Radio Map Estimation: A Data-Driven Approach to Spectrum Cartography

Daniel Romero and Seung-Jun Kim.

Abstract

Radio maps can be utilized to characterize a parameter of interest in a communication channel, such as the received signal strength, at every point of a certain geographical region. This article presents an introductory tutorial to radio map estimation, where radio maps are constructed using spatially distributed measurements. After describing the applications of this kind of maps, this article delves into estimation approaches. Starting by simple regression techniques, gradually more sophisticated algorithms are introduced until reaching state-of-the-art estimators. The presentation of this versatile toolkit is accompanied with toy examples to build up intuition and gain insight into the foundations of radio map estimation. As a secondary objective, this article attempts to reconcile the sometimes conflicting terminology in the literature and to connect multiple bodies of literature and sub-communities that have been working separately in this context.

Index Terms

Radio environment maps, spectrum cartography, radio map estimation.

I. Introduction

Spectrum cartography comprises a collection of techniques to construct and maintain radio maps, which provide useful information on the RF landscape, such as the received signal power,

D. Romero is with the Dept. of Information and Communication Technology, University of Agder, Jon Lilletunsvei 9, 4879 Grimstad, Norway. Email: daniel.romero@uia.no. S.-J. Kim is with the Dept. of Computer Science and Electrical Engineering, University of Maryland, Baltimore County, 1000 Hilltop Circle, Baltimore, MD 21250, USA. Email: sjkim@umbc.edu

This research has been funded in part by the Research Council of Norway under IKTPLUSS grant 311994.

interference power, power spectral density (PSD), electromagnetic absorption, and channel gain across a geographic area; see e.g. [1]–[4].

Besides the applications related to localization [2] and radio tomography [5], [6], radio maps find a myriad of applications in wireless communications and networking, such as network planning, interference coordination and mitigation, power control, resource allocation, handoff, multi-hop routing, dynamic spectrum access, and cognitive radio networking tasks; see [7], [8] and the references therein.

This article provides a tutorial introduction to spectrum cartography by guiding the reader from the foundations and main applications of radio map estimation to the recent advances in this rapidly growing research area.

II. Types of Radio Maps and Motivating Applications

The role of radio maps is to characterize the RF spectrum environment over space. As we know, the signal received at a certain location is determined by i) the transmitted signal; and ii) the communication channel between the transmitter and the receiver locations. Depending on which of these two ingredients are given more focus, two families of radio maps can be considered, namely *occupancy maps* and *propagation maps*. Both families are described next along with their major properties and motivating applications.

A. Occupancy Maps

The occupancy maps focus on the metrics of the received signal, or the aggregate effect of the channel and the signals transmitted by all active sources. This is the case, for instance, if the goal is to map interference power levels. Constructing such maps does not require the knowledge of the number, locations, and power of the transmitters, which is appealing in scenarios involving a large number of mobile transmitters, such as device-to-device communication or the cellular uplink. The term "occupancy" is brought here from the cognitive radio literature, where it refers to the aggregate contribution of all transmissions at the same time, frequency, and spatial location [9]. Different kinds of occupancy maps are presented next in the order of the amount of detail that the maps can capture.

1) Coverage Maps: The coarsest characterization of the radio environment can be provided by a map that takes only binary values. Specifically, let $p(\boldsymbol{x})$ denote the signal power that a radio receives at a spatial location $\boldsymbol{x} \in \mathcal{X}$, where $\mathcal{X}$ represents the geographical region of interest, typically a subset of $\mathbb{R}$, $\mathbb{R}^2$, or $\mathbb{R}^3$. A coverage map is a function $s : \mathcal{X} \to \{0, 1\}$ that takes

the value $s(\boldsymbol{x}) = 1$ if $p(\boldsymbol{x}) \geq \gamma$ and 0 otherwise, where γ is a given threshold. This threshold may correspond to the minimum signal power necessary to guarantee a prescribed minimum communication rate. It also makes sense to replace $p(\boldsymbol{x})$ with the *signal-to-noise-power ratio* (SNR) or the *signal-to-interference-plus-noise-power ratio* (SINR).

Coverage maps have been often used by cellular and TV broadcast network operators, to detect the areas of weak coverage and therefore to determine where new base stations and relay antennas need to be deployed. More recent application involves planning a mission for autonomous mobile robots or vehicles that require network connectivity. With a coverage map, important problems such as minimizing the time and distance traversed in the areas with no connectivity can be tackled.

2) Outage Probability Maps: A soft version of the coverage maps can be constructed by adopting a probabilistic perspective, as the effects of the channel, such as fading and shadowing, are often modeled as random. An outage probability map $s(\boldsymbol{x})$ provides the probability that $p(\boldsymbol{x}) < \gamma$. Since the outage probability map captures more detailed information than the coverage map, the former can be employed in any application of the latter. However, the additional information provided by outage probability maps allows more sophisticated decision-making procedures, such as the route planning [10].

3) Power Maps: A substantially finer characterization of the occupancy is obtained by power maps, which return the received power $p(\boldsymbol{x})$ directly at every spatial location $\boldsymbol{x} \in \mathcal{X}$. This tacitly assumes that such received power does not change significantly within a certain time interval which depends on the application and mobility of the terminals. As the information contained in the power maps is richer than that in the coverage or the outage probability maps, the power maps can be used not only for such tasks as network planning and trajectory optimization, but also for localization tasks. For instance, the locations of the transmitters can be determined from the power map [2]. In the fingerprint-based localization, a mobile device can measure the received powers of nearby access points and determine its position by matching the measurement with the predicted values in the map.

4) PSD Maps: One is sometimes interested not only in the power distribution across space but also across the frequency domain. A PSD map is a function $p(\boldsymbol{x}, f)$ that provides the PSD of the received signal at each location $\boldsymbol{x} \in \mathcal{X}$. In practice one discretizes the frequency variable f. Then the PSD map can be thought of as a collection of power maps $p(\boldsymbol{x}, f_1), \ldots, p(\boldsymbol{x}, f_{N_f})$, for a set of different frequency bins or bands $\{f_n\}$.

In addition to the applications mentioned for the previous kinds of occupancy maps, PSD maps enable multiple additional use cases. For example, they can be used for speeding up handoff procedures in cellular networks by providing the quality of the relevant channels at a given location, which saves the need for time-consuming channel measurement or feedback processes when it comes to determine which one is the strongest channel. Furthermore, since PSD maps characterize the distribution of the RF signal power per channel over space, they can play a major role in increasing frequency reuse and mitigating interference by enabling interference coordination techniques that are aware of the user locations. Additional use cases emerge in cognitive radio networks, where the so-called secondary users exploit underutilized spectrum resources in the space/frequency/time domains. These users can therefore benefit from PSD power maps since they unveil underutilized "white spaces" [11].

B. Propagation Maps

Whereas the occupancy maps capture the aggregate effect of the transmitted signals and the channels, propagation maps focus on the channel effects. In principle, one can conceive different kinds of propagation maps for different channel parameters of interest. *Channel gain maps*, described next, constitute the simplest kind of propagation maps. Suppose that p^{RX} denotes the power received at location $\boldsymbol{x}^{\text{RX}}$ from a transmitter at location $\boldsymbol{x}^{\text{TX}}$ whose transmit power is p^{TX}. A *channel gain map* is nothing but the (power) gain $p^{\text{RX}}/p^{\text{TX}}$ seen as a function $h(\boldsymbol{x}^{\text{TX}}, \boldsymbol{x}^{\text{RX}})$ of the transmitter and receiver locations.

Clearly, given a propagation map $h(\boldsymbol{x}^{\text{TX}}, \boldsymbol{x}^{\text{RX}})$ together with the locations $\boldsymbol{x}_1^{\text{TX}}, \ldots, \boldsymbol{x}_S^{\text{TX}}$ and transmit-powers $p_1^{\text{TX}}, \ldots, p_S^{\text{TX}}$ of the S sources in a region, one can obtain the power map as $p(\boldsymbol{x}) = \sum_s h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}) p_s^{\text{TX}}$, provided that the signals transmitted by the sources are uncorrelated. Thus, one can argue that propagation maps offer more detailed information than occupancy maps such as power maps. Their construction, however, is more challenging, as discussed later. The changes in the locations and transmit-powers of the sources give rise to the changes in the occupancy maps, whereas the propagation maps may remain unaffected. On the other hand, alterations in the scattering environment, such as the construction of new buildings or seasonal changes of foliage, affect the propagation maps, and in turn, the occupancy maps. Thus, the time scale of variations in occupancy maps is never greater than that of the propagation maps.

Propagation maps can serve in many of the applications of occupancy maps provided that the locations and transmit power of the sources are known. However, the fact that occupancy maps capture the aggregate effects of all sources renders propagation maps more suitable in certain

use cases. For example, an occupancy map may provide the total interference at each location, whereas a propagation map can be used to determine the contribution of each source to that interference. This results in enhanced flexibility in tasks such as interference coordination or network planning.

Furthermore, due to the slower variation of propagation maps relative to occupancy maps alluded to earlier, the former can be used to construct occupancy maps in highly dynamic setups, such as the uplink of a cellular network, where mobile users rapidly change their positions and activity patterns. To this end, information on the locations and transmit power of the users would need to be collected.

Propagation maps can also help address the classical problem of predicting the potential interference inflicted to passive receivers. This need arises in the context of cognitive radios. For example, when reusing the TV spectrum [11], the challenge is to allow transmissions without introducing detrimental interference to the TV receivers. With a propagation map, one can ensure that no receiver in a certain area will be negatively affected without the need for knowing its precise location.

Yet another application of propagation maps is the problem of *aerial base station placement* [12]. There, a propagation map of the air-to-ground channels can be constructed to determined the best set of locations where base stations mounted on unmanned aerial vehicles (UAVs) need to be deployed to serve the users on the ground.

III. Radio Map Estimation

Arguably, spectrum cartography can be traced back to the application of Maxwell's equations to characterize the propagation of radio waves across space. However, in the absence of computational power, the approach was confined to problems involving relatively simple geometries such as determining the electromagnetic field radiated by a dipole. To analyze more complex environments, a vast number of empirical models have been developed, such as the well-known P-recommendations from the International Telecommunication Union-Radiocommunication Sector (ITU-R).

With the advent of the modern computational power, finite-element analysis and ray-tracing techniques paved the way for effectively approximating the solutions of Maxwell's equations for complex environments. Besides their high computational complexity, the main limitation of such approaches is that they require an accurate description of the propagation environment by means of the 3D models of all objects and obstacles along with their electromagnetic properties.

To mitigate such limitations, RME was proposed, originally in the context of cognitive radios [1]. In RME, a collection of measurements acquired by spatially distributed sensors is used together with their locations to construct a map of the relevant RF descriptors, typically by applying some form of interpolation algorithms. As this does not require physical modeling of the propagation environment, it constitutes a *data-driven* alternative to the model-based approaches described earlier. Since its conception, a sizable body of literature has emerged on the estimation of a variety of kinds of radio maps for a wide range of application scenarios; see e.g. [2], [3], [7], [8], [13], [14] and the references therein. In the last few years, the work in this area has intensified thanks to the boom of deep learning techniques [15]–[18].

In the typical RME formulation, the goal is to construct a radio map using a certain number N of measurements together with their locations. For occupancy maps, the n-th measurement m_n is acquired by a sensor at location $\boldsymbol{x}_n$. For example, in the case of power maps, m_n is typically the power measured in a certain band within a certain time interval. The problem becomes constructing the desired occupancy map given the pairs $\{(\boldsymbol{x}_n, m_n)\}_{n=1}^N$.

For propagation maps, the n-th measurement is collected by a pair of radios, one at location $\boldsymbol{x}_n$ and the other at $\boldsymbol{x}'_n$. The radio at $\boldsymbol{x}_n$ transmits a signal of a predetermined power and the radio at $\boldsymbol{x}'_n$ measures the received power to obtain m_n as an estimate of the channel gain. The RME problem is then posed to construct the propagation map given $\{(\boldsymbol{x}_n, \boldsymbol{x}'_n, m_n)\}_{n=1}^N$.

It is worth noting that each sensor may collect measurements at multiple locations provided that the measurements are taken within the time scale of variations of the target map. Thus, the number of sensors may be much smaller than N. Furthermore, a sensor need not be a special-purpose device. A user terminal in cellular networks may function as a sensing device. In fact, the RME formulation can be extended to accommodate the decision on where to acquire the next measurements given the past measurements and their locations [19]. This *spectrum surveying* task will be discussed in Sec. VI.

IV. Estimation of Occupancy Maps

Here we present some of the main approaches for constructing occupancy maps. We will focus on the power map estimation to illustrate the main challenges and trade-offs.

A. Estimation of Power Maps

As indicated earlier, when it comes to estimating power maps, measurements are typically obtained by measuring power in a certain band within a certain time interval. However, due to

the finite averaging time, the n-th measurement m_n will not generally coincide with the true power $p(\boldsymbol{x}_n)$. It is thus customary to model the measurements as $m_n = p(\boldsymbol{x}_n) + z_n$, where z_n denotes measurement error, also known as noise. This section describes how a power map p can be estimated from a set of N measurements defined in this way. Following standard mathematical notation, p refers to a *function* (in this case a map) whereas $p(\boldsymbol{x})$ denotes the *value* of function p at point $\boldsymbol{x}$.

1) Linear Parametric RME: Let us start by considering the simplest possible scenario of estimating a power map, which arises when there is a single transmitter with known location $\boldsymbol{x}_1^{\text{TX}}$ in free space. As per Friis' transmission equation, the received power at location $\boldsymbol{x}$ is inversely proportional to the squared distance $\|\boldsymbol{x} - \boldsymbol{x}_1^{\text{TX}}\|^2$. In other words, $p(\boldsymbol{x})$ can be written as $p(\boldsymbol{x}) = \alpha_1 \psi_1(\boldsymbol{x})$, where $\psi_1(\boldsymbol{x}) := 1/\|\boldsymbol{x} - \boldsymbol{x}_1^{\text{TX}}\|^2$ and α_1 depends on the (unknown) transmit power. To know $p(\boldsymbol{x})$ everywhere, it therefore suffices to obtain α_1. In this simplistic scenario, one would be able to estimate the entire map with a single noiseless measurement: if $\boldsymbol{x}_1$ is the measurement location, then $m_1 = p(\boldsymbol{x}_1) = \alpha_1 \psi_1(\boldsymbol{x}_1)$ implies $\alpha_1 = m_1/\psi_1(\boldsymbol{x}_1)$.

Similarly, if S transmitters with known locations $\boldsymbol{x}_1^{\text{TX}}, \ldots, \boldsymbol{x}_S^{\text{TX}}$ are active in a certain region, one can model $p(\boldsymbol{x})$ as

$$p(\boldsymbol{x}) = \alpha_1 \psi_1(\boldsymbol{x}) + \ldots + \alpha_S \psi_S(\boldsymbol{x}) \tag{1}$$

where $\psi_s(\boldsymbol{x}) := 1/\|\boldsymbol{x} - \boldsymbol{x}_s^{\text{TX}}\|^2$. In this case, one will typically be able to estimate the S coefficients from S noiseless measurements by solving the system of equations

$$\begin{aligned} m_1 &= \alpha_1 \psi_1(\boldsymbol{x}_1) + \ldots + \alpha_S \psi_S(\boldsymbol{x}_1) \\ &\vdots \\ m_S &= \alpha_1 \psi_1(\boldsymbol{x}_S) + \ldots + \alpha_S \psi_S(\boldsymbol{x}_S) \end{aligned} \tag{2}$$

for $\alpha_1, \ldots, \alpha_S$. In practice, however, measurements are noisy and one may use $N > S$ of them to estimate the coefficients $\boldsymbol{\alpha} := [\alpha_1, \ldots, \alpha_S]^\top$. Upon defining $\boldsymbol{m} := [m_1, \ldots, m_N]^\top$, $(\boldsymbol{\Psi})_{n,s} := \psi_s(\boldsymbol{x}_n)$ and $\boldsymbol{z} := [z_1, \ldots, z_N]^\top$, (2) can be written as $\boldsymbol{m} = \boldsymbol{\Psi}\boldsymbol{\alpha} + \boldsymbol{z}$. Thus, a least squares (LS) estimate is obtained as $\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \|\boldsymbol{m} - \boldsymbol{\Psi}\boldsymbol{\alpha}\|^2$. Fig. 1 illustrates a setup where a map needs to be estimated on a line, i.e. the region of interest is given by $\mathcal{X} = \mathbb{R}$, which may correspond to a road or railway. This figure compares the true map with the estimated map obtained by substituting $\hat{\boldsymbol{\alpha}}$ into the right-hand side (RHS) of (1). The map is seen to be reasonably accurate and can be shown to converge to the true map for $N \to \infty$ under mild conditions.

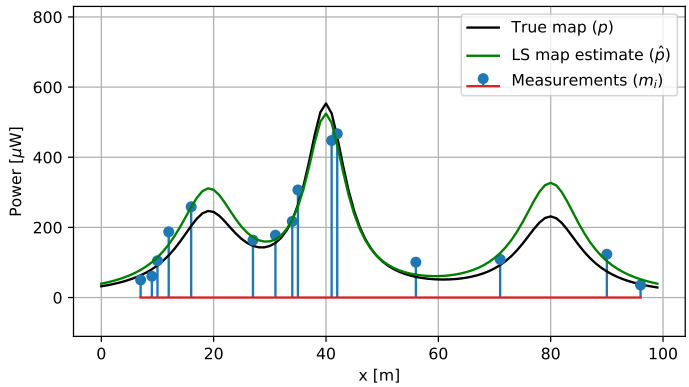


Fig. 1: Example of map estimation in 1D using a parametric estimator that knows the transmitter locations. The estimate is reasonably accurate despite the low number of measurements.

Because the number S of parameters to be estimated does not depend on the number of measurements N, this approach is termed *parametric*. Further parametric and *non-parametric* estimators are discussed in the rest of this section.

The natural next step in this direction is to consider generic basis functions. Note that, so far, it was assumed that propagation takes place in free space. If this is not the case, then the above basis functions $\psi_s(\boldsymbol{x}) = 1/\|\boldsymbol{x} - \boldsymbol{x}_s^{\text{TX}}\|^2$ may not lead to a satisfactory fit. Although one can in principle adopt other families of basis functions, such as those determined by the well-known Okumura-Hata model, it is clear that the flexibility of such an approach is rather limited. Besides, the location of the sources is required, which may not be a realistic assumption in certain applications. These observations suggest generalizing the expansion in (1) as

$$p(\boldsymbol{x}) = \alpha_1 \psi_1(\boldsymbol{x}) + \ldots + \alpha_B \psi_B(\boldsymbol{x}) \tag{3}$$

where $\psi_b(\boldsymbol{x})$ is not necessarily as before: it can take an arbitrary form determined by the user and need not even be linked to any particular transmitter. For example, in the case where a map needs to be constructed on a line, $\{\psi_b(\boldsymbol{x})\}_b$ could form a polynomial basis by setting $\psi_b(\boldsymbol{x}) = \psi_b(x) = x^{b-1}$. The coefficients $\alpha_1, \ldots, \alpha_B$ can be found similarly by using LS estimation. However, despite the appealing simplicity of this approach, the quality of the estimates is oftentimes poor, as illustrated by Fig. 2 for the same setup as Fig. 1. Comparing Figs. 2 and 1 suggests that regression methods are sensitive to the choice of the basis functions.

2) Kernel-Based Learning: The main limitation of the parametric approach is the difficulty to select suitable basis functions. This is even more challenging for higher dimensions such as

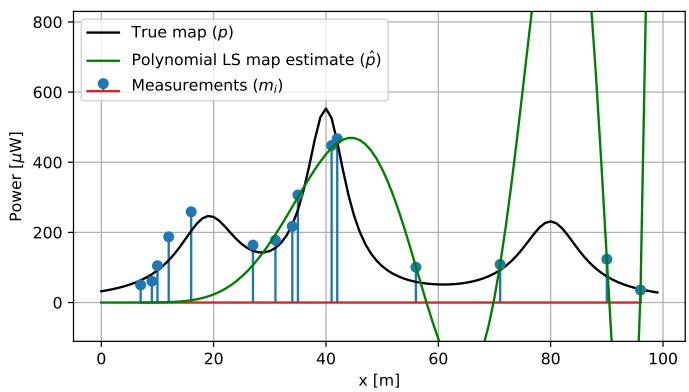


Fig. 2: Example of map estimation by fitting a polynomial of degree 13 via LS. The estimate is clearly unsatisfactory despite the fact that the estimate fits accurately most of the measurements.

$\mathcal{X} = \mathbb{R}^2$ or $\mathcal{X} = \mathbb{R}^3$. To sidestep this, kernel-based learning has been extensively used in RME, mainly due to its simplicity, universality, and good performance [20].

The starting point is to postulate a family of functions $\mathcal{G}$ where an estimate $\hat{p}$ of function p is a member. The goal is to use the data $\{(\boldsymbol{x}_n, m_n)\}_{n=1}^N$ to select $\hat{p}$ in $\mathcal{G}$ that satisfies $\hat{p}(\boldsymbol{x}) \approx p(\boldsymbol{x})\ \forall \boldsymbol{x}$. In kernel-based learning, $\mathcal{G}$ is a special class of functions named *reproducing-kernel Hilbert space* (RKHS), which is given by:

$$\mathcal{G} := \left\{ g : g(\boldsymbol{x}) = \sum_{i=1}^{\infty} \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i'),\ \boldsymbol{x}_i' \in \mathcal{X},\ \alpha_i \in \mathbb{R} \right\} \tag{4}$$

where $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a *reproducing kernel* [20, Ch. 2], i.e., a function that is (i) symmetric, meaning that $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \kappa(\boldsymbol{x}', \boldsymbol{x})\ \forall \boldsymbol{x}', \boldsymbol{x}$; and (ii) positive definite, meaning that the matrix $\bar{\boldsymbol{K}}$ with entries $(\bar{\boldsymbol{K}})_{i,j} = \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is positive definite for any set of points $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$. Although any reproducing kernel can be used, a common choice is the so-called Gaussian *radial basis function* (RBF) $\kappa(\boldsymbol{x}, \boldsymbol{x}') := \exp\left(-\|\boldsymbol{x} - \boldsymbol{x}'\|^2/2\sigma^2\right)$, where $\sigma > 0$ is a prescribed parameter. Observe that the set in (4) contains a function for every choice of the *centroids* $\boldsymbol{x}_i'$ and coefficients α_i. When the kernel is a Gaussian RBF, $\kappa(\boldsymbol{x}, \boldsymbol{x}_i')$ seen as a function of $\boldsymbol{x}$ is a bell shape centered at $\boldsymbol{x}_i'$. Thus, a function in $\mathcal{G}$ is a (possibly infinite) sum of Gaussian bells with different centers and amplitudes. An example is shown in Fig. 3.

In view of (4), finding a suitable estimate $\hat{p}$ in $\mathcal{G}$ amounts to determining a set of coefficients $\{\alpha_i\}$ and centroids $\{\boldsymbol{x}_i\}$. To this end, the typical approach is to seek $\hat{p}$ as the solution of

$$\hat{p} = \arg\min_{g \in \mathcal{G}} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}\left(\boldsymbol{x}_n, m_n, g(\boldsymbol{x}_n)\right) + \lambda \|g\|_{\mathcal{G}}^2 \tag{5}$$

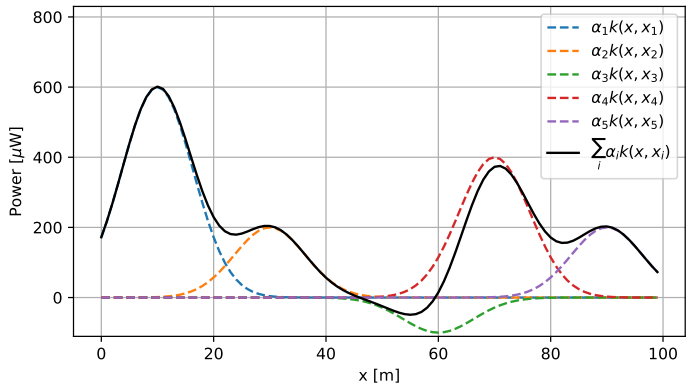


Fig. 3: Example of a function in an RKHS obtained with the expansion in (4) with only 5 terms.

where $\lambda > 0$ is a pre-determined regularization parameter, $\mathcal{L}$ is a loss function quantifying the deviation between the observations $\{m_n\}_{n=1}^N$ and the predictions $\{g(\boldsymbol{x}_n)\}_{n=1}^N$ produced by a candidate g, and $\|g\|_{\mathcal{G}}$ is a norm, which for a RKHS function $g(\boldsymbol{x}) = \sum_{i=1}^{\infty} \alpha_i \kappa(\boldsymbol{x}, \boldsymbol{x}_i')$ is given by

$$\|g\|_{\mathcal{G}}^2 := \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \alpha_i \alpha_j \kappa(\boldsymbol{x}_i', \boldsymbol{x}_j'). \tag{6}$$

A typical loss adopted is the *square loss* $\mathcal{L}(m_n, \boldsymbol{x}_n, g(\boldsymbol{x}_n)) = (m_n - g(\boldsymbol{x}_n))^2$, in which case (5) is referred to as the *kernel ridge regression* (KRR) [20, Ch. 4].

To understand the intuition behind (5), one can start by noting that $\mathcal{L}$ is typically selected so that $\mathcal{L}(\boldsymbol{x}_n, m_n, g(\boldsymbol{x}_n))$ is zero whenever $g(\boldsymbol{x}_n) = m_n$ and positive otherwise, which means that the first term in the RHS of (5) becomes 0 when g perfectly fits the data. In the absence of the second term, the infinite degrees of freedom of g (cf. (4)) generally make it possible to find a solution to (5) where the first term is 0. Unfortunately, the resulting estimate $\hat{p}$ would be very irregular and likely to differ significantly from p away from the measurement locations $\boldsymbol{x}_n$ because of the noise present in the measurements. This is termed *overfitting* and therefore needs to be avoided. Since some of the α_i of overfitted solutions are likely to be large, a way to reduce overfitting is by penalizing solutions with large values of these coefficients. This is precisely the role of the second term in the RHS of (5), since the fact that κ is positive definite implies that $\|g\|_{\mathcal{G}}^2$ in (6) becomes large as the coefficients α_i become large. Thus, the terms in the RHS of (5) pose a tradeoff and λ needs to be adjusted to select the desired "sweet spot": on the one hand, if λ is too small, the solution will fit well the data but will likely incur overfitting. Conversely, if λ is too large, the resulting $\hat{p}$ will be a *smooth* function but $\hat{p}(\boldsymbol{x}_n)$ may significantly differ from

$p(\boldsymbol{x}_n)$.

To solve (5), one could initially think of substituting the expansion (4) into (5) and optimizing with respect to the infinitely many coefficients α_i and centroids $\boldsymbol{x}_i'$. However, this approach is obviously intractable. Instead, one may resort to the so-called *representer theorem* [20, Th. 4.2], which states that the solution to (5) must be of the form:

$$\hat{p}(\boldsymbol{x}) = \sum_{n=1}^{N} \alpha_n \kappa(\boldsymbol{x}, \boldsymbol{x}_n) \tag{7}$$

for some $\{\alpha_n\}_{n=1}^N$. Observe that the centroids in (7) are precisely the measurement locations. This effectively reduces an optimization problem with infinitely many variables into a problem with just the N variables $\alpha_1, \ldots, \alpha_N$. Now, one can substitute (7) into (5) and solve

$$\hat{\boldsymbol{\alpha}} = \arg\min_{\boldsymbol{\alpha}} \frac{1}{N} \|\boldsymbol{m} - \boldsymbol{K}\boldsymbol{\alpha}\|^2 + \lambda \boldsymbol{\alpha}^\top \boldsymbol{K} \boldsymbol{\alpha} \tag{8}$$

where, with some abuse of notation, $\boldsymbol{\alpha} := [\alpha_1, ..., \alpha_N]^\top$, and $\boldsymbol{K}$ is a positive-definite $N \times N$ matrix whose (i, j)-th entry is $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Observe that, as opposed to the methods described in Sec. IV-A1, the number of parameters to be determined here does depend on the number of measurements N, which is why the method is *non-parametric*. Problem (8) can be readily solved in closed-form as

$$\hat{\boldsymbol{\alpha}} = (\boldsymbol{K} + \lambda N \mathbf{I}_N)^{-1} \boldsymbol{m}. \tag{9}$$

The estimate $\hat{p}$ solving (5) for kernel ridge regression can be recovered by substituting the entries of $\hat{\boldsymbol{\alpha}}$ into (7).

Figs. 4 and 5 show KRR estimates in the same setup as the previous figures. Observe that, as the number of measurements increases, the estimate becomes closer to the true map. It is worth mentioning that, in RME, kernel methods are suited for scenarios where no prior knowledge about the propagation environment is available due to their ability to approximate spatial fields with arbitrarily high accuracy.

A brief discussion on the extensions to KRR is in order. One can combine the flexibility of the non-parametric estimator in this section with the ability of the parametric estimator in Sec. IV-A1 to capture prior information by means of basis functions grounded on domain knowledge. To this end, one can assume that p is the sum of a function of the form of the RHS of (3) and a function in an RKHS [21]. This approach also generalizes the so-called *thin-plate spline regression*, which has well-documented merits in RME [7], [14].

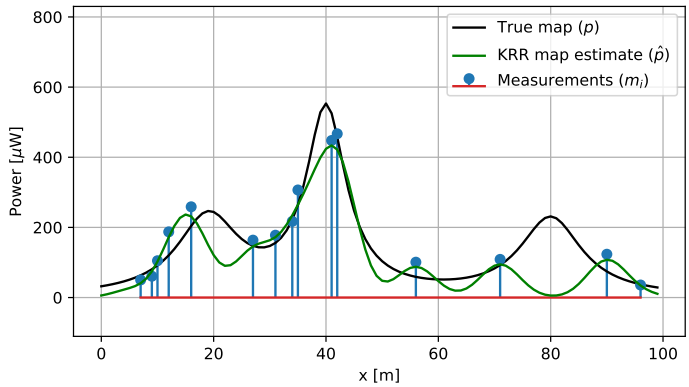


Fig. 4: Example of KRR estimate. As expected, the quality of the fit is higher in regions with higher measurement density.

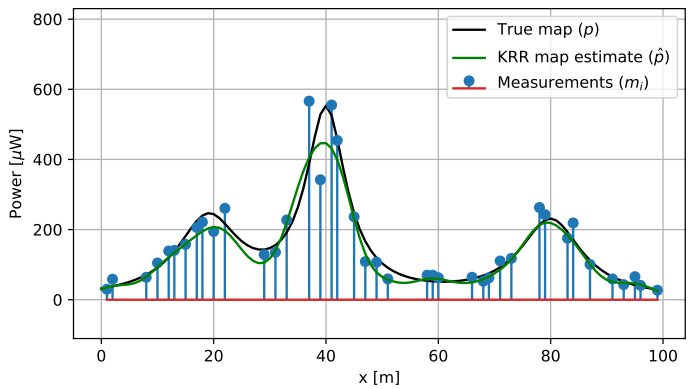


Fig. 5: Example of KRR estimate with more measurements than in Fig. 4. The fit is considerably better.

Another possible extension deals with the choice of the kernel. The reason is that estimation performance may be significantly affected by the adopted kernel or its parameters. Using *multi-kernel* learning, one specifies a dictionary of kernels and a suitably designed algorithm uses the data to construct a kernel by combining the kernels in the dictionary.

3) Kriging: Another approach to radio map estimation arises by adopting a probabilistic model. In this context, *kriging* is probably the simplest and most popular approach [1], [9], [19]. There are several ways to arrive at the kriging estimator. This section pursues one of them that models $p(\boldsymbol{x})$ as a random variable by relying on customary channel assumptions.

Assume for simplicity that there is a single transmitter at location $\boldsymbol{x}^{\mathrm{TX}}$ that transmits with power p^{TX}. The received power in logarithmic units can be clearly written as $p(\boldsymbol{x}) = p^{\mathrm{TX}} + h(\boldsymbol{x}^{\mathrm{TX}}, \boldsymbol{x})$,

where $h(\boldsymbol{x}^{\text{TX}}, \boldsymbol{x})$ is expressed in dB. A common decomposition for the latter is $h(\boldsymbol{x}^{\text{TX}}, \boldsymbol{x}) = h^{\text{PL}}(\boldsymbol{x}) - a^{\text{SF}}(\boldsymbol{x}) - a^{\text{FF}}(\boldsymbol{x})$, where $h^{\text{PL}}(\boldsymbol{x})$ is the path loss, $a^{\text{SF}}(\boldsymbol{x})$ is the attenuation due to shadow fading, and $a^{\text{FF}}(\boldsymbol{x})$ is the attenuation due to (fast) fading. The dependence of the RHS on $\boldsymbol{x}^{\text{TX}}$ has been omitted to simplify notation. Recall that shadow fading is produced by obstructions in the line of sight between the transmitter and the receiver, whereas fast fading owes to the constructive and destructive interference between the different multipath components arriving at the receiver.

Under this model, it is reasonable to assume that $h^{\text{PL}}(\boldsymbol{x})$ is a deterministic quantity since it only depends on $\boldsymbol{x}$ and $\boldsymbol{x}^{\text{TX}}$, which are given. On the other hand, one may model $a^{\text{SF}}(\boldsymbol{x})$ and $a^{\text{FF}}(\boldsymbol{x})$ as uncorrelated (or even independent) random variables with certain means μ^{SF} and μ^{FF}.

To model the spatial structure of a^{SF}, one may resort to the well-known Gudmundson correlation model [22], which prescribes that $\text{Cov}[a^{\text{SF}}(\boldsymbol{x}), a^{\text{SF}}(\boldsymbol{x}')] = \sigma^2_{\text{SF}} 2^{-\|\boldsymbol{x}-\boldsymbol{x}'\|/d^{\text{SF}}}$. Here, σ^2_{SF} is a constant and d^{SF} is the distance at which the correlation decays by 50%.

On the other hand, given the rapid spatial variability of $a^{\text{FF}}(\boldsymbol{x})$, it makes sense to assume that this quantity is spatially independent. This implies that $\text{Cov}[a^{\text{FF}}(\boldsymbol{x}), a^{\text{FF}}(\boldsymbol{x}')] = \sigma^2_{\text{FF}} \delta[\boldsymbol{x} - \boldsymbol{x}']$, where σ^2_{FF} is a given constant and function δ takes the value 1 when its argument is 0 and the value 0 otherwise.

Another implication of the spatial independence assumption of $a^{\text{FF}}(\boldsymbol{x})$ is that there is no hope that this component can be estimated given measurements at other locations. Thus, it is more appropriate to regard $a^{\text{FF}}(\boldsymbol{x})$ as a perturbation around the shadow fading and deterministic terms. Specifically, one may adopt the decomposition $p(\boldsymbol{x}) = \mu_p(\boldsymbol{x}) - \bar{a}^{\text{SF}}(\boldsymbol{x}) - \bar{a}^{\text{FF}}(\boldsymbol{x})$ where $\mu_p(\boldsymbol{x}) := p^{\text{TX}} + h^{\text{PL}}(\boldsymbol{x}) - \mu^{\text{SF}} - \mu^{\text{FF}}$, $\bar{a}^{\text{SF}}(\boldsymbol{x}) := a^{\text{SF}}(\boldsymbol{x}) - \mu^{\text{SF}}$, and $\bar{a}^{\text{FF}}(\boldsymbol{x}) := a^{\text{FF}}(\boldsymbol{x}) - \mu^{\text{FF}}$.

In short, it follows that the above power map model is fully determined by $\text{E}[p(\boldsymbol{x})] = \mu_p(\boldsymbol{x})$ and $\text{Cov}[p(\boldsymbol{x}), p(\boldsymbol{x}')] = \sigma^2_{\text{SF}} 2^{-\|\boldsymbol{x}-\boldsymbol{x}'\|/d^{\text{SF}}} + \sigma^2_{\text{FF}} \delta[\boldsymbol{x} - \boldsymbol{x}']$. Furthermore, recalling that the measurements are given by $m_n = p(\boldsymbol{x}_n) + z_n$, one can conclude that $\text{E}[m_n] = \mu_p(\boldsymbol{x}_n)$ and $\text{Cov}[m_n, m_{n'}] = \sigma^2_{\text{SF}} 2^{-\|\boldsymbol{x}_n - \boldsymbol{x}_{n'}\|/d^{\text{SF}}}$ if $n \neq n'$ and $\text{Cov}[m_n, m_{n'}] = \sigma^2_{\text{SF}} + \sigma^2_{\text{FF}} + \sigma^2_z$ otherwise, where it has been assumed that z_n is zero mean, uncorrelated across n, uncorrelated with $p(\boldsymbol{x}_{n'})$ for all $n \neq n'$, and has variance σ^2_z.

Under this model, the problem of estimating $p(\boldsymbol{x})$ given the measurements $\boldsymbol{m} := [m_1, \ldots, m_N]^\top$ can be solved using the well-known *linear minimum mean square error* (LMMSE) estimator, which is given by

$$\hat{p}(\boldsymbol{x}) = \mu_p(\boldsymbol{x}) + \text{Cov}[p(\boldsymbol{x}), \boldsymbol{m}] \text{Cov}^{-1}[\boldsymbol{m}, \boldsymbol{m}](\boldsymbol{m} - \text{E}[\boldsymbol{m}]). \tag{10}$$

Observe that all quantities in the RHS are given by the model above. Remarkably, if the random variables involved in this problem are Gaussian, which follows to a large extent from the usual assumption that shadow fading is log-normal, then the aforementioned estimator is also the *minimum mean square error* (MMSE) estimator.

It is worth comparing (10) with (7), where the coefficients in the latter equation are given by (9). It can be easily seen that, except for the mean terms in (10), the estimators provided by (10) and (7) coincide if one sets $\kappa(\boldsymbol{x}, \boldsymbol{x}') = \mathrm{Cov}[p(\boldsymbol{x}), p(\boldsymbol{x}')]$ and adjusts λ properly. This is a manifestation of the general fact that a reproducing kernel can be thought of as a generalization of covariance.

4) Leveraging Sparsity: In many practical RME problems, the map estimation performance can be significantly improved by incorporating useful prior information. The sparsity prior has played a critical role in compressive sensing (CS), in which framework the RME problems can often be formulated. Moreover, depending on the choice of the basis functions, the sparsity prior can be physically interpreted in terms of the spatial, temporal, and spectral scarceness of the RF energy distribution [2], [3], [14].

Going back to the linear parametric RME model (1), but rather than assuming that the number S and the locations $\{\boldsymbol{x}_s^{\mathrm{TX}}\}$ of the transmitters are known, simply discretize the map area using N_g grid points $\{\boldsymbol{x}_{n_g}^{\mathrm{grid}}\}$ representing the possible locations of transmitters. Then, upon defining $\tilde{\boldsymbol{\alpha}} := [\tilde{\alpha}_1, \ldots, \tilde{\alpha}_{N_g}]^\top$ and $\tilde{\boldsymbol{\Psi}} \in \mathbb{R}^{N \times N_g}$ with $(\tilde{\boldsymbol{\Psi}})_{n,n_g} = \psi_{n_g}(\boldsymbol{x}_n) := 1/\|\boldsymbol{x}_n - \boldsymbol{x}_{n_g}^{\mathrm{grid}}\|^2$ for $n = 1, \ldots, N$ and $n_g = 1, \ldots, N_g$, one has the model $\boldsymbol{m} = \tilde{\boldsymbol{\Psi}}\tilde{\boldsymbol{\alpha}} + \boldsymbol{z}$. Now, it is expected that only a small subset of the grid points are actually occupied by transmitters in practical scenarios, yielding $N_g \gg S$. Thus, one can impose the sparsity prior on $\tilde{\boldsymbol{\alpha}}$. For example, a Lasso problem can be formulated as $\hat{\tilde{\boldsymbol{\alpha}}} := \arg\min_{\tilde{\boldsymbol{\alpha}}} \|\boldsymbol{m} - \tilde{\boldsymbol{\Psi}}\tilde{\boldsymbol{\alpha}}\|_2^2 + \lambda\|\tilde{\boldsymbol{\alpha}}\|_1$, where the term $\|\tilde{\boldsymbol{\alpha}}\|_1 := \sum_{n_g=1}^{N_g} |\tilde{\alpha}_{n_g}|$ is known to promote sparsity in $\tilde{\boldsymbol{\alpha}}$. The non-zero entries of the obtained $\hat{\tilde{\boldsymbol{\alpha}}}$ reveal the (grid-based) locations $\{\boldsymbol{x}_s^{\mathrm{TX}}\}$ and the number S of the transmitters. Then, one can reconstruct the desired power map $p(\boldsymbol{x})$ using (1).

In using the basis functions $\psi_{n_g}(\boldsymbol{x}) = 1/\|\boldsymbol{x} - \boldsymbol{x}_{n_g}^{\mathrm{grid}}\|^2$, the propagation characteristic is assumed to be known deterministically [2], [3]. The restrictive assumption can be alleviated via a sparse total least-square (TLS) approach [13], kernel-based learning [14], and sparse Bayesian learning [23]. In [23], the locations of the transmitters were directly estimated without relying on a set of grid points.

5) Matrix Completion: Another useful framework for RME is low-rank matrix completion. For instance, consider building a power map over a rectangular area $\mathcal{R} \subset \mathbb{R}^2$. By discretizing $\mathcal{R}$ using a regular grid $\{\boldsymbol{x}^{\text{grid}}_{(i,j)} : i = 1, \ldots, I,\ j = 1, \ldots, J\}$, one can obtain a power map matrix $\boldsymbol{P} \in \mathbb{R}^{I \times J}$ where $(\boldsymbol{P})_{i,j} := p(\boldsymbol{x}^{\text{grid}}_{(i,j)})$. Of course, only a small subset of the entries will be actually observed by the sensors. However, when the grid is dense enough compared to the spatial variability of the map, the entries of $\boldsymbol{P}$ will be correlated, which will, in turn, manifest itself as the approximate rank deficiency of $\boldsymbol{P}$, that is, $\text{rank}(\boldsymbol{P}) \ll \min\{I, J\}$. The matrix completion thus tries to estimate the unobserved entries of $\boldsymbol{P}$ under a low-rank prior. A tractable formulation can be obtained by substituting the nonconvex matrix rank with the nuclear norm of the matrix, which is the sum of the singular values. Denote the set of indices of the observed entries as $\mathcal{O} \subset \{1, \ldots, I\} \times \{1, \ldots, J\}$ and the nuclear norm of $\boldsymbol{P}$ as $\|\boldsymbol{P}\|_*$. Also, let $\boldsymbol{M}$ be the matrix with the (i, j)-element equal to the sensor measurement at $\boldsymbol{x}^{\text{grid}}_{(i,j)}$ if $(i, j) \in \mathcal{O}$ and 0 otherwise. A matrix completion problem for the power map can be posed as

$$\underset{\boldsymbol{P}}{\text{minimize}}\, \frac{1}{2} \sum_{(i,j) \in \mathcal{O}} \left[(\boldsymbol{P})_{(i,j)} - (\boldsymbol{M})_{(i,j)} \right]^2 + \lambda \|\boldsymbol{P}\|_*. \tag{11}$$

With a sufficient number of observed entries, which depends on the rank and the incoherence of $\boldsymbol{P}$, the desired map can be reconstructed reliably. When the map area is large, the rank of the map matrix may increase, as the power distribution may become decorrelated. In this case, local matrix completion on the submatrices of $\boldsymbol{P}$ may be a viable approach [24]. The matrix completion idea can also be extended to tensors, when the maps in a 3-D space are desired [25] or other domains such as the time and frequency are considered together with space [26].

6) Dictionary Learning: When it is desired to capture the temporal variations of the power map, e.g., to exploit the unused spectral resources over both time and space, it is useful to learn a *library* of power maps, from which the suitable one can be chosen to explain the power distribution at a given time. Dictionary learning is an unsupervised learning method that seeks a set of possibly overcomplete bases, so that the data can be represented via a linear combination of a small number of basis vectors from the dictionary. Dictionary learning has been successfully applied to various problems in signal processing and machine learning.

Denote the power measurements at N sensors at time t as $\boldsymbol{m}(t) := [m_1(t), \ldots, m_N(t)]^\top$ for $t = 1, \ldots, T$. Dictionary learning postulates that $\{\boldsymbol{m}(t)\}$ can be represented using a dictionary $\boldsymbol{D} \in \mathbb{R}^{N \times Q}$ as $\boldsymbol{m}(t) \approx \boldsymbol{D}\boldsymbol{s}(t)$, where $\boldsymbol{s}(t) \in \mathbb{R}^Q$ is a *sparse* vector of coefficients for the measurements at time t. The columns of $\boldsymbol{D}$ are called the *atoms*. Collecting the data samples into a matrix $\boldsymbol{M} := [\boldsymbol{m}(1), \ldots, \boldsymbol{m}(T)] \in \mathbb{R}_+^{N \times T}$, one can appreciate that dictionary learning can

also be viewed as matrix factorization: $\boldsymbol{M} \approx \boldsymbol{D}\boldsymbol{S}$, where $\boldsymbol{S} := [\boldsymbol{s}(1), \ldots, \boldsymbol{s}(T)]$ is a sparse matrix. There are various optimization formulations to learn $\boldsymbol{D}$ from $\boldsymbol{M}$; see [27] and the references therein.

In the present context of power map estimation, consider the case where the sensors do not report their measurements every time, due to energy-saving sleep modes or congested signaling channels. Thus, the network controller must apply an appropriate interpolation technique to estimate the missing observations. A helpful piece of side information is the topology of the network of sensors, which is typically maintained for various network control tasks such as routing. To leverage this topology information, let $\boldsymbol{A} \in \{1, 0\}^{N \times N}$ be the adjacency matrix of the network topology, i.e., the (n, n')-th entry of $\boldsymbol{A}$ is equal to 1 if nodes n and n' are neighbors, and 0 otherwise. The Laplacian matrix $\boldsymbol{L}$ is defined as $\boldsymbol{L} := \mathrm{diag}\{\boldsymbol{A}\boldsymbol{1}\} - \boldsymbol{A}$, where $\boldsymbol{1}$ is the all-one vector. At each time t, a subset $\mathcal{N}^{obs}(t) \subset \mathcal{N} := \{1, \ldots, N\}$ of sensors acquire power measurements, which are stacked in vector $\boldsymbol{m}^{obs}(t) \in \mathbb{R}_+^{|\mathcal{N}^{obs}(t)|}$. Also, let $\boldsymbol{O}(t)$ denote the matrix consisting of the n-th row of an $N \times N$-identity matrix, where $n \in \mathcal{N}^{obs}(t)$. Then, upon defining

$$f(\boldsymbol{s}, \boldsymbol{D}; \boldsymbol{m}^{obs}(t), \boldsymbol{O}(t)) := \frac{1}{2}\|\boldsymbol{m}^{obs}(t) - \boldsymbol{O}(t)\boldsymbol{D}\boldsymbol{s}\|_2^2 + \lambda_s \|\boldsymbol{s}\|_1 + \frac{\lambda_L}{2}\boldsymbol{s}^\top \boldsymbol{D}^\top \boldsymbol{L}\boldsymbol{D}\boldsymbol{s} \tag{12}$$

the dictionary can be learned via

$$\hat{\boldsymbol{D}} := \underset{\boldsymbol{D} \in \mathcal{D}, \{\boldsymbol{s}(t)\}}{\arg\min} \sum_{t=1}^{T} f(\boldsymbol{s}(t), \boldsymbol{D}; \boldsymbol{m}^{obs}(t), \boldsymbol{O}(t)) \tag{13}$$

where $\mathcal{D} := \{[\boldsymbol{d}_1, \ldots, \boldsymbol{d}_Q] \in \mathbb{R}^{N \times Q} : \|\boldsymbol{d}_q\|_2^2 \leq 1,\ q = 1, \ldots, Q\}$. The first term in (12) promotes the fitness of the reconstruction to the training datum in a LS sense, and the second term with an adjustable weight $\lambda_s > 0$ is an ℓ_1-norm-based regularizer encouraging sparsity in $\boldsymbol{s}$, and the third term with the weight $\lambda_L \geq 0$ captures a priori information that the power levels at the neighboring sensor nodes should be similar, since it holds that $\boldsymbol{v}^\top \boldsymbol{L}\boldsymbol{v} = \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N} a_{nn'}(v_n - v_{n'})^2$ for any $\boldsymbol{v} \in \mathbb{R}^N$. The problem can be solved efficiently via a block coordinate descent (BCD) algorithm [28]. Once the dictionary $\hat{\boldsymbol{D}}$ is obtained, given a (new) set of measurements $\bar{\boldsymbol{m}}^{obs}$ and the corresponding observation matrix $\bar{\boldsymbol{O}}$ (corresponding to the observation set $\bar{\mathcal{N}}^{obs}$), one first finds the sparse coefficients by solving $\bar{\boldsymbol{s}} := \arg\min_{\boldsymbol{s}} f(\boldsymbol{s}, \hat{\boldsymbol{D}}; \bar{\boldsymbol{m}}^{obs}, \bar{\boldsymbol{O}})$. Then, the missing power levels for sensors $n \in \bar{\mathcal{N}}^{miss} := \mathcal{N} \backslash \bar{\mathcal{N}}^{obs}$ can be obtained by first reconstructing the whole $\hat{\bar{\boldsymbol{m}}} = \hat{\boldsymbol{D}}\bar{\boldsymbol{s}}$ and extracting the entries $\{\hat{\bar{m}}_n\}_{n \in \bar{\mathcal{N}}^{miss}}$. In [28], an online dictionary learning algorithm is also derived, as well as application of the framework for predicting future power levels. An online and distributed algorithm for a related formulation has been proposed in [29].

7) Deep Learning: A (possibly deep) neural network is a function $g_{\boldsymbol{w}}$ that can be expressed as the composition of more elementary functions called *layers*. These functions are parameterized by a vector $\boldsymbol{w}$ and *learning* or *training* means finding a suitable $\boldsymbol{w}$, typically so that $g_{\boldsymbol{w}}$ fits a certain data set. Relative to other generic estimators such as kernel-based estimators, deep neural networks (DNNs) feature a large learning capacity, can be trained efficiently using stochastic optimization methods, and excel at exploiting spatial structure when utilizing *convolutional layers*, in which case the DNN is called *convolutional neural network* (CNN).

As described next, there are multiple possibilities to use DNNs for occupancy map estimation.

a) Pointwise DNN Estimators: Possibly the simplest approach to apply a DNN is along the same lines as with kernel methods. The input to the network may be the sensor location and the output can be a single scalar that represents the power at that location. This approach has been pursued in [15], where the input is encoded using the spherical coordinates of the sensor relative to a coordinate system located at the (single) transmitter. Since the dimensionality of the input is small, the network architecture may be kept simple and the resulting estimator is not affected by the so-called *curse of dimensionality* [20, Sec. 4.3]. However, this input encoding is not amenable to CNN architectures, which therefore relinquishes their full potential to exploit spatial information. Besides, the network needs to be trained for each specific environment and, therefore, it does not learn from experience, that is, it cannot benefit from measurements collected in multiple scenarios, such as different cities.

b) Local DNN Estimators: To alleviate the limitations of pointwise DNN estimators, one possibility is to replace the aforementioned network input with a collection of matrices that capture information about the local environment of the sensor. These matrices, typically stacked as slabs of a tensor, can be thought of as *local maps* centered at the sensor. One of such maps may contain the distance from each point in a grid of points around the sensor to the transmitter or to the sensor [16], [17]. It is also possible to include a local terrain map that indicates the altitude of the terrain at each grid point around the sensor. Further kinds of local maps include building indicator maps [30], building height maps [31], [32], or foliage maps [31]. One could even think of using a picture of the surroundings of the sensor as a local map. Fig. 6 provides an illustration of this kind of setup.

This input format lends itself to CNN architectures that leverage spatial information in the vicinity of the sensor to predict the received power. To be able to learn across environments where the transmitters use possibly different transmit power, one can set the output of the network to

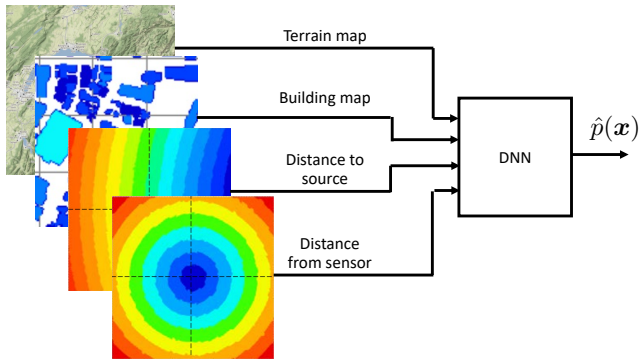


Fig. 6: Illustration of a local DNN estimator, which provides $\hat{p}(\boldsymbol{x})$ at a single $\boldsymbol{x}$.

be the gain between each transmitter and the sensor and work out the received power afterwards. This effectively sets this approach halfway between occupancy and propagation map estimation.

The main limitation of this approach is that it requires knowledge of the locations (and transmit power if one wishes to estimate the gain) of all transmitters and the measurements should be obtained separately for each transmitter. Besides, it only exploits information in the surroundings of the sensor but, in practice, obstacles or scatterers far away from the sensor may also affect the received power significantly. In addition, networks designed in this way provide the received power (or channel gain) only at a single location per evaluation (also known as forward pass). If the goal is to construct an entire map, the estimator needs to be evaluated repeatedly for a (possibly large) number of points on a grid, which may result in significant computational complexity.

c) Global DNN Estimators: To accommodate global, rather than local, environment information, one can create a regular grid across the region where the map needs to be constructed and formulate the RME problem as a matrix or tensor completion task [18], [30]–[33].

To this end, each measurement is associated with the nearest grid point and a matrix is constructed with an entry per grid point. If a single measurement is assigned to a grid point, then the corresponding entry contains the value measured there. If multiple measurements are assigned to a given grid point, the corresponding matrix entry contains their average. Those points with no associated measurements can be filled with zeros [33], [34], or with a constant value [32]. The resulting matrix condenses the measurement information in the entire environment and can be fed to the network to produce another matrix of the same size whose entries are the desired map values. However, this forces the network to learn the special meaning of the filling value. For

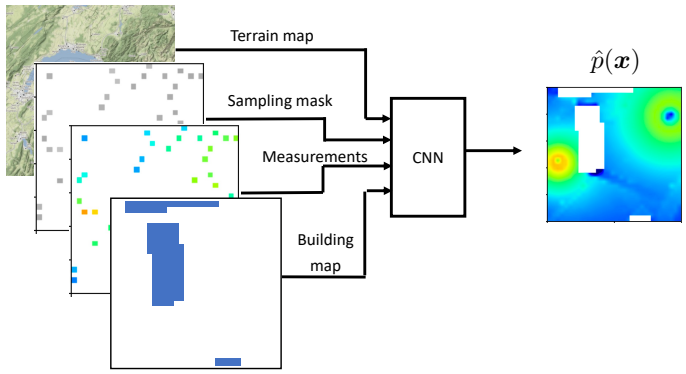


Fig. 7: Illustration of a global DNN estimator, which provides $\hat{p}(\boldsymbol{x})$ for all values of $\boldsymbol{x}$ on a grid.

example, if the filling value is 0, then the network needs to learn that 0 means no measurement, whereas 0.0001 means a small measured power. To avoid learning such discontinuities, a sampling mask is appended to the input in [30], [35], which is a matrix whose entries equal 1 at the measurement locations and 0 elsewhere.

Other maps with side information can be stacked to the aforementioned measurement matrix and sampling mask to form an input tensor to the network. The same kind of maps as for local DNN estimators (terrain, building, vegetation, etc) can be used, but now these maps are *global* since they describe the entire region of interest. A global DNN estimator is illustrated in Fig. 7.

The most common architectures are autoencoders [30], [34] and UNets [18], [32]. The motivation for the former is described in Box 1.

Besides the fact that global DNN estimators exploit information about the entire environment, the formulation they adopt is amenable to CNN architectures. In addition, unlike local DNN estimators, a single evaluation (forward pass) of the DNN here suffices to construct the entire map. Since each training instance will generally correspond to a different environment, these approaches naturally learn from experience.

The downside is that collecting a sufficiently large data set to train such a network may be challenging. To alleviate that difficulty, one may resort to data augmentation or to generate data using ray-tracing simulators; see e.g. [30]. Another limitation deals with the spatial resolution of the constructed maps. If one wishes a high resolution, then the density of the grid needs to be increased, which may significantly affect computational complexity.

Box 1: **Manifold structure of power maps.**

Autoencoder networks are attuned to situations where the data lies on a low-dimensional manifold embedded in a high-dimensional space. To see that this is the case of radio maps, consider the values of a power map produced by 2 sources radiating with a fixed height and power in free space. A data set can be generated where each map is obtained by placing the sources at random locations on the horizontal plane. Each map is therefore uniquely identified by the 4 scalars corresponding to the locations of the sources. If the maps are defined on a 32×32 grid, they comprise $32^2 = 1024$ points, which means that these maps lie on a manifold of dimension 4 embedded in a space of dimension 1024.

This observation is corroborated in [30] by training an autoencoder on the aforementioned data set. The autoencoder is the concatenation of an encoder and a decoder. The encoder takes a 32×32 map and produces a code vector $\boldsymbol{\lambda}$ of length 4 named *code*. The decoder takes this vector at its input and aims at reconstructing the original 32×32 map. For properly trained encoder and decoder, the output of the decoder resembles closely the input of the encoder, which means that the code effectively condenses the information of the map in just 4 numbers.

Each value of the code identifies a point in the manifold. The top panel of Fig. 8 shows the output of the decoder when its input equals the average of the codes associated with each map in the data set. The rest of panels show the output of the decoder applied to the result of perturbing the entries of this average code indicated by $\mathcal{S}$ by an amount equal to the standard deviation of that entry across the data set. This procedure yields different points in the manifold. All panels approximately correspond to maps of the kind composing the data set, which supports the manifold hypothesis.

If propagation does not take place in free space or if the power or height of the sources is variable, a longer code needs to be utilized to capture all information in the maps. Experiments with other data sets reveal that, in presence of propagation phenomena such as shadowing and fading, radio maps lie *close* to a manifold of low dimension [30].

B. Estimation of PSD Maps

As described earlier, many applications demand PSD maps, which describe how power distributes not only across space but also across the frequency domain. To estimate a PSD map $p(\boldsymbol{x}, f)$, most schemes assume that the sensors measure the power that they receive at a set of fre-

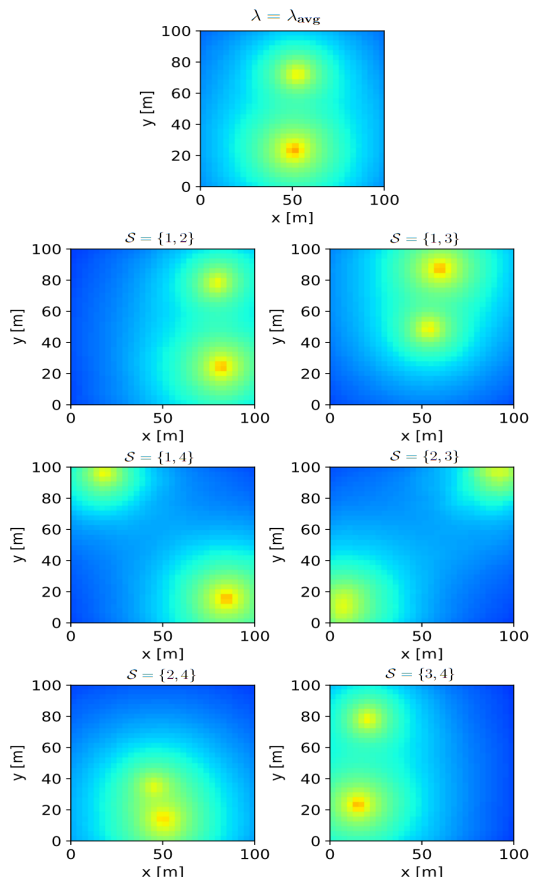


Fig. 8: Decoder outputs for the average and perturbed average codes obtained with an autoencoder with code length 4 [30]. See Box 1. [This figure should go inside Box 1.]

quencies $f_1, \ldots, f_{N_f}$. The n-th measurement is therefore a vector $\boldsymbol{m}_n = [\tilde{p}(\boldsymbol{x}_n, f_1), \ldots, \tilde{p}(\boldsymbol{x}_n, f_{N_f})]^\top$, where $\tilde{p}(\boldsymbol{x}, f)$ denotes the measured power at location $\boldsymbol{x}$ and frequency f, possibly obtained using a periodogram or similar technique, such as the well-known Welch method. Relying on these measurements, the goal is to obtain a PSD map estimate $\hat{p}$ such that $\hat{p}(\boldsymbol{x}, f)$ is as close to $p(\boldsymbol{x}, f)$ as possible. To this end, several alternatives are explored next.

1) Separate Estimation per Frequency: The most immediate possibility is to consider each frequency separately and essentially decompose the problem of estimating a PSD map at N_f frequencies as N_f problems of estimating a single power map [34]. More specifically, the n-th power map is estimated from the measurements $p(\boldsymbol{x}_1, f_n), \ldots, p(\boldsymbol{x}_N, f_n)$ using the techniques described earlier. The main limitation of this approach is that it totally disregards any structure in the frequency domain, which therefore makes it more sensitive to measurement noise than other schemes explored later. On the upside, these approaches are simple and do not require any

assumptions about the channel or transmit PSDs. Besides, a twofold benefit arises when it comes to data-driven estimators such as the deep learning estimators described in a previous section: on the one hand, if a data set of measurements across space and frequency is given and propagation phenomena can be assumed to affect all frequencies in a similar way, considering each frequency separately will increase the number of training examples by a factor of N_f. On the other, the number of inputs of the estimator is reduced by the same factor, which results in a decrease of the number of parameters to be learned [30, Sec. III-C1].

2) Estimation in Narrowband Channels: When the width of the band of interest is small or moderate, it makes sense to assume that the channel is not frequency selective [26], [35]. This means that the true PSD map can be written as $p(\boldsymbol{x}, f) = \sum_s h_s(\boldsymbol{x}) p_s^{\text{TX}}(f)$, where $h_s(\boldsymbol{x}) = h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x})$ is the channel gain from the s-th transmitter to location $\boldsymbol{x}$. In this way, the measurements essentially provide N_f noisy linear combinations of the S functions $h_1(\boldsymbol{x}), \ldots, h_S(\boldsymbol{x})$. Whenever $N_f \gg S$, this is therefore an effective means to exploit structure in the frequency domain and, therefore, reduce sensitivity to measurement noise.

One of the main benefits of this approach is that no knowledge of the transmit PSD is required, as it can be estimated using tools such as non-negative matrix factorization [35].

3) Estimation in Wideband Channels: If one is interested in estimating a PSD map over a wide frequency band, assuming that the channel is frequency flat is not generally realistic. However, frequency-domain structure can still be exploited upon realizing that the transmit PSD within each subchannel is typically known up to a scaling factor, namely the power. Specifically, note that the PSD of communication waveforms is typically determined by spectrum regulations and standards, which specify bandwidth, carrier frequencies, transmission masks, roll-off factors, number of sub-carriers, and so forth; see [36] and references therein. This means that the transmit PSD of a source can be approximated by a *basis expansion model* as $p_s^{\text{TX}}(f) = \sum_c \beta_{s,c} \phi_c(f)$, where ϕ_c denotes the c-th basis function and $\beta_{s,c}$ is a nonnegative quantity. This decomposition is illustrated in Fig. 9.

The received PSD at a location $\boldsymbol{x}$, which in general is given by $p(\boldsymbol{x}, f) = \sum_s h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, f) p_s^{\text{TX}}(f)$ with $h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, f)$ the channel gain at frequency f, can be expressed for the aforementioned parameterization as $p(\boldsymbol{x}, f) = \sum_c \sum_s \beta_{s,c} h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, f) \phi_c(f)$. If the bandwidth of the basis functions is small relative to the entire band, it is reasonable to assume that h is approximately frequency flat in the bandwidth of each basis function. This yields $h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, f)\phi_c(f) \approx h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, \tilde{f}_c)\phi_c(f)$, where $\tilde{f}_c$ is the central frequency of the c-th basis function. With this approximation, one can

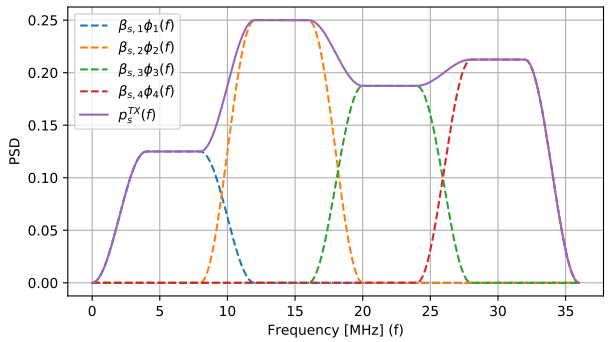


Fig. 9: A basis expansion model can be used to decompose a PSD as a linear combination of functions in a basis. In this case, the basis functions are square root raised cosine functions.

write $p(\boldsymbol{x}, f) = \sum_c p_c(\boldsymbol{x})\phi_c(f)$, where $p_c(\boldsymbol{x}) = \sum_s h(\boldsymbol{x}_s^{\text{TX}}, \boldsymbol{x}, \tilde{f}_c)\beta_{s,c}$ constitutes the power captured by the c-th basis function at location $\boldsymbol{x}$.

Observe that introducing the basis expansion model has reduced the problem of estimating N_f power maps to the problem of estimating the $C \ll N_f$ power maps $p_1, \ldots, p_C$. Clearly, the smaller C, the smaller the sensitivity to measurement noise. The approaches in the previous two subsections are extreme cases of the choice of C: whereas the former is recovered for $C = N_f$, the latter is obtained for $C = 1$.

To estimate a PSD map, the aforementioned technique can be used in combination with virtually any of the approaches for power map estimation described earlier; see e.g. [2], [7], [13], [14]. A recent example is [30], where a basis expansion model is used to design the last layer of a DNN for RME.

V. Estimation of Propagation Maps

The goal here is to predict the channel attenuation for links between arbitrary pairs of locations where no sensors may be deployed [8]. As indicated earlier, the n-th measurement is collected by a pair of sensors, one at location $\boldsymbol{x}_n$ and the other at location $\boldsymbol{x}'_n$. The channel gain of the link between them is measured using possibly pilot signals. The resulting measurement can be expressed as $m_n = h(\boldsymbol{x}_n, \boldsymbol{x}'_n) + z_n$, where h is the true map and z_n represents measurement noise. The RME problem is to obtain an estimate $\hat{h}$ of h given $\{(\boldsymbol{x}_n, \boldsymbol{x}'_n, m_n)\}_{n=1}^N$. A good estimation algorithm should have good generalization properties, meaning that $\hat{h}(\boldsymbol{x}, \boldsymbol{x}') \approx h(\boldsymbol{x}, \boldsymbol{x}')$ for all location pairs $(\boldsymbol{x}, \boldsymbol{x}')$, even for those between which no measurement has been collected.

Like occupancy RME, estimating propagation maps is also a function estimation problem. Techniques described earlier for estimating occupancy maps, such as parametric and kernel-based estimators, carry over with minor modifications. The difference is that now the function to be estimated depends on two location inputs rather than on a single one. If $\boldsymbol{x}$ denotes a location in 3D space, it is clear that h is a function of 6 scalar inputs, namely the entries of $\boldsymbol{x}$ and $\boldsymbol{x}'$. This means that the number of measurements necessary to attain a given accuracy may be considerably greater than for estimating an occupancy map, which is a function of just 3 scalar inputs. This is a manifestation of the so-called *curse of dimensionality* [20, Sec. 4.3]. As described in the rest of this section, a number of algorithms have been tailor-made for propagation RME to alleviate such a difficulty.

A. Non-tomographic Approaches

In the non-tomographic approaches, channel gains are directly modeled based on basic wireless propagation models without postulating an underlying auxiliary map. In order to maintain tractability, however, the RME problem is often simplified by fixing one end of a link. For example, a set of maps $\{h_n(\boldsymbol{x}) := h(\boldsymbol{x}, \boldsymbol{x}_n)\}_n$ are pursued for fixed locations $\{\boldsymbol{x}_n\}_{n=1}^N$, where the spectrum sensors are located. The individual functions $\{h_n(\boldsymbol{x})\}$ can be estimated using the various methods employed for the occupancy maps [8], [37].

For a concrete example, and also extending the RME problem to include the time domain to capture the temporal variation of the channel gains, let us consider the channel gain $h_n(\boldsymbol{x}, t)$ between locations $\boldsymbol{x}$ and $\boldsymbol{x}_n$ at time t [8]. Suppose that the small-scale fading effect has been averaged out, allowing $h_n(\boldsymbol{x}, t)$ in dB to be expressed as

$$h_n(\boldsymbol{x}, t) = h_n^{\text{PL}}(\boldsymbol{x}) - a_n^{\text{SF}}(\boldsymbol{x}, t) \tag{14}$$

where $h_n^{\text{PL}}(\boldsymbol{x})$ is the free space gain from $\boldsymbol{x}_n$ to $\boldsymbol{x}$ and $a_n^{\text{SF}}(\boldsymbol{x}, t)$ is the shadow fading in dB. Note that $h_n^{\text{PL}}(\boldsymbol{x})$ is known since both $\boldsymbol{x}_n$ and $\boldsymbol{x}$ as well as the antenna gains are known. Thus, the main issue is to track the time-varying shadow fading map $a_n^{\text{SF}}(\boldsymbol{x}, t)$. To do this, shadow fading measurements are needed, which can be obtained by subtracting the transmit power and the path loss from the received power measurement. Thus, at time t, the sensor at $\boldsymbol{x}_n$ obtains noisy measurements $\{\breve{a}_n^{\text{SF}}(\boldsymbol{x}_j, t)\}_j$ of shadow fading using the pilot signals sent from the radios at $\{\boldsymbol{x}_j\}_{j \in \mathcal{N}_{-n}}$, where $\mathcal{N}_{-n} := \{1, \ldots, n-1, n+1, \ldots, N\}$, which can be expressed as

$$\breve{a}_n^{\text{SF}}(\boldsymbol{x}_j, t) = a_n^{\text{SF}}(\boldsymbol{x}_j, t) + z_n(\boldsymbol{x}_j, t), \;\; j \in \mathcal{N}_{-n} \tag{15}$$

where $z_n(\boldsymbol{x}_j,t)$ is zero-mean Gaussian measurement noise. Upon defining $\breve{\boldsymbol{a}}_n^{\rm SF}(t) := [\breve{a}_n^{\rm SF}(\boldsymbol{x}_1,t),\ldots,\breve{a}_n^{\rm SF}(\boldsymbol{x}_{n-1},t),\breve{a}_n^{\rm SF}(\boldsymbol{x}_{n+1},t),\cdots,\breve{a}_n^{\rm SF}(\boldsymbol{x}_N,t)]^\top$, our goal is to estimate $h_n(\boldsymbol{x},t)$ for arbitrary $\boldsymbol{x}$, based on the cumulative measurements $\breve{\mathcal{A}}_n^{\rm SF}(t) := \{\breve{\boldsymbol{a}}_n^{\rm SF}(\tau)\}_{\tau=1}^t$ up to time t.

The problem can be tackled in the framework of kriged Kalman filtering, also known as space-time Kalman filtering [38]. Employing the log-normal shadowing model, it is assumed that $a_n^{\rm SF}(\boldsymbol{x},t)$ is a Gaussian process, whose spatio-temporal dynamics can be described by [8], [38]

$$a_n^{\rm SF}(\boldsymbol{x},t) = \mu_n^{\rm SF}(\boldsymbol{x},t) + \nu_n(\boldsymbol{x},t) \tag{16}$$

$$\mu_n^{\rm SF}(\boldsymbol{x},t) = \int w_n(\boldsymbol{x},\boldsymbol{u})\mu_n^{\rm SF}(u,t-1)d\boldsymbol{u} + \eta_n(\boldsymbol{x},t) \tag{17}$$

where $\mu_n^{\rm SF}(\boldsymbol{x},t)$ is the spatio-temporally correlated component with $w_n(\boldsymbol{x},\boldsymbol{u})$ capturing the interaction of the component at location $\boldsymbol{x}$ at time t and at location $\boldsymbol{u}$ at time $(t-1)$; and $\nu_n(\boldsymbol{x},t)$ and $\eta_n(\boldsymbol{x},t)$ are spatially correlated but temporally white zero-mean Gaussian processes. Process $\nu_n(\boldsymbol{x},t)$ is uncorrelated with $z_n(\boldsymbol{u},\tau)$, and $\eta_n(\boldsymbol{x},t)$ is uncorrelated with $\nu_n(\boldsymbol{u},\tau)$ or $z_n(\boldsymbol{u},\tau)$, for all $\boldsymbol{u}$ and τ. Moreover, $\mathbb{E}\{\nu_n(\boldsymbol{x},t)\mu_n^{\rm SF}(\boldsymbol{u},t)\} = \mathbb{E}\{\eta_n(\boldsymbol{x},t)\mu_n^{\rm SF}(\boldsymbol{u},t-1)\} = 0$ for all $\boldsymbol{x},\boldsymbol{u}$ and t. For stability, it is also assumed that $|\int w_n(\boldsymbol{x},\boldsymbol{u})d\boldsymbol{u}| < 1$.

Since the state-space model in (16)–(17) is infinite-dimensional, for tractability, adopt a basis expansion approach. For a set of K orthonormal bases $\{\psi_k(\boldsymbol{x})\}_k$, $\mu_n^{\rm SF}$ and w_n are approximated as $\mu_n^{\rm SF}(\boldsymbol{x},t) \approx \sum_{k=1}^K \alpha_{n,k}(t)\psi_k(\boldsymbol{x})$ and $w_n(\boldsymbol{x},t) \approx \sum_{k=1}^K \beta_{n,k}(t)\psi_k(\boldsymbol{x})$, respectively, where $\{\alpha_{n,k}(t)\}$ and $\{\beta_{n,k}(t)\}$ are the expansion coefficients. By substituting these to (15)–(17) and sampling the equations at $\{\boldsymbol{x}_j\}_{j\in\mathcal{N}_{-r}}$, one can obtain the state-space model in finite dimensions as

$$\breve{\boldsymbol{a}}_n^{\rm SF}(t) = \boldsymbol{\Psi}_n\boldsymbol{\alpha}_n(t) + \boldsymbol{\nu}_n(t) + \boldsymbol{z}_n(t) \tag{18}$$

$$\boldsymbol{\alpha}_n(t) = \boldsymbol{\Psi}_n^\dagger\boldsymbol{B}_n\boldsymbol{\alpha}_n(t-1) + \boldsymbol{\Psi}_n^\dagger\boldsymbol{\eta}_n(t). \tag{19}$$

Here, let $\boldsymbol{\psi}(\boldsymbol{x}) := [\psi_1(\boldsymbol{x}),\ldots,\psi_K(\boldsymbol{x})]^\top$, $\boldsymbol{\alpha}_n(t) := [\alpha_{n,1}(t),\ldots,\alpha_{n,K}(t)]^\top$, and $\boldsymbol{\beta}_n(t)$ is defined similarly. $\boldsymbol{B}_n$ and $\boldsymbol{\Psi}_n$ are matrices constructed by arranging $\boldsymbol{\beta}_n(\boldsymbol{x}_j)^\top$ and $\boldsymbol{\psi}(\boldsymbol{x}_j)^\top$ as rows, respectively, for $j \in \mathcal{N}_{-n}$. Vectors $\boldsymbol{\nu}_n(t)$, $\boldsymbol{z}_n(t)$, and $\boldsymbol{\eta}_n(t)$ are similarly constructed from $\{\nu_n(\boldsymbol{x}_j,t)\}_{j\in\mathcal{N}_{-n}}$, $\{z_n(\boldsymbol{x}_j,t)\}_{j\in\mathcal{N}_{-n}}$, and $\{\eta_n(\boldsymbol{x}_j,t)\}_{j\in\mathcal{N}_{-n}}$, respectively.

Based on (18)–(19), the MMSE estimate $\hat{\boldsymbol{\alpha}}_n(t|t)$ of $\boldsymbol{\alpha}_n(t)$ given $\breve{\mathcal{A}}_n^{\rm SF}(t)$ can be obtained via ordinary Kalman filtering, from which the temporally dynamic component $\mu_n^{\rm SF}(\boldsymbol{x},t)$ can be estimated as $\mathbb{E}\{\mu_n^{\rm SF}(\boldsymbol{x},t)|\breve{\mathcal{A}}_n^{\rm SF}(t)\} = \boldsymbol{\psi}(\boldsymbol{x})^\top\hat{\boldsymbol{\alpha}}_n(t|t)$. To capture $\nu_n(\boldsymbol{x},t)$ as well, one needs to employ a Kriging estimator, which is a linear spatial interpolator. Overall, the MMSE estimate $\hat{a}_n^{\rm SF}(\boldsymbol{x},t) := \mathbb{E}\{a_n^{\rm SF}(\boldsymbol{x},t)|\breve{\mathcal{A}}_n^{\rm SF}(t)\}$ can be obtained exploiting the correlation structure [8]. Once

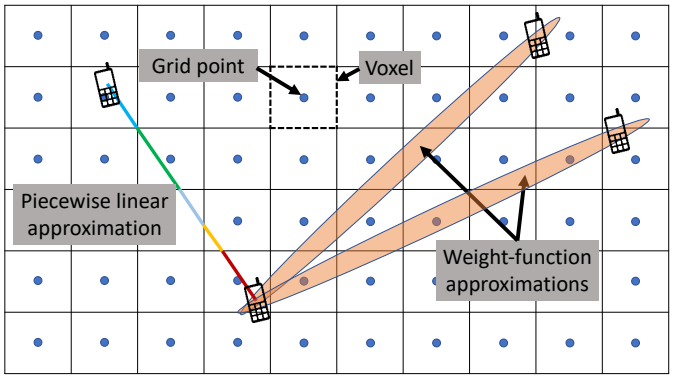


Fig. 10: Illustration of possible approximations of the tomographic integral.

$\hat{a}_n^{\text{SF}}(\boldsymbol{x}, t)$ is obtained, the channel gain map estimate $\hat{h}_n(\boldsymbol{x}, t)$ can be constructed as $\hat{h}_n(\boldsymbol{x}, t) = h_n^{\text{PL}}(\boldsymbol{x}) - \hat{a}_n^{\text{SF}}(\boldsymbol{x}, t)$; cf. (14).

B. Radio Tomographic Approaches

Most works on propagation RME rely on the so-called *radio tomographic* model; see e.g. [39], [40]. This model, whose name already suggests its similarity with the models used for tomographic imaging in other areas such as medical imaging, prescribes that the attenuation due to shadowing can be expressed in terms of the line integral of a non-negative function termed *spatial loss field* (SLF), which is specific to each propagation scenario [41]. Specifically, decompose the channel gain (expressed in logarithmic units) as $h = h^{\text{PL}} - a^{\text{SF}} - a^{\text{FF}}$, where h^{PL} is the path loss, a^{SF} is the so-called *shadow fading* attenuation, caused by obstructions in the propagation path of the radio wave, and a^{FF} is the *fast fading* attenuation, caused by the constructive and destructive interference among multiple paths through which the radio signal travels.

The scale of spatial variations of the last term is in the order of the wavelength, which for contemporary communication systems ranges in the order of a few millimeters or centimeters. Since this is much smaller than the typical error of localization systems such as GPS used to determine the measurement locations, estimating a^{FF} may be challenging in practice. Thus, it can be regarded as a perturbation and absorbed into the measurement noise. On the other hand, the term h^{PL} can be considered known since it is determined by the endpoints of the communication link as well as by the antenna gains and additional known factors.

These observations mean that estimating h amounts to estimating a^{SF}. The fundamental principle underpinning the radio tomography model is that closely located radio links exhibit similar

shadowing due to the presence of common obstructions. According to the tomographic model, the latter is given by [41]

$$a^{\mathrm{SF}}(\boldsymbol{x}, \boldsymbol{x}') = \frac{1}{\sqrt{\|\boldsymbol{x} - \boldsymbol{x}'\|}} \int_{\boldsymbol{x}}^{\boldsymbol{x}'} F(\bar{\boldsymbol{x}}) d\bar{\boldsymbol{x}}, \tag{20}$$

where $F(\bar{\boldsymbol{x}})$ represents the value of the SLF at location $\bar{\boldsymbol{x}}$ and quantifies how much a radio wave attenuates at $\bar{\boldsymbol{x}}$. As described later, the SLF needs to be estimated for each scenario. SLFs are not only useful for RME: they reveal the location of obstacles and, for this reason, find applications such as device-free passive localization [5], environmental monitoring for surveillance or intrusion detection [42], and through-the-wall imaging [43]. For example, these techniques may enable the police or emergency services to locate persons in burning buildings, survivors in rescue operations, or kidnappers in hostage situations.

In order to manipulate F, this function needs to be discretized, customarily by storing its values on a 2D or 3D regular grid of spatial locations. Given such a discretization, two main approaches have been utilized to approximate the integral in (20). The most common approximation replaces the line integral with a weighted sum of the values of the SLF on the grid points that lie inside an ellipse (or ellipsoid in 3D) whose foci are $\boldsymbol{x}$ and $\boldsymbol{x}'$. This is illustrated by the right two links on Fig. 10. The intuition behind this approximation is that the attenuation between two points should be heavily affected by the obstacles near the line of sight or, more specifically, within the so-called *Fresnel zone*, which is an ellipse whose geometry is dictated by the wavelength. Several functions have been proposed in the literature to assign weights to the grid points lying inside the ellipse; see e.g. [42] for an overview. While these functions are based on heuristic considerations, blind schemes have also been developed to learn the weight function from data [44].

Although this approximation is easy to implement, it suffers from certain limitations. First, the approximated $a^{\mathrm{SF}}(\boldsymbol{x}, \boldsymbol{x}')$ is not a continuous function of $\boldsymbol{x}$ and $\boldsymbol{x}'$ since a small change in either of these vectors may lead to a change in the set of grid points that lie within the ellipse. In fact, it may well happen that the approximation is just 0, as it occurs when the ellipse misses all grid points; see the left ellipse on Fig. 10. To attain a reasonable accuracy, this implies that the separation between grid points must be small enough, which in turn may result in a prohibitively large number of grid points.

This motivates an alternative approximation where the integral in (20) is approximated by the integral of a piecewise constant function, which takes the same value at every point inside each voxel [12]. The integral is therefore a linear combination of the values of the SLF at the voxels that the line of sight traverses. The coefficients in this linear combination correspond to

the distance traversed in each voxel. This is illustrated by the colored line on Fig. 10. It can be seen that this approximation involves less computational complexity than the one based on ellipsoids, it results in continuous approximations, and does not vanish unless the SLF vanishes. This relaxes the need for a dense grid, which is especially convenient when the region of interest has dimension 3, as occurs for instance in aerial communications [12].

Fixed $\boldsymbol{x}$ and $\boldsymbol{x}'$, both of the aforementioned approximations are linear functions of the values of F at the grid points. This means that the measurements are affine functions of these values, which means that the latter can be estimated, in principle, via (possibly non-negative) LS. However, this approach requires that the number of measurements is significantly larger than the number of grid points, which therefore poses a practical constraint. A possible alternative to relax these requirements is to adopt regularized estimators capable of solving underdetermined systems of equations [45]. Yet another possibility is to pursue a Bayesian approach, where the need for data is counteracted by the introduction of prior distributions; see e.g. [46]

VI. Spectrum Surveying

In certain applications such as network planning, it is necessary to deploy a sensor and use it to collect measurements with the purpose of constructing an occupancy map of a certain geographical region. Traditionally, a team of technicians would physically move to a set of locations to collect measurements. However, with the advances in mobile robotics of the last decade, it is now possible to equip an autonomous UAV with an on-board sensor and use it to collect the desired measurements. This is clearly more efficient in terms of time and personnel cost.

The question that arises is, therefore, where such an autonomous UAV must collect measurements. A common possibility adopted nowadays is to define a grid and collect a measurement at each grid point. The main limitation of such an approach is clearly that the UAV needs to traverse the entire grid, which will be highly time consuming, especially if the grid is reasonably dense. Besides inefficient, this is problematic since the battery capacity typically constrains the flight time to around 30 minutes. Thus, a more sensible approach would be to collect measurements at a small set of highly informative locations and apply an estimator such as those described so far to construct the entire map. To this end, the work in [19] proposes radio map estimators that not only provide the estimated map but also an uncertainty map that indicates how informative a measurement would be at each location given the measurements collected so far. Using such an uncertainty map, a route planning algorithm determines a trajectory through areas of high

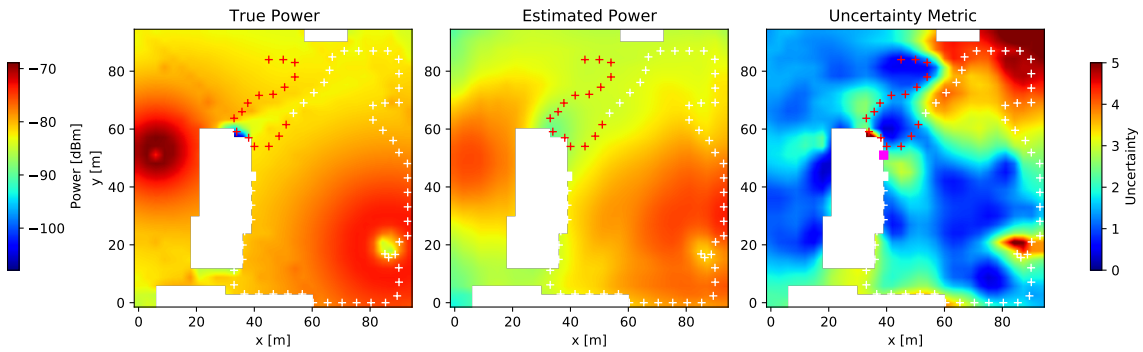


Fig. 11: Example of surveying operation with an autonomous UAV in an urban environment seen from above. White boxes denote buildings. Red and white crosses denote measurement locations.

uncertainty. This approach leads to estimates of a much higher quality in the same surveying time as alternative approaches such as the one described above, where the UAV collects measurements at all points of a grid. Conversely, a much shorter time is required to obtain an estimate of a target accuracy.

Fig. 11 illustrates an example of a surveying operation using a ray-tracing data set in a region of downtown Rosslyn, Virginia. The three panels show the UAV trajectory seen from above. White boxes correspond to space occupied by buildings, where no measurements can be taken. Red and white crosses denote measurement locations. The left-most panel shows the ground truth power map in a setup with two transmitters. The middle and right panels respectively show the estimated power map and the uncertainty map when only the measurements marked by red crosses have been collected. At that point in time, the UAV plans a trajectory through areas of high uncertainty, represented by white crosses. The estimator in this case is a global DNN estimator capable of learning the nature of propagation phenomena from a data set, as explained earlier.

VII. Practical Considerations

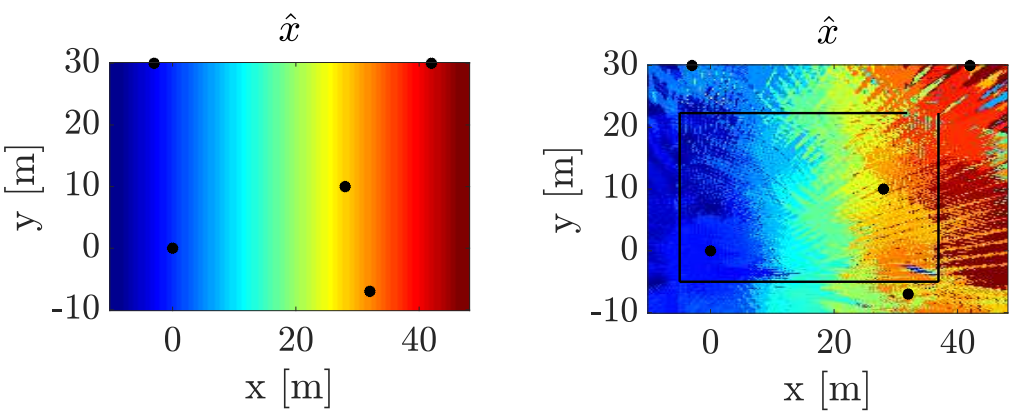
Following the theoretical exposition, this section will present a number of challenges for implementing the RME techniques in practical setups. Specifically, we will discuss the impact of the localization error and how to counter it via *location-free cartography*, which comprises techniques that essentially construct radio maps that are not indexed by spatial coordinates but by *channel-state information* (CSI) [47], [48]. Afterwards, we will explore decentralized

implementations [2], [39] and how to reduce the bandwidth required by the sensors to report measurements [7].

A. Localization Errors

The RME schemes described in the previous sections require accurate knowledge of the measurement locations. In practice, to know its location, a sensor needs to rely on localization systems such as GPS. The operation of these systems is generally as follows: a number of transmitters at known locations, typically at satellites or cellular base stations, regularly transmit signals termed *localization pilots*. Each sensor then measures certain *features* of these signals to estimate its location. For example, measuring the received signal strength or the delay provides information about the distance to the transmitters. An algorithm can then apply geometric principles to fuse these distances and produce a location estimate.

Unfortunately, this task may be challenged in practice due to propagation phenomena affecting the pilot signals. Specifically, multipath propagation, especially important in indoor and dense urban scenarios, distorts the pilot features and, consequently, the distance estimates obtained by the sensors. This may result in localization errors in the order of tens of meters.



(a) Free space.

(b) Scenario with 4 walls.

Fig. 12: The color of each point indicates the x-coordinate of the location estimate obtained by a sensor at that location. The black circles indicate the positions of the transmitters. The estimate accurately matches the true coordinate when there is no multipath; cf Fig. 12a (thus, Fig. 12a serves as a color bar). On the other hand, the estimation error is large in the presence of multipath; cf Fig. 12b. (Both figures were taken from [47].)

This phenomenon is illustrated in Fig. 12, which compares the x-coordinate of the location estimates in a scenario without multipath (Fig. 12a) and with multipath (Fig. 12b). The localization algorithm is based on the time difference of arrival between the pilot signals arriving from each pair of transmitters; see details in [47]. The poor quality of the location estimates in Fig. 12b hinders the application of the techniques presented so far.

The localization error therefore limits the resolution that can be achieved in RME. With typical errors in the order of a few meters at best, there is no hope that one can estimate fine details of a radio map such as those created by fast fading. This means that, in practice, radio maps can only capture phenomena that exert spatial changes in a scale larger than the localization error, such as shadow fading and path loss. In any case, it is worth noting that a relatively low localization error may not entail a loss of resolution since the latter can already be limited by the spatial density of the measurements. For example, if the separation between measurements is in the order of tens of meters, as may occur in practice, details in the order of meters cannot be reconstructed anyway regardless of how small the localization error is. For this reason, accurately mapping the fast-fading component in practice may be challenging: its coherence distance is much smaller than the distance between measurements.

In short, the schemes presented so far can be applied whenever the localization error is sufficiently small. When this is not the case, one needs to resort to the so-called *location-free (LocF) cartography* framework [47]. To introduce this framework, it is instructive to consider power maps. Recall that a power map p returns the power $p(\boldsymbol{x})$ received at a location $\boldsymbol{x}$. Since the location is the input variable, it is expected that localization errors propagate to the output of this function. The key realization is, therefore, that $\boldsymbol{x}$ is not a suitable variable to act as the "index" of the map.

To determine a convenient input variable, it is worth stepping back and noting that the location estimates are the result of applying a localization algorithm on the pilot features. A possibility is therefore to bypass this step and directly use pilot features to index the map. Since these features evolve more smoothly across space than the location estimates, a map can be effectively learned [47]. To evaluate the map at a certain location there are two approaches: if a sensor is present at that location, then it just needs to extract the features of the pilot signals. If no sensor is present at that location, one can still interpolate these features using the notion of channel charting [49].

The downside is that the number of scalar inputs of the map function will generally be larger,

which means that a larger number of measurements is required relative to the location-based approach with no localization error. Another difficulty is that the availability of each feature may depend on the location, since not all pilot signals can be received everywhere. However, algorithms that can cope with these misses have also been developed [47].

Finally, instead of relying on features extracted from localization pilot signals, it is also possible to index the map using features extracted from regular communication signals, for example by the signals broadcast by a cellular base station [48].

B. Decentralized Implementation

Different from conventional spectrum sensing techniques, which often assume a common spectrum occupancy over the entire area of interest, spectrum cartography accounts for spatial variability. Thus, it is necessary that the measurements are obtained at various locations $\{\boldsymbol{x}_n\}_{n=1}^N$ within the region, which then must be processed jointly. While this can be achieved in theory by collecting the measurements at a fusion center (FC) for centralized processing, the feedback overhead and the associated delay can be significant in practice. Moreover, the FC must operate with higher resource and security requirements. An alternative is to employ a distributed in-network processing, where the sensors collaboratively perform map estimation, each sensor $n \in \mathcal{N} := \{1, \ldots, N\}$ at $\boldsymbol{x}_n$ exchanging information only with the set of single-hop neighbors $\mathcal{N}_n$ [2], [8], [14], [39], [50]. The key idea is that the RME task often boils down to a regression problem of the form

$$\underset{\boldsymbol{\theta}}{\text{minimize}}\, \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta}\|_2^2 + \psi(\boldsymbol{\theta}) \tag{21}$$

where $\boldsymbol{y} \in \mathbb{R}^M$ and $\boldsymbol{X} \in \mathbb{R}^{M \times \Theta}$ represent the targets and the regressors, respectively, $\boldsymbol{\theta} \in \mathbb{R}^\Theta$ is the regression coefficients, and $\psi(\cdot)$ is a convex regularizer that captures prior information; see, e.g., (8). It is often the case that the data $\boldsymbol{X}$ and $\boldsymbol{y}$ consist of the collection of the data $\{\boldsymbol{X}_n\}$ and $\{\boldsymbol{y}_n\}$ from the individual sensors. That is, $\boldsymbol{y} = [\boldsymbol{y}_1^\top, \ldots, \boldsymbol{y}_N^\top]^\top$, where $\boldsymbol{y}_n \in \mathbb{R}^{M_n}$, $n \in \mathcal{N}$, with $\sum_{n=1}^N M_n = M$. Likewise, $\boldsymbol{X} = [\boldsymbol{X}_1^\top, \ldots, \boldsymbol{X}_N^\top]^\top$ with $\boldsymbol{X}_n \in \mathbb{R}^{M_n \times \Theta}$, $n \in \mathcal{N}$. Thus, (21) represents the centralized problem solved by a FC.

In order to solve (21) in a distributed manner, consider first an undirected graph $\mathcal{G} := (\mathcal{N}, \mathcal{E})$ representing the network of sensors, with the vertex set $\mathcal{N}$ and the edge set $\mathcal{E}$, where edge $(n, n') \in \mathcal{E}$ whenever sensors n and n' are single-hop neighbors. If $\mathcal{G}$ is connected, i.e., there is a possibly multi-hop path between every pair of sensors, it can be easily shown that (21) is

equivalent to

$$\underset{\{\boldsymbol{\theta}_n,\boldsymbol{\gamma}_n,\boldsymbol{\gamma}_n^{n'}\}}{\text{minimize}} \sum_{n=1}^{N} \left[\frac{1}{2}\|\boldsymbol{y}_n - \boldsymbol{X}_n\boldsymbol{\gamma}_n\|_2^2 + \frac{1}{N}\psi(\boldsymbol{\theta}_n)\right] \tag{22}$$

$$\text{subject to } \boldsymbol{\gamma}_n = \boldsymbol{\theta}_n,\ n \in \mathcal{N} \tag{23}$$

$$\boldsymbol{\theta}_n = \boldsymbol{\gamma}_{(n,n')} = \boldsymbol{\theta}_{n'},\ n' \in \mathcal{N}_n,\ n \in \mathcal{N}. \tag{24}$$

where $\{\boldsymbol{\gamma}_n\}$ and $\{\boldsymbol{\gamma}_{(n,n')}\}$ are auxiliary variables. Per (23), $\boldsymbol{\gamma}_n$ is just a copy of $\boldsymbol{\theta}_n$. $\{\boldsymbol{\gamma}_{(n,n')}\}$ facilitate the derivation of simple update rules and are eventually eliminated. The decentralized algorithm can be derived by applying the alternating direction method of multipliers (ADMM) to (22)–(24). Following steps similar to those in [14, App. D], one can obtain the decentralized update rules for iteration k as

$$\boldsymbol{u}_n^{[k]} = \boldsymbol{u}_n^{[k-1]} + \rho \sum_{n' \in \mathcal{N}_n} \left(\boldsymbol{\theta}_n^{[k]} - \boldsymbol{\theta}_{n'}^{[k]}\right) \tag{25}$$

$$\boldsymbol{\lambda}_n^{[k]} = \boldsymbol{\lambda}_n^{[k-1]} + \rho \left(\boldsymbol{\theta}_n^{[k]} - \boldsymbol{\gamma}_n^{[k]}\right) \tag{26}$$

$$\boldsymbol{\theta}_n^{[k+1]} = \arg\min_{\boldsymbol{\theta}} \frac{1}{Nc_n}\psi(\boldsymbol{\theta}_n) + \frac{1}{2}\|\boldsymbol{\theta}_n - \boldsymbol{a}_n\|_2^2 \tag{27}$$

$$\boldsymbol{\gamma}_n^{[k+1]} = \left(\rho \boldsymbol{I}_\Theta + \boldsymbol{X}_n^\top \boldsymbol{X}_n\right)^{-1} \left(\boldsymbol{X}_n^\top \boldsymbol{y}_n + \rho \boldsymbol{\theta}_n^{[k+1]} + \boldsymbol{\lambda}_n^{[k]}\right) \tag{28}$$

where $\rho > 0$, $c_n := \rho(1 + 2|\mathcal{N}_n|)$, and

$$\boldsymbol{a}_n := \frac{1}{c_n}\left(\rho \sum_{n' \in \mathcal{N}_n} \left(\boldsymbol{\theta}_n^{[k]} + \boldsymbol{\theta}_{n'}^{[k]}\right) + \rho \boldsymbol{\gamma}_n^{[k]} - \boldsymbol{u}_n^{[k]} - \boldsymbol{\lambda}_n^{[k]}\right) \tag{29}$$

for $n \in \mathcal{N}$. As can be seen in (25) and (29), the updates involve only local communication with the neighbors. The proximal problem in (27) often admits a closed form solution for various choices of $\psi(\cdot)$. It can be proved that the iterate $\boldsymbol{\theta}_n^{[k]}$ for any $n \in \mathcal{N}$ converges to the solution to (21) as $k \to \infty$ [14].

C. Rate Constraints

In order to track the changes of a map over time, each sensor needs to send new measurements every certain time either to a fusion center or to other sensors, depending on whether the operation is centralized or decentralized. In certain cases, for example when constructing PSD maps that change frequently over time, the bandwidth required to report these measurements may be significant. Since sensing and communication functionalities may coexist in the same device, it is worth minimizing the bandwidth used to report measurements so that more bandwidth can

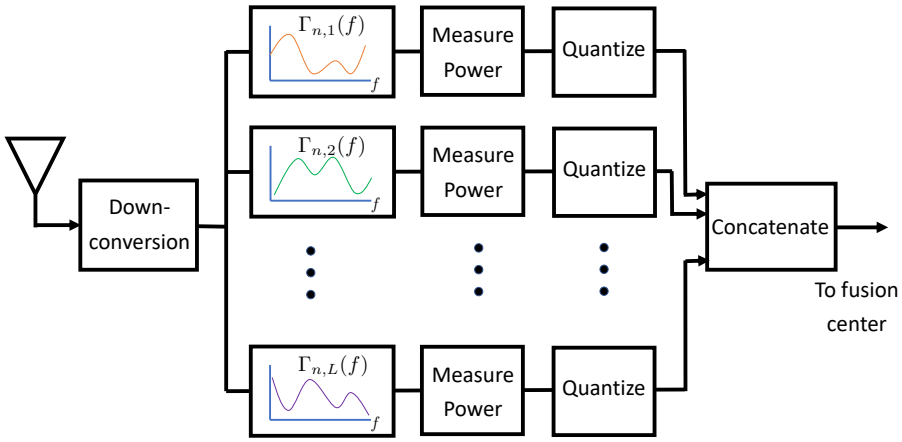


Fig. 13: To reduce the rate necessary to report measurements, sensors may use a bank of random filters. The energy of each filter is measured, quantized, and sent to a fusion center that performs RME.

be used to transmit user data. This section explains how this goal can be accomplished by means of compression [7].

The idea is that, instead of computing periodograms or other forms of PSD estimates, each sensor measures the power at the outputs of a filter bank acting on the received signal and sends the result of quantizing those measurements; see Fig. 13. To simplify the explanation, it will be first assumed that each sensor uses a filter bank with a single filter; the extension to multiple filters is discussed later.

As before, let $p(\boldsymbol{x}, f)$ denote the PSD at location $\boldsymbol{x}$. If the received signal is filtered by a sensor at location $\boldsymbol{x}_n$, the output of such a filter has an (ensemble) power $\bar{p}_n := \int p(\boldsymbol{x}_n, f)|\Gamma_n(f)|^2 df$, where $\Gamma_n(f)$ denotes the frequency response. This power is measured, which will yield a value $\tilde{p}_n$ that is generally different from $\bar{p}_n$, and the result is quantized. The quantized measurement, say m_n, is then sent to the fusion center or other sensors, which requires a much smaller bandwidth than sending e.g. periodograms.

To see how the map can be estimated from these linearly compressed and quantized measurements, it is worth recalling the decomposition $p(\boldsymbol{x}, f) = \sum_c p_c(\boldsymbol{x})\phi_c(f)$ used earlier. Since the basis functions ϕ_c are known, this decomposition reduces the problem of estimating function p to the problem of estimating the C functions $p_1, \ldots, p_C$. It also follows from the afore-

mentioned decomposition that $\bar{p}_n$ can be written as $\bar{p}_n = \sum_c p_c(\boldsymbol{x}_n) \int \phi_c(f)|\Gamma_n(f)|^2 df = [p_1(\boldsymbol{x}_n), \ldots, p_C(\boldsymbol{x}_n)]\boldsymbol{\phi}_n$, where the c-th entry of vector $\boldsymbol{\phi}_n$ is $\int \phi_c(f)|\Gamma_n(f)|^2 df$. In words, $\bar{p}_n$ is a linear combination of the values that the functions $p_1, \ldots, p_C$ take at $\boldsymbol{x}_n$.

To arrive at an estimator, suppose first that $\bar{p}_1, \ldots, \bar{p}_N$ are known. In that case, it makes sense to seek estimates of $p_1, \ldots, p_C$ as functions $g_1, \ldots, g_C$ in an RKHS that satisfy $[g_1(\boldsymbol{x}_n), \ldots, g_C(\boldsymbol{x}_n)]\boldsymbol{\phi}_n = \bar{p}_n$. The framework of kernel-based learning can be used to select a suitable estimate among the infinitely many functions that satisfy this equality.

Now consider a less idealized setup where instead of knowing $\bar{p}_1, \ldots, \bar{p}_N$, one knows the quantized measurements $m_1, \ldots, m_N$, but it holds that $\tilde{p}_n = \bar{p}_n$ for all n. In other words, there is quantization but no measurement noise. Each m_n therefore indicates which quantization interval contains $\bar{p}_n$. Upon denoting the endpoints of the interval as $a(m_n)$ and $b(m_n)$, it makes sense to now seek estimates as functions $g_1, \ldots, g_C$ that satisfy $[g_1(\boldsymbol{x}_n), \ldots, g_C(\boldsymbol{x}_n)]\boldsymbol{\phi}_n \in [a(m_n), b(m_n)]$ for all n.

Finally, in the case where there is measurement noise, one has that $\tilde{p}_n$ is, in general, different from $\bar{p}_n$. If the noise is small relative to the width of the quantization interval, the result of quantizing both values will be often the same, but not always. This means that one cannot impose that $[g_1(\boldsymbol{x}_n), \ldots, g_C(\boldsymbol{x}_n)]\boldsymbol{\phi}_n$ necessary falls in the quantization interval $[a(m_n), b(m_n)]$. Instead, it makes more sense to promote this condition in a soft manner by penalizing deviations from the interval. Interestingly, by doing so in a linear fashion, it can be shown that the resulting estimates can be found using a *support vector machine* for regression [7].

The previous considerations can be extended to the case where the filter bank at each sensor contains $L > 1$ filters, as depicted in Fig. 13. Observe that two subscripts are necessary to index each branch. For example, the power at the l-th branch of the sensor at $\boldsymbol{x}_n$ is given by $\bar{p}_{n,l} = [p_1(\boldsymbol{x}_n), \ldots, p_C(\boldsymbol{x}_n)]\boldsymbol{\phi}_{n,l}$. Since all the vectors $\boldsymbol{\phi}_{n,1}, \ldots, \boldsymbol{\phi}_{n,L}$ multiply the same $[p_1(\boldsymbol{x}_n), \ldots, p_C(\boldsymbol{x}_n)]$, the values $\bar{p}_{n,l}$ are not fully informative about $p_1, \ldots, p_C$ unless $\boldsymbol{\phi}_{n,1}, \ldots, \boldsymbol{\phi}_{n,L}$ are linearly independent. This imposes a design constraint on the filters. For this reason, filters with pseudorandom impulse responses may be utilized, which are expected to yield linearly independent vectors $\boldsymbol{\phi}_{n,1}, \ldots, \boldsymbol{\phi}_{n,L}$ so long as $L \leq C$.

VIII. Future Directions

Although RME has been the subject of a sizable research body, a number of open issues still remain. First of all, the potential of radio maps to endow applications with radio situational awareness is yet to be exploited. A large part of the progress in this regard has taken place in the

context of device-free localization (see e.g. [45]) and UAV communications (see e.g. [12] and references therein), but a number of tasks arising in cellular networks such as resource allocation are yet to be explored. Radio maps can also be used as priors for enhanced channel estimation in mobile communications. Massive MIMO and mmWave networks may also benefit from reduced times for beam selection by means of RME.

Improving inference biases in data-driven radio map estimators is also necessary. This can be achieved by collecting extensive data sets in multiple bands, since most works so far rely on synthetic data generated with ray-tracing software. Such data sets would also open the door to devising improved uncertainty metrics. Remarkably, these can be used for improving spectrum surveying techniques [19].

Recent developments adopt machine learning algorithms to predict the CSIs of desired multi-antenna channels based on pilot CSIs. Thus, the approach can capture the characteristics of small-scale fading, going beyond the channel gain maps. In [48], the pilot CSIs are obtained from a set of links that are *different* from the target link. The optimal transmit-beampattern of the desired link is predicted based on the acquired CSIs. When the source and the target links are not collocated, the traditional assumption is that the CSIs are statistically independent. In reality, they can show significant dependency, since the CSIs depend on the propagation geometry such as the transceiver locations, line-of-sight path, and other multipaths, within the coherence time of the channels. Thus, given sufficiently rich pilot CSI measurements that can capture the relevant geometry, an appropriate nonlinear mapping (e.g. via a DNN) can exploit the dependency. Related, the idea of channel charting obtains even in an unsupervised fashion the low-dimensional embeddings of the high-dimensional CSIs, where the embeddings indeed show strong correlation with the actual spatial locations of the radio [49].

IX. Conclusion

Radio maps characterize important metrics on the RF spectrum landscape across a geographical area. Two types of radio maps were considered based on whether the aggregate received signal metrics or the propagation channel effects are of interest, namely, the occupancy maps and the propagation maps, respectively, together with their representative applications. Tutorial expositions of various data-driven methods for RME have been presented, ranging from parametric, non-parametric, and probabilistic approaches, to recent powerful deep learning techniques, incorporating useful priors such as sparsity, low rank, and union-of-subspace structures. Practical issues related to the spectrum surveying, noisy location estimates, decentralized implementation,

and limited-rate measurements were also discussed. With the advent of ultra-dense and ultra-dynamic deployment scenarios often envisioned in the future wireless networking, the role of data-driven spectrum cartography enabled via sophisticated RME techniques will likely grow.

REFERENCES

[1] A. Alaya-Feki, S. B. Jemaa, B. Sayrac, P. Houze, and E. Moulines, "Informed spectrum usage in cognitive radio networks: Interference cartography," in *Proc. IEEE Int. Symp. Personal, Indoor Mobile Radio Commun.*, Cannes, France, Sep. 2008, pp. 1–5.

[2] J.-A. Bazerque and G. B. Giannakis, "Distributed spectrum sensing for cognitive radio networks by exploiting sparsity," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1847–1862, Mar. 2010.

[3] B. A. Jayawickrama, E. Dutkiewicz, I. Oppermann, G. Fang, and J. Ding, "Improved performance of spectrum cartography based on compressive sensing in cognitive radio networks," in *Proc. IEEE Int. Commun. Conf.*, Budapest, Hungary, Jun. 2013, pp. 5657–5661.

[4] H. B. Yilmaz, T. Tugcu, F. Alagöz, and S. Bayhan, "Radio environment map as enabler for practical cognitive radio networks," *IEEE Commun. Mag.*, vol. 51, no. 12, pp. 162–169, Dec. 2013.

[5] N. Patwari and P. Agrawal, "Effects of correlated shadowing: Connectivity, localization, and RF tomography," in *Proc. Int. Conf. Info. Process. Sensor Networks*, St. Louis, MO, Apr. 2008, pp. 82–93.

[6] D. Romero, D. Lee, and G. B. Giannakis, "Blind radio tomography," *IEEE Trans. Signal Process.*, vol. 66, no. 8, pp. 2055–2069, Jan. 2018.

[7] D. Romero, S-J. Kim, G. B. Giannakis, and R. López-Valcarce, "Learning power spectrum maps from quantized power measurements," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2547–2560, May 2017.

[8] S.-J. Kim, E. Dall'Anese, and G. B. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 1, pp. 24–36, Feb. 2011.

[9] A. Agarwal and R. Gangopadhyay, "Predictive spectrum occupancy probability-based spatio-temporal dynamic channel allocation map for future cognitive wireless networks," *Trans. Emerging Telecommun. Technol.*, vol. 29, no. 8, pp. e3442, 2018.

[10] D. Romero and G. Leus, "Non-cooperative aerial base station placement via stochastic optimization," in *Proc. IEEE Mobile Ad-hoc Sensor Netw.*, Shenzhen, China, Dec. 2019, pp. 131–136.

[11] E. Axell, G. Leus, and E. G. Larsson, "Overview of spectrum sensing for cognitive radio," in *Proc. Cognitive Inf. Process.*, 2010, pp. 322–327.

[12] D. Romero, P. Q. Viet, and G. Leus, "Aerial base station placement leveraging radio tomographic maps," in *arXiv:2109.07372*, 2021.

[13] E. Dall'Anese, J.-A. Bazerque, and G. B. Giannakis, "Group sparse lasso for cognitive network sensing robust to model uncertainties and outliers," *Phy. Commun.*, vol. 5, no. 2, pp. 161–172, Jun. 2012.

[14] J.-A. Bazerque, G. Mateos, and G. B. Giannakis, "Group-lasso on splines for spectrum cartography," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4648–4663, Oct. 2011.

[15] C. Parera, Q. Liao, I. Malanchini, C. Tatino, A. E. C. Redondi, and M. Cesana, "Transfer learning for tilt-dependent radio map prediction," *IEEE Trans. Cognitive Commun. Networking*, vol. 6, no. 2, pp. 829–843, Jan. 2020.

[16] T. Imai, K. Kitao, and M. Inomata, "Radio propagation prediction model using convolutional neural networks by deep learning," in *Proc. IEEE European Conf. Antennas Propag.*, Krakow, Poland, Apr. 2019, pp. 1–5.

[17] M. Iwasaki, T. Nishio, M. Morikura, and K. Yamamoto, "Transfer learning-based received power prediction with ray-tracing simulation and small amount of measurement data," *arXiv preprint arXiv:2005.00833*, 2020.

[18] R. Levie, Ç. Yapar, G. Kutyniok, and G. Caire, "RadioUNet: Fast radio map estimation with convolutional neural networks," *IEEE Trans. Wireless Commun.*, vol. 20, no. 6, pp. 4001–4015, 2021.

[19] R. Shrestha, D. Romero, and S. P. Chepuri, "Spectrum surveying: Active radio map estimation with autonomous uavs," in *arXiv:2201.04125*, 2022.

[20] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*, MIT Press, 2002.

[21] D. Romero, S.-J. Kim, and G. B. Giannakis, "Stochastic semiparametric regression for spectrum cartography," in *Proc. IEEE Int. Workshop Comput. Advan. Multi-Sensor Adapt. Process.*, Cancun, Mexico, Dec. 2015, pp. 513–516.

[22] M. Gudmundson, "Correlation model for shadow fading in mobile radio systems," *Electron. Letters*, vol. 27, no. 23, pp. 2145–2146, Nov. 1991.

[23] D.-H. Huang, S.-H. Wu, W.-R. Wu, and P.-H. Wang, "Cooperative radio source positioning and power map reconstruction: A sparse Bayesian learning approach," *IEEE Trans. Veh. Technol.*, vol. 64, no. 6, pp. 2318–2332, Jun. 2015.

[24] B. Khalfi, B. Hamdaoui, and M. Guizani, "AirMAP: Scalable spectrum occupancy recovery using local low-rank matrix approximation," in *Proc. IEEE GLOBECOM*, Abu Dhabi, UAE, Dec. 2018.

[25] D. Schäufele, R. L. G. Cavalcante, and S. Mtanczak, "Tensor completion for radio map reconstruction using low rank and smoothness," in *Proc. IEEE SPAWC*, Cannes, France, Jul. 2019.

[26] G. Zhang, X. Fu, J. Wang, and M. Hong, "Coupled block-term tensor decomposition based blind spectrum cartography," in *Proc. Asilomar Conf. Signal, Syst., Comput.*, Pacific Grove, CA, Nov. 2019, pp. 1644–1648.

[27] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Process.*, vol. 37, no. 23, pp. 3311–3325, Dec. 2006.

[28] S.-J. Kim and G. B. Giannakis, "Cognitive radio spectrum prediction using dictionary learning," in *Proc. IEEE Global Commun. Conf.*, Atlanta, GA, Dec. 2013, pp. 3206 – 3211.

[29] S.-J. Kim and G. B. Giannakis, "Dynamic learning for cognitive radio sensing," in *Proc. of the 5th IEEE Int. Workshop on Comp. Adv. Multi-Sensor Adaptive Process.*, St. Martin, French Caribbean, Dec. 2013, pp. 388–391.

[30] Y. Teganya and D. Romero, "Deep completion autoencoders for radio map estimation," *IEEE Trans. Wireless Commun.*, 2021.

[31] V. V. Ratnam, H. Chen, S. Pawar, B. Zhang, C. J. Zhang, Y.-J. Kim, S. Lee, M. Cho, and S.-R. Yoon, "FadeNet: Deep learning-based mm-wave large-scale channel fading prediction and its applications," *IEEE Access*, vol. 9, pp. 3278–3290, 2020.

[32] E. Krijestorac, S. Hanna, and D. Cabric, "Spatial signal strength prediction using 3D maps and deep learning," in *Proc. IEEE Int Conf. Commun.* IEEE, 2021, pp. 1–6.

[33] Q. Niu, Y. Nie, S. He, N. Liu, and X. Luo, "RecNet: A convolutional network for efficient radiomap reconstruction," in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–7.

[34] X. Han, L. Xue, F. Shao, and Y. Xu, "A power spectrum maps estimation algorithm based on generative adversarial networks for underlay cognitive radio networks," *Sensors*, vol. 20, no. 1, pp. 311, Jan. 2020.

[35] S. Shrestha, X. Fu, and M. Hong, "Deep generative model learning for blind spectrum cartography with NMF-based radio map disaggregation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* IEEE, 2021, pp. 4920–4924.

[36] D. Romero and G. Leus, "Wideband spectrum sensing from compressed measurements using spectral prior information," *IEEE Trans. Signal Process.*, vol. 61, no. 24, pp. 6232–6246, Dec. 2013.

[37] S.-J. Kim, N. Jain, and G. B. Giannakis, "Joint link learning and cognitive radio sensing," in *Proc. of the 45th Asilomar Conf. on Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2011, pp. 1415–1419.

[38] K. V. Mardia, C. Goodall, E. J. Redfern, and F. J. Alonso, "The Kriged Kalman filter," *Test*, vol. 7, no. 2, pp. 217–285, Dec. 1998.

[39] E. Dall'Anese, S.-J. Kim, and G. B. Giannakis, "Channel gain map tracking via distributed kriging," *IEEE Trans. Veh. Technol.*, vol. 60, no. 3, pp. 1205–1211, 2011.

[40] D. Lee, S.-J. Kim, and G. B. Giannakis, "Channel gain cartography for cognitive radios leveraging low rank and sparsity," *IEEE Trans. Wireless Commun.*, vol. 16, no. 9, pp. 5953–5966, Jun. 2017.

[41] N. Patwari and P. Agrawal, "NeSh: A joint shadowing model for links in a multi-hop network," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Las Vegas, NV, Mar. 2008, pp. 2873–2876.

[42] B. R. Hamilton, X. Ma, R. J. Baxley, and S. M. Matechik, "Propagation modeling for radio frequency tomography in wireless networks," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 1, pp. 55–65, Feb. 2014.

[43] J. Wilson and N. Patwari, "See-through walls: Motion tracking using variance-based radio tomography networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 5, pp. 612–621, May 2011.

[44] D. Romero, D. Lee, and G. B. Giannakis, "Blind radio tomography," *IEEE Trans. Signal Process.*, vol. 66, no. 8, pp. 2055–2069, Apr. 2018.

[45] J. Wilson, N. Patwari, and O. G. Vasquez, "Regularization methods for radio tomographic imaging," in *Virginia Tech Symp. Wireless Personal Commun.*, Blacksburg, VA, Jun. 2009.

[46] D. Lee, D. Berberidis, and G. B. Giannakis, "Adaptive Bayesian channel gain cartography," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Calgary, Canada, Apr. 2018, pp. 3555–3558.

[47] Y. Teganya, D. Romero, L. M. Lopez-Ramos, and B. Beferull-Lozano, "Location-free spectrum cartography," *IEEE Trans. Signal Process.*, vol. 67, no. 15, pp. 4013–4026, Aug. 2019.

[48] Z. Jiang, S. Chen, A. F. Molisch, R. Vannithamby, S. Zhou, and Z. Niu, "Exploiting wireless channel state information structures beyond linear correlations: A deep learning approach," *IEEE Commun. Magazine*, vol. 57, no. 3, pp. 28–34, Mar. 2019.

[49] J. Deng, O. Tirkkonen, J. Zhang, X. Jiao, and C. Studer, "Network-side localization via semi-supervised multi-point channel charting," in *Proc. Int. Wireless Commun. Mobile Computing*, 2021, pp. 1654–1660.

[50] S.-J. Kim, N. Jain, G. B. Giannakis, and P. Forero, "Joint link learning and cognitive radio sensing," in *Proc. Asilomar Conf. Signal, Syst., Comput.*, Pacific Grove, CA, Nov. 2011, pp. 1415 –1419.