Graph Scale-Space Theory for Distributed Peak and Pit Identification

Andreas Loukas[†] Marco Cattani[†] Marco Zuniga[†] Jie Gao[‡]
Delft University of Technology[†], Stony Brook University[‡]
{a.loukas, m.cattani, m.a.zunigazamalloa}@tudelft.nl[†], jgao@cs.sunysb.edu[‡]

ABSTRACT

Graph filters are a recent and powerful tool to process information in graphs. Yet despite their advantages, graph filters are limited. The limitation is exposed in a filtering task that is common, but not fully solved in sensor networks: the identification of a signal's peaks and pits. Choosing the correct filter necessitates *a-priori* information about the signal and the network topology. Furthermore, in sparse and irregular networks graph filters introduce distortion, effectively rendering identification inaccurate, even when signal-specific information is available. Motivated by the need for a multi-scale approach, this paper extends classical results on scale-space analysis to graphs. We derive the family of scale-space kernels (or filters) that are suitable for graphs and show how these can be used to observe a signal at all possible scales: from *fine* to *coarse*. The gathered information is then used to distributedly identify the signal's peaks and pits. Our graph scale-space approach diminishes the need for *a-priori* knowledge, and reduces the effects caused by noise, sparse and irregular topologies, exhibiting: (i) superior resilience to noise than the state-of-the-art, and (ii) at least 20% higher precision than the best graph filter, when evaluated on our testbed.

1. INTRODUCTION

Recently, there has been a surge of research focusing on the processing of graph data. One of the breakthroughs of the community has been the design of *graph filters*, distributed algorithms with applications to sensor, transportation, social and biological networks [21, 23]. Similar to how classical filters operate on time signals and images, graph filters operate on *graph signals*, *i.e.*, signals defined on the nodes of irregular graphs [24]. Being abstract representations of graph data, graph signals can be used in a variety of contexts. In sensor networks for example, the graph models the communication network between wireless devices, whereas the signal represents the data that devices sense. One of the benefits of graph filters is that they allow one to observe graph data at different scales. For example, Figure 1 shows that a signal filtered with a low scale parameter ($s = 0$), exposes fine details, while coarse signal trends are observed at higher scales ($s = 14$). Based on the scale parameter, a low-pass graph filter controls the size of observable signal structures, attenuating structures of small size, such as noise [28]. Graph filters are also useful for revealing communities [25], identifying event-regions (band-pass) [18], and detecting anomalies (high-pass) [21].

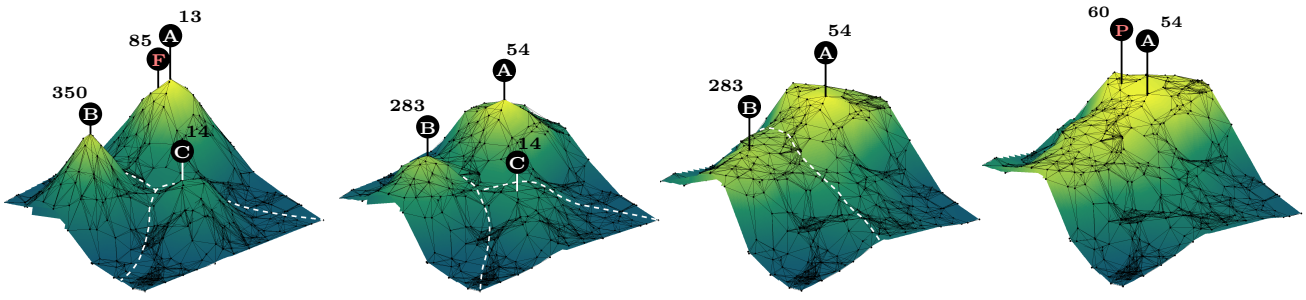
Yet, despite their theoretical guarantees and distributed computational efficiency, graph filters are also limited. A common task that exposes their shortcomings is the identification of the *peaks and pits* of a graph signal. Beyond giving us insights about the signal itself, the peaks (and pits) of a signal appear recurrently in a wide range of applications in sensor networks. Peaks are implicitly used by event and target tracking algorithms [1, 2, 8, 10] and form the basis of topological methods for signal mapping and compression, such as surface networks [11], iso-contour maps [22], and Morse-Smale complexes [29]. Furthermore, peaks are implicitly used by gradient-based navigation [9, 19], where a discovered path is only useful if it leads to a true peak. An accurate identification of peaks is thus a necessary prior step for the proper operation of gradient-based methods.

On the surface, identifying the peaks and pits of a signal appears deceptively simple: a node is at the summit of a peak if its value is the largest amongst its neighbors (local maximum). Equivalently, a node is at the bottom of a pit if its value is the smallest amongst its neighbors (local minimum). In practice however, the accurate identification of peaks and pits is challenging [10, 11]. The challenge arises due to two key problems. First, extrema are inherently tied to the local signal derivative and thus notoriously sensitive to noise. Second, extrema are affected by how the network is connected, and occur more often in sparse irregular networks. For these two reasons, graph signals often contain *false extrema*—maxima and minima that do not correspond to the real peaks and pits of the physical signal.

Though identifying peaks/pits is a filtering problem, graph filters exhibit several drawbacks: (i) First, the filtering efficiency depends on the correct choice of scale. For Figure 1, this drawback maps to the following question: what scale gives the most truthful representation of the underlying signal? To choose the scale of observation correctly, one must have *a-priori* information about the observed phenomenon, as well as of the instrument of observation—in our case, the network topology. (ii) Second, even the correct choice of scale results in loss of information. Every scale conveys use-

IPSN'15 April 14 - 16, 2015, Seattle, WA, USA

ACM 978-1-4503-3475-4/15/04$15.00.
http://dx.doi.org/10.1145/2737095.2737101.



(a) $s = 0$ (b) $s = 2$ (c) $s = 7$ (d) $s = 14$

Figure 1: The maxima of a graph signal across four different scales s. A number indicates the id of each maximum. The three peaks are indicated with letters A-C. F and P indicate a false and a phantom extremum, respectively. The sequence of signals across all scales gives rise to the signal's scale-space.

ful information about a signal: coarse scales describe large structures, whereas fine scales reveal details. Enforcing a single scale of observation can lead us to ignore valuable information. (*iii*) Third, this paper shows that filtering over *irregular* graphs, such as those found in real wireless networks, can cause *phantom extrema*, *i.e.*, extrema that are not present on the signal, but are an artifact of the filtering process. These phantom extrema severely hamper identification in practice, even when the scale is chosen correctly.

1.1 Related work

One of the standard approaches to overcoming the limitations of filtering is the *scale-space approach* [26]. According to scale-space theory, to capture the full set of features present in a signal, we must examine it across all possible scales of observation. Scale-space approaches are widely used in computer vision for extracting image features, such as extrema, saddles, and corner pixels, as well as for image smoothing and edge detection [16, 27]. The central question of scale-space is identifying the scale-space kernel (or filter) that abides to a set of conditions, referred to as scale-space axioms [17]. We currently know that, in both continuous and discrete settings, the only kernel that abides to the scale-space axioms is the heat kernel. However, even though there are indications that the same kernel also applies to graphs [12, 27], a rigorous examination of graph scale-space is -to this point- missing. We prove that in graphs the heat kernel is not the only kernel that abides to the scale-space axioms; others do too. We also propose an efficient way to compute them distributedly, which is especially relevant in wireless sensor networks, due to the limited time and resources available to communicate and process sensor data.

Scale-space is not the only approach to identify peaks and pits: (*i*) In sensor networks, the first to recognize that false extrema exist (also referred to as weak peaks/pits) and to propose an algorithm for their identification were Jeong *et al.* [10, 11]. We found that their algorithm does improve identification accuracy, but only to a limited extend. Especially when the signal is noisy, the identification is inaccurate. (*ii*) In the continuous setting, the peak identification problem is referred to as mode seeking [5]. From that perspective, our methods are generalizations of a graph-based mean-shift to multiple scales. Our evaluation reveals that even the best single-scale method (*i.e.*, graph-based mean-shift with optimal bandwidth) cannot cope with the abundance of phantom extrema present in wireless networks. A multi-scale approach is necessary.

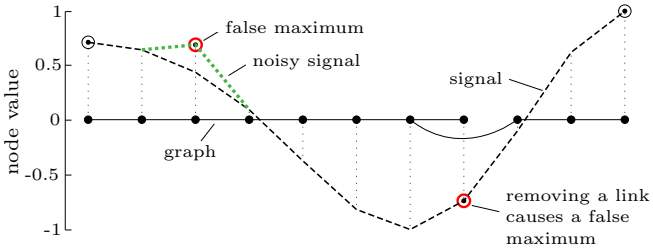
1.2 Contributions

Within this context, we provide three main contributions:

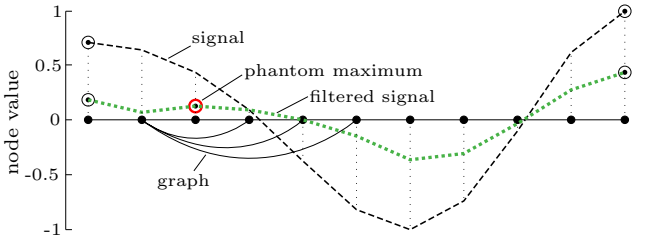
Contribution 1. Extending scale-space theory to graphs (Section 3). We address two fundamental questions: *What are the scale-space kernels that are appropriate for graphs and how efficiently can we compute them distributedly?* To answer this question, we first identify the properties that graph scale-space kernels must have, and then, evaluate the pros and cons of three candidate kernels. We also show that in practice, *synchronous* implementations of graph scale-space kernels seem to be the only viable option in terms of time complexity. Our analysis suggests that synchrony is a fundamental requirement of scale-space kernels for graphs—in the sense that currently known asynchronous algorithms exhibit much higher complexity. Last, we show that scale-space kernels are essentially graph filters. Our insight allows us to draw interesting connections between scale-space theory and signal processing on graphs.

Contribution 2. Using the scale-space approach to identify signals' peaks and pits (Sections 4-5). Similar to images, instead of selecting a single scale, we observe a signal at every possible scale and use the combined information to identify the peaks and pits [17]. Intuitively, this method follows a *survival of the fittest* approach, where the longer a peak (or pit) survives across scale, the higher the likelihood of being identified as a 'true' feature. Overall, the scale-space approach entails three steps: (*i*) Using a scale-space kernel, we progressively simplify the sensed signal (see Figure 1). The sequence of scaled signals, each simpler than the previous, gives rise to the signal's *scale-space*. (*ii*) All the while, we track how the simplification changes the signal extrema. The captured information, which is referred to as the signal's *deep structure* [13], contains the location and lifetime of each extremum across scale (*cf.* Figure 3). (*iii*) We use the deep structure to discern whether an extremum is true, false or phantom, essentially identifying the peaks and pits of the underlying signal. All three steps are computed locally within the network, require no location information, and incur a computational overhead similar to that of a graph filter.

Contribution 3. Defining the challenges of peak identification in real-world sensor networks (Section 6). We implemented both state-of-the-art and scale-space algorithms in a testbed consisting of (up to) 99 wireless sensor



(a) false extrema due to noise (left) and sparsity (right)



(b) phantom extrema are artifacts of filtering

Figure 2: Even slight changes to the connectivity of a path graph (black solid lines) and to the values of the physical signal (black dashed line) can cause false and phantom extrema to appear. On the top, the noisy sensed signal (green dotted line) contains two false maxima, caused by noise and irregular connectivity. On the bottom, the filtered signal (green dotted line) contains a phantom maximum.

nodes, and used the gathered information to benchmark the accuracy of each method. To the best of our knowledge, we are the first to evaluate such mechanisms in real-word scenarios, thus including radio-specific effects like irregular coverage, asymmetric links, packet loss and temporal link variability. We discovered that these phenomena have a drastic impact on the accuracy of peak/pit identification and should be considered in the evaluation of future mechanisms. Moreover, we show that the superior resilience of scale-space methods to radio effects makes our approach far more resilient to noise than the state-of-the-art and at least 20% more precise than the best graph filter.

2. PRELIMINARIES

We start by describing the problem of peak and pit identification in sensor networks and discussing its main challenges. Section 2.2 presents an overview of our approach.

2.1 Peak and pit identification

Consider a sensor network $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ of n nodes and m links that is monitoring its environment. Each node $u \in \mathcal{V}$ is situated in some (possibly unknown) physical location, and samples a physical signal present in a Euclidean space and imbued with noise of unknown characteristics, such as mean, variance, and type. The sensed information is captured by a graph signal $x : \mathcal{V} \to \mathbb{R}$, which assumes a real value $x(u)$ on each node u. Our objective is to identify the peaks and the pits of the underlying physical signal.

Challenges. Identifying the extrema of the sensed graph signal x is easy: a node u is a *maximum* if $x(u) > x(v)$ for all nodes $v \sim u$ in its communication vicinity ($v \sim u$ denotes that v is a neighbor of u). Correspondingly, u is a *minimum* if $x(u) < x(v)$ for all $v \sim u$. Nevertheless, the maxima and minima of x do not necessarily correspond to the peaks and the pits of the underlying physical signal. In Figure 1(a) for example, there are four extrema but only three peaks (A,B,C). These mismatches occur for three main reasons, the first two reasons are well known in the community, while the latter is an insight obtained through our work:

(*i*) *Noise.* Whether because of spatio-temporal perturbations or sensor imprecision, signals can fluctuate significantly around their original value, leading to false positives and false negatives (*cf.* Figure 2(a)).

(*ii*) *Graph irregularities.* It is commonly assumed that whenever two nodes are placed within a given radius, the nodes are joined by a valid link. In practice, links exhibit high spatial and temporal variability, leading to false positive extrema: a low-valued node connected to a high-valued node would not select itself as a maximum, but if the link disappears, it will. In Figure 2(a) for example, the fourth node from the left perceives itself as a maximum only because it has no link to the node on its right. This phenomenon is pervasive in real networks and particularly acute in graphs with small average degree. Proposition 2 in the appendix shows that the expected number of extrema arising from volatile links grows with an exponential trend as the graph becomes sparser[1]. This trend is later observed in our evaluation.

(*iii*) *Phantom effects.* Broadly speaking, graph filters alter a signal by diffusing it locally. The filtering process is thus determined by the underlying connectivity. If a regular lattice is used –as is common in classical signal processing– no bias is introduced by the topology in the filtering process. In real deployments however, the node density varies across the network and highly dense areas tend to 'accumulate' a higher share of the diffused mass. This creates phantom extrema: extrema that are neither present on the underlying physical signal nor in the sensed signal, but they are an artifact of filtering over an irregular topology (*cf.* Figure 2(b)).

2.2 The scale-space approach

In this paper, we use scale-space theory to identify the peaks and pits of a signal. Our approach boils down to three steps, each computed distributedly within the network:

Step 1. Extracting a signal's scale-space. The central idea of scale-space is that we can learn more about a signal by systematically examining it across different scales of observation: coarse scales give us the big picture (Figure 1(d)), whereas in fine scales details prevail (Figure 1(a)). More concretely, the scale-space of a graph signal x consists of a sequence $\{y_s\}$ of scaled signals, each simpler than the previous. The scale-space is constructed by filtering x with a family of $n \times n$ kernels $\mathbf{K}_s$ as

$$y_s = \mathbf{K}_s x, \tag{1}$$

where the *scale parameter* s denotes that information is diffused within the s-hop neighborhood. Scale determines the

[1]Note that the number of maxima (minima) is at most equal to the size of the graph's maximal independent set.

maximum size of structures that can be observed. In Section 3, we identify appropriate scale-space kernels $\mathbf{K}_s$ and give local algorithms for their computation.

Step 2. Tracking extrema across scale. As a signal is progressively simplified, its extrema evolve. Peak B for instance, is born on node 350 in Figure 1(a), moves to node 283 in Figure 1(b), and dies in Figure 1(d). This process is compactly captured by the signal's *deep structure*, which visualizes the scale-trajectory of each extremum e (see Figure 3). The distributed computation and analysis of the deep structure is given in Section 4.

Step 3. Extremum selection. In the last step, we use the trajectory of each maximum (minimum) to infer the nature and importance of the corresponding peak (pit). We focus on two aspects: First, the extremum's *lifetime*, which is defined as the length of the scale-period over which e exists. Second, we focus on *birth events*. As explained before, in contrast to images, graph filters distort signals in a way that is unique to the graph topology, leading to phantom extrema. In Section 5.2 we will see that phantom extrema are easy to spot from the deep structure because they are always born in large scales.

3. SCALE-SPACE THEORY ON GRAPHS

This section sets the necessary theoretical basis of scale-space analysis in graphs. Though the ideas are rooted upon the scale-space theory for continuous signals and images [14, 17], our analysis deviates from the original. In contrast to the classic setting, graphs have irregular connectivity. Additionally, in networks each node can only directly exchange information with its neighbors. This motivates us to ask: *What type of kernels are appropriate for graphs and how efficiently can we compute them distributedly?*

We address this question in three steps: *In Section 3.1, we focus on the existence of graph scale-space kernels.* We derive necessary and sufficient conditions for a matrix to be a scale-space kernel and we present three candidates kernels. *In Section 3.2, we focus on computation.* We show that scale-space kernels are locally computable if synchrony is assumed, but that currently known practical asynchronous algorithms are not local. *Last, in Section 3.3 we expose the connection between graph scale-space kernels and graph filters.* We show that each of the candidate graph scale-space kernels forms a one-parameter family of graph filters. The observation gives insight into the operation of kernels and to their relation to the graph spectrum.

3.1 Axiomatic scale-space theory

The objective of scale-space theory is to provide a scale-invariant observation of a signal x. This is achieved by diffusing the signal with a family of kernels $\mathbf{K}_s$, where the *scale parameter* s determines the size of observed peaks and pits. But which kernel should one use? One of the breakthroughs of the scale-space community has been the axiomatization of the theory [17]. A scale-space kernel is then the one that satisfies the three main scale-space axioms[2]:

Axiom 1 (Linearity)**.** *Kernels $\mathbf{K}_s$ are linear operators.*

Axiom 2 (Semi-group property)**.** *The family of scale-space kernels forms a* semi-group*: $\mathbf{K}_{s_1}\mathbf{K}_{s_2} = \mathbf{K}_{s_1+s_2}$.*

Axiom 3 (Non-enhancement)**.** *The absolute value of any extremum in $\mathbf{K}_s x$ must always decrease in $\mathbf{K}_{s+1}x$.*

The first axiom asserts that a scale-space kernel is not signal specific, in the sense that it abides to the superposition principle: for any two signals x_1 and x_2, $\mathbf{K}_s(x_1 + x_2) = \mathbf{K}_s x_1 + \mathbf{K}_s x_2$. Additionally, $\mathbf{K}_s$ should act in the same way at all scales. If the semi-group property is not met, diffusion deforms the signal in a scale-specific way, which is undesirable. Last, according to the non-enhancement axiom, a kernel must always simplify existing signal extrema. Note that guaranteeing the simplification of *existing* extrema at higher scales (Axiom 3) is not the same as guaranteeing that no new extrema appear. Signals often contain hidden structures, which diffusion reveals[3]. Thus, non-enhancement represents the weaker alternative of guaranteeing that existing extrema are always simplified.

Even though the fundamental question of scale-space theory *–which scale-space kernel satisfies the scale-space axioms?–* has been answered for the continuous and discrete settings [14, 17], the answer does not directly apply to graphs. Graphs exhibit irregular connectivity and are not, in general, a metric space. This poses an analytical challenge as, unlike images, differentiation is impossible. These differences motivate us to address scale-space theory from a graph-theoretic perspective. In the following, we show that a graph scale-space kernel $\mathbf{K}_s$ that satisfies the scale-space axioms exists and that it is connected to the graph spectrum.

Central to our discussion is the notion of an h-local matrix. Intuitively, an h-local matrix is an operation that uses signal information at most h hops away from each node.

Definition 1 (h-local matrix)**.** *An arbitrary matrix M is h-local if, for all u_i and u_j in $\mathcal{V}$ with shortest-path distance $d(u_i, u_j) > h$, $M_{ij} = 0$.*

We proceed with our main result.

Theorem 1. *A kernel $\mathbf{K}_s$ satisfies the scale-space axioms if and only if $\mathbf{K}_s = S^s$, where the scale-space matrix S is 1-local and non-negative.*

Proof. We start by noticing that, for any kernel satisfying Axioms 1 and 2, it must be that $\mathbf{K}_s = S\mathbf{K}_{s-1}$, where $S = \mathbf{K}_1$. Applying this recursively, we obtain $\mathbf{K}_s = S^s$. The conditions that S is 1-local and non-negative are imposed by Axiom 3 and are given in Lemmas 1 and 2, respectively. □

Lemma 1. *Kernel $\mathbf{K}_s = S^s$ satisfies the non-enhancement axiom only if S is a 1-local matrix.*

Proof. We show by method of contradiction that, kernel $\mathbf{K}_s = S^s$ satisfies Axiom 3 only if S is 1-local. Any matrix S is always h-local for h equal to the network diameter; the question is whether $h = 1$. For sake of contradiction assume that $h > 1$. We show that, if this is true, a signal x always exists (independently of $\mathcal{G}$) that violates the

[2]Though different combinations of axioms have been used in the literature [17], the three mentioned here suffice for the characterization of graph scale-space kernels.

[3]For example, consider two peaks joined by a thin bridge. If the lowest point of the bridge is slightly taller than the lowest peak, then the signal has only one maximum. The second maximum is however revealed when, due to diffusion, the bridge collapses.

kernel	name	S	sym.	col.-st.	row-st.
$\mathbf{H}_t$	heat	$I - \mathcal{L}$	yes	no	no
$\mathbf{T}_t$	random-walk	T	no	yes	no
$\mathbf{P}_t$	consensus	P	no	no	yes

Table 1: Candidate scale-space kernels, along with the properties of their scale-space matrices S. 'Row/col.-st' refers to row- and column-stochasticity, respectively, whereas 'sym.' is a shorthand for symmetric.

non-enhancement axiom. Therefore the assumption is incorrect and $h = 1$. The construction is as follows: Choose two nodes u_i, u_j with $d(u_i, u_j) = h > 1$ such that $S_{ij} \neq 0$. Assign values $x(u_i) = 1$ and $x(v) = 0$ otherwise. Clearly, u_i is a maximum and its value should decrease in in the next iteration $(Sx)(u_i)$. Nevertheless, a real number $\beta \gg 1$ always exists for which, if $x(u_j) = \beta \operatorname{sign}(S_{ij})$, then $(Sx)(u_i) > 1$, which is a contradiction. □

Lemma 2. *Kernel $\mathbf{K}_s = S^s$ satisfies the scale-space axioms if and only if S is a 1-local and non-negative matrix.*

Proof. We begin by establishing that, if the two conditions hold, the scale-space axioms are satisfied. Let u_i be a maximum of x. By definition, its neighbors u_j have strictly smaller values. It is easy to see that, if the non-negative condition $S_{ij} \geq 0$ holds, then $(Sx)(u_i) \leq x(u_i)$, which is exactly the non-enhancement axiom. As we can see, when S is 1-local and non-negative it always satisfies the three axioms. The two proposed conditions are therefore sufficient. As we show next, since the opposite is not always true, the two conditions are also necessary. The proof is done by contradiction: if we assume that $S_{ij} < 0$, we can always construct a signal that breaks the non-enhancement axiom. That is simply by assigning $x(u_i) = 1$, and for each neighbor u_j with $S_{ij} < 0$, $x(u_j) = -\beta$, where $\beta \gg 1$. The non-negativity of S's diagonal guarantees that extrema do not oscillate between being maxima and minima as scale increases. □

Remark. Though it is not requested by the scale-space axioms, it also useful to impose that $\|\mathbf{K}_s x\|$ never grows infinitely large. For this reason, in the following we only consider scale-space matrices S with spectral radius smaller or equal to one.

Candidate kernels. Though Theorem 1 establishes the properties of graph scale-space kernels, it does not provide an explicit form. In other words, many possible kernels may exist that satisfy the scale-space axioms. We have identified three such candidate kernels (*cf.* Table 1): The first is the *heat kernel* $\mathbf{H}_s = (I - \mathcal{L})^s$, where $\mathcal{L}$ is Chung's normalized Laplacian matrix. Additionally to satisfying the conditions of Theorem 1, the heat kernel is symmetric, which suits applications working over undirected graphs. In wireless networks however this property is not satisfied.

The next two kernels under consideration are asymmetric. But to validate their suitability for peak identification, we need to evaluate two important properties: *column-* and *row-stochasticity*: (*i*) Column-stochasticity warrants that the mass of x remains constant as it is being diffused, *i.e.*, $\mathbb{1}^\top \mathbf{K}_s x = \mathbb{1}^\top x$ for all s, where $\mathbb{1}$ is the all ones vector. The random walk kernel $\mathbf{T}_s = T^s = (AD^{-1})^s$, where D is the diagonal degree matrix and A the adjacency matrix, is column stochastic. For our purposes however column-stochasticity is not a desirable property, because the distribution of mass gets strongly biased towards well connected nodes. Considering the high irregularity of wireless networks, column-stochasticity would exacerbate the phantom extrema effect. On the other hand, this kernel is suitable for purely graph-based signals, such as web page centrality, and it has also been used for graph partitioning by Chung [6]. (*ii*) Row-stochasticity governs the behavior of a signal at very large scales. Consider the *consensus kernel* $\mathbf{P}_s = P^s = (D^{-1}A)^s$, our third scale-space kernel. Being row-stochastic, the consensus kernel flattens signals completely as $s \to \infty$. This property is particularly useful for filtering physical signals, such as the measurements of a sensor network, because it progressively eliminates (smooths) structures based on their size. Given that the consensus kernel $\mathbf{P}_s$ has the required properties for peak and pit identification, we used it as the default kernel for the rest of this paper.

Remark. The candidate kernels are easily generalized to consider additional adjacency information, such as edge weights. For instance, a node may choose to diffuse its value with a weight that is inversely proportional to the physical distance to its neighbors. Nevertheless, when only the communication graph is available, a simple average presents the most viable choice.

3.2 Distributed computation

We proceed to examine how graph scale-space kernels can be computed efficiently in a distributed network. We show that synchronous algorithms are the best option for practical implementations. In the asynchronous case, currently known practical algorithms are non-local.

Computational models. For convenience, we will assume that the computation proceeds in rounds t, during which nodes exchange *exactly one scalar* with each of their neighbors. Based on whether rounds of neighboring nodes overlap or not, we distinguish two versions: the asynchronous and the synchronous model. The main assumption posed, *i.e.*, that at least one message is exchanged with each neighbor, can be implemented in either of two ways: deterministically by using a local schedule and probabilistically by random beaconing. We quantify the computation cost in terms of the algorithm's *time complexity* (total number of rounds).

Synchronous algorithms. From Theorem 1, we can derive that, in the synchronous model, $\mathbf{K}_s$ is computed by the well known recursion:

$$\underbrace{y^{(t+1)}(u) \leftarrow \sum_{v \sim u} [S]_{uv} y^{(t)}(v)}_{\text{repeat every round}} \quad \text{and} \quad \underbrace{y^{(0)}(u) \leftarrow x(u)}_{\text{initialization}}, \qquad (2)$$

where each exchanged packet contains exactly one scalar ($y^{(t)}(v)$). Since each node needs s rounds to compute the desired information, the time complexity is $O(s)$.

Asynchronous algorithms. How efficiently can scale-space kernels be computed by a network without round-level synchronization? To answer this question, we examine the (only) three algorithmic approaches known to compute graph kernels asynchronously. We then show that, even though some of these approaches can be used to compute

scale-space kernels asynchronously, doing so incurs complexity much higher than $O(s)$.

The *first algorithmic approach* is straightforward: one can synchronize the network via packet exchanges. Unfortunately, enforcing synchronization requires global knowledge and takes at least as long as Ω(diameter) rounds [4]. Considering that s needs to be roughly of the same order as the diameter of the largest structure of interest in the signal, we could have $s \ll$ diameter. The first algorithm therefore exhibits increased time-complexity and is not local. The *second approach* follows from a simple observation: according to Theorem 1, $\mathbf{K}_s$ is s-local. This means that each node u needs only the values and connectivity of the nodes in its s-hop vicinity to compute $(\mathbf{K}_s x)(u)$. Communicating once with each neighbor in the s-hop vicinity is possible in $O(s)$ rounds if the packet-size can grow arbitrarily large. However, in our model the size of each packet is bounded to one scalar, and the time complexity is $\Omega(\delta^{s-1})$, where δ is the minimum node degree. Its exponential complexity renders the second algorithm impractical (especially in dense networks). The *third approach* is slightly more complex: instead of computing a graph kernel using power iteration (as in (2)), we will use an alternative recursion that -converges- linearly to the output without being affected by asynchrony. Though such recursions have been shown to hold great promise for graph kernels in general [18, 19], Theorem 2 shows that they cannot be used in our case:

Theorem 2. *No 1-st order recursion converges to* $\mathbf{K}_s$ *in the asynchronous model.*

Proof. Given in the Appendix. □

The intuition of the proof is that, whereas $(\mathbf{K}_s x)(u)$ is truncated, *i.e.*, it takes into account -at most- the values in an s-hop neighborhood of u, any kernel computed by the third approach decays asymptotically with the number of hops. Therefore, no 1-st order recursion converges exactly to $\mathbf{K}_s$. Though more tedious, the same argument applies for showing that no recursion of *any* order converges to $\mathbf{K}_s$ in the asynchronous model. Notice that, our results do not suffice to prove that $\mathbf{K}_s$ are not locally computable in the asynchronous model, but rather that no currently known such algorithm exists.

3.3 Connection to graph filters

We give an alternative interpretation to graph scale-space by noticing that each of the three candidate kernels is a low-pass graph filter [24]. This brings forth two main insights: *First, scale acts as the parameter of a low-pass graph filter.* Therefore, the way that a scale-space kernel attenuates peaks/pits is affected by the spectral graph properties. *Second, scale-space theory reveals the design characteristics of a scale-invariant graph filter*, *i.e.*, a filter which does not distort signals in a scale-specific manner.

First, we will show that the three candidate kernels are low-pass graph filters [24]. Showing that $\mathbf{H}_s$ is a graph filter is straightforward—any power series of a generalized Laplacian matrix is a valid graph filter. The same also holds for kernels $\mathbf{T}_s$ and $\mathbf{P}_s$ (up to a normalization factor). To see this notice that, $\mathbf{P}_s = P^s = D^{-1/2}(I-\mathcal{L})^s D^{1/2}$ and equivalently, $\mathbf{T}_s = T^s = D^{1/2}(I-\mathcal{L})^s D^{-1/2}$. Both $\mathbf{T}_s$ and $\mathbf{P}_s$ are therefore power series of the normalized Laplacian $\mathcal{L}$, normalized by (diagonal) matrices $D^{\pm 1/2}$. Furthermore, the frequency response of all three candidate kernels is given by $r(\lambda_k; s) = (1-\lambda_k)^s$, where λ_k is the k-th eigenvalue of $\mathcal{L}$.

Insights. Our first insight concerns the attenuation properties of the scale parameter s. The frequency response $r(\lambda_k; s)$ of all three filters attenuates phenomena as a monotonically decreasing function of λ_k. Taking into account that high(er) eigenvalues are associated with high(er) frequency events (such as noise), the larger s is, the simpler the graph signal becomes. The way a scale-space kernel alters a signal is therefore inherently tied to the spectral graph properties. Intuitively, this means that peaks lying on sparse subgraphs and containing many bottleneck links are attenuated first. For a more in-depth characterization of the influence of the graph spectrum, the reader is directed to the relevant literature on graph filters [18, 23].

Our analysis started from the premise that a signal should be observed across all scales. However even if a single scale is used, such as in graph filters, our analysis still carries a valuable insight. As a consequence of Theorem 1, the only filters that are signal- (Axiom 1) and scale-invariant (Axiom 2) must have a frequency response of $r(\lambda_k; s) = (1-\lambda_k)^s$, for λ_k the eigenvalue of some generalized Laplacian. These two properties are very desirable when one needs to filter a signal without introducing scale-specific distortion.

4. TRACKING EXTREMA ACROSS SCALE

In this section we identify and track the extrema of graph signals across scale. The computed trajectory, is a tree-like abstraction, called the signal's *deep structure*, reveals information about the relation and relative importance of peaks and pits.

Deep structure. As a signal is diffused, its extrema evolve. But what does this reveal about the signal's peaks and pits? We distinguish two cases:

Drift. When a peak/pit changes shape, the corresponding extremum drifts. In Figures 1(a) and 1(b) for example, as peak B is flattened, the maximum drifts from node 350 to 283. In this case, we say that extremum 350 is *unstable*. The movement of an unstable extremum e is captured in its trajectory $(u_b, \ldots, u_s, \ldots u_d)$, where u_s is the node e resides on at scale s, b is the scale over which e is born and d over which it dies. Notice that, even though the location of the extremum can change, i.e. $u_s \neq u_{s+1}$, the extremum continues to correspond to a *single* peak (or pit).

Collapse. When a peak/pit collapses, the corresponding extremum dies. A collapse thus entails the destruction of a signal structure. Consider for example peak C in Figure 1(b). By $s = 7$, the peak has collapsed and extremum 14 dies. It is also easy to see that, after peak C collapses, it is drawn into peak B: a gradient ascent on y_7 starting on node 14 ends up on node 283.

Diffusion results in a tree-hierarchy of collapses, each corresponding to a structure merging with a larger nearby structure. This process is compactly captured by the signal's deep structure. As shown in Figure 3, the deep structure visualizes the trajectory $(u_b, \ldots, u_d)$ of each extremum e using a black horizontal line. The trajectory spans the scales over which e lives: it starts at scale b and ends at d. Collapses on the other hand are depicted using dotted lines. Extremum

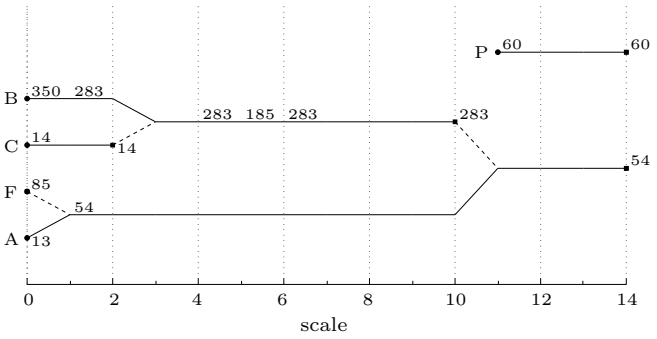


Figure 3: The deep structure (of Figure 1) describes the trajectory of extremal points (horizontal black lines) across scale. Dotted lines correspond to collapses. For each extremum we give the ids of the nodes in its trajectory.

14, for example, collapses at $s = 3$ in Figure 3, and is drawn into extremum 283.

Tracking algorithm. We next present an algorithm for the computation of the signal's deep structure. In our approach, the knowledge is distributed amongst the signal extrema. For this reason, our algorithm's time complexity $O(d_x)$ does not depend on the network diameter d (no node knows the complete deep structure), but on the diameter d_x of the largest peak/pit in the signal's scale-space[4].

We track the trajectory of extrema by forwarding information greedily along the diffused signal's gradient. The algorithms are better understood from an extremum's perspective: Consider an extremum e which at scale s resides on node u_s. W.l.o.g., assume that e is a maximum[5]. If u_s is also a maximum at scale $s+1$, then no change occurred and $u_{s+1} = u_s$. On the other hand, if drift or collapse occurs, a neighbor v will have a larger value than u_s. While v may not be necessary a maximum, the maximum u_{s+1} is easily found by following the gradient ascent path, *i.e.*, by iteratively forwarding a tracking query to the neighbor with the largest value (see Corollary 1 in the Appendix).

Even though the tracking algorithm is the same whether the extremum drifts or collapses, it is easy to distinguish the two cases locally: in a collapse, two distinct extrema merge into one. Thus whenever a maximum receives more than one gradient ascent (tracking) query, a collapse has occurred.

But if two extrema merge, which of the two collapsed? Suppose that $e_u = (\ldots, u_s, w_{s+1}, \ldots, w_d)$ and $e_v = (\ldots, v_s, w_{s+1}, \ldots, w_d)$ are the trajectories of two maxima, which at scale $s+1$ merge at node w_{s+1}. To figure out which maximum collapsed we use the following procedure: From Corollary 2 in the Appendix, we know that the absolute value of an *unstable* extremum always decreases. Thus, if the value of maximum e_u increased between scales s and $s+1$, and the value of e_v decreased *i.e.*, if

$$y_s(v_s) \geq y_{s+1}(w_{s+1}) > y_s(u_s), \tag{3}$$

then e_u collapsed and e_v was unstable. But what if the value of both e_u and e_v decreased from scale s to $s+1$, *i.e.*,

$$y_s(u_s) > y_{s+1}(w_{s+1}) \text{ and } y_s(v_s) > y_{s+1}(w_{s+1})? \tag{4}$$

This phenomenon can occur in noisy signals, when a peak is 'hidden' under nearby extrema (caused by noise): as the signal is diffused and the extrema merge, their values jointly decrease. In this case, there is no reliable way to distinguish which extremum collapsed[6]. Thus the scale-space method can discern that the merged extrema jointly form a peak, but can not trace back the extremum to the most relevant node at scale zero. In Figure 3, this case would be depicted as two dotted lines merging into a new extremum.

5. EXTREMUM SELECTION

We now describe how we use the signal's deep structure to identify peaks and pits. We consider two criteria for extremum selection: The *lifetime criterion* aims at filtering out false extrema, while the *birth criterion* aims at discarding phantom extrema.

5.1 The lifetime criterion

One of the fundamental results of scale-space theory is that the lifetime of extrema is a measure of the importance of the corresponding peaks and pits [15]. Though a number of definitions exist, the most natural defines the lifetime l_e of an extremum e as the length of scale-period over which e exists: $l_e = d_e - b_e$. Here, b_e and d_e denote the scale over which e is born and dies, respectively. The lifetime criterion is useful in separating small structures (false extrema) –caused by small signal perturbations– from inherent signal trends (true extrema): an extremum is true if $l_e \geq l$, where l is a user-defined lifetime threshold, and false otherwise. As we will see in the following, by setting l we can (roughly) retain peaks/pits that are bigger than l hops.

According to classic scale-space theory, the lifetime criterion is successful because it captures two key properties of the signal's peaks and pits: *volume* and *span*. Volume is a property that mainly depends on the signal, while span is also affected by the graph topology. Important structures in a signal have large volume. In Figure 1(a) for example, peak A is the largest because it is tall and decreases slowly at each direction. The volume in an s-hop neighborhood around e is captured by $y_s(e) = (\mathbf{K}_s x)(e)$ and is a weighted average of all the information residing at most s hops away from e. Additionally, a peak is important if it spans multiple hops. In Figure 1(a), peak A is more significant than B because A's width covers more hops. Intuitively, lifetime gives the size of the largest s-hop neighborhood centered around e.

5.2 The birth criterion

As depicted in Figure 2(b), a scale-space kernel introduces distortion when the underlying graph is unevenly connected. We call any effects that are present on the filtered signal, but not on the underlying physical and sensed signal, *phantom*. Phantom effects are particularly harmful as they obscure real data (*cf.* Section 6). Nevertheless, in the particular case of phantom extrema, there is an easy way to recognize them from the signal's deep structure. As we show in the following, phantom effects occur only in large scales. As a consequence, a phantom extremum e is always born at large scales and $b_e > 0$. This is the birth criterion.

[4]Though usually $d_x \ll d$, it is possible to construct an example in which $d_x = d$.

[5]For ease of presentation, we focus our discussion on maxima. Our results however also hold for minima. Tracking minima is equivalent to tracking the maxima of $-x$.

[6]In fact, one could argue that all extrema collapsed.

We examine phantom effects analytically by comparing filtering of the same signal in two different topological spaces: $\mathcal{G}$ models the network topology (*e.g.*, an irregular and sparse graph), whereas $\overline{\mathcal{G}}$ models the space over which the signal is defined (*e.g.*, an ideal unbiased lattice). To demonstrate the dependency between phantom effects, topology and scale, we examine how the absolute quadratic error

$$e_s = \left| \frac{x^\top (\overline{\mathbf{K}}_s - \mathbf{K}_s) x}{x^\top x} \right| \tag{5}$$

of the corresponding scale-space kernels $\overline{\mathbf{K}}_s = \overline{S}^s$ and $\mathbf{K}_s = S^s$ changes as the signal is filtered. We are interested in how the error behaves when scale s increases. Using simple operations we can see that $0 = e_0 \leq e_\infty$, which indicates that the error has an increasing trend. The following result bounds the error between the extremes of $s = 0$ and $s \to \infty$, when S is symmetric (all candidate kernels have symmetric scale matrices).

Proposition 1. *For any two graphs $\overline{\mathcal{G}} = (\mathcal{V}, \overline{\mathcal{E}})$ and $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, a signal $x \in \mathbb{R}^n$, and a scale $s \in \mathbb{N}$,*

$$\max(0,\, e_\infty - 2\lambda_2^s) \leq e_s \leq 1 - \lambda_{min}^s + \epsilon_s$$

where $\epsilon_s = (2s^s)/(s+1)^{s+1}$ when s is odd and zero otherwise, $-1 < \lambda_{min} < 0$ is the smallest amongst the eigenvalues of $\overline{S}$ and S, and $0 < \lambda_2 < 1$ is the smallest of the second eigenvalues of $\overline{S}$ and S.

Proof. We split the proof of Proposition 1 in two parts: The upper bound is given in Lemma 3 and the lower bound in Lemma 4. Both can be found in the Appendix. □

Note that both the lower and upper bounds increase with s. Moreover, for all $s \leq \log(\lambda_{min})(1 - e_\infty)$, $e_s < e_\infty$. This suggests that phantom effects become more likely as scale increases. We have to remark that, though unlikely, it is possible that an extremum e born at a scale larger than zero is not phantom. As discussed in Section 3, we can construct a signal which contains hidden peaks; *i.e.*, peaks revealed by filtering. In that sense, the birth criterion is necessary but not sufficient.

6. EVALUATION

We split our evaluation in two parts. Section 6.1 quantifies the identification accuracy of our method and compares it with the state-of-the-art. Section 6.2 evaluates a proof-of-concept implementation in a large-scale wireless testbed.

6.1 Simulations

We conducted Matlab simulations using a test-set of ten synthetic physical signals. Each signal was a mixture of seven multi-variate Gaussian distributions with random mean and covariance. The signal was sampled by 1024 nodes deployed at random. The objective of the simulations was to quantify the efficiency of identifying the seven Gaussian peaks. We measured identification accuracy using the standard metrics of *precision* and *recall*. Precision measures the fraction of true peaks present among all retrieved peaks, and recall the fraction of true peaks with respect to the ground truth. The ideal method therefore retrieves all true peaks (recall = 1) and discards peaks caused by false/phantom extrema (precision = 1).

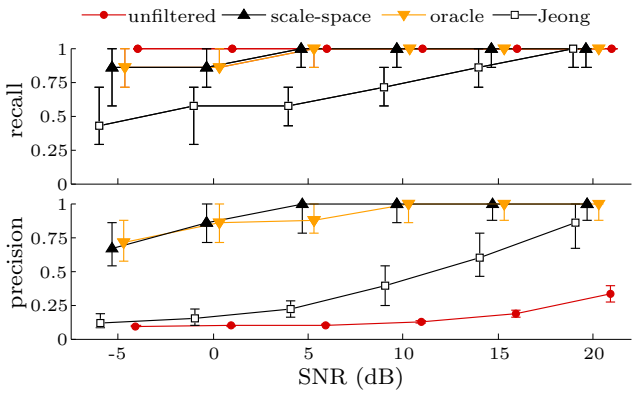


Figure 4: Precision/recall median and standard deviation for different methods and SNRs. Both oracle and scale-space filters achieved significantly better recall than state-of-the-art. Scale-space did so with a fixed lifetime thresholds ($l = 7$). To diminish overlap, we perturbed the horizontal positioning of data points.

To evaluate our algorithms, we compared the scale-space approach (*scale-space*) to the state-of-art method for identifying peaks and pits (*Jeong*) [11]. Additionally, we compared our method to the 'raw' unfiltered signal (*unfiltered*) and to the single-scale graph filter with the best possible performance (*oracle*). The latter was computed offline by exhaustively searching, for each experiment, the scaled signal y_s that maximized the minimum value of the metrics: $\max\min_s\{precision(y_s), recall(y_s)\}$. While it is infeasible to compute the oracle filter online, it serves as a benchmark for judging the benefit of the scale-space approach (multi-scale) over the best possible graph filter (vs. best single scale)

Because graph filters (and the scale-space approach) identify the peaks and pits of a signal by varying the scale of observation, an extremum of a scaled signal is not necessarily located at the same node as in the physical signal. To verify that an extremum in a large scale corresponded to a peak, we checked whether the node that the extremum resided on at the sensed signal was close to a true extremum. When extrema merged without collapsing, we checked whether one of them was close to a true extremum. Otherwise, the identification was deemed a false positive. Analogously, peaks/pits missed were deemed false negatives.

The effect of noise. To evaluate the influence of noise, we perturbed the sensed signal with random Gaussian noise of zero mean and progressively larger variance. This experiment focused on well-connected topologies (average degree between 13.7 and 14.3) and was devoid of sparsity effects. The average diameter of the graph was 25.4. Figure 4 summarizes our results over ten different random topologies, ten signals, and six signal-to-noise ratios (SNR).

We have two main observations: *First, the method by Jeong* et al. *can be inaccurate for small SNR.* Even though the method improved over the unfiltered signal, it exhibited a median precision of 0.12 for SNR = −5 dB. *Second, the accuracy of the scale-space approach, in both precision and recall, is on par with the oracle filter.* It should not be a surprise that a hand-tuned graph filter is exceptionally effi-

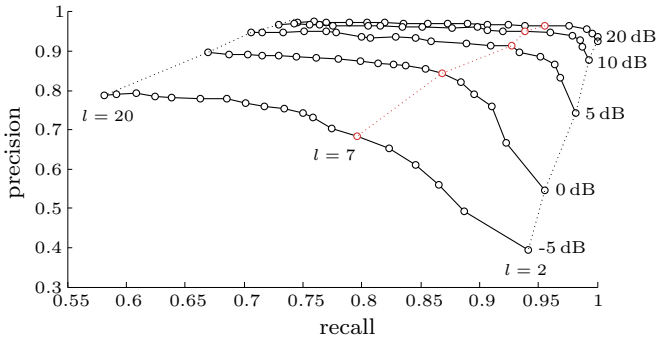


Figure 5: Precision/recall trade-off of scale-space for different lifetime thresholds l and SNRs. The points annotated in red correspond to the lifetime threshold used in our evaluation ($l = 7$).

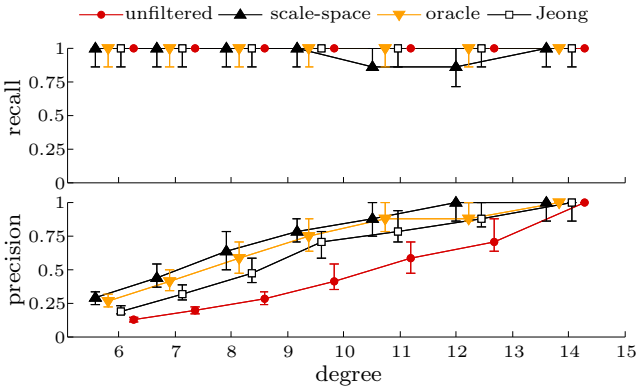


Figure 6: Precision/recall median and standard deviation for different methods and average node degrees. To diminish overlap, we perturbed the horizontal positioning of data-points. The dip in the scale-space recall (top figure, average degree between 11 and 12) corresponds to the smallest perceivable error from our experimental setup and is attributed to experimental variability.

cient in removing noise. It is however noteworthy that the scale-space approach could achieve low error over so diverse noise-levels with a single parametrization, a lifetime threshold of $l = 7$. Extremum lifetime is little affected by small signal perturbations. In fact, to demonstrate the robustness of our approach to incorrect parametrization, we used the same threshold ($l = 7$) in all simulations and experiments.

But how does the lifetime threshold l affect the identification accuracy in noisy signals? In Figure 5 we depict five precision-recall (PR) curves of the same experiment—each curve corresponding to one SNR. The value of l ranges from 2 to 20. The maximum value of 20 is way larger than required to identify any of the seven peaks, but it was used to be exhaustive in our exploration. As usual, a trade-off exists between being too selective (high l) and not selective enough (small l): a threshold that achieves high precision suffers in recall, and vice versa. *Nevertheless, we found that our method improved upon state-of-the-art over all reasonable parameterizations.* The scale-space approach outperformed the method by Jeong *et al.* in precision for all tested l, and in recall for $2 \leq l \leq 10$. Note that, setting $l > 10$ would imply a peak diameter close to 20, which is significantly larger than in our experiments.

The effect of sparsity. As shown by Proposition 2, sparsity presents a major challenge by introducing false extrema, *i.e.*, extrema not present in the physical signal. We evaluated the impact of sparsity by progressively decreasing the transmission radius of nodes (no noise was introduced). This resulted in increasingly sparser networks with average degree varying from 5.61 to 14.3. Figure 6 summarizes our results over 700 runs (7 radii × 10 physical signals × 10 runs).

To begin with, the results confirm that sparsity severely affects precision. We found that the number of false extrema increases exponentially as the average node degree decreases. The exponential trend has a devastating effect on identification accuracy. A case in point is that, for the smallest transmission radius (leftmost errorbars in the figure) the unfiltered signal contained ≈ 50 maxima (average) compared to the seven peaks of the physical signal. Hence, even in the absence of noise, some filtering is necessary. *We can also see that the scale-space approach achieves a slightly higher precision than the state-of-the-art, while not significantly sacrificing recall.* In the absence of noise, the scale-space approach and Jeong's method trade-off precision for recall. Given the overwhelming number of false extrema (false positives) present in sparse graphs, optimizing for precision is very desirable. To give a sense of perspective, when the average degree was 8.25 the scale-space approach had zero false negatives and four false positives, whereas the method by Jeong *et al.* had no false negatives but eight false positives (in median).

It is important to remark the lack of accuracy of the oracle filter. If the scale of the oracle filter is hand-picked to maximize its precision/recall, why does it behave so poorly? Furthermore, how does the scale-space approach improve upon it (even slightly)? Sparsity affects a graph filter in two distinct ways. Whereas in small scales a signal is overwhelmed by false extrema, in larger scales filtering causes phantom extrema. This is a challenge that a simple filter cannot overcome: no single scale depicts the extrema of the physical signal clearly. On the contrary, the scale-space approach filters out false extreme at low scales while discarding phantom extrema at higher scales.

6.2 Experiments

Contrary to common assumption, the connectivity of wireless networks is highly irregular. Especially indoors, wireless links exhibit high spatial and temporal variability, phenomena which significantly affect peak and pit identification. To evaluate the accuracy of identification methods in real-world wireless networks, we implemented scale-space kernels in Contiki [7] and ran an extensive set of experiments in a 100-node wireless testbed deployed above the ceiling tiles of our office building. The scale-space computation was built on top of a simple CSMA MAC protocol with round-level synchronization (as discussed in Section 3.2, synchrony is a fundamental requirement of scale-space analysis). Experiments were conducted using 10 synthetic signals over 7 noise levels (from -5 dB to 25 dB), resulting in 70 distinct runs. Due to the limited size of our testbed, we inserted in each synthetic signal three well-separated Gaussian distributions

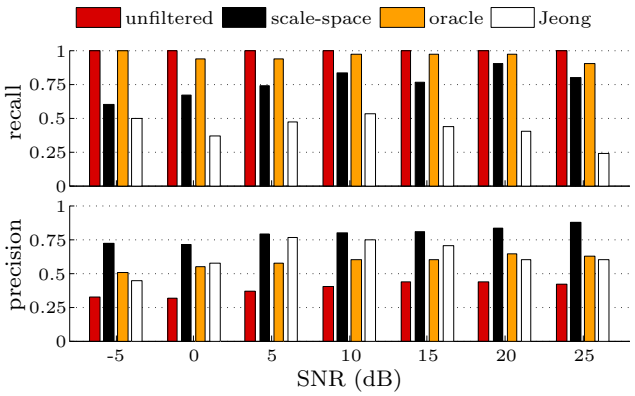


Figure 7: Average precision/recall for each noise level in our testbed. The precision of the oracle filter is significantly hampered by the poor correlation of wireless links with the underlying physical space.

with random mean and variance (as compared to seven in our simulations). The number of distributions a network can observe is directly related to its size. Intuitively, a node can observe only peaks that span an area larger than that of a neighborhood. We found experimentally that, even in perfect conditions, our testbed could not reliably detect the peaks of signals with more than three distributions. We used the extracted data (the signals' scale-space $\{y_s\}$ and per-round connectivity) to evaluate and compare the aforementioned methods offline. The number of nodes varied from 73 to 99, resulting in a network diameter between 9 and 10 hops, and an average degree between 7.17 and 9.5 neighbors. Figure 7 summarizes our results.

We provide three main observations: *First, the identification precision does not significantly increase when the SNR increases.* For the unfiltered signal, at 25 dB the average precision was only 0.1 higher than at -5 dB. The fact that the precision was low even at high SNR is due to the testbed being not only (relatively) sparse, but also differently connected from the random geometric graphs used in our simulations. We found that a significant number of false extrema were caused by graph irregularities, as depicted in Figure 2(a). Nodes spatially adjacent to a true extremum were not always in wireless proximity, thus wrongly identifying themselves as extrema. At low SNR, the number of false positives remained bounded (up to 15), which is not surprising because the number of false extrema (either maxima or minima) is at most equal to the size of the graph's maximal independent set. *Second, the identification recall of Jeong's method was much smaller than in simulations.* We found that the close proximity between peaks, as well as link asymmetry, caused false negatives. We argue that, by testing existing methods in a real-world scenario, we were able to observe behaviors (problems) never observed before. We therefore suggest that future mechanisms should include testbed experiments in order to be evaluated in conditions as diverse as possible. *Third, the scale-space approach was significantly more precise than the oracle filter.* Different from our simulation, the irregular patterns of real wireless links can cause lots of phantom extrema (*cf.* Section 5.2). In fact, phantom extrema can be so severe, that noise effects pale in comparison. In our experiments, phantom extrema were abundant at almost all scales, severely hampering the precision of any (single-scale) graph filter, and thus of the oracle filter. In contrast, the scale-space approach was able to distinguish and discard phantom extrema using the birth criterion. In summary, our testbed experiments demonstrated that, in real-word conditions, it is essential for a method to be resilient to phantom and false extrema.

7. CONCLUSIONS

According to scale-space theory, to understand the structures of a signal, one must observe it at all possible scales. But can we do so when the signal is defined on an irregular graph? By identifying the scale-space kernels appropriate for graphs and studying their computation, this paper in effect extends scale-space theory to graphs. We demonstrated the value of the scale-space approach by applying it to the problem of peak and pit identification in sensor networks. Beyond peak identification, we believe that principles of scale-space analysis can be beneficial for other problems in sensor networks. For instance, the deep structure could be used to extend current topological methods for signal mapping and compression to multiple scales [22, 29], thus making distributed pattern recognition possible. A second potential application is event-region detection [18]. Tracking event-boundaries across scale has the potential to improve the resilience of current algorithms to phantom effects.

Our work takes a step towards the distributed scale-invariant analysis of graph signals. However, it also opens up new challenges. The first challenge concerns the extension of the theory to multiple dimensions. The results in this paper are limited to graph signals with one value at each node ($x \in \mathbb{R}^n$). It is still an open question how the theory can be applied to higher dimensional graph signals (*e.g.*, $x \in \mathbb{R}^{k\times n}$ or $\mathbb{C}^n$). We also found that filtering in graphs can cause phantom effects and we gave bounds on the induced error. Yet the process is little understood. For instance how does this error relate to the properties of the underlying graph?

Acknowledgements. Andreas Loukas and Marco Zuniga were supported by the Dutch Technology Foundation STW and the Technology Program of the Ministry of Economic Affairs, Agriculture and Innovation (D2S2 project). Jie Gao would acknowledge the federal government support through NSF DMS-1418255, NSF DMS-1221339, NSF CNS-1217823, and AFOSR FA9550-14-1-0193. Last, we would like to thank the anonymous reviewers and our shepherd Urbashi Mitra for providing us valuable feedback.

References

[1] Y. Baryshnikov and R. Ghrist. Target enumeration via integration over planar sensor networks. *Robotics: Science and Systems IV*, 2008.

[2] P. Basu, A. Nadamani, and L. Tong. Extremum tracking in sensor fields with spatio-temporal correlation. In *Int. Conf. on Militaty Communications*. IEEE, 2010.

[3] S. P. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] A. Carzaniga, C. Hall, and M. Papalini. Fully decentralized estimation of some global properties of a network. In *INFOCOM*, 2012.

[5] Y. Cheng. Mean shift, mode seeking, and clustering. *Trans. on Pattern Analysis and Machine Intelligence*, 1995.

[6] F. Chung. The heat kernel as the pagerank of a graph. *National Academy of Sciences*, 2007.

[7] A. Dunkels, B. Gronvall, and T. Voigt. Contiki-a lightweight and flexible operating system for tiny networked sensors. In *Int. Conf. on Local Computer Networks*. IEEE, 2004.

[8] Q. Fang, F. Zhao, and L. Guibas. Lightweight sensing and communication protocols for target enumeration and aggregation. In *Int. Symp. on Mobile ad hoc networking and Computing*. ACM, 2003.

[9] J. Faruque, K. Psounis, and A. Helmy. Analysis of gradient-based routing protocols in sensor networks. In *DCOSS*. Springer, 2005.

[10] M.-H. Jeong and M. Duckham. A coordinate-free, decentralized algorithm for monitoring events occurring to peaks in a dynamic scalar field. In *ISSNIP*. IEEE, 2013.

[11] M.-H. Jeong, M. Duckham, and A. Kealy. Decentralized and coordinate-free computation of critical points and surface networks in a discretized scalar field. *Int. Journal of Geographical Information Science*, 2014.

[12] R. I. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete input spaces. In *Int. Conf. on Machine Learning*, 2002.

[13] A. Kuijper. *The deep structure of Gaussian scale space images*. PhD thesis, Utrecht University, 2002.

[14] T. Lindeberg. Scale-space for discrete signals. *Trans. on Pattern Analysis and Machine Intelligence*, 1990.

[15] T. Lindeberg. Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention. *Int. Journal of Computer Vision*, 1993.

[16] T. Lindeberg. *Scale-space theory in computer vision*. Springer, 1993.

[17] T. Lindeberg. *Generalized Gaussian Scale-Space Axiomatics Comprising Linear Scale-Space, Affine Scale-Space and Spatio-Temporal Scale-Space*. Springer, 2010.

[18] A. Loukas, M. A. Zúñiga, I. Protonotarios, and J. Gao. How to identify global trends from local decisions? event region detection on mobile networks. In *INFOCOM*. IEEE, 2014.

[19] A. Loukas, M. A. Zúñiga, M. Woehrle, M. Cattani, and K. Langendoen. Think globally, act locally: On the reshaping of information landscapes. In *IPSN*. ACM/IEEE, 2013.

[20] L.-Z. Lu and C. E. M. Pearce. Some new bounds for singular values and eigenvalues of matrix products. *Annals of Operations Research*, 2000.

[21] A. Sandryhaila and J. M. Moura. Discrete signal processing on graphs: Frequency analysis. *Trans. on Signal Processing*, 2014.

[22] R. Sarkar, X. Zhu, J. Gao, L. J. Guibas, and J. S. Mitchell. Iso-contour queries and gradient descent with guaranteed delivery in sensor networks. In *INFOCOM*. IEEE, 2008.

[23] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst. The Emerging Field of Signal Processing on Graphs: Extending High-Dimensional Data Analysis to Networks and Other Irregular Domains. *IEEE Signal Processing Magazine*, 2013.

[24] D. I. Shuman, P. Vandergheynst, and P. Frossard. Chebyshev polynomial approximation for distributed signal processing. In *DCOSS*. IEEE, 2011.

[25] A. J. Smola and R. Kondor. Kernels and regularization on graphs. In *Learning theory and kernel machines*. Springer, 2003.

[26] A. P. Witkin. Scale-space filtering: A new approach to multi-scale description. In *ICASSP*. IEEE, 1984.

[27] F. Zhang and E. R. Hancock. Image scale-space from the heat kernel. In *Progress In Pattern Recognition, Image Analysis And Applications*. Springer, 2005.

[28] F. Zhang and E. R. Hancock. Graph spectral image smoothing using the heat kernel. *Pattern Recognition*, 2008.

[29] X. Zhu, R. Sarkar, and J. Gao. Topological data processing for distributed sensor networks with morse-smale decomposition. In *INFOCOM*. IEEE, 2009.

APPENDIX

Proposition 2. *The expected number of maxima (minima) X appearing if k out of m edges of a random geometric graph are deleted at random is $\mathrm{E}[X] > n\,(e^{k/m} - e^{-m/n} - \epsilon)$, where ϵ becomes negligibly small when $k > m/n$.*

Proof. The event that a node u becomes an extremum after deleting k edges is described by an indicator r.v. $X_u = \{0, 1\}$. X_u depends on the number Y_u of neighbors $v \sim u$ with $x(v) \geq x(u)$ (correspondingly $x(v) \leq x(u)$). Because u becomes a maximum (minimum) if all edges to neighbors with larger (smaller) values are deleted, we have that

$$\mathrm{E}[X_u = 1 | Y_u = y] = \mathrm{P}(X_u = 1 | Y_u = y) = \binom{k}{y}\binom{m}{y}^{-1}.$$

Moreover, from independence, the expected number of appearing maxima (minima) X is

$$\mathrm{E}[X] = \sum_{u \in \mathcal{V}} \mathrm{E}[X_u] = n \sum_{y=1}^{k} \mathrm{P}(X_u = 1 | Y_u = y)\mathrm{P}(Y_u = y).$$

Since Y_u is always smaller than the node degree, we obtain a lower bound by substituting Y_u with the degree r.v. D. In particular, for a random geometric graph with average degree m/n,

$$\begin{aligned}\mathrm{E}[X] &> n \sum_{i=1}^{k} \binom{k}{i}\binom{m}{i}^{-1} \frac{(m/n)^i e^{-m/n}}{i!} \\ &= n\,(e^{k/m} - e^{-m/n} - \epsilon),\end{aligned}$$

where $\epsilon = 1 - \Gamma(k, m/n)/\Gamma(k, 0)$ is negligibly small for $k > m/n$ and Γ is the incomplete gamma function. □

Theorem 2. *No 1-st order recursion computes $\mathbf{K}_s$ in the asynchronous model.*

Proof. In the most general sense, 1-st order recursions are given by

$$y^{(t+1)} = A^{(t)} y^{(t)} + B^{(t)} x \;\text{ and }\; y^{(0)} = x, \tag{6}$$

where, because $\{A^{(t)}\}$ and $\{B^{(t)}\}$ are sequences of 1-local matrices, nodes only directly communicate with their neighbors. By standard arguments, (6) is stable if, $|\lambda_{\max}(A^{(t)})| < 1$ for all t. To ensure feasibility in the asynchronous model we have to additionally impose that no τ exists for which $A^{(t)} = 0_{n\times n}$ or $B^{(t)} = 0_{n\times n}$ and for all $t > \tau$. Indeed, if such τ exists system (6) degenerates to (2), which is synchronous. The steady state is of the system is

$$\mathbf{K}'x = \lim_{t\to\infty} \sum_{\tau=0}^{t-1} \Phi(t-1, t-\tau) B^{(t-\tau-1)} x, \tag{7}$$

where $\Phi(t_1, t_2) = A^{(t_1)} A^{(t_1-1)} \dots A^{(t_2)}$ when $t_1 \geq t_2$ and $\Phi(t_1, t_2) = I$ otherwise.

To deduce that $\mathbf{K}' \neq \mathbf{K}_s$, we will consider the signal information each kernel depends on. We will show that, whereas $\mathbf{K}_s$ only considers the signal in an s-hop vicinity around each node, $\mathbf{K}'$ depends on the entire signal. The two matrices therefore cannot be equal. From Theorem 1, $\mathbf{K}_s$ is exactly s-local. But do matrix sequences $\{A^{(t)}\}$ and $\{B^{(t)}\}$ exist such that $\mathbf{K}'$ is s-local? Being a product of $(t_1 - t_2 + 1)$ 1-local matrices, $\Phi(t_1, t_2)$ is $(t_1 - t_2 + 1)$-local. Furthermore, it is non-zero: if $A^{(k)} \neq 0$ for every $k \in [t - \tau, t - 1]$,

$$\|\Phi(t-1, t-\tau)\| \geq \prod_{k=t-\tau}^{t-1} \sigma_{min}(A^{(k)}) > 0,$$

where σ_{min} is the minimum singular value [20]. Terms $\Phi(t-1, t-\tau)B^{(t-\tau-1)}x$ thus violate the locality requirement whenever $t \geq s + \tau - 2$. Thus $\mathbf{K}'$ can only be s-local if the coefficients of successive terms cancel out, which is impossible as (from Theorem 1) all coefficients are non-negative. □

Corollary 1. *If a node u_s has a locally maximal value at scale s, but not at scale $s+1$, the gradient ascent path from u_s to the new local maximum u_{s+1} can have arbitrary length.*

Proof. We construct an example in which u_s and u_{s+1} are k hops away. Because of the semi-group axiom, it is sufficient to give an example for $s = 0$, $y_s = x$ and $y_{s+1} = y_1$. Let $\mathcal{G}$ a line graph and x a signal with strictly monotonically increasing values $y_s(u_{i+1}) = y_s(u_i) + B$ for all $u_i \in \mathcal{V}$. Create a maximum at u_i by setting $y_s(u_i) = y_s(u_{i+1}) + \beta$ for some small positive β. We will show that for any scale space kernel, $y_1(u_{i-1}) < y_1(u_i) < y_1(u_{i+1})$ if $B > \beta/3$. The maximum will therefore move until it reaches node u_n ($k = n - i$ hops) at the end of the line graph. The value of a node u_i at scale 1 is $\alpha_l x(u_{i-1}) + \alpha_c x(u_i) + \alpha_r x(u_{i+1})$, where α are kernel dependent positive coefficients for the left (α_l), center (α_c), and right (α_r) neighbors. Furthermore, $\alpha_l = \alpha_r$ as the nodes in a line-graph have no way of distinguishing between their left and right neighbors. Inequality $y_1(u_{i-1}) < y_1(u_i)$ holds if: (*i*) $\alpha_l x(u_{i-2}) < \alpha_r x(u_{i+1})$ and (*ii*) $(\alpha_c - \alpha_l)x(u_{i-1}) < (\alpha_c - \alpha_r)x(u_{i-2})$. Due to our construction, both inequalities are satisfied. Following the same reasoning, inequality $y_1(u_i) < y_1(u_{i+1})$ holds if $3B = x(u_{i+2}) - x(u_{i-1}) > x(u_i) - x(u_{i+1}) = \beta$. □

Corollary 2. *The absolute value of an extremum always decreases, $|y_s(u_s)| \geq |y_{s+1}(u_{s+1})|$, where u_s is the node the extremum resides on at scale s.*

Proof. Let $u_s \neq u_{s+1}$ the nodes over which an unstable extremum e resides on at scales s and $s+1$, respectively. W.l.o.g., the extremum is a maximum and the claim is that $y_s(u_s) \geq y_{s+1}(u_{s+1})$. By Theorem 1, the value of a node in scale $s+1$ is bounded by the maximum value at most 1-hop away from it in scale s (this includes the node itself). Thus, the only way that the inequality does not hold is that there exists node v at most 1-hop away from u_{s+1} with $y_s(v) > y_s(u_s)$. This implies that a second extremum e' exists near u_{s+1} which is reachable by gradient ascent. Moreover, e collapses to e' at $s+1$, a contradiction. □

Lemma 3. *For any two graphs $\overline{\mathcal{G}} = (\mathcal{V}, \overline{\mathcal{E}})$ and $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, signal $x \in \mathbb{R}^n$, and scale $s \in \mathbb{N}$, $e_s \leq 1 - \lambda_{min}^s + \epsilon_s$, where $\epsilon_s = (2s^s)/(s+1)^{s+1}$ when s is odd and zero otherwise. Furthermore, $-1 < \lambda_{min} < 1$ is the smallest amongst the eigenvalues of $\overline{S}$ and S.*

Proof. Denote by $\lambda_{\max}(S)$ and $\lambda_{\min}(S)$ the largest and smallest eigenvalue of a matrix S. It is well known that for any symmetric matrix S, $\lambda_{\min}(S)x^\top x \leq x^\top S x \leq \lambda_{\max}(S)x^\top x$ [3]. Furthermore, as S's spectral radius is smaller or equal to one, $\lambda_{\max}(S) = 1$ and $-1 < \lambda_{\min}(S) < 1$. Combining the inequalities for matrices $\overline{S}^s$ and S^s we have that

$$\lambda_{\min}^s(\overline{S}) - 1 \leq \frac{x^\top(\overline{S}^s - S^s)x}{x^\top x} \leq 1 - \lambda_{\min}^s(S).$$

Setting $\lambda_{min} = \min(\lambda_{\min}(S), \lambda_{\min}(\overline{S}))$, we get a first upper bound $e_s \leq 1 - \lambda_{min}^s$. When s is odd, $1 - \lambda_{min}^s > 1$, which is loose. We can improve upon it using the triangle inequality:

$$\begin{aligned} e_{s+1} &= \left| \frac{x^\top\left(S^{s+1} - S^s + \overline{S}^s - \overline{S}^{s+1} + S^s - \overline{S}^s\right)x}{x^\top x} \right| \\ &\leq \left| \frac{x^\top(S^{s+1} - S^s)x}{x^\top x} \right| + \left| \frac{x^\top(\overline{S}^{s+1} - \overline{S}^s)x}{x^\top x} \right| + e_s \end{aligned} \quad (8)$$

Term $x^\top(S^{s+1} - S^s)x$ captures the distance traveled in one step by each of the two dynamical systems and is

$$\begin{aligned} \left| \frac{x^\top(S^{s+1} - S^s)x}{x^\top x} \right| &= \left| \frac{\sum_{k=1}^n \left(\lambda_k^{s+1} - \lambda_k^s\right)(\psi_k^\top x)^2}{x^\top x} \right| \\ &\leq \frac{\left| \sum_{k=1}^n \left(\left(\frac{s}{s+1}\right)^{s+1} - \left(\frac{s}{s+1}\right)^s \right)(\psi_k^\top x)^2 \right|}{x^\top x} \\ &= \frac{s^s}{(s+1)^{s+1}}. \end{aligned} \quad (9)$$

Above, ψ_k is the k-th eigenvector of S and, in the second step, we bounded the convex function $\lambda_k^{s+1} - \lambda_k^s$ using its minimum value. Substituting (9) into (8), we obtain the desired bound. □

Lemma 4. *For any two graphs $\mathcal{G}$ and $\overline{\mathcal{G}}$, signal $x \in \mathbb{R}^n$, and scale $s \in \mathbb{N}$, $e_{s+1} \geq \max(0,\, e_\infty - 2\lambda_2^s)$ where $0 < \lambda_2 = \min(|\lambda_2(S)|, |\lambda_2(\overline{S})|) < 1$ is the smallest of the second eigenvalues of matrices S and $\overline{S}$.*

Proof. We will express $S^s - \overline{S}^s$ in terms of the e_∞. Because scale-space kernels are marginally stable, e_∞ is bounded. Furthermore, it is easily computable when scale-space matrices S and $\overline{S}$ are known.

$$\begin{aligned} e_s &= \left| \frac{x^\top\left(S^\infty - \overline{S}^\infty + \overline{S}^\infty - \overline{S}^s - S^\infty + S^s\right)x}{x^\top x} \right| \\ &\geq e_\infty - \left| \frac{x^\top(\overline{S}^\infty - \overline{S}^s)x}{x^\top x} \right| - \left| \frac{x^\top(S^\infty - S^s)x}{x^\top x} \right| \end{aligned} \quad (10)$$

Note that above we abuse notation and refer to $\lim_{s\to\infty} S^s$ as S^∞. For both the second and third terms in (10) the following simple upper bound holds:

$$\left| \frac{x^\top(S^\infty - S^s)x}{x^\top x} \right| = \left| \frac{\sum_{s=2}^n \lambda_i^s(S)(\psi_i^\top x)^2}{x^\top x} \right| \leq |\lambda_2^s(S)| \quad (11)$$

Substituting (11) into (10), we obtain the lower bound

$$e_s \geq e_\infty - \lambda_2(S)^s - \lambda_2(\overline{S})^s \geq e_\infty - 2\lambda_2^s \quad (12)$$

which holds as long as $s \geq \log(\lambda_2)\, e_\infty$. Otherwise, we use the trivial lower bound $e_s \geq 0$. □