} = [\mathbf{p}_{k_1} \mathbf{p}_{k_2} \cdots \mathbf{p}_{k_K}]$, where $\mathbf{p}_{k_i} \in \Phi$, and $k_i \ne k_j, \forall i \ne j$. The *sorted order-K Voronoi cell* of vector $\mathbf{v}$, denoted as $V(\mathbf{v}|\Phi)$, consists of all points in $\mathbb{R}^2$ for which $\mathbf{p}_{k_1}$ is the closest particle, and $\mathbf{p}_{k_2}$ is the second closest particle, and so on, up to particle $\mathbf{p}_{k_K}$. Formally, if we let $\Psi_{k_i} = \Phi \backslash \{\mathbf{p}_{k_1}, \mathbf{p}_{k_2}, \ldots, \mathbf{p}_{k_i}\}$ (i.e., Ψ_{k_i} is the

set containing all particles from Φ minus those that appear before $\mathbf{p}_{k_{i+1}}$ in $\mathbf{v}$), we define

$$V(\mathbf{v}|\Phi) = \{\mathbf{p} \in \mathbb{R}^2 : \|\mathbf{p}_{k_1} - \mathbf{p}\| \le \|\mathbf{p}_j - \mathbf{p}\|, \forall \mathbf{p}_j \in \Psi_{k_1}$$
$$\|\mathbf{p}_{k_2} - \mathbf{p}\| \le \|\mathbf{p}_j - \mathbf{p}\|, \forall \mathbf{p}_j \in \Psi_{k_2}$$
$$\vdots$$
$$\|\mathbf{p}_{k_K} - \mathbf{p}\| \le \|\mathbf{p}_j - \mathbf{p}\|, \forall \mathbf{p}_j \in \Psi_{k_K}\}. \quad (8)$$

Having defined the Sorted order-K Voronoi cell as above, it is quite clear to see that they can also be expressed in terms of an intersection of K simple Voronoi cells that are computed with respect to the sets Ψ_{k_i} as

$$V(\mathbf{v}|\Phi) = V(\mathbf{p}_{k_1}|\Psi_{k_1}) \cap V(\mathbf{p}_{k_2}|\Psi_{k_2}) \cap \cdots \cap V(\mathbf{p}_{k_K}|\Psi_{k_K}). \quad (9)$$

As we will see in the following, the above property can be exploited in deriving an algorithm for computing the sorted order-K cells, using algorithms that compute simple Voronoi cells.

Since $\mathbf{v}$ can be any of the $\binom{N}{K} K!$ possible sorted vectors with elements from Φ and length K, $V(\mathbf{v}|\Phi)$ may, of course, not exist. Thus, it is convenient to make the following definition.

Definition 2: Let $\mathbf{v}$ be a vector of points from Φ, defined as previously. If $V(\mathbf{v}|\Phi) \ne \emptyset$, then we will say that $\mathbf{v}$ is a *feasible sorting of points*. Also, if $V(\mathbf{v}|\Phi) = \emptyset$, we will say that $\mathbf{v}$ is an *unfeasible sorting of points*.

Similarly to the Voronoi diagram and the order-K Voronoi diagram, we may define the *sorted order-K Voronoi diagram* as the set of all nonempty sorted order-K Voronoi cells $V(\mathbf{v}|\Phi)$, for all possible vectors $\mathbf{v}$ that have K elements from Φ. We denote this diagram as

$$V_K(\Phi) = \{V(\mathbf{v}|\Phi) : \mathbf{v} = [\mathbf{p}_{k_1} \mathbf{p}_{k_2} \cdots \mathbf{p}_{k_K}], \mathbf{p}_{k_i} \in \Phi, k_i \ne k_j\}. \quad (10)$$

It is obvious that $V_1(\Phi)$ is identical to the simple Voronoi diagram $V(\Phi)$. In Fig. 1 we demonstrate the simple Voronoi diagram, the order-2 Voronoi diagram, and the sorted order-2 Voronoi diagram for five example points on $\mathbb{R}^2$.

It is important to make clear that the sorted order-K Voronoi cells $V(\mathbf{v}|\Phi)$ of a vector $\mathbf{v}$, are different from the order-K Voronoi cells $V(H|\Phi)$. In fact, the difference between the geometric construct explored here and the order-K Voronoi cells lies in the sorting of the points used in their definition: While an order-K Voronoi cell $V(H|\Phi)$ is defined in terms of a set H, where elements in H are not ordered, $V(\mathbf{v}|\Phi)$ is defined in terms of a vector $\mathbf{v}$ whose elements are ordered. We proceed in the following by giving some interesting properties for $V(\mathbf{v}|\Phi)$.

B. Properties of Sorted Order-K Voronoi Cells

As pointed out in the previous subsection, some vectors $\mathbf{v}$ may be unfeasible sortings of points. This fact reflects the geometric constraints related to the particular placement of the points of Φ over the plane $\mathbb{R}^2$. The following two Lemmas provide us a clue on how many feasible vectors exist.

Lemma 1: For any set of points H ($|H| = K$), for which the order-K Voronoi cell $V(H|\Phi)$ is nonempty, there exists at least one feasible sorting of points $\mathbf{v}$ with $V(\mathbf{v}|\Phi) \subseteq V(H|\Phi)$.

Proof: Since $V(H|\Phi)$ is nonempty, there exists at least one point $\mathbf{p}_0 \in V(H|\Phi)$. Consider now the Euclidean distances between point $\mathbf{p}_0$ and the kth element of H as $\rho_k = \|\mathbf{p}_0 - \mathbf{h}_k\|$, where $\mathbf{h}_k$ denotes the kth element of set H. If we sort these distances in ascending order as $\rho_{k_1} \leq \rho_{k_2} \leq \cdots \leq \rho_{k_K}$ and consider the vector of points $\mathbf{v}_0 = [\mathbf{h}_{k_1} \mathbf{h}_{k_2} \cdots \mathbf{h}_{k_K}]$, then by definition, $\mathbf{p}_0 \in V(\mathbf{v}_0|\Phi)$. Thus $V(\mathbf{v}_0|\Phi)$ is nonempty and $\mathbf{v}_0$ is a feasible sorting of points. Furthermore, all points in $V(\mathbf{v}_0|\Phi)$ have the same K nearest neighbors, i.e., the elements of H. Thus, $V(\mathbf{v}_0|\Phi)$ is a subset of $V(H|\Phi)$. ∎

Lemma 2: Given a set Φ of N points, where there are no more than three cocircular points, the total number N_K of feasible sortings $\mathbf{v}$ with K elements, satisfies $N_K \geq (2K-1)N - (K^2-1) - \sum_{k=1}^{K} S_{k-1}$, for $N \geq K \geq 1$, where S_k is the number of unbounded cells of the order-k Voronoi diagram of Φ and $S_0 = 0$.

Proof: The proof is a direct consequence of [17, Lemma 1 and Theorem 2] ([17, Theorem 2] is also generalized in [18, Theorem 4]). In particular, from Lemma 1 we have that for each nonempty $V(H|\Phi)$ there exists at least one nonempty $V(\mathbf{v}|\Phi)$ which is always a subset of $V(H|\Phi)$. Thus, the number of cells $V(\mathbf{v}|\Phi)$ is at least equal to the number of cells $V(H|\Phi)$. The number of cells $V(H|\Phi)$ is given by [17, Theorem 2] as $(2K-1)N - (K^2-1) - \sum_{k=1}^{K} S_{k-1}$. ∎

Having given the above properties of sorted order-K cells, we proceed in the following section by pointing out their connection to the model-independent localization problem.

IV. MODEL-INDEPENDENT LOCALIZATION

A. Voronoi Cells and Model-Independent Localization

The connection between sorted order-K Voronoi cells and the source localization problem, can be stated by means of the following Lemma.

Lemma 3: Consider a sorted vector of sensor locations $\mathbf{v} = [\mathbf{r}_{k_1} \mathbf{r}_{k_2} \cdots \mathbf{r}_{k_K}]$, and a source located at $\mathbf{r}$. Then, the event in which sensor k_1 is the closest sensor to the source, and sensor k_2 is the second closest sensor to the source, and so on, up to sensor k_K being the K-th closest sensor to the source, can only be possible if $V(\mathbf{v}|\Phi)$ is not the empty set, that is, if $\mathbf{v}$ is a feasible sorting of points, where now $\Phi = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$. Also, $\mathbf{r}$ must be a point in $V(\mathbf{v}|\Phi)$. Moreover, in this case any monotone decreasing function $g(\cdot)$ satisfies

$$g(\|\mathbf{r} - \mathbf{r}_{k_1}\|) \geq g(\|\mathbf{r} - \mathbf{r}_{k_2}\|) \geq \cdots \geq g(\|\mathbf{r} - \mathbf{r}_{k_K}\|) \quad (11)$$

and $g(\|\mathbf{r} - \mathbf{r}_{k_K}\|) \geq g(\|\mathbf{r} - \mathbf{r}_m\|)$ for all m where $\mathbf{r}_m$ is not an element of $\mathbf{v}$.

Proof: Using the definition of $V(\mathbf{v}|\Phi)$ we have that the points $\mathbf{r} \in V(\mathbf{v}|\Phi)$ are the only ones that satisfy

$$\|\mathbf{r} - \mathbf{r}_{k_1}\| \leq \|\mathbf{r} - \mathbf{r}_{k_2}\| \leq \cdots \leq \|\mathbf{r} - \mathbf{r}_{k_K}\| \quad (12)$$

and $\|\mathbf{r} - \mathbf{r}_{k_K}\| \leq \|\mathbf{r} - \mathbf{r}_m\|$ for all m where $\mathbf{r}_m$ is not an element of $\mathbf{v}$. Also, considering any function $g(\cdot)$ that is monotone decreasing and inserting it into the previous inequalities, we have that points $\mathbf{r} \in V(\mathbf{v}|\Phi)$ are the only ones satisfying the given property.

The above Lemma is, in fact, a geometric constraint that links the source location $\mathbf{r}$ to the sorting of the distances between the source and the sensor nodes of the network. Thus, if we knew the correct sorting of the distances then we would be able to restrict the space in which the source may lie, and this space would be the cell $V(\mathbf{v}_o|\Phi)$ that corresponds to the correct sorting $\mathbf{v}_o$. ∎

B. A Cost Function for Localization

It can be shown [5] that when the energy decay model is known and under the assumption that w_n are zero mean Gaussian random variables with the same variance, then Maximum Likelihood estimation is accomplished by minimizing the cost function

$$C(\mathbf{r}, \alpha) = \sum_{l=l_1}^{l_L} (y_l - \alpha g(\|\mathbf{r} - \mathbf{r}_l\|))^2 \quad (13)$$

which is defined in terms of the measurements and locations of active nodes in $\mathcal{A} = \{l_1, l_2, \ldots, l_L\}$. In the case explored here, where the energy decay function $g(\cdot)$ is not known, the above cost function cannot be used. Instead, a cost function that tries to identify a proper energy decay function as well as the location of the source should be used. Consider the cost function

$$J(\mathbf{r}, h(\cdot)) = \sum_{l=l_1}^{l_L} (y_l - h(\|\mathbf{r} - \mathbf{r}_l\|))^2 \quad (14)$$

where $\mathbf{r}$ can be any point in the plane and $h(\cdot)$ can be any monotone decreasing function. Consider now that we are able to find a point $\mathbf{r}_0$ and a monotone decreasing function $h_{\mathbf{r}_0}(\cdot)$ for which (14) attains its minimum value, $J(\mathbf{r}_0, h_{\mathbf{r}_0}(\cdot))$. Then, $\mathbf{r}_0$ is an optimal solution for the model-independent localization problem, in the sense that there exists an energy decay model $h_{\mathbf{r}_0}(\cdot)$ that best describes the acquired measurements. Also, if there exists another pair $\{\mathbf{r}_1, h_{\mathbf{r}_1}(\cdot)\}$ with $J(\mathbf{r}_1, h_{\mathbf{r}_1}(\cdot))$ equal to $J(\mathbf{r}_0, h_{\mathbf{r}_0}(\cdot))$, then point $\mathbf{r}_1$ is also an optimal solution. Thus, in general, (14) may be minimized for $\mathbf{r} \in \mathcal{L}$, where $\mathcal{L}$ is the locus of points for which there exists a monotone decreasing function

that minimizes (14). Although minimization of (14) seems intractable, there exists a condition under which the optimal locus of points $\mathcal{L}$ can be computed. This result is given in terms of the following Theorem.

Theorem 1: Consider the sorting of the location vectors of active nodes $\mathbf{v}_A = [\mathbf{r}_{k_1} \mathbf{r}_{k_2} \cdots \mathbf{r}_{k_L}]$ determined by the sorting of the measurements $y_{k_1} > y_{k_2} > \cdots > y_{k_L}$, assuming that there are no equal measurements. Then, if $V(\mathbf{v}_A | \Phi)$ has positive area, where $\Phi = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$, the optimal solutions of (14) are internal points of $V(\mathbf{v}_A | \Phi)$ and *vice versa*.

Proof: Consider any point $\mathbf{r}$ that is an internal point of $V(\mathbf{v}_A | \Phi)$. By the definition of $V(\mathbf{v}_A | \Phi)$, we have that

$$\|\mathbf{r} - \mathbf{r}_{k_1}\| < \|\mathbf{r} - \mathbf{r}_{k_2}\| < \cdots < \|\mathbf{r} - \mathbf{r}_{k_L}\| \quad (15)$$

and $\|\mathbf{r} - \mathbf{r}_{k_L}\| < \|\mathbf{r} - \mathbf{r}_m\|$ for all m where $\mathbf{r}_m$ is not an element of $\mathbf{v}_A$. Thus, among the monotone decreasing functions characterized by Lemma 3, there exists a monotone decreasing function $h_{\mathbf{r}}(\cdot)$ satisfying $h_{\mathbf{r}}(\|\mathbf{r} - \mathbf{r}_{k_1}\|) = y_{k_1}$, $h_{\mathbf{r}}(\|\mathbf{r} - \mathbf{r}_{k_2}\|) = y_{k_2}$, ... and $h_{\mathbf{r}}(\|\mathbf{r} - \mathbf{r}_{k_L}\|) = y_{k_L}$. Substituting into (14), we obtain $J(\mathbf{r}, h_{\mathbf{r}}(\cdot)) = 0$. Consider now any sorting of the location vectors of active nodes different from $\mathbf{v}_A$. This sorting will correspond to a sorting of the measurements y that is not monotone decreasing. If the sequence of measurements is not monotone decreasing, then we cannot find a monotone decreasing function giving a squared error equal to zero, thus (14) attains its minimum value for all points inside $V(\mathbf{v}_A | \Phi)$. ∎

In fact, we can generalize the aforementioned Theorem so as to include also the case of equal measurements. If there is a set of pairs of nodes with equal measurements, then the locus of points $\mathcal{L}$ for which (14) can be equal to zero, would be non empty only if the intersection of all the lines that are defined as the perpendicular bisectors of these pairs of nodes exists. However, we consider this event as quite rare, and thus we are interested only in the case covered by the previous Theorem.

C. Centralized Localization Algorithms

In this section, we propose two centralized algorithms that try to exploit Theorem 1 in order to localize the source of interest. The algorithms are centralized, in the sense that all the measurements acquired by the active sensors must be transmitted to a so-called "Fusion Center" for further processing. Furthermore, the Fusion Center does not have the processing and memory limitations that the nodes of the sensor network usually have, and thus it can perform a more elaborate processing. A distributed localization algorithm is considered in the next subsection.

Using the proof of Theorem 1 we can easily see that a model-independent localization algorithm, should try to construct a sorted order-K Voronoi cell P, so that for all points $\mathbf{r} \in P$ the cost function in (14) would be equal to zero. If this is possible, then K would be equal to L, the number of active nodes. Of course, in the case of noisy measurements, this might not be possible since the sorting of the measurements of active nodes in descending order might not reflect the correct sorting of the distances from the source, in ascending order. Thus, in such a case, the localization algorithm should at least try to maximize the order K of the cell P. Consider now the algorithm A_1 in

TABLE I
CENTRALIZED LOCALIZATION ALGORITHM A_1

```
Input: A, Φ, and y_l for all l ∈ A
Output: A convex polygon P in which the source may be located

1. Sort the measurements of active nodes as y_{k_1} > y_{k_2} > ··· > y_{k_L}
2. P = V(r_{k_1}|Φ\{r_{k_1}})
3. FOR l=2 TO L
       P_l = V(r_{k_l}|Φ\{r_{k_1}, r_{k_2}, ..., r_{k_l}})
       IF P ∩ P_l = ∅
         Stop and output P
       ELSE
         Update P = P ∩ P_l
       END IF
   END FOR
```

Table I. The algorithm starts by sorting the measurements of active nodes in descending order, and initializes the polygon P to the Voronoi cell of the node with the maximum energy measurement. In the sequel, using the property of (9) regarding the incremental construction of a sorted order-K cell, the algorithm tries to increase the order of the cell P, by visiting the other nodes of the network, in order of decreasing measurement. If such an order increase is possible, the algorithm computes the new cell. In the opposite case, the algorithm stops.

Consider that A_1 stops at a sorted order-K Voronoi cell $V([\mathbf{r}_{k_1} \mathbf{r}_{k_2} \cdots \mathbf{r}_{k_K}] | \Phi)$ with $K < L$. It is quite easy to see that, in this case, the algorithm has detected the optimal solution for a "truncated" form of the cost function in (14) which is given by

$$J'(\mathbf{r}, h(\cdot)) = \sum_{l=k_1}^{k_K} (y_l - h(\|\mathbf{r} - \mathbf{r}_l\|))^2 \quad (16)$$

and has only K out of the L terms of (14). In other words, algorithm A_1 will find the optimal solution in the case where the set of active nodes was defined in terms of a threshold T satisfying $y_{k_{K+1}} < T \leq y_{k_K}$.

Let us now relax the requirement for the cost function being equal to zero. This means that we may allow a node with a smaller measurement appear before a node with a higher measurement, as long as the resulting sorting is a feasible one. However, it is reasonable to allow this, only in the case where the placement of the node with higher measurement before the other one would lead to an unfeasible sorting. The algorithm that implements this scheme appears in Table II, and we denote it as algorithm A_2. This algorithm executes a number of steps, and at each one it tries to increase the order of the cell P. To this end, it maintains a set O of the labels of the nodes which have not yet been taken into account, as well as a set I that contains the labels of the nodes already used. An update step is executed as long as there exists an active node, not yet taken into account, whose incorporation would lead to a feasible sorting. If many such nodes exist, the one with highest measurement is preferred. It is quite easy to verify that A_2 will always take into account a number of active nodes which is at least equal to that used by A_1.

There are a number of interesting properties that are satisfied by both A_1 and A_2:

1) In the case that one is interested in a single point estimate $\hat{\mathbf{r}}$ for the source location, rather than a convex polygon, we should include a final step to the algorithms A_1 and A_2 during which such a single point would be computed. In

Input: $\mathcal{A}$, Φ, and y_l for all $l \in \mathcal{A}$
Output: A convex polygon P in which the source may be located
1. Sort the measurements of active nodes as $y_{k_1} > y_{k_2} > \cdots > y_{k_L}$
2. $P = V(\mathbf{r}_{k_1}\|\Phi\backslash\{\mathbf{r}_{k_1}\})$
3. $O = \{k_2, k_3, \ldots, k_L\}$
4. $I = \{k_1\}$
4. WHILE $F = \{k \in O : P \cap V(\mathbf{r}_k\|\Phi\backslash\{\mathbf{r}_n : n \in I\}) \neq \emptyset\}$ not empty
$\quad$ Select k with maximum y_k among all $k \in F$
$\quad$ Update $P = P \cap V(\mathbf{r}_k\|\Phi\backslash\{\mathbf{r}_n : n \in I\})$
$\quad$ Update $O = O\backslash\{k\}$
$\quad$ Update $I = I \cup \{k\}$
$\quad$ END WHILE

the case where the output polygon is bounded, a reasonable choice for this task is to compute the so-called Fermat-Weber center of that polygon. The Fermat-Weber center of a planar object P, is a point in the plane, such that the average distance from it to the points in P is minimal. In [19], a linear-time approximation scheme for finding the Fermat-Weber center of a convex object was derived.

2) It can be seen that there is no need to transmit the exact values of the measurements to the Fusion Center for processing. Rather, if the Fusion Center knows only the sorting of the measurements it can execute A_1 or A_2 in the same manner as if the exact measurements were available. This property suggests that energy savings are possible if the nodes first cooperate to determine their sorting, and then transmit to the Fusion Center. For example, the algorithms of [20] can be used to determine the sorting of the measurements of the nodes.

3) Another interesting property is that, in the noise free case, A_1 and A_2 become equivalent and furthermore, both yield the optimal solution that minimizes (14). This is easy to verify, since in the noise-free case, the sorting of the measurements will yield the correct sorting of the distances.

4) Another interesting property of the derived algorithms has to do with noise robustness. In particular, consider that the algorithm (either A_1 or A_2) has managed to identify the correct sorting of the first K_1 nodes, while there are still $K - K_1$ nodes whose sorting is not the correct one. This scenario is a very common one, since nodes near the source usually have better signal to noise ratio, and thus, can be ordered correctly with higher probability than nodes away from the source. In such a case, the algorithm will identify the optimal polygon P during the K_1 update step. It will then start making erroneous decisions. Nevertheless, since all subsequent polygons are subsets of the correct polygon identified at the K_1 step, it follows that the maximum error (Euclidean distance between the source and its estimate) cannot be greater than the diameter of the convex polygon P, even if we select the final estimate anywhere inside the output polygon. Note that the diameter of a convex polygon P is defined as the maximum distance between any pair of points than belong to the set P.

Regarding the complexity of the proposed algorithms, we note that the initial Voronoi tesselation, requires $O(N \log N)$ operations [17]. Computing a higher order cell requires only local processing, since after removal of a particular point $\mathbf{p}$ from

Φ, only the cells having point $\mathbf{p}$ as Voronoi neighbor must be updated. Also, the computation of the intersection of two convex polygons can be computed in linear time [21].

D. Distributed Localization Algorithm

In this section, we derive a simple distributed algorithm for model-independent localization. The algorithm is termed as distributed since it does not require the transmission of the measurements to a Fusion Center for processing. Instead, the nodes of the sensor network exchange local messages and update the location estimate accordingly. The distributed algorithm is based on the property given by (9), which can be written in terms of the active nodes of the network as

$$V(\mathbf{v}\|\Phi) = V(\mathbf{r}_{k_1}\|\Psi_{k_1}) \cap V(\mathbf{r}_{k_2}\|\Psi_{k_2}) \cap \cdots \cap V(\mathbf{r}_{k_L}\|\Psi_{k_L}) \quad (17)$$

where $\Phi = \{\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N\}$, $\Psi_{k_i} = \Phi\backslash\{\mathbf{r}_{k_1}, \mathbf{r}_{k_2}, \ldots, \mathbf{r}_{k_i}\}$ and $\mathbf{v} = [\mathbf{r}_{k_1}\mathbf{r}_{k_2}\cdots\mathbf{r}_{k_L}]$.

Consider now that each active node k_i has computed the respective set $V(\mathbf{r}_{k_i}\|\Psi_{k_i})$. In such a case, the method of projections onto convex sets (POCS, [8]) can be used to compute a point on a boundary of $V(\mathbf{v}\|\Phi)$, if it is not empty. This algorithm would be similar to the algorithm proposed in [8], with the only difference that now the involved convex sets are different. In particular, while the algorithm of [8] uses disks centered at the locations of active nodes with radii equal to the estimated source-sensor distances, the convex sets used for model-independent localization are the sets $V(\mathbf{r}_{k_i}\|\Psi_{k_i})$.

Of course, computation of the exact sorted order-K Voronoi cell $V(\mathbf{v}\|\Phi)$ would require a large number of nodes, since the sets Ψ_{k_i} contain almost all of the node locations in Φ. However, if each active node k_i uses the algorithm of [22] to compute the Voronoi cell $V(\mathbf{r}_{k_i}\|\Psi_{k_i})$, then POCS could be used to compute $V(\mathbf{v}\|\Phi)$ as the intersection of these cells, i.e., exact distributed computation is, in this case, possible.

However, in this work we propose an approximate solution that results in a much simpler algorithm. In particular, an approximation to a Voronoi cell $V(\mathbf{r}\|\Phi)$ can be obtained as $V(\mathbf{r}\|\Phi')$, for $\Phi' \subseteq \Phi$. In fact, if Φ' is large enough so as to contain all the Voronoi neighbors of $\mathbf{r}$, then $V(\mathbf{r}\|\Phi') = V(\mathbf{r}\|\Phi)$[22]. Consider now the sets of node locations

$$\Psi'_{k_i}(\rho) = \{\mathbf{r}_l \in \Phi : 0 < \|\mathbf{r}_{k_i} - \mathbf{r}_l\| \leq \rho, y_{k_i} > y_l\} \quad (18)$$

which are subsets of the respective sets Ψ_k. In particular, the set $\Psi'_{k_i}(\rho)$ contains the locations of the nodes that are close to the active node k_i (i.e., within distance ρ), and have smaller energy measurement. According to the above, an approximation of the cell $V(\mathbf{v}\|\Phi)$ is given by

$$\hat{V}(\mathbf{v}\|\Phi) = V(\mathbf{r}_{k_1}\|\Psi'_{k_1}) \cap V(\mathbf{r}_{k_2}\|\Psi'_{k_2}) \cap \cdots \cap V(\mathbf{r}_{k_L}\|\Psi'_{k_L}) \quad (19)$$

where we have omitted the index ρ from the sets $\Psi'_{k_i}(\rho)$ for simplicity of notation. Of course, computation of the sets $\Psi'_{k_i}(\rho)$ requires the exchange of some messages among the nodes of the sensor network. As an example, each active node might broadcast its measurement in a time division multiple access scheme, and wait for a one-bit response from nearby nodes with smaller measurement. In any case, such communication is required only

TABLE III
DISTRIBUTED LOCALIZATION ALGORITHM A_3 AT NODE k

Parameters: Threshold T, radius ρ

1. Acquire measurement y_k
 2. IF $y_k \geq T$
     Compute $\Psi'_k(\rho)$ using local messages
     REPEAT a number of times
       Listen for a message θ, from a nearby active node
       FOR the elements $\mathbf{r} \in \Psi'_k(\rho)$
         IF $2(\mathbf{r}_k - \mathbf{r})^T \theta - \|\mathbf{r}_k\|^2 + \|\mathbf{r}\|^2 < 0$
           Update θ to $\theta - \frac{2(\mathbf{r}_k-\mathbf{r})^T\theta - \|\mathbf{r}_k\|^2 + \|\mathbf{r}\|^2}{2\|(\mathbf{r}_k-\mathbf{r})\|^2}(\mathbf{r}_k - \mathbf{r})$
         END IF
       END FOR
       Send θ to the next active node
     END REPEAT
     ELSE
       Do not participate into the estimation
 END IF

for the initialization of the localization algorithm and each message must be transmitted at a distance that can be at most ρ.

Thus, if each active node k_i of the network is able to compute the set $V(\mathbf{r}_{k_i}|\Psi'_{k_i})$, the POCS algorithm can be used to compute an approximate solution. Furthermore, we note that active nodes do not need to construct any Voronoi tesselation: The set $V(\mathbf{r}_{k_i}|\Psi'_{k_i})$ can be decomposed further, into an intersection of half-planes, as

$$V(\mathbf{r}_{k_i}|\Psi'_{k_i}) = \bigcap_{k \in \Psi'_{k_i}} h(\mathbf{r}_{k_i}, \mathbf{r}_k). \tag{20}$$

Thus, using the above equation, each node k_i can compute a projection of a point $\boldsymbol{\theta}$ onto $V(\mathbf{r}_{k_i}|\Psi'_{k_i})$, by simply computing projections of $\boldsymbol{\theta}$ onto the half-planes (convex sets) that define $V(\mathbf{r}_{k_i}|\Psi'_{k_i})$, in a POCS fashion.

It is easy to verify that $\boldsymbol{\theta} \in h(\mathbf{r}_i, \mathbf{r}_j)$ is equivalent to

$$2(\mathbf{r}_i - \mathbf{r}_j)^T \boldsymbol{\theta} - \|\mathbf{r}_i\|^2 + \|\mathbf{r}_j\|^2 \geq 0 \tag{21}$$

and that if a point $\boldsymbol{\theta} \notin h(\mathbf{r}_i, \mathbf{r}_j)$, then the projection of $\boldsymbol{\theta}$ onto $h(\mathbf{r}_i, \mathbf{r}_j)$ is the point

$$\boldsymbol{\theta}_\perp = \boldsymbol{\theta} - \frac{2(\mathbf{r}_i - \mathbf{r}_j)^T \boldsymbol{\theta} - \|\mathbf{r}_i\|^2 + \|\mathbf{r}_j\|^2}{2\|(\mathbf{r}_i - \mathbf{r}_j)\|^2}(\mathbf{r}_i - \mathbf{r}_j). \tag{22}$$

Thus, using the above formulas, the POCS method can be used to obtain a point on the boundary of $\hat{V}(\mathbf{v}|\Phi)$, in the case where it is not empty, in a decentralized fashion. The distributed algorithm that summarizes the above, is shown in Table III.

Although the previous algorithms have been developed considering a two dimensional region of space, they can easily be extended into the three dimensional space. In particular, algorithms A_1 and A_2 can be used in three dimensions without any modification, as long as the geometric operations they rely upon (Voronoi tesselation, intersection of convex polyhedra, etc.) are implemented in three dimensions. Also, algorithm A_3 can be modified to compute projections onto half-spaces rather that half-planes.

V. AN ANALYSIS OF THE PROPOSED ALGORITHMS

In order to analyze the performance of the proposed localization method, we will assume that the nodes of the network are uniformly deployed over the plane $\mathbb{R}^2$. Thus, the locations of the nodes define a homogeneous Poisson process in the plane, and the respective random Voronoi diagram is the well known *Poisson-Voronoi tessellation* of the plane $\mathbb{R}^2$ [23], [24]. The intensity λ of the Poisson process is a measure of the density of the sensor network, i.e., the expected number of nodes per unit of area. Similarly to the Poisson-Voronoi tessellation, we may define the *order-K Poisson-Voronoi tessellation* of $\mathbb{R}^2$, as well as the *sorted order-K Poisson-Voronoi tessellation* of $\mathbb{R}^2$, as the respective random tessellations in case there are infinite points in Φ uniformly deployed over the plane, with intensity λ.

Since the derived algorithms A_1 and A_2 compute a convex polygon in which the source may lie, it is reasonable to study the area of such polygons as a measure of the accuracy of this localization methodology. Also, we will not delve into the particular details of A_1 or A_2, rather, we will focus on the general case where an algorithm has detected the correct sorting of at least K nodes of the network, and we will try to characterize the expected performance of the method in terms of the parameters K and λ. Thus, let us define the random variables $\mathcal{X}_K$, $K = 1, 2, \ldots$, where $\mathcal{X}_K$ describes the area of the sorted order-K Voronoi cell in a sorted order-K Poisson-Voronoi tessellation generated by a point process with intensity λ.

Lemma 4: The first two moments of the random variables $\mathcal{X}_K$, satisfy the following:

$$E[\mathcal{X}_1] = \frac{1}{\lambda}, \quad E[\mathcal{X}_1^2] \approx \frac{1.280176}{\lambda^2} \tag{23}$$

and for any finite K

$$0 < E[\mathcal{X}_K] \leq E[\mathcal{X}_1] \tag{24}$$

$$0 < E[\mathcal{X}_K^2] \leq E[\mathcal{X}_1^2] \tag{25}$$

Proof: First, the equations in (23) were proven in [24] considering simple Poisson-Voronoi tesselations, and they apply to our case since for $K = 1$ the sorted order-K Poisson-Voronoi tesselation is equivalent to the simple Poisson-Voronoi tesselation. Concerning now the inequalities in (24) and (24), we note that since the random variables $\mathcal{X}_K$ are nonnegative, and we can always find a strictly positive instance of $\mathcal{X}_K$, we have that $0 < E[\mathcal{X}_K]$ and $0 < E[\mathcal{X}_K^2]$. Consider now a realization of Φ, and define N_1 instances of the random variable $\mathcal{X}_1$ as $X_{1,1}, X_{1,2}, \ldots, X_{1,N_1}$. Then each one of the polygons associated with each one of the N_1 such instances of $\mathcal{X}_1$, will be decomposed into a number of polygons in the sorted order-K tessellation. Let us denote the aggregate of the areas of these polygons as $X_{K,1}, X_{K,2}, \ldots, X_{K,N_K}$. These areas are instances of the random variable $\mathcal{X}_K$. Furthermore, we have that the sums of the two sets of instances will be equal to each other, and that $N_K \geq N_1$. Thus

$$E[\mathcal{X}_K] = \lim_{N_K \to \infty} \left(\frac{1}{N_K} \sum_{n_K=1}^{N_K} X_{K,n_K} \right)$$

$$= \lim_{N_K, N_1 \to \infty} \left(\frac{N_1}{N_K} \frac{1}{N_1} \sum_{n_1=1}^{N_1} X_{1,n_1} \right)$$

$$= \lim_{N_K, N_1 \to \infty} \left(\frac{N_1}{N_K} \right) E[\mathcal{X}_1] \tag{26}$$

since the limit

$$\lim_{N_K, N_1 \longrightarrow \infty} \left(\frac{N_1}{N_K} \right) \leq 1 \tag{27}$$

exists, because $N_K \geq N_1$. Combining now (25) and (26), we have that $E[\mathcal{X}_K] \leq E[\mathcal{X}_1]$. The proof for the second moment is similar. ∎

Using the above Lemma, the performance of the proposed localization method is characterized by the following Theorem.

Theorem 2: Consider a sorted order-K Poisson-Voronoi tessellation with intensity λ, and a uniformly distributed point $\mathbf{r}$ in the plane. The expected area of the sorted order-K Poisson-Voronoi cell of this process that contains the point $\mathbf{r}$ is given by

$$E_K = \frac{E[\mathcal{X}_K^2]}{E[\mathcal{X}_K]}. \tag{28}$$

Proof: Fix one instance of Φ and consider a number N_K of instances of sorted order-K Voronoi cells. Denote the areas of each such instance as $X_{K,1}, X_{K,2}, \ldots, X_{K,N_K}$. The expected area of the cell that contains point $\mathbf{r}$, given that point $\mathbf{r}$ is in one of the N_K cell instances, is given by

$$E_{K,N_K} = \sum_{n_K=1}^{N_K} p_{n_K} X_{K,n_K} \tag{29}$$

where p_{n_K} is the probability that $\mathbf{r} \in X_{K,n_K}$. Since point $\mathbf{r}$ is uniformly deployed over the plane, p_{n_K} will be proportional to the area of cell X_{K,n_K}, thus

$$p_{n_K} = \frac{X_{K,n_K}}{\sum_{n_K=1}^{N_K} X_{K,n_K}}. \tag{30}$$

Substituting into the above equation and considering all possible instances, we have that

$$\begin{aligned}
E_K &= \lim_{N_K \longrightarrow \infty} \sum_{n_K=1}^{N_K} \frac{X_{K,n_K}^2}{\sum_{n_K=1}^{N_K} X_{K,n_K}} \\
&= \frac{\displaystyle\lim_{N_K \longrightarrow \infty} \frac{\sum_{n_K=1}^{N_K} X_{K,n_K}^2}{N_K}}{\displaystyle\lim_{N_K \longrightarrow \infty} \frac{\sum_{n_K=1}^{N_K} X_{K,n_K}}{N_K}} \\
&= \frac{E[\mathcal{X}_K^2]}{E[\mathcal{X}_K]}
\end{aligned} \tag{31}$$

where we have divided numerator and denominator by N_K, and the limits exist from Lemma 4. ∎

Using Lemma 4 and the above Theorem, we have that

$$E_1 \approx \frac{1.280176}{\lambda}. \tag{32}$$

However, computing values of E_K for $K > 1$ can be rather difficult. Thus, we turn our attention to the properties of $E[\mathcal{X}_K]$.

Theorem 3: The expected area of the sorted order-K Poisson-Voronoi cell, generated by a process with intensity λ, is bounded by

$$E[\mathcal{X}_K] \leq \frac{1}{(2K-1)\lambda} \tag{33}$$

for any finite K.

Proof: Consider (24) and (26) from the proof of Lemma 4. Using Lemma 2, we have that

$$\frac{N_1}{N_k} \leq \frac{N_1}{(2K-1)N_1 - (K^2-1) - \sum_{k=1}^{K} S_{k-1}}. \tag{34}$$

Dividing By N_1 the numerator and the denominator at the right hand side of the above inequality, we have that

$$\lim_{N_K, N_1 \longrightarrow \infty} \left(\frac{N_1}{N_k} \right) \leq \frac{1}{2K-1} \tag{35}$$

since $(K^2-1)/N_1$ Will converge to zero, and

$$\lim_{N_k, N_1 \longrightarrow \infty} \frac{\sum_{k=1}^{K} S_{k-1}}{N_1} = 0. \tag{36}$$

the above can be verified by considering each of the terms in the summation as

$$\frac{S_k}{N_1} = \frac{S_k}{N_k} \frac{N_k}{N_1} \tag{37}$$

where, in the limit, $S_k/N_k = \Pr(\text{Unbounded Order} - k \text{ Cell})$. Thus, since any order-$k$ Voronoi cell is a subset of the union of the k Voronoi Cells of the k particles that define the order-k set, and we know that Poisson-voronoi cells are bounded with probability 1, then it follows that

$$\lim_{N_k, N_1 \longrightarrow \infty} \frac{S_k}{N_k} = 0 \tag{38}$$

for any finite k. Furthermore, from Lemma 4, we have that the limit of N_1/N_k cannot be zero, because in that case we would have that $E[\mathcal{X}_k] = 0$, which is not true. Thus, the limit of N_k/N_1 is a finite number, and multiplied by the limit of (37), will yield a value zero. Combining (23), (25), and (34), yields the desired result. ∎

As a practical example for the bound in (32), consider a wireless sensor network with a density of one node per square meter. Then, in a scenario with three correctly sorted node measurements ($K = 3$), we have an expected area smaller than 1/5 square meters in which the source may lie. At this point, it should be mentioned that the bound in (32) is expected to be quite loose for large K. This is due to the fact that properties of the order-K diagram were used, rather than properties of the sorted order-K diagram. Using simulations, we were able to see that

$$E[\mathcal{X}_K] \approx \frac{1}{(K^3 - K^2 + K)\lambda} \tag{39}$$

but we were unable to prove this equation for $K > 2$. For $K = 1$, the well known result $E[\mathcal{X}_1] = 1/\lambda$ is obtained. For $K = 2$,

using (26), and the fact that the typical Voronoi cell has 6 edges [24], we have that

$$\lim_{N_1, N_2 \to \infty} \frac{N_1}{N_2} = \frac{1}{6} \qquad (40)$$

which is consistent with (39).

Of course, in the case where the sensor nodes of the network are not uniformly deployed over the field of interest, some performance degradation may be noticeable. However, since the proposed algorithms A_1 and A_2 compute a region of space in which the source may be located, the area of the polygon can be used as a measure of the accuracy of the estimation. This fact, in some cases, may constitute an advantage over existing localization methods.

VI. SIMULATION RESULTS

A. Verification of the Analytical Results

In order to verify the results derived in the previous Section, we performed the following two experiments: (A) We simulated 50.000 instances of a Poisson point process, having 100 points uniformly distributed in the unit square. Thus, this process has $\lambda = 100$, in the examined area. For each instance of the point process, a point source $\mathbf{r}$ was placed at the center of the unit square, so as to assure that the Voronoi cell in which $\mathbf{r}$ lies will be bounded. For each such instance, we computed the sorted order-K cell in which $\mathbf{r}$ lies, for $K = 1, 2, \ldots, 20$, and kept the area of each one cell. (B) In the second experiment, 50.000 instances of a point process were simulated as in the previous, but this time no source was placed on the plane. Rather, a bounded sorted order-K cell was chosen randomly, and its area was stored.

Fig. 2 shows the results of the previous two experiments. In particular, the result of experiment (A) appears in Fig. 2 as the line labeled "Average area of the cell with the source", as a function of the order K. Using data from experiment (B), we give three curves in Fig. 2: The average area of the randomly selected cell (labeled as "Mean of $\mathcal{X}_K$"), the average of the squared areas of these cells (labeled as "Mean of $\mathcal{X}_K^2$"), and their ratio. Finally, we also plot the bound of (32). Thus, from Fig. 2 we verify the following two results:

1) The bound of (33) is greater than the mean of the area of the typical sorted order-K cell, however this bound is quite loose.
2) From Fig. 2, we can verify the validity of Theorem 2. In particular, we observe that the performance of the localization method, measured by experiment (A), is very close to the term $E_K = E[\mathcal{X}_K^2]/E[\mathcal{X}_K]$, which can be approximated using experiment (B).

B. Performance Comparison With the Maximum Likelihood (ML) Estimator

In order to assess the performance of the proposed algorithms, and compare it to the performance of previously proposed estimators including the ML estimator of [5] (which assumes a known energy decay model), we performed some numerical simulations. In particular, N nodes were uniformly deployed
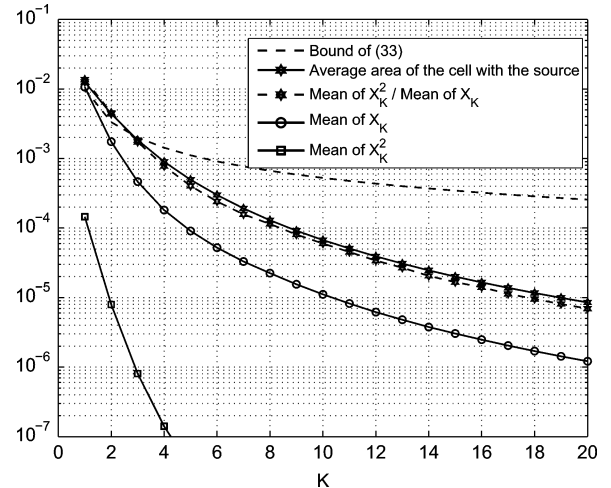

Fig. 2. Verification of the analytical results.

over a $100 \text{ m} \times 100 \text{ m}$ field, where N was increased from 300 to 3100 in 200 increments. A signal source with $\alpha = 100$ was placed at $\mathbf{r} = [50\,50]^T$, i.e., at the center of the deployment field. The RSS measurements at the sensor nodes were corrupted by zero mean additive white Gaussian noise (AWGN) with variance $\sigma^2 = 4/5$. A threshold $T = 5$ was used to detect active nodes, i.e., only sensors whose SNR (y_n/σ^2) is greater than 8 dB take part in the estimation procedure. Fig. 3(a) demonstrates the root mean square (RMS) localization error of various algorithms (computed as the average of the errors of 10.000 different realizations), as a function of the average number of active nodes. The energy decay model used for this experiment is given by $g(x) = 1/x^2$. Also, in order to also include the performance of the Maximum Likelihood estimator of [5], only experiments that resulted in more than 3 nodes being active were taken into account, for all algorithms. Fig. 3(b) demonstrates the average numbers of nodes used by algorithms A_1 and A_2, which are smaller than the number of active nodes (dashed line).

For the two POCS based algorithms (i.e., the POCS with known model and A_3), we performed 10 iterations over the set of active nodes, with a constant relaxation sequence equal to 1, while the final estimate was given as the average of the estimates of the last iteration [8]. Also, the final estimate of algorithms A_1 and A_2 was given as the average of the vertex locations of the output polygon. Replacing this estimate with the so-called Fermat-Weber center of the output polygon, would certainly lead to further performance improvement. Finally, for the ML estimator, we used exhaustive search over a 101×101 square grid, close to the true source location [i.e., ranging from (48,48) to (52,52)].

From Fig. 3, we make the following observations:

1) Concerning our distributed algorithm A_3, we notice that it obtains better localization accuracy than other distributed model-independent algorithms, such as the CPA and the Averaging [12] estimators. A neighborhood radius of 10 meters, used for local negotiations among nodes, is found to yield very good results for the network densities examined, since further increase of this radius does not offer a significant performance improvement.
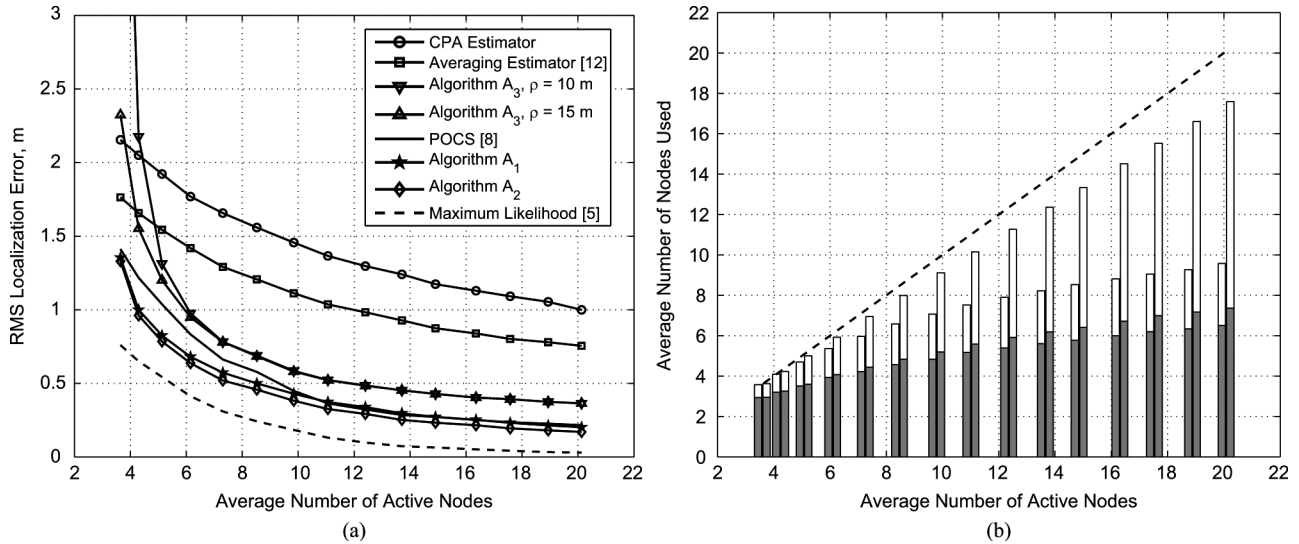
Fig. 3. (a) RMS error as a function of the average number of active nodes, at least 3 nodes active, $g(x) = 1/x^2$. (b) Average number of nodes used by A_1 (left bars) and A_2 (right bars), the gray portions of the bars are the average numbers of correctly sorted nodes.

2) Surprisingly, our model-independent centralized algorithms A_1 and A_2 offer performance that is somewhat better than the performance of the POCS estimator of [8], even though the latter algorithm has perfect knowledge of the power of the source and the energy decay model. Also, A_2 offers little performance improvements relative to A_1.

3) It is quite clear that the distributed algorithm A_3 does not obtain the performance of its centralized counterparts, A_1 and A_2, even when a large value of ρ is used. This is mainly due to the fact that the distributed algorithm does not take into account the geometric feasibility constraints that are exploited by the centralized algorithms.

4) From Fig. 3(b), we can see that the centralized algorithms A_1 and A_2 use only a subset of the active nodes. As expected, A_2 uses more nodes than A_1. Also, A_2 is shown to be able to increase the number of nodes that are sorted correctly [gray portion of the bars in Fig. 3(b)], however, this has only negligible performance gains, as shown in Fig. 3(a).

In order to further explore the relationship between the performance of the proposed localization method and the performance of the ML estimator, we tested the ML estimator in the case where accurate knowledge of the energy decay exponent β is not available. Fig. 4 presents the performance of the ML estimator when the assumed value for β is varied from 2 to 3 in increments of 0.2 while the correct value is equal to 2. From this Figure, we can see that the performance of the ML estimator deteriorates as the assumed β is increased. Furthermore, in many cases, the proposed algorithm A_2 offers better performance.

C. Performance for Various Models

In this section, we demonstrate the performance of the proposed algorithms without the inclusion of the ML estimator of [5], thus we allow all experiments that result in at least one active node. All the other parameters used in this experiment are the same as those mentioned in the previous. Fig. 5 presents the RMS localization error performance in this scenario, where in
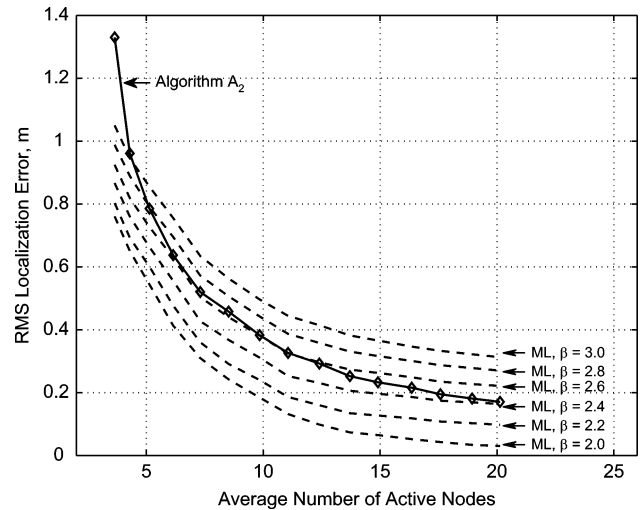

Fig. 4. RMS error as a function of the average number of active nodes, at least 3 nodes active, $g(x) = 1/x^2$. Comparison between algorithm A_2 and maximum likelihood for various assumed values for β.

Fig. 5(a) $g(x) = 1/x^2$, and in Fig. 5(b) $g(x) = 1/x^3$ is used as the energy decay function. However, both models result in similar conclusions, which can be summarized in the following.

1) Our centralized localization algorithms A_1 and A_2 offer better localization accuracy compared to the accuracy of the CPA and the Averaging estimators, when more than two nodes are on average active.

2) A_1 and A_2 offer almost indistinguishable performance, which is better than that of the POCS estimator of [8], although the latter assumes a known energy decay model and source power. The POCS estimator, approaches the performance of A_1 and A_2 when more than 10 nodes are on average active.

3) The distributed algorithm A_3, offers better localization performance than the CPA and the Averaging estimators, when more than 5 nodes are on average active.
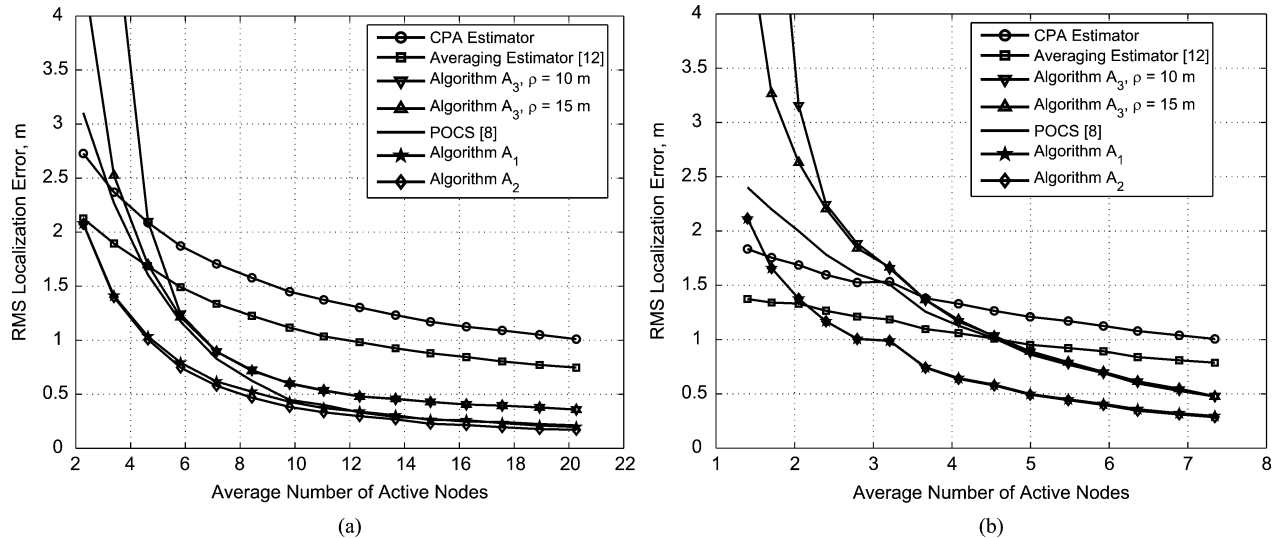
Fig. 5.   RMS error as a function of the average number of active nodes, at least 1 node active: (a) $g(x) = 1/x^2$. (b) $g(x) = 1/x^3$.

VII. CONCLUSION

In this paper, a cost function for model-independent source localization using energy measurements was considered. Also, a condition under which the optimal solution can be computed was given. This condition, involves a geometric construct called the sorted order-K Voronoi diagram, whose properties were examined. Also, centralized and distributed algorithms that exploit the derived condition for localization were proposed. Some theoretical aspects regarding the performance of the algorithms were explored. Finally, simulation results verified the effectiveness of the derived algorithms.

REFERENCES

[1] F. Zhao and L. Guibas, *Wireless Sensor Networks: An Information Processing Approach*.   San Francisco, CA: Morgan Kaufmann, 2004.

[2] S. Haykin, *Array Signal Processing*.   Englewood Cliffs, NJ: Prentice-Hall, 1985.

[3] N. Owsley and G. Swope, "Time delay estimation in a sensor array," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 3, pp. 519–523, 1981.

[4] D. Li and Y.-H. Hu, "Energy-based collaborative source localization using acoustic microsensor array," *EURASIP J. Appl. Signal Process.*, vol. 2003, no. 1, pp. 321–337, 2003.

[5] X. Sheng and Y.-H. Hu, "Maximum likelihood multiple-source localization using acoustic energy measurements with wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 44–53, Jan. 2005.

[6] W. Wang, V. Srinivasan, K.-C. Chua, and B. Wang, "Energy-efficient coverage for target detection in wireless sensor networks," in *IPSN '07: Proc. 6th Int. Conf. Inf. Process. Sens. Netw.*, New York, 2007, pp. 313–322.

[7] M. G. Rabbat and R. D. Nowak, "Decentralized source localization and tracking," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2004, pp. 921–924.

[8] D. Blatt and A. O. Hero, "Energy-based sensor network source localization via projection onto convex sets," *IEEE Trans. Signal Process.*, vol. 54, no. 9, pp. 3614–3619, Sep. 2006.

[9] B. Fidan, S. Dasgupta, and B. D. O. Anderson, "Conditions for guaranteed convergence in sensor and source localization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2007, pp. II-1081–II-1084.

[10] C. Meesookho, U. Mitra, and S. Narayanan, "On energy-based acoustic source localization for sensor networks," *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 365–377, Jan. 2008.

[11] D. Ampeliotis and K. Berberidis, "Linear least squares based acoustic source localization utilizing energy measurements," in *Proc. 5th IEEE Sens. Array Multichannel Signal Process. Workshop*, 2008.

[12] M. G. Rabbat, R. D. Nowak, and J. Bucklew, "Robust decentralized source localization via averaging," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2005, pp. 1057–1060.

[13] D. Ampeliotis and K. Berberidis, "Energy-based model-independent source localization in wireless sensor networks," in *Proc. 16th Eur. Signal Process. Conf. (EUSIPCO)*, 2008.

[14] K. Kaemarungsi and P. Krishnamurthy, "Modeling of indoor positioning systems based on location fingerprinting," in *Proc. INFOCOM 2004 23rd Ann. Joint Conf. IEEE Comput. Commun. Soc.*, Mar. 2004, vol. 2, pp. 1012–1022.

[15] F. Aurenhammer, "Voronoi diagrams—a survey of a fundamental geometric data structure," *ACM Comput. Surv.*, vol. 23, no. 3, pp. 345–405, 1991.

[16] M. I. Shamos and D. Hoey, "Closest-point problems," in *Proc. 1975. 16th Ann. Symp. Found. Comput. Sci.*, Oct. 1975, pp. 151–162.

[17] D.-T. Lee, "On k-nearest neighbor Voronoi diagrams in the plane," *IEEE Trans. Comput.*, vol. C-31, no. 6, pp. 478–487, Jun. 1982.

[18] D. Schmitt and J.-C. Spehner, "Order-k Voronoi diagrams, k-sections, and k-sets," in *Proc. JCDCG '98: Revised Papers from the Japan. Conf. Discrete and Computat. Geometry*, London, U.K., 2000, pp. 290–304.

[19] P. Carmi, S. Har-Peled, and M. J. Katz, "On the Fermat-Weber center of a convex object," *Comput. Geom.*, vol. 32, no. 3, pp. 188–195, 2005.

[20] M. Singh and V. K. Prasanna, "Energy-optimal and energy-balanced sorting in a single-hop wireless sensor network," in *Proc. First IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, 2003.

[21] J. O'Rourke, C. B. Chien, T. Olson, and D. Naddor, "A new linear algorithm for intersecting convex polygons," *Comput. Graphics Image Process.*, vol. 19, pp. 384–391, 1982.

[22] B. A. Bash and P. J. Desnoyers, "Exact distributed Voronoi cell computation in sensor networks," in *Proc. 6th Int. Conf. Inf. Process. Sens. Netw. IPSN'07*, New York, 2007, pp. 236–243.

[23] R. E. Miles, "Random polygons determined by random lines in a plane," in *Proc. Nat. Acad. Sci. (USA)*, 1964, vol. 52, pp. 901–907.

[24] A. Hayen and M. P. Quine, "Areas of components of a Voronoi polygon in a homogeneous Poisson process in the plane," *Adv. Appl. Probabil.*, vol. 34, pp. 281–291, 2002.

Dimitris Ampeliotis (S'05) was born in Athens, Greece, in 1979. He received the Diploma degree in computer engineering and informatics, and the Masters degree in signal processing from the University of Patras, Greece, in 2002 and 2004, respectively.

He is currently pursuing the Ph.D. degree in signal processing and communications at the University of Patras. Since 2003, he has been a Research Assistant with the Research Academic Computer Technology Institute (RACTI), Patras, and as a Teaching Assistant with the Computer Engineering and Informatics Department, University of Patras. His research interests include sensor networks, signal processing for communications and medical image processing.

Mr. Ampeliotis is a member of the Technical Chamber of Greece and a member of EURASIP.

Kostas Berberidis (S'87–M'90–SM'07) received the Diploma degree in electrical engineering from DUTH, Greece, in 1985, and the Ph.D. degree in signal processing and communications from the University of Patras, Greece, in 1990.

From 1986 to 1990, he was a Research Assistant with the Research Academic Computer Technology Institute (RACTI), Patras, and a Teaching Assistant with the Computer Engineering and Informatics Department (CEID), University of Patras. During 1991, he was with the Speech Processing Laboratory of the National Defense Research Center. From 1992 to 1994 and from 1996 to 1997, he was a Researcher with RACTI. During 1994–1995, he was a Postdoctoral Fellow with CCETT, Rennes, France. Since December 1997, he has been with CEID, University of Patras, where he is currently a Professor and Head of the Signal Processing and Communications Laboratory. His research interests include adaptive filtering, signal processing for communications, and sensor networks.

Dr. Berberidis has served as a member of scientific and organizing committees of several international conferences, as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING, and as a Guest Editor for the EURASIP *JASP*. He is currently serving as an Associate Editor for the IEEE SIGNAL PROCESSING LETTERS and for the EURASIP *Journal on Advances in Signal Processing*. He is also a Member of the Technical Chamber of Greece and a member of EURASIP.