

# On the game $p$ -Laplacian on weighted graphs with applications in image processing and data clustering<sup>†</sup>

A. ELMOATAZ<sup>1</sup>, X. DESQUESNES<sup>2</sup> and M. TOUTAIN<sup>3</sup>

<sup>1</sup>Université de Caen Normandie and the ENSICAEN in the GREYC Laboratory, Image Team, 6 Boulevard Maréchal Juin, F-14050 Caen Cedex, France  
email: [abderrahim.elmoataz-billah@unicaen.fr](mailto:abderrahim.elmoataz-billah@unicaen.fr)

<sup>2</sup>University of ORLEANS, Orleans, France  
email: [xavier.desquesnes@univ-orleans.fr](mailto:xavier.desquesnes@univ-orleans.fr)

<sup>3</sup>Datexim SAS, 51 Avenue de la Côte de Nacre, 14000 Caen Cedex, France  
email: [matthieu.toutain@unicaen.fr](mailto:matthieu.toutain@unicaen.fr)

(Received 19 October 2016; revised 10 May 2017; accepted 16 May 2017;  
first published online 3 July 2017)

Game-theoretic  $p$ -Laplacian or normalized  $p$ -Laplacian operator is a version of classical variational  $p$ -Laplacian which was introduced recently in connection with stochastic games called Tug-of-War with noise (Peres *et al.* 2008, Tug-of-war with noise: A game-theoretic view of the  $p$ -laplacian. *Duke Mathematical Journal* **145**(1), 91–120). In this paper, we propose an adaptation and generalization of this operator on weighted graphs for  $1 \leq p \leq \infty$ . This adaptation leads to a partial difference operator which is a combination between 1-Laplace, infinity-Laplace and 2-Laplace operators on graphs. Then we consider the Dirichlet problem associated to this operator and we prove the uniqueness and existence of the solution. We show that the solution leads to an iterative non-local average operator on graphs. Finally, we propose to use this operator as a unified framework for interpolation problems in signal processing on graphs, such as image processing and machine learning.

**Key words:** Game  $p$ -Laplacian, graph signal processing, image processing, machine learning, Tug-of-War games

## 1 Introduction

### 1.1 Introduction and motivations

The local and non-local  $p$ -Laplace and infinity-Laplace operators have been well studied in both the discrete and continuous setting. These operators in the local continuous setting play an important role in geometry and Partial Differential Equations (PDEs) and enter in many mathematical problems and models for applications in physics, mechanics, image processing or computer vision. For an introduction and survey on these topics,

<sup>†</sup> This work was funded under a doctoral grant supported by the Coeur et Cancer association, the regional council of Normandy, the European FEDER Grant (PLANUCA Project) and the project ANR GRAPH SIP.

see [17, 29] and references therein. Non-local diffusion problems involving non-local continuous version of  $p$ -Laplacian and infinity-Laplacian are of great interest of pure mathematical problems [4, 13], they arise also in a wide variety of applications including biology, image processing and computer vision. For theory and applications, see [27] and the recent book [4] and references therein. On the other hand, the discrete Laplacian or graph Laplacian has been extensively used in machine learning for clustering [14] and dimension reduction [12]. The graph  $p$ -Laplacian, generalization of the standard Laplacian has started to attract attention from mathematical, machine learning, image and manifolds processing community. See [8, 9, 20, 28] and references therein.

The game theoretic  $p$ -Laplacian is another version of the continuous  $p$ -Laplacian, it was recently introduced in connection with the stochastic game model called Tug-of-War with noise [36]. A part of interest of this class of operator is that it contains, as particular cases, the  $\infty$ -Laplace operator for  $p = \infty$ , the mean curvature operator for  $p = 1$  and a multiple of the ordinary Laplace operator for  $p = 2$ . In the homogeneous case, the game  $p$ -Laplacian equation coincides with the variational  $p$ -Laplacian equation, for which several approximations have been proposed. Some of these schemes are based on finite element [6]. Several finite difference approximation schemes were proposed for the game  $p$ -Laplacian for  $p = 1$ ,  $p = \infty$  and  $p \geq 2$  [35]. More recently, an approximation scheme based on the semi-Lagrangian scheme was proposed in [24]. One can also quote the approximations of the normalized  $p$ -Laplacian for  $1 \leq p \leq \infty$  by statistical operators [38]. However, the most of these discretizations are proposed for regular domains. Recently, there has been a growing interest in the transcription and the adaptation of PDEs on general domains, or graph of arbitrary topology. First, as a subject of its own interest. Second, for their potential and existing applications.

Indeed, by now, more and more contemporary applications deal with a large amount of data which are collected in the form of graphs, networks and functions on these graphs (images, surfaces, meshes, social networks,...). Processing and analysing these kinds of data is a major challenge for both image and machine learning communities. Hence, it is very important to transfer many mathematical tools such as PDEs or wavelets, which are initially developed on usual Euclidean space and proven to be efficient for many problems and applications dealing with usual image and signal domains, to graph and networks.

In previous works, we have used the framework of Partial difference Equations (PdEs) which enables us to translate and adapt continuous PDEs and variational methods to the setting of graphs [7, 9, 20, 40, 41]. Conceptually, the idea of introducing PdEs is to mimic continuous PDEs on graph structures by consistently adapting important mathematical concepts, e.g., integration and differentiation. By doing so, one is able to directly translate most of the established techniques for PDEs and in particular for the  $p$ -Laplace operator to graphs. Moreover, in case of evolution PdEs, this framework provides models in which spatial integration and temporal evolution can be handled separately. Indeed, in this case, there is no need for a spatial discretization and one gains a unification of local and non-local models [21]. In our previous works, we have also introduced a family of  $p$ -Laplacian operators (isotropic and anisotropic) on graphs in divergence form [9] within this framework. For general formulation of  $p$ -Laplacian and  $\infty$ -Laplacian on graphs with gradient terms, see also [22, 23].

In this paper, we propose an adaptation and generalization of the game  $p$ -Laplacian on weighted graphs of arbitrary topology using the framework of PdEs [20, 40]. This adaptation leads to a new class of  $p$ -Laplacian on graph in non-divergence form which interpolates between non-local 1-Laplacian, non-local  $\infty$ -Laplacian and non-local 2-Laplacian, on graphs. We show the connections of this operator with local and non-local continuous normalized  $p$ -Laplacian. Then we consider the Dirichlet problem associated with these operators and we prove the existence and the uniqueness of the solution. We show that our proposed operator is an extension on weighted graphs of the well-known numerical approximation of Oberman of the 1-Laplacian and  $\infty$ -Laplacian [35]. We also show the connection with local and non-local Tug-of-War games. In the continuous domain,  $p$ -harmonic functions are used to interpolate functions defined on Euclidean domains. Their extension on graph allows to express  $p$ -harmonic functions on irregular domains and to use these functions as a unified framework to interpolate and solve various problems both in image processing (such as inpainting) and in machine learning (e.g., semi-supervised classification). Finally, we propose to use the game  $p$ -Laplacian on graph as a framework to solve many interpolation problems on graph with applications in image processing and machine learning.

### 1.2 Game $p$ -Laplacian

We recall that the variational  $p$ -Laplacian of a function  $f : \Omega \subset \mathbb{R}^N \rightarrow \mathbb{R}$  is given for  $1 \leq p < \infty$  as

$$\Delta_p f = \operatorname{div}(|\nabla f|^{p-2} \cdot \nabla f). \tag{1.1}$$

In the case where  $p = \infty$ , it is traditionally given by  $\Delta_\infty f = \sum_{i,j}^n \frac{\partial f}{\partial x_i} \frac{\partial f}{\partial x_j} \frac{\partial^2 f}{\partial x_i \partial x_j}$ .

The game  $p$ -Laplacian or normalized  $p$ -Laplacian, recently introduced in [37] to model a stochastic game called Tug-of-War game is written as, for  $1 \leq p < \infty$ :

$$\Delta_p^N f = \frac{1}{p} |\nabla f|^{2-p} \cdot \operatorname{div}(|\nabla f|^{p-2} \cdot \nabla f). \tag{1.2}$$

When  $p = \infty$ , the normalized  $p$ -Laplacian is written as

$$\Delta_\infty^N f = |\nabla f|^{-2} \Delta_\infty f. \tag{1.3}$$

$\Delta_p^N f$  is called normalized since it is homogeneous of degree 1, i.e.,  $\Delta_p^N(t f) = t \cdot \Delta_p^N f$  for  $t \in \mathbb{R}$  in contrast to the variational  $p$ -Laplacian which is homogeneous of degree  $p - 1$ . Thus, the parabolic problems involving the normalized  $p$ -Laplacian are scaling invariant. This is a useful attribute in the context of image processing. If  $f$  is a smooth function, equation (1.2) can be rewritten as

$$\begin{aligned} \Delta_p^N f &= \frac{(p-2)}{p} \Delta_\infty^N f + \frac{1}{p} \Delta f, \\ &= \frac{(p-2)}{p} \Delta_\infty^N f + \frac{2}{p} \Delta_2^N f, \\ &= \alpha(p) \Delta_\infty^N f + \beta(p) \Delta_2^N f, \end{aligned} \tag{1.4}$$

with  $\alpha(p) = (p - 2)/p$  and  $\beta(p) = 2/p$ .

The game  $p$ -Laplacian for  $p = 1$  can be written as

$$\Delta_1^N f = \operatorname{div}\left(\frac{\nabla(f)}{|\nabla(f)|}\right)|\nabla(f)|. \tag{1.5}$$

As  $\Delta_1^N f = \Delta - \Delta_\infty^N f$ , equation (1.4) can be rewritten as

$$\Delta_p^N f = \alpha'(p)\Delta_2^N f + \beta'(p)\Delta_1^N f, \tag{1.6}$$

with  $\alpha'(p) = \frac{2(p-1)}{p}$  and  $\beta'(p) = \frac{2-p}{p}$ .

One can remark that for the equation (1.4), to get both the  $\alpha$  and  $\beta$  functions positive, we set the value interval for  $p$  as  $2 \leq p \leq \infty$ . Similarly, for (1.6), the positive value interval for  $\alpha'$  and  $\beta'$  is  $1 \leq p \leq 2$ .

The game  $p$ -Laplacian for  $1 \leq p \leq \infty$  can be rewritten as

$$\Delta_p^N f = \begin{cases} \frac{2}{p}\Delta_2^N f + \frac{p-2}{p}\Delta_\infty^N f & \text{for } 2 \leq p \leq \infty \\ \frac{2(p-1)}{p}\Delta_2^N f + \frac{2-p}{p}\Delta_1^N f & \text{for } 1 \leq p \leq 2. \end{cases} \tag{1.7}$$

### 1.3 Paper organization

The rest of the paper will be organized as follows. In Section 2, we provide definitions and notations used in this work. In Section 3, we present a new family of  $p$ -Laplacian on graphs, as non-local normalized  $p$ -Laplacian. Then, we study existence and uniqueness of the Dirichlet problem associated with the  $p$ -Laplacian equation. Section 4 presents several applications to some interpolation problems in image and high-dimensional data. Finally, section 5 concludes the paper.

## 2 Partial difference equations on graphs

In this section, we review some definitions of operators involved in this paper. All these definitions are borrowed from [19, 20, 40].

### 2.1 Notations

Let us consider the general situation where any discrete domain can be seen as a weighted graph. A weighted graph  $G = (V, E, w)$  consists in a finite set  $V$  of  $N$  vertices and in a finite set  $E \subseteq V \times V$  of edges. Let  $(u, v)$  be the edge that connects vertices  $u$  and  $v$ . An undirected graph is weighted if it is associated with a weight function  $w : V \times V \rightarrow [0, 1]$ . The weight function represents a similarity measure between two vertices of the graph. According to the weight function, the set of edges is defined as  $E = \{(u, v) | w(u, v) > 0\}$ . The degree of a vertex  $u$  is defined as  $\mu(u) = \sum_{v \sim u} w(u, v)$  where notation  $v \sim u$  means that the vertex  $v$  is adjacent to  $u$ . The neighbourhood of a vertex  $u$  (i.e., the set of vertices adjacent to  $u$ ) is denoted  $N(u)$ . Let  $\mathcal{H}(V)$  be the Hilbert space of real-valued functions on the vertices of the graph. Each function  $f : V \rightarrow \mathbb{R}$  of  $\mathcal{H}(V)$  assigns a real value  $f(u)$  to each vertex

$u \in V$ . Similarly, let  $\mathcal{H}(E)$  be the Hilbert space of real-valued functions defined on the edges of the graph. These two spaces are endowed with the following inner products:  $\langle f, g \rangle_{\mathcal{H}(V)} = \sum_{u \in V} f(u)g(u)$  with  $f, g \in \mathcal{H}(V)$ , and  $\langle F, G \rangle_{\mathcal{H}(E)} = \sum_{(u,v) \in E} F(u,v)G(u,v)$  where  $F, G \in \mathcal{H}(E)$ .

**2.2 Difference, divergence on graphs**

Let us fix a weighted graph  $G = (V, E, w)$ . The *difference operator*  $\mathcal{G}_w : \mathcal{H}(V) \rightarrow \mathcal{H}(E)$  is given for all  $f \in \mathcal{H}(V)$  and  $(u, v) \in E$  by  $(\mathcal{G}_w f)(u, v) = \sqrt{w(u, v)}(f(v) - f(u))$ .

The *adjoint operator* of the difference operator, denoted by  $\mathcal{G}_w^* : \mathcal{H}(E) \rightarrow \mathcal{H}(V)$ , is defined by  $\langle \mathcal{G}_w f, H \rangle_{\mathcal{H}(E)} = \langle f, \mathcal{G}_w^* H \rangle_{\mathcal{H}(V)}$ , with  $f \in \mathcal{H}(V)$  and  $H \in \mathcal{H}(E)$ . Using the definitions of the inner products in  $\mathcal{H}(V)$  and  $\mathcal{H}(E)$  and the definition of the difference operator, we obtain easily the expression  $\mathcal{G}_w^*$  at a vertex  $u : (\mathcal{G}_w^* H)(u) = \sum_{v \sim u} \sqrt{w(u, v)}(H(v, u) - H(u, v))$ .

The *divergence operator*, defined by  $\text{div}_w = -\mathcal{G}_w^*$ , measures the net outflow of a function in  $\mathcal{H}(E)$  at each vertex of  $V$ .

The *directional derivative* (or edge derivative) of a function  $f$  at a vertex  $u$  along an edge  $e = (u, v)$ , is defined as  $\partial_v f(u) = \sqrt{w(u, v)}(f(v) - f(u))$ .

Two *weighted directional difference operators* can be defined. The weighted directional external and internal difference operators are, respectively,

$$\begin{aligned} (\partial_v^+ f)(u) &= \sqrt{w(u, v)}(f(v) - f(u))^+ \text{ and} \\ (\partial_v^- f)(u) &= \sqrt{w(u, v)}(f(v) - f(u))^- \end{aligned} \tag{2.1}$$

with  $(x)^+ = \max(0, x)$  and  $(x)^- = -\min(0, x)$ .

**2.3 Gradient operators and  $p$ -Laplacian**

The *weighted gradient* of a function  $f \in \mathcal{H}(V)$  at vertex  $u$  is the vector of all edge derivatives:

$$(\nabla_w f)(u) = ((\partial_v f)(u))_{v \sim u}^T \tag{2.2}$$

Two discrete formulations of *weighted morphological gradients* on graphs are defined. The weighted external  $\nabla_w^+$  and internal  $\nabla_w^-$  gradient operators are, respectively,

$$(\nabla_w^+ f)(u) = ((\partial_v^+ f)(u))_{v \sim u}^T \tag{2.3}$$

$$(\nabla_w^- f)(u) = ((\partial_v^- f)(u))_{v \sim u}^T \tag{2.4}$$

We define the length of these gradients as follows:

$$\|(\nabla_w f)(u)\|_p = \left[ \sum_{v \in V} \sqrt{w(u, v)}^p |f(v) - f(u)|^p \right]^{\frac{1}{p}} \tag{2.5}$$

$$\|(\nabla_w^\pm f)(u)\|_p = \left[ \sum_{v \in V} \sqrt{w(u, v)}^p |(f(v) - f(u))^\pm|^p \right]^{\frac{1}{p}} \tag{2.6}$$

Another form of the length of gradients can be defined as

$$\|(\nabla_w f)(u)\|_\infty = \max_{v \in V} \left( \sqrt{w(u,v)} |f(v) - f(u)| \right), \tag{2.7}$$

$$\|(\nabla_w^\pm f)(u)\|_\infty = \max_{v \in V} \left( \sqrt{w(u,v)} |f(v) - f(u)|^\pm \right). \tag{2.8}$$

The non-local anisotropic  $p$ -Laplace operator of  $f \in \mathcal{H}(V)$  is defined for a vertex  $u \in V$  for  $1 \leq p < \infty$  as [21]:

$$\begin{aligned} \Delta_{w,p}(f)(u) &\stackrel{\text{def.}}{=} \operatorname{div}_w(|\mathcal{G}_w f|^{p-2} \mathcal{G}_w f)(u) \\ &= \sum_{v \sim u} \sqrt{w(u,v)}^p |f(v) - f(u)|^{p-2} (f(v) - f(u)). \end{aligned} \tag{2.9}$$

For  $p = 2$ , we obtain the 2-Laplacian as follows:

$$\Delta_{w,2}(f)(u) = \sum_{v \sim u} w(u,v)(f(v) - f(u)). \tag{2.10}$$

For  $p = 1$ , we obtain the following 1-Laplacian on graphs:

$$\Delta_{w,1}(f)(u) \stackrel{\text{def.}}{=} \sum_{v \sim u} \sqrt{w(u,v)} \operatorname{sign}(f(v) - f(u)), \tag{2.11}$$

where

$$\operatorname{sign}(x) \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{otherwise} \end{cases}. \tag{2.12}$$

The  $\infty$ -Laplacian on graph is defined as [18]

$$\Delta_{w,\infty}(f)(u) \stackrel{\text{def.}}{=} \frac{1}{2} \left( \|\nabla_w^+ f(u)\|_\infty - \|\nabla_w^- f(u)\|_\infty \right). \tag{2.13}$$

which can be rewritten as

$$\begin{aligned} \Delta_{w,\infty}(f)(u) &= \frac{1}{2} \left[ \max_{v \sim u} \left( \sqrt{w(u,v)} \max((f(v) - f(u)), 0) \right) \right. \\ &\quad \left. + \min_{v \sim u} \left( \sqrt{w(u,v)} \min((f(v) - f(u)), 0) \right) \right]. \end{aligned} \tag{2.14}$$

**Remark** As in the continuous case, this operator can be formally derived by the minimization of the following energy on graphs (for  $p \rightarrow \infty$ ):

$$J_{w,p}(f) = \frac{1}{2p} \sum_{u \in V} \|(\nabla_w f)(u)\|_p^p. \tag{2.15}$$

This is an extension of the well-known Oberman numerical approximation [34].

### 3 Game $p$ -Laplacian on graph

In this section, we propose an extension of the game  $p$ -Laplacian on weighted graphs. For this, we first introduce statistical operators, that will be used to define the normalized

$p$ -Laplacian we propose. We make the connection with various  $p$ -Laplacian formulations in the continuous local and non-local setting. Then we study the existence and uniqueness of the solution of the homogeneous normalized  $p$ -Laplacian equation on graphs with Dirichlet boundary conditions. We also show the connection of our newly defined operator with the Tug-of-War game.

### 3.1 Non-local statistical operators on weighted graphs

To propose a definition of the normalized  $p$ -Laplacian on graphs, we first define the following average operators, which are extensions of the classical average operators (mean, max, min, midrange, median):

$$\begin{aligned}
 \text{NLMean}(f)(u) &= \frac{\sum_{v \sim u} w(u,v)f(v)}{\sum_{v \sim u} w(u,v)}, \\
 \text{NLMax}(f)(u) &= \max_{v \sim u} (\sqrt{w(u,v)} \max(f(v) - f(u), 0)) + f(u), \\
 \text{NLMin}(f)(u) &= \max_{v \sim u} (\sqrt{w(u,v)} \max(f(u) - f(v), 0)) + f(u), \\
 \text{NLMidrange}(f)(u) &= \frac{1}{2}(\text{NLMin}(f)(u) + \text{NLMax}(f)(u)), \\
 \text{NLMedian}(f)(u) &= \text{median}((\nabla_w f)(u)) + f(u),
 \end{aligned}
 \tag{3.1}$$

where median is the classical discrete median operator defined as follows:  
 For  $x_i \in \mathbb{R}$  and  $i = 1 \dots, m$

$$\text{median}\{x_j\}_{1 \leq j \leq m} = \begin{cases} y_{\frac{m+1}{2}} & \text{if } m \text{ is odd} \\ \frac{y_{\frac{m}{2}} + y_{\frac{m}{2}+1}}{2} & \text{if } m \text{ is even,} \end{cases}
 \tag{3.2}$$

with  $\{y_1, \dots, y_m\}$  a non-decreasing arrangement of  $\{x_1, \dots, x_m\}$ .

One can see that by setting  $w(u,v) = 1$ , we recover the classical statistical mean, midrange and median filters.

**Definition** We propose to define the normalized version of the non-local 1-Laplacian, 2-Laplacian and  $\infty$ -Laplacian as

$$\begin{aligned}
 \Delta_{w,2}^N(f)(u) &= \text{NLMean}(f)(u) - f(u), \\
 \Delta_{w,1}^N(f)(u) &= \text{NLMedian}(f)(u) - f(u), \\
 \Delta_{w,\infty}^N(f)(u) &= \text{NLMidrange}(f)(u) - f(u).
 \end{aligned}
 \tag{3.3}$$

We can remark that these operators are related with partial operators on graphs, indeed

$$\begin{aligned}
 \Delta_{w,2}^N(f)(u) &= \frac{1}{\mu(u)} \Delta_{w,2}(f)(u), \\
 \Delta_{w,\infty}^N(f)(u) &= \Delta_{w,\infty}(f)(u), \\
 \Delta_{w,1}^N(f)(u) &= \text{median}(\nabla_w(f)(u)).
 \end{aligned}
 \tag{3.4}$$

### 3.2 Game $p$ -Laplacian on graph

With a simple discretization of (1.7) and using the discrete version of game  $p$ -Laplacian with  $p = 1$ ,  $p = 2$  and  $p = \infty$ , we propose a non-local version of the game  $p$ -Laplacian on graph. This is given by the following equations:

$$\Delta_{w,p}^N(f)(u) = \begin{cases} \frac{2}{p} \Delta_{w,2}^N(f)(u) + \frac{p-2}{p} \Delta_{w,\infty}^N(f)(u) & \text{for } 2 \leq p \leq \infty, \\ \frac{2(p-1)}{p} \Delta_{w,2}^N(f)(u) + \frac{2-p}{p} \Delta_{w,1}^N(f)(u) & \text{for } 1 \leq p \leq 2. \end{cases} \tag{3.5}$$

Using the definitions (3.3) and (3.5), the game  $p$ -Laplacian formulation on weighted graphs can be rewritten as

$$\Delta_{w,p}^N(f)(u) = \text{NLA}(f)(u) - f(u), \tag{3.6}$$

where  $\text{NLA}(f)(u)$  is a non-local average operator as

$$\text{NLA}(f)(u) = \begin{cases} \frac{2}{p} \text{NLMean}(f)(u) + \frac{p-2}{p} \text{NLMidrange}(f)(u), & \text{for } 2 \leq p \leq \infty, \\ \frac{2(p-1)}{p} \text{NLMean}(f)(u) + \frac{2-p}{p} \text{NLMedian}(f)(u), & \text{for } 1 \leq p \leq 2. \end{cases} \tag{3.7}$$

### 3.3 Connection with local and non-local PDEs

In this paragraph, we show that the normalized  $p$ -Laplacian on graphs we propose is a discrete version of certain continuous local and non-local PDEs.

Let  $\Omega$  be a bounded, smooth and convex domain in  $\mathbb{R}^N$ ,  $f : \Omega \rightarrow \mathbb{R}$  a given function, the normalized anisotropic  $p$ -Laplacian can be expressed as

$$\Delta_p^N f = \frac{1}{p} |\nabla f|^{2-p} \sum_{i=1}^N \frac{\partial}{\partial x_i} \left[ \left| \frac{\partial f}{\partial x_i} \right|^{p-2} \frac{\partial f}{\partial x_i} \right]. \tag{3.8}$$

If we discretize this expression with finite centred difference,

$$D_i(f)(x) \approx \frac{\partial}{\partial x_i} f(x) = \frac{f(x + h_i/2) - f(x - h_i/2)}{h_i}, \tag{3.9}$$

with  $p = 2$ , and by using a constant space discretization step  $h_i = h$  we get the following operator:

$$\Delta_2^N f(x) = \frac{1}{2h^2} \left[ \sum_{i=1}^N f(x_i + h_i) + f(x_i - h_i) \right] - \frac{N}{h^2} f(x). \tag{3.10}$$

Given a weighted graph  $G(V, E, w)$  that represents an  $m$ -dimensional grid. Let  $u$  be a vertex associated to an  $m$ -dimensional vector of spatial coordinates:  $u = (i_1 h_1, \dots, i_m h_m)^T$ , where  $i_j \in \mathbb{N}$  and  $h_j$  is the grid spacing size with  $j = 1, \dots, m$ . The neighbourhood of  $u$  can be defined as  $N(u) = \{v : v = u \pm h_j e_j\}_{j=1, \dots, m}$  where  $e_j = (q_k)_{k=1, \dots, m}^T$  is the vector such

that  $q_k = 1$  if  $j = k$  and  $q_k = 0$  otherwise. In the sequel, we call this graph construction a Euclidean graph.

We consider the case where  $p = 2$  and replacing vertices in the graph by their spatial coordinates, and setting  $w_1(u, v_i) = 1$ , we get the following formulation:

$$\begin{aligned} \Delta_{w_1,2}^N(f)(u) &= \frac{\sum_{v \sim u} w_1(u, v)f(v)}{\sum_{v \sim u} w_1(u, v)} - f(u) \\ &= \frac{1}{2N} \left[ \sum_{i=1}^N f(v_i^+) + f(v_i^-) \right] - f(u), \end{aligned} \tag{3.11}$$

with  $v_i^\pm = u \pm h_i e_i$ .

Considering a constant spatial step  $h_i = h \forall i \in \{1, \dots, N\}$ , one can see that we recover the discrete formulation of the normalized  $p$ -Laplacian, by the following relation:

$$\Delta_{w_1,2}^N(f) = \frac{h^2}{N} \Delta_2^N f. \tag{3.12}$$

Now, if we consider  $G = (V, E, w)$  a complete graph, for  $p = 2$  equation (3.6) can be rewritten as

$$\Delta_{w,2}^N(f)(u) = \frac{\sum_{v \in V} w(u, v)(f(v) - f(u))}{\sum_{v \in V} w(u, v)}. \tag{3.13}$$

Now, let us recall the non-local continuous  $p$ -Laplacian formulation (using  $p = 2$ ) of a function  $u : \Omega \rightarrow \mathbb{R}$ , where  $\Omega$  is a bounded domain of  $\mathbb{R}^N$ , introduced by [3]:

$$\mathcal{L}_2 u(x, t) = \int_{\Omega} J(x - y)(u(y, t) - u(x, t)) dy. \tag{3.14}$$

One can see that our formulation (equation (3.13)) is the discrete analogue of equation (3.14), normalized by the degree:

$$\Delta_{w,2}^N(f)(u) = \frac{\mathcal{L}_2 f(u)}{\sum_{v \in V} w(u, v)}. \tag{3.15}$$

In the case where  $p = 1$ , the analogy can be done using the recent results of [38]. Let  $\Omega \subset \mathbb{R}^N$ ,  $x \in \Omega$ ,  $u : \Omega \rightarrow \mathbb{R}$  a smooth function,  $\epsilon = \sqrt{2h}$ ,  $B(x, \epsilon)$  a ball of radius  $\epsilon$  and centre  $x$ , and a non-vanishing gradient of  $u$  at point  $x$ . The author of [38] shows the following relation:

$$u(x) - \operatorname{median}_{y \in \partial B(x, \epsilon)}(u(y)) = -\frac{h}{N - 1} \Delta_1^N u(x) + o(h) \tag{3.16}$$

which can be rewritten formally as

$$\Delta_1^N u(x) = \frac{N - 1}{h} (\operatorname{median}_{y \in \partial B(x, \epsilon)}(u(y)) - u(x)) + o(h). \tag{3.17}$$

By using our formulation of  $\Delta_{w,1}^N$ , we can do the analogy with this formula. Now, let us set  $w_2(u, v)$  as

$$w_2(u, v) = \begin{cases} \frac{(N-1)^2}{h^2}, & \text{if } v \in \partial B(u, \epsilon) \\ 0, & \text{otherwise.} \end{cases} \tag{3.18}$$

If we replace it into  $\Delta_{w,1}^N$ , we get formally

$$\Delta_{w_2,1}^N(f)(u) = \frac{N-1}{h} (\text{median}_{v \sim u}(f(v)) - f(u)). \tag{3.19}$$

As one can see, this formulation recovers formally the equation (3.17) proposed in [38].

Similarly, in the case where  $p = \infty$ , we can do the analogy with the Oberman discretization of infinity-Laplace equation [35]: Let  $f(x)$  be a smooth function with non-vanishing gradient at  $x$ . The discretization performed in [35] is

$$\Delta_\infty f(x) = \min_{|y-x|=\epsilon} \frac{(f(y) - f(x))}{\epsilon^2} + \max_{|y-x|=\epsilon} \frac{(f(y) - f(x))}{\epsilon^2} + O(\epsilon^2). \tag{3.20}$$

To recover this formulation, we use the Euclidean graph construction, but setting  $w(u, v)$  as

$$w_3(u, v) = \begin{cases} \frac{4}{\epsilon^4}, & \text{if } v \in \partial B_\epsilon(u) \\ 0, & \text{otherwise.} \end{cases} \tag{3.21}$$

If we replace it in  $\Delta_{w,\infty}^N$ , we get

$$\begin{aligned} \Delta_{w_3,\infty}^N(f)(u) &= \frac{1}{2} [\max_{v \sim u}(\sqrt{w_3(u, v)}(f(v) - f(u))^+) - \max_{v \sim u}(\sqrt{w_3(u, v)}(f(v) - f(u))^-)] \\ &= \frac{1}{\epsilon^2} [\max_{v \sim u}(f(v) - f(u))^+ - \max_{v \sim u}(f(v) - f(u))^-]. \end{aligned} \tag{3.22}$$

If we set the neighbourhood of  $u$  as  $N(u) \cup \{u\}$ , we get

$$\begin{aligned} \Delta_{w_3,\infty}^N(f)(u) &= \frac{1}{\epsilon^2} [\max_{v \sim u}(f(v) - f(u)) - \max_{v \sim u}(f(u) - f(v))], \\ &= \frac{1}{\epsilon^2} [\max_{v \sim u}(f(v) - f(u)) + \min_{v \sim u}(f(v) - f(u))], \\ &= \max_{v \sim u} \frac{f(v) - f(u)}{\epsilon^2} + \min_{v \sim u} \frac{f(v) - f(u)}{\epsilon^2}. \end{aligned} \tag{3.23}$$

Still using  $p = \infty$ , but by considering a complete graph, our formulation corresponds to the recently proposed Hölder infinity-Laplacian equation proposed in [13].

$$\begin{aligned} \Delta_{w_4,\infty}(f)(x) &= \frac{1}{2} \left[ \max_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^s} \right) \right. \\ &\quad \left. + \min_{y \in \Omega, y \neq x} \left( \frac{f(y) - f(x)}{|y - x|^s} \right) \right], \end{aligned} \tag{3.24}$$

with

$$w_4(x, y) = \begin{cases} \frac{1}{|y-x|^{2s}}, & \text{if } y \in \Omega \text{ and } y \neq x \\ 0, & \text{otherwise.} \end{cases} \quad (3.25)$$

This operator is formally derived from the minimization of an energy of the form

$$\int_{\Omega} \int_{\Omega} \frac{|f(y) - f(x)|^p}{|x - y|^{p \times s}} dx dy \quad (3.26)$$

as  $p$  goes to infinity. In [13], it has been proved that the following equation

$$\begin{cases} \Delta_{w_4, \infty}^N(f)(x) = 0, & x \in \Omega \\ f(x) = g(x), & x \in \partial\Omega \end{cases} \quad (3.27)$$

(and with some conditions on  $g$ ), has a unique solution.

### 3.4 Study of the existence and uniqueness of the game $p$ -Laplacian equation

Now, we will study the Dirichlet problem associated to this non-local normalized  $p$ -Laplacian equation (3.5).

**Theorem 1** *Given a connected graph  $G = (V, E, w)$ , a set  $A \subset V$  and a function  $g : \partial A \rightarrow \mathbb{R}$  where  $\partial A$  is the boundary of  $A$ . Then, there exists a unique function  $f \in \mathcal{H}(V)$  such that  $f$  verifies the following equation:*

$$\begin{cases} \Delta_{w, p}^N(f)(u) = 0 & u \in A \\ f(u) = g(u) & u \in \partial A. \end{cases} \quad (3.28)$$

**Proof of Theorem 1** Using the NLA operator defined in equation (3.7), we can rewrite (3.28) as

$$\begin{cases} \text{NLA}(f)(u) - f(u) = 0 & u \in A \\ f(u) = g(u) & u \in \partial A. \end{cases} \quad (3.29)$$

First, let us prove the uniqueness of the solution by using the comparison principle.

Given two functions  $f$  and  $h$ , we will prove that if  $f = \text{NLA}(f)$  and  $h = \text{NLA}(h)$  with  $f \leq h$  on  $\partial A$ , then  $f \leq h$  on the whole domain  $V$ . By the argument of contradiction, we assume that there exists  $M$  such that

$$M = \sup_V (f - h) > 0.$$

Let  $B = \{u \in A : f(u) - h(u) = M\}$ . By construction, we have  $B \neq \emptyset$  and  $B \cap \partial A = \emptyset$ . We claim that there exists  $u_\lambda \in B$  and  $v_\lambda \in N(u_\lambda)$ , such that  $v_\lambda \notin B$ . Otherwise, if for each  $u \in A$  and for each  $v \in N(u)$  we have  $v \notin B$ , then it implies that  $B \cap \partial A \neq \emptyset$ , since the

graph is connected: there is a contradiction. Then, from the definition of  $M$ , we have

$$\begin{aligned} f(u_\lambda) - h(u_\lambda) &\geq f(u) - h(u) \quad \forall u \in N(u_\lambda), \\ h(u) - h(u_\lambda) &\geq f(u) - f(u_\lambda) \quad \forall u \in N(u_\lambda). \end{aligned}$$

In particular, we can write

$$h(v_\lambda) - h(u_\lambda) > f(v_\lambda) - f(u_\lambda).$$

From these inequalities, and using definitions of NLMin and NLMax, we have

$$\begin{aligned} &\max_{u \sim u_\lambda} \left( \sqrt{w(u_\lambda, u)} \max(h(u) - h(u_\lambda), 0) \right) \\ &\geq \\ &\max_{u \sim u_\lambda} \left( \sqrt{w(u_\lambda, u)} \max(f(u) - f(u_\lambda), 0) \right) \\ \text{NLMax}(h)(u_\lambda) - h(u_\lambda) &\geq \text{NLMax}(f)(u_\lambda) - f(u_\lambda), \end{aligned} \tag{3.30}$$

similarly,

$$\begin{aligned} &\max_{u \sim u_\lambda} \left( \sqrt{w(u_\lambda, u)} \min(h(u) - h(u_\lambda), 0) \right) \\ &\geq \\ &\max_{u \sim u_\lambda} \left( \sqrt{w(u_\lambda, u)} \min(f(u) - f(u_\lambda), 0) \right) \\ h(u_\lambda) - \text{NLMin}(h)(u_\lambda) &\geq f(u_\lambda) - \text{NLMin}(f)(u_\lambda). \end{aligned} \tag{3.31}$$

By the same reasoning, using the NLMean operator, we have

$$\begin{aligned} \frac{\sum_{u \sim u_\lambda} w(u_\lambda, u)h(u)}{\sum_{u \sim u_\lambda} w(u_\lambda, u)} - h(u_\lambda) &\geq \frac{\sum_{u \sim u_\lambda} w(u_\lambda, u)f(u)}{\sum_{u \sim u_\lambda} w(u_\lambda, u)} - f(u_\lambda) \\ \text{NLMean}(h)(u_\lambda) - h(u_\lambda) &> \text{NLMean}(f)(u_\lambda) - f(u_\lambda). \end{aligned} \tag{3.32}$$

This previous inequality is strict because we know there is  $v \in N(u_\lambda)$  such that  $h(v) - h(u_\lambda) > f(v) - f(u_\lambda)$ .

Finally, using the NLMedian operator, we have

$$\begin{aligned} \sqrt{w(u_\lambda, u)}(h(u) - h(u_\lambda)) &\geq \sqrt{w(u_\lambda, u)}(f(u) - f(u_\lambda)) \\ \text{median}(\sqrt{w(u_\lambda, u)}(h(u) - h(u_\lambda))) &\geq \text{median}(\sqrt{w(u_\lambda, u)}(f(u) - f(u_\lambda))) \\ \text{NLMedian}(h)(u_\lambda) - h(u_\lambda) &\geq \text{NLMedian}(f)(u_\lambda) - f(u_\lambda). \end{aligned} \tag{3.33}$$

From relations (3.30)–(3.33), we can write the following inequality for  $2 \leq p \leq \infty$ :

$$\begin{aligned} & \frac{p-2}{p} \text{NLMidrange}(h)(u_\lambda) + \frac{2}{p} \text{NLMean}(h)(u_\lambda) - h(u_\lambda) \\ & > \\ & \frac{p-2}{p} \text{NLMidrange}(f)(u_\lambda) + \frac{2}{p} \text{NLMean}(f)(u_\lambda) - f(u_\lambda) \end{aligned}$$

and for  $1 \leq p \leq 2$

$$\begin{aligned} & \frac{2-p}{p} \text{NLMedian}(h)(u_\lambda) + \frac{2(p-1)}{p} \text{NLMean}(h)(u_\lambda) - h(u_\lambda) \\ & > \\ & \frac{2-p}{p} \text{NLMedian}(f)(u_\lambda) + \frac{2(p-1)}{p} \text{NLMean}(f)(u_\lambda) - f(u_\lambda). \end{aligned}$$

So that we can write for  $1 \leq p \leq \infty$

$$\begin{aligned} \text{NLA}(h)(u_\lambda) - h(u_\lambda) &> \text{NLA}(f)(u_\lambda) - f(u_\lambda) \\ h(u_\lambda) - h(u_\lambda) &> f(u_\lambda) - f(u_\lambda) \\ 0 &> 0. \end{aligned}$$

This shows a contradiction and concludes the proof.

Now, let us prove existence. First, we recall the Brouwer fixed point theorem: A continuous function from a convex, compact subset of a Euclidean space to itself has a fixed point.

Then, we identify  $\mathcal{H}(V)$  as  $\mathbb{R}^N$  and consider the set  $K = \{f \in \mathcal{H}(V) \mid f(u) = g(u) \forall u \in \partial A, \text{ and } m \leq f(u) \leq M \forall u \in A\}$ , where  $m = \min_{\partial A}(g(u))$  and  $M = \max_{\partial A}(g(u))$ . By definition,  $K$  is a convex and compact subset of  $\mathbb{R}^N$ .

It is easy to show that the map  $f \rightarrow \text{NLA}(f)$  is continuous and take from  $K$  to  $K$ . So, by the Brouwer fixed point theorem, the map  $\text{NLA}$  has a fixed point that is solution of  $\text{NLA}(f) = f$ . This completes the proof. □

### 3.5 Connection with Tug-of-War games

Many local PDEs ( $p$ -Laplacian (for  $p \geq 2$ ), infinity-Laplacian) are related to a stochastic game called Tug-of-War game. We show that our newly introduced partial difference operator also recovers value functions of this game. In particular, Tug-of-War game and Tug-of-War game with noise [33, 36].

Let us briefly review the notion of Tug-of-War game introduced by [36]. Let  $\Omega \subset \mathbb{R}^N$  be a Euclidean space, and  $g : \Omega \rightarrow \mathbb{R}$  a function. Fix a number  $\varepsilon > 0$ . The dynamics of the game are as follows. A token is placed at an initial position  $x_0 \in \Omega$ . At the  $k$ th stage of the game, Player I and Player II select points  $x_k^I$  and  $x_k^{II}$ , respectively, each belonging to a specified set  $B_\varepsilon(x_{k-1}) \subseteq \Omega$  (where  $B_\varepsilon(x_{k-1})$  is the  $\varepsilon$ -ball centred in  $x_{k-1}$ ). The game token is then moved to  $x_k$ , where  $x_k$  is chosen so that  $x_k = x_k^I$  with probability  $P = \frac{1}{2}$ .

In other words, a fair coin is tossed to determine where the token is placed. After the  $k$ th stage of the game, if  $x_k \in \Omega$ , then the game continues to stage  $k + 1$ . Otherwise, if  $x_k \in \partial\Omega$ , the game ends and Player II pays Player I the amount  $g(x_k)$ . Player I attempts to maximize the payoff while Player II attempts to minimize it. According to the dynamic programming principle (see [33, 36]), the value functions for Player I and Player II for standard  $\varepsilon$ -turn Tug-of-War satisfy the relation

$$f(x) = \frac{1}{2} \left[ \max_{y \in B_\varepsilon(x)} f(y) + \min_{y \in B_\varepsilon(x)} f(y) \right] \text{ on } \Omega, \tag{3.34}$$

with  $f(x) = g(x)$  on  $\partial\Omega$ .

Given a Euclidean graph  $G(V, E, w)$ , with  $V = \Omega \subset \mathbb{R}^N$ ,  $E = \{(x, y) \in V \times V \mid w_5(x, y) > 0\}$  and

$$w_5(x, y) = \begin{cases} 1, & \text{if } |y - x| \leq \varepsilon \\ 0, & \text{otherwise,} \end{cases} \tag{3.35}$$

and using the following relations easily obtained from definitions of the  $\mathcal{L}_\infty$  norms of morphological gradients (2.7)

$$\begin{aligned} \max_{y \in B_\varepsilon(x)} f(y) &= \|\nabla_{w_5}^+ f(x)\|_\infty + f(x), \\ \min_{y \in B_\varepsilon(x)} f(y) &= f(x) - \|\nabla_{w_5}^- f(x)\|_\infty. \end{aligned} \tag{3.36}$$

One can see that by replacing max and min in equation (3.34) by their formulation of (3.36), we get

$$\begin{aligned} f(x) &= \frac{1}{2} [\|\nabla_{w_5}^+ f(x)\|_\infty(x) - \|\nabla_{w_5}^- f(x)\|_\infty] + f(x) \\ \frac{1}{2} [\|\nabla_{w_5}^+ f(x)\|_\infty(x) - \|\nabla_{w_5}^- f(x)\|_\infty] &= 0, \end{aligned} \tag{3.37}$$

which coincides with our formulation (3.5) of the infinity-Laplacian on graphs:

$$\Delta_{w_5, \infty}^N(f)(x) = 0. \tag{3.38}$$

For a general Euclidean weighted graph, and with  $p = \infty$ , our formulation is connected to the following non-local Tug-of-War game. This is the same game as previously described, except that the  $\varepsilon$ -ball is replaced by a neighbourhood  $N(x_{k-1}) \subset \Omega$  defined as

$$N(x_{k-1}) = \{x \in \Omega \mid w(x, x_{k-1}) > 0\}. \tag{3.39}$$

In this version of the game, the game token is then moved to  $x_k$ , where  $x_k$  is chosen randomly so that  $x_k = x_k^I$  with a probability

$$P = \frac{\sqrt{w(x_{k-1}, x_k^I)}}{\sqrt{w(x_{k-1}, x_k^I)} + \sqrt{w(x_{k-1}, x_k^{II})}} \tag{3.40}$$

and that  $x_k = x_k^{II}$  with a probability  $1 - P$ . According to the dynamic programming principle, the value functions for Player I and Player II for this game satisfies the relation

$$\begin{aligned} & \max_{y \in N(x)} \sqrt{w(x,y)}(f(y) - f(x)) \\ & + \min_{y \in N(x)} \sqrt{w(x,y)}(f(y) - f(x)) = 0, \end{aligned} \tag{3.41}$$

which is simply

$$\Delta_{w,\infty}^N(f)(x) = 0. \tag{3.42}$$

**Tug-of-War game with noise:** If the game is modified as follows: at point  $x$  in  $\Omega$ , player I and player II play  $\epsilon$ -step Tug-of-War game with probability  $\alpha$ , and with probability  $\beta$  such that  $\alpha + \beta = 1$ , a random point in ball of radius  $\epsilon$  centred at  $x$  is chosen. The value functions of the game satisfy the Dynamic Programming Principle

$$f(x) = \frac{\alpha}{2} \left[ \max_{y \in B_\epsilon(x)} f(y) + \min_{y \in B_\epsilon(x)} f(y) \right] + \frac{\beta}{|B_\epsilon(x)|} \int_{B_\epsilon(x)} f(y) dy, \tag{3.43}$$

with  $f(x) = g(x)$  for  $x$  in  $\Omega$  and  $\alpha$  and  $\beta \in \mathbb{R}^+$  such that  $\alpha + \beta = 1$ . A detailed proof for existence and uniqueness of these types of functions was shown in [33].

By using the same graph construction as for the local Tug-of-War game and the same weighting function  $w_5$ , we can rewrite equation (3.43) in the context of PdEs on graphs: we have already shown that the first right-hand term of (3.43) ( $1/2 [\max_{y \in B_\epsilon(x)} f(y) + \min_{y \in B_\epsilon(x)} f(y)]$ ) can be rewritten as equation (3.37). For the second term, we can rewrite it as

$$\frac{1}{|N(u)|} \int_{v \in N(u)} f(v) dv = \Delta_{w_5,2}^N(f)(u) + f(u), \tag{3.44}$$

then equation (3.43) can be rewritten as

$$f(x) = \alpha(\Delta_{w_5,\infty}^N(f)(u) + f(u)) + \beta(\Delta_{w_5,2}^N(f)(u) + f(u)), \tag{3.45}$$

which is equivalent to

$$\alpha \Delta_{w_5,\infty}^N(f)(u) + \beta \Delta_{w_5,2}^N(f)(u) = 0. \tag{3.46}$$

One can see here that we recover our formulation on graphs of the normalized  $p$ -Laplacian for  $2 \leq p \leq \infty$ :

$$\Delta_{w_5,p}^N(f)(u) = 0. \tag{3.47}$$

### 4 Experiments

In this section, we illustrate the behaviour of the normalized  $p$ -Laplacian presented in this paper, through some inverse problems as functions interpolation on graphs. The experiments provided are not here to solve a particular application but to illustrate the potentialities of our proposal.

Many tasks in image processing, computer vision and machine learning can be formulated as interpolation problems. Image and video colourization, inpainting and semi-supervised segmentation or clustering are examples of these interpolation problems. Interpolating data consists in constructing new values for missing data in coherence with a set of known data. In this paper, we propose to use the non-local normalized  $p$ -Laplacian as a unified framework for both the tasks of semi-supervised segmentation or clustering and image inpainting. For this, we solve the following Dirichlet problem:

$$\begin{cases} \Delta_{w,p}^N(f)(u) = 0 & u \in A \\ f(u) = g(u) & u \in V - A = \partial A, \end{cases} \tag{4.1}$$

where  $A \subset V$  is the subset of vertices associated to the missing information. The initial value function  $g$  is application dependent and will be defined for each application in the sequel.

To solve (4.1), we make use of the following associated evolution equation problem:

$$\begin{cases} \frac{\partial}{\partial t} f(u, t) = \Delta_{w,p}^N(f)(u, t), & u \in A, \\ f(u, t) = g(u), & u \in \partial A, \\ f(u, t = 0) = f_0(u), & u \in A, \end{cases} \tag{4.2}$$

for which  $f_0$  is an initial function that is also application dependent.

To solve (4.2) iteratively, we use an explicit forward Euler time discretization:

$$\frac{\partial f}{\partial t}(u, t) = \frac{f^{n+1}(u) - f^n(u)}{\Delta t}, \tag{4.3}$$

with  $f^n(u) = f(u, n\Delta t)$ .

Hence, we can try to solve (4.2) by the following iteration scheme:

$$\begin{cases} f^{n+1}(u) = f^n(u) + \Delta t \Delta_{w,p}^N(f^n)(u), & u \in A, \\ f^{n+1}(u) = g(u), & u \in \partial A, \\ f^0(u) = f_0(u), & u \in A. \end{cases} \tag{4.4}$$

Using  $\Delta_{w,p}^N = NLA(f) - f$  and setting  $\Delta t = 1$ , we get the following non-local average filter, which is a convex combination of a dilation, an erosion and a non-local mean processes:

$$\begin{cases} f^{n+1}(u) = \begin{cases} \frac{2}{p} \text{NLMean}(f^n)(u) + \frac{p-2}{p} \text{NLMidrange}(f^n)(u), & \text{for } 2 \leq p \leq \infty \\ \frac{2(p-1)}{p} \text{NLMean}(f^n)(u) + \frac{2-p}{p} \text{NLMedian}(f^n)(u) & \text{for } 1 \leq p \leq 2 \end{cases} \\ f^{n+1}(u) = g(u), & u \in \partial A, \\ f^0(u) = f_0(u), & u \in A. \end{cases} \tag{4.5}$$

**Proposition** *If the iterative filtering (4.5) converges to a function  $f^*$ , then  $f^*$  satisfy  $\Delta_{w,p}^N(f^*)(u) = 0$  for  $u \in V$ .*

**Proof** Let  $f^*$  be the limit process. Since functions NLMean, NLMidrange and NLMedian are continuous, then we have,

$$f^*(u) = \begin{cases} \frac{2}{p}\text{NLMean}(f^*)(u) + \frac{p-2}{p}\text{NLMidrange}(f^*)(u), & \text{for } 2 \leq p \leq \infty \\ \frac{2(p-1)}{p}\text{NLMean}(f^*)(u) + \frac{2-p}{p}\text{NLMedian}(f^*)(u) & \text{for } 1 \leq p \leq 2. \end{cases} \tag{4.6}$$

It easy to show that (4.6) is equivalent to

$$\Delta_{w,p}^N(f^*)(u) = 0. \tag{4.7}$$

□

### 4.1 Weighted graph construction

There exists several popular methods to transform discrete data  $\{x_1, \dots, x_n\}$  into a weighted graph structure. Considering a set of vertices  $V$  such that data are embedded by functions of  $\mathcal{H}(V)$ , the construction of such graph consists in modelling the neighbourhood relationships between the data through the definition of a set of edges  $E$  and using a pairwise distance measure  $\mu : V \times V \rightarrow \mathbb{R}^+$ . In the particular case of images, the ones based on geometric neighbourhoods are particularly well-adapted to represent the geometry of the space, as well as the geometry of the function defined on that space. One can quotes following:

- *Grid graphs* which are most natural structures to describe an image with a graph. Each pixel is connected by an edge to its adjacent pixels. Classical grid graphs are 4-adjacency grid graphs and 8-adjacency grid graphs. Larger adjacency can be used to obtain non-local graphs.
- *Region adjacency graphs* (RAG) which provide very useful and common ways of describing the structure of a picture: vertices represent regions and edges represent region adjacency relationship.
- *k-neighbourhood graphs* ( $k$ -NNG) where each vertex  $v_i$  is connected with its  $k$ -nearest neighbours according to  $\mu$ . Such construction implies to build a directed graph, as the neighbourhood relationship is not symmetric. Nevertheless, an undirected graph can be obtained while adding an edge between two vertices  $u$  and  $v$  if  $u$  is among the  $k$ -nearest neighbour of  $v$  or if  $v$  is among the  $k$ -nearest neighbour of  $u$ .
- *k-Extended RAG* ( $k$ -ERAG) which are RAGs extended by a  $k$ -NNG. Each vertex is connected to adjacent regions vertices and to its  $k$  most similar vertices of  $V$ .

The similarity between two vertices is computed according to a measure of similarity  $s : E \rightarrow \mathbb{R}^+$ , which satisfies

$$w(u, v) = \begin{cases} s(u, v) & \text{if } (u, v) \in E \\ 0 & \text{otherwise.} \end{cases}$$

Usual similarity functions are as follows:

$$s_0(u, v) = 1,$$

$$s_1(u, v) = \exp\left(-\mu(f^0(u), f^0(v))/\sigma^2\right) \text{ with } \sigma > 0,$$

where  $\sigma$  depends on the variation of the function  $\mu$  and control the similarity scale.

Several choices can be considered for the expression of the feature vectors, depending on the nature of the features to be used for the graph processing. In the context of image processing, one can quote the simplest gray scale or colour feature vector  $F_u$ , or the patch feature vector  $F_u^\tau = \bigcup_{v \in \mathcal{W}^\tau(u)} F_v$  (i.e., the set of values  $F_v$  where  $v$  is in a square window  $\mathcal{W}^\tau(u)$  of size  $(2\tau + 1) \times (2\tau + 1)$  centred at a vertex pixel  $u$ ), in order to incorporate non-local features.

### 4.2 Semi-supervised segmentation and classification

In the case of image semi-supervised segmentation, graph-based approaches have become very popular in recent years. Many graph-based algorithms for image segmentation have been proposed. Interested reader should refer to recent works [15, 16] and references therein.

In this paper, we propose to consider this problem as an interpolation problem, where the function to interpolate is the label function. Using equation (4.1), and considering two classes  $A$  and  $B$ , the initial value label function  $g$  is defined as follows:

$$\begin{cases} g(u) = -1 \text{ if } u \in A \\ g(u) = 1 \text{ if } u \in B \\ g(u) = 0 \text{ otherwise .} \end{cases} \tag{4.8}$$

At convergence, the class membership can be easily computed by a simple threshold on the sign of  $f$ .

**Remark 1** *In the case of more than two classes, multi-classes segmentation can be performed by several segmentation of one class versus the others.*

#### 4.2.1 Natural image segmentation using non-local graph

In this paragraph, we show the benefits of non-local schemes compared to local ones for semi-supervised image segmentation, especially for images that contain fine and repetitive structures.

Figure 1 presents several segmentation results of a natural tiger image, with two different (local and non-local) graph constructions and different values for the parameter  $p$ . The local graph is built as a usual 4-adjacency grid graph where each pixel is characterized by its colour feature vector. The non-local one is built using a  $k$ nns graph in a  $51 \times 51$  neighbourhood window and each pixel is characterized by a  $5 \times 5$  patch of colour feature vectors.

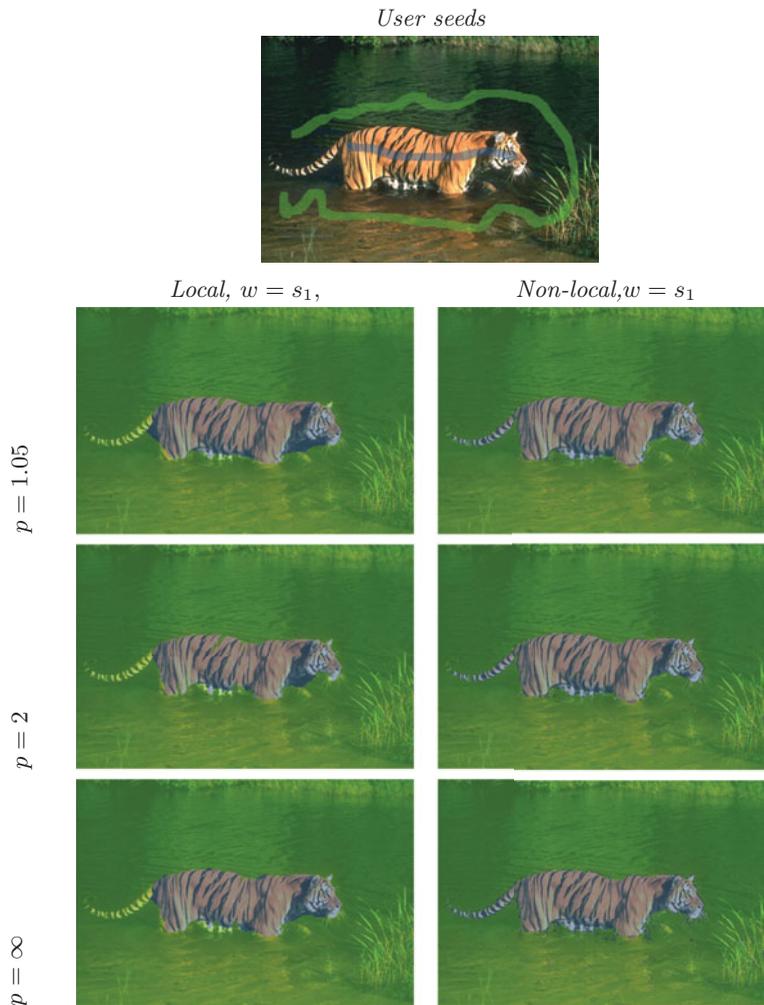


FIGURE 1. Semi-supervised natural image segmentation with  $\Delta_{w,p}^N$ . First column presents results with a local 4-adjacency grid graph where each pixel is characterized by its colour feature vector. Second column presents non-local results obtained with a larger neighbourhood (each pixel  $u$  is linked with any pixel in a  $51 \times 51$  window centred on  $u$ ) and pixels are characterized by patches of size  $5 \times 5$ . In both cases, results are provided for  $p = 1.05$ ,  $p = 2$  and  $p = \infty$ . See text for more details.

#### 4.2.2 Real data clustering

In this paragraph, we present experiments on data semi-supervised classification.

We have considered label diffusion using the game  $p$ -Laplacian on three standard state-of-the-art databases: MNIST [32], OPTDIGITS [2] and PENDIGITS [1]. These databases are composed of hand-written digits. For these databases, we have merged both the training and the test set (as performed in [10]), resulting in datasets of 70,000, 5,620 and 10,992 instances, for MNIST, OPTDIGITS and PENDIGITS, respectively. For the OPTDIGITS and PENDIGITS databases, we have used a preprocessed version of the data, giving constant size feature vectors, and giving invariance to small distortions (see [2]

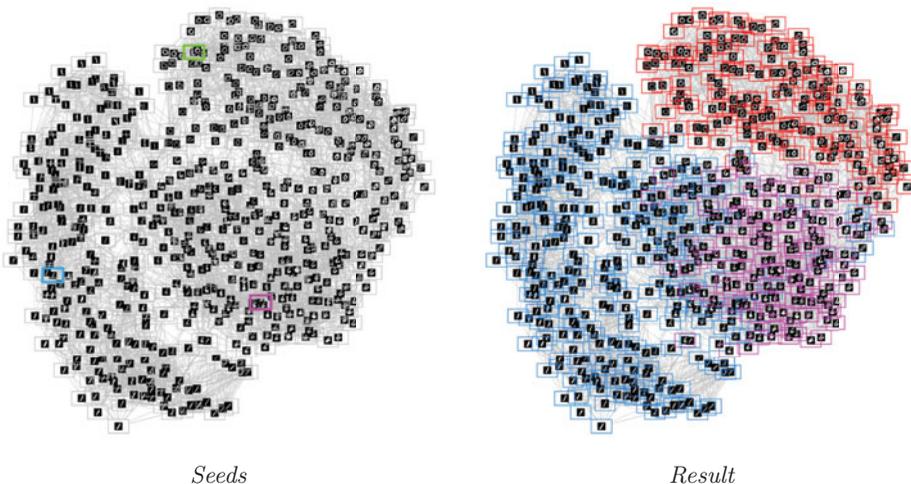


FIGURE 2. Semi-supervised data clustering illustration. Left figure (Seeds) shows a graph built on a database of hand-written numbers (0's, 1's and 6's) with some nodes initially labeled (in green, blue and purple) in each class. The figure on the right shows the result of the label diffusion on the graph. See text for more details.

and [1] for more details on the preprocessing routines). For MNIST, we use the data from the original database, where the digits are stored as small images. As the authors of [10], we have constructed a  $k$ -nearest neighbour graph on the merged datasets, with  $k = 10$ . For the MNIST dataset, we constructed the graph using the two-sided tangent distance [31], giving invariance to small affine transforms, and for the two others we used the Euclidean distance between each data points. To compute weights between each vertex, we have used the well-known Gaussian Kernel similarity:  $w(u, v) = s_1(u, v)$ . Since the parameter  $\sigma$  (in  $s_1$ ) is a strong bottleneck of graph-based methods, we consider strategies for computing automatically its value. We tried two strategies: using a global scaling parameter  $\sigma$ , and using a  $\sigma_i$  local to each vertex, as in [42], to have a local scaling weight function. In this particular case, the similarity function becomes  $w(u, v) = \exp \frac{-\mu(u, v)^2}{\sigma_u \sigma_v}$ , with  $\sigma_u$  the local scaling parameter at vertex  $u$ . We computed each  $\sigma_u$  as the distance to the  $M$ th closest vertex to  $u$ .

To estimate a global  $\sigma$  parameter, we used the method described in [30]. These authors proposed a robust method to estimate a global and a local  $\sigma$  parameter. In this work, we have used the global estimation, which is  $\hat{\sigma} = 1.4826 \text{median}(|\mathcal{E}_S| - \text{median}|\mathcal{E}_S|)$ , where  $\mathcal{E}_S$  is the set of local residuals in the graph, computed as  $\mathcal{E}_u = (\sum_{v \sim u} f(u) - f(v)) / \sqrt{|v \sim u|^2 + |v \sim u|}$ , for a vertex  $u$ . For further details and justifications, see [30].

As a test protocol, we make 10 runs for each algorithm, and we use a percentage of already labeled vertices, settled randomly each time. A typical labelling result is shown in Figure 2.

Classification results are shown in Figures 3–5. We only present here results that have provided the best average classification rates.

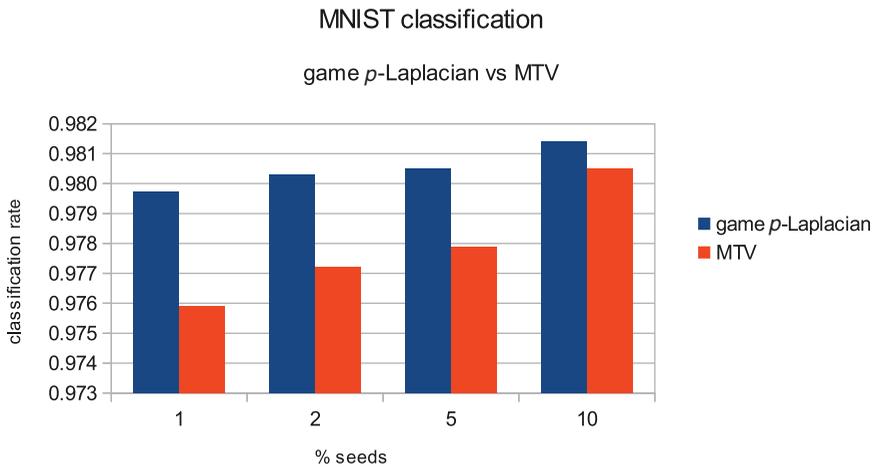


FIGURE 3. Semi-supervised classification results for the MNIST database.  $X$ -axis is the amount of labeled vertices and  $Y$ -axis is the classification rate. Best results were achieved using a local scaling parameter ( $\sigma$ ) and  $p$  close to one. See text for more details.

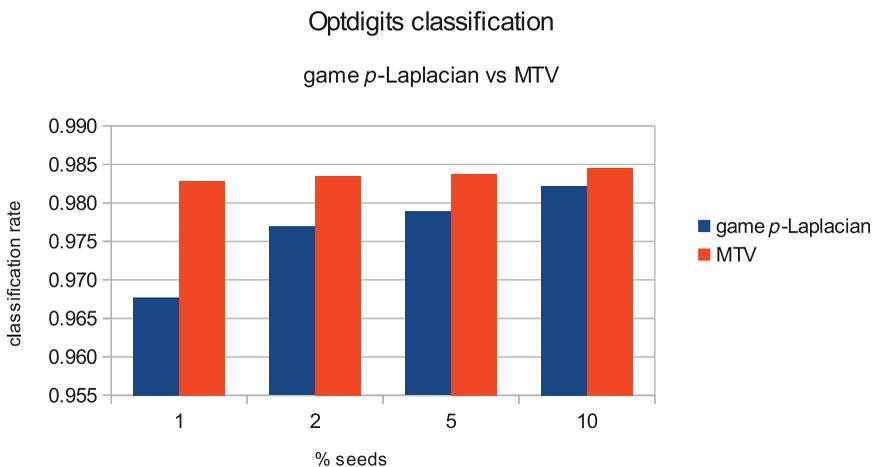


FIGURE 4. Semi-supervised classification results. Like the MNIST database, best results were achieved using a local scaling parameter ( $\sigma$ ) and  $p$  close to one. See text for more details.

To evaluate our method, we compare ourselves with the most recent and efficient method called Multiclass Total Variation clustering [10]. As it can be seen from the results, for MNIST (Figure 3) and PENDIGITS (Figure 5) datasets, our method outperforms the state-of-the-art while for OPTDIGITS (Figure 4) it compares well with.

To show the behaviour of our operator, we plotted classification rates varying the parameter  $p$  in Figure 6–7. As one can see the closer we get from  $p = 1$ , the better the rates.

The number of iteration to reach the convergence vary with the database (for Pendigits : 500 iterations, Optigits : 500 iterations, Mnist : 100 iterations). All processing have been implemented in C++ and have been run on a current workstation with a GNU/Linux operating system.

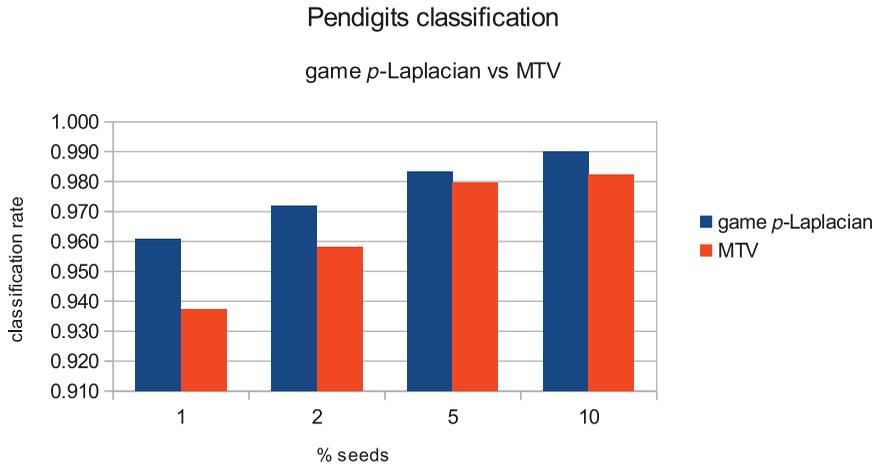


FIGURE 5. Semi supervised classification results for the PENDIGITS database. Like the MNIST and OPTDIGIT databases, best results were achieved using a local scaling parameter ( $\sigma$ ) and  $p$  close to one. See text for more details.

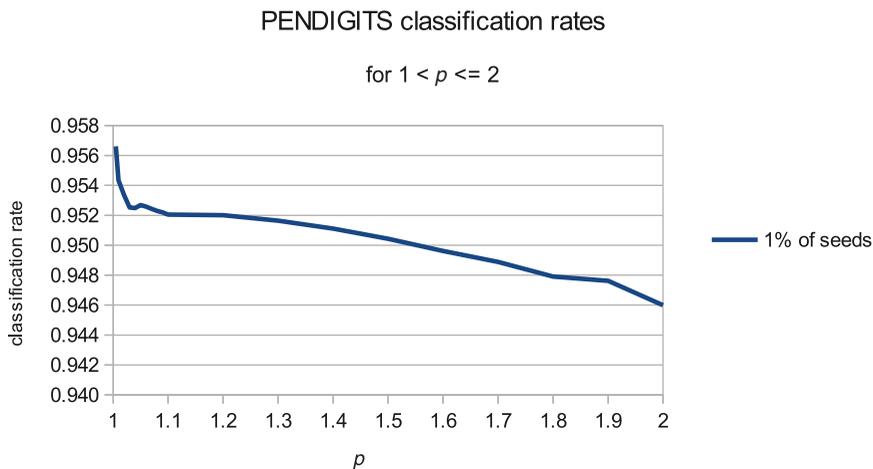


FIGURE 6. Semi-supervised classification rates for  $1 < p \leq 2$  on the PENDIGITS dataset. See text for more details.

### 4.3 Non-local image inpainting

Digital inpainting is a fundamental problem in image processing and has many applications in different fields. It can be simply resumed as reconstructing a damaged or incomplete image by filling the missing information in incomplete regions.

In recent years, many methods have been developed for interpolating the geometry, the texture or both geometry and texture. Among the methods of interpolation that have been proposed, a number of methods are based on PDEs or variational methods, see [5, 39] and references therein.

Since the work of [11] on non-local filtering, many non-local methods for image inpainting have gained considerable attention these recent years. This is in part due to

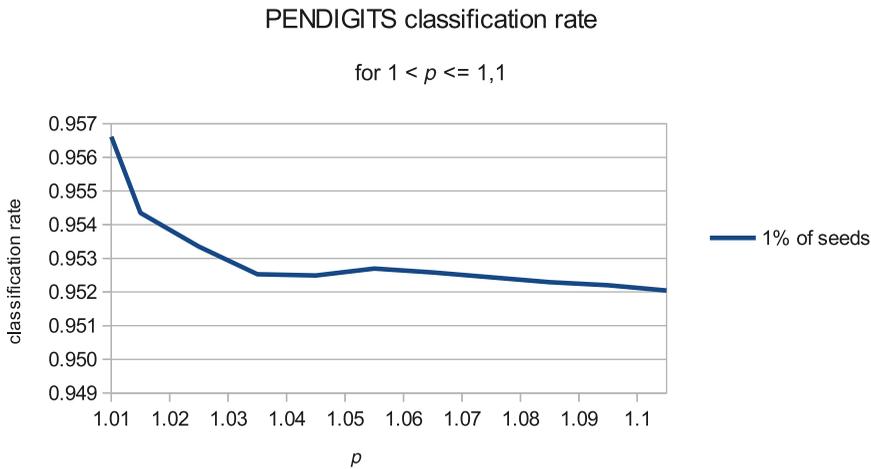


FIGURE 7. Semi-supervised classification rates for  $1 < p \leq 1.1$  on the PENDIGITS dataset. See text for more details.

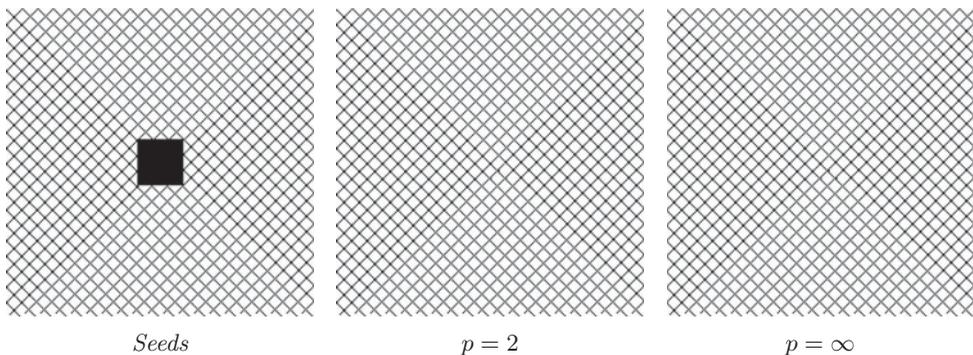


FIGURE 8. Texture image Inpainting. See text for details.

their superior performance in textured images, known weakness of purely local methods. Recent works tend to unify the local and non-local interpolation approaches [26]. A variational framework for non-local image inpainting has been presented in [5]. In [25], a discrete non-local regularization framework for image and manifold processing has been proposed, and has been used to present a unifying approach of local geometric methods and non-local exemplar-based ones for video inpainting.

Considering equation (4.1), inpainting problem can be summarized as follows:  $V_0$  is the set of pixels with missing information.  $g : V \rightarrow \mathcal{H}(V)$  represents the known information.  $f : V \rightarrow \mathcal{H}(V)$  represents the image to reconstruct.

This is illustrated in Figures 8–10. The first one (Figure 8) illustrates the behaviour of the algorithm on a texture image, using a non-local graph construction with a  $31 \times 31$  neighbourhood window and  $15 \times 15$  patches, with data-dependent weight function. Figure 9 illustrate the algorithm, using local (8-adjacency graph) and non-local ( $k$ nn graph in a

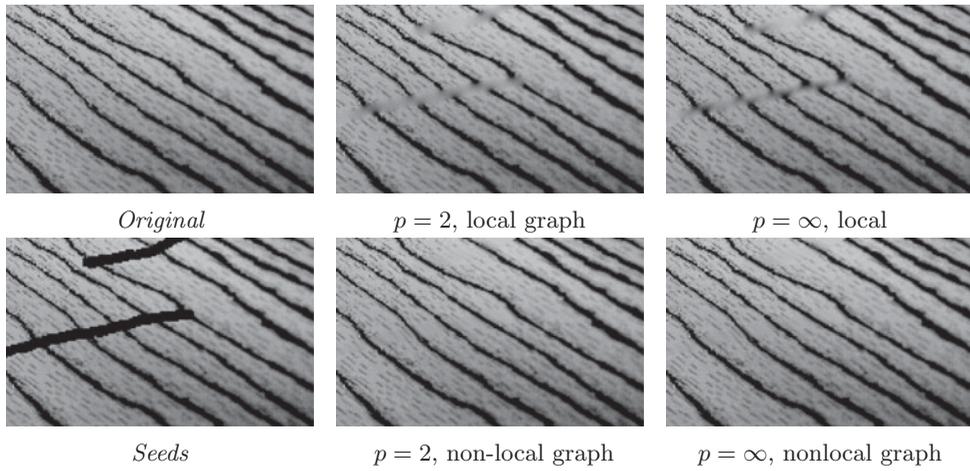


FIGURE 9. Image inpainting using local and non-local graph construction. See text for details.

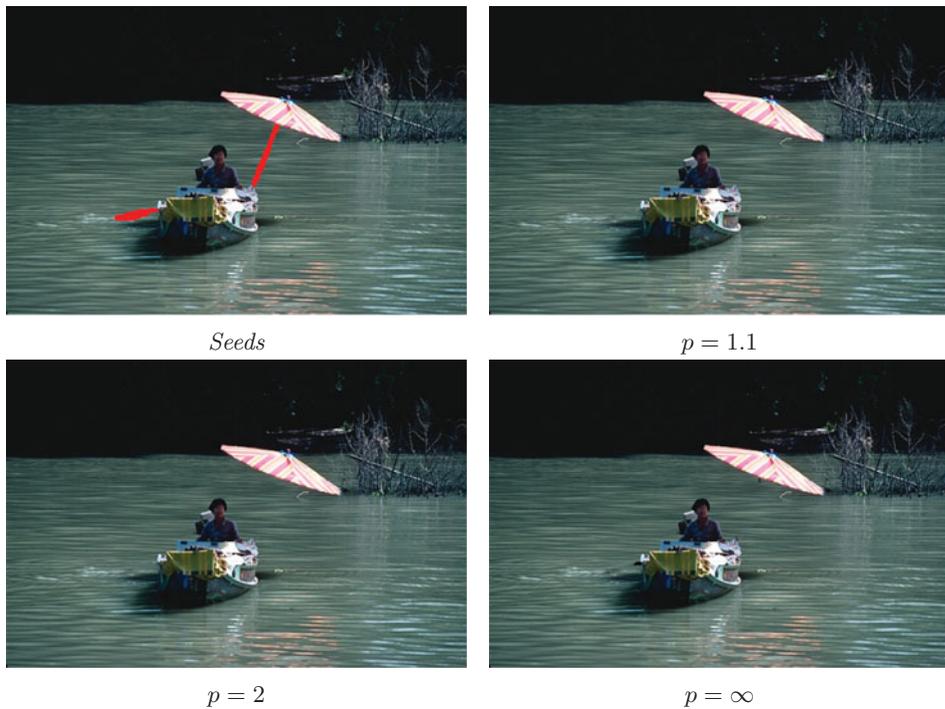


FIGURE 10. Natural colour image inpainting using non-local graph construction. See text for details.

$31 \times 31$  neighbourhood window and  $15 \times 15$  patches) graph construction. Figure 10 shows the results on a natural colour image, using non-local graph construction, and varying the  $p$  parameter of  $\Delta_{w,p}^N$ .

## 5 Conclusion

In this paper, we have introduced a new class of normalized  $p$ -Laplacian operators as a discrete adaptation of the game-theoretic  $p$ -Laplacian on weighted graphs. This class is based on new partial difference operator which interpolate between normalized 2-Laplacian, 1-Laplacian and  $\infty$ -Laplacian on graphs. This operator is also connected to non-local average operators such as non-local mean, non-local median and non-local midrange. It generalizes the normalized  $p$ -Laplacian on graphs for  $1 \leq p \leq \infty$ . We have shown the connections with local and non-local PDEs of  $p$ -Laplacian types and Tug-of-War games. We have proved existence and uniqueness of the Dirichlet problem involving operators of this new class. Finally, we have illustrated the interest and behaviour of such operators in some inverse problems in image processing and machine learning.

## References

- [1] ALPAYDIN, E. & ALIMOGLU, F. (1998) Pen-based recognition of handwritten digits data set. University of California, Irvine, Machine Learning Repository.
- [2] ALPAYDIN, E. & KAYNAK, C. (1995) Optical Recognition of Handwritten Digits Data Set. UCI Machine Learning Repository.
- [3] ANDREU, F., MAZÓN, J., ROSSI, J. & TOLEDO, J. (2008) A nonlocal  $p$ -laplacian evolution equation with neumann boundary conditions. *J. Math. pures et Appl.* **90**(2), 201–227.
- [4] ANDREU-VALLO, F., MAZÓN, J. M., ROSSI, J. D. & TOLEDO-MELERO, J. J. (2010) *Nonlocal Diffusion Problems*, Mathematical surveys and monographs; Vol. 165, American Mathematical Society.
- [5] ARIAS, P., FACCILOLO, G., CASELLES, V. & SAPIRO, G. (2011) A variational framework for exemplar-based image inpainting. *Int. J. Comput. Vis.* **93**(3), 319–347.
- [6] BARRETT, J. W. & LIU, W. (1993) Finite element approximation of the  $\infty$ -laplacian. *Math. Comput.* **61**(204), 523–537.
- [7] BERTOZZI, A. L. & FLENNER, A. (2016) Diffuse interface models on graphs for classification of high dimensional data. *SIAM Rev.* **58**(2), 293–328.
- [8] BOUGLEUX, S., ELMOATAZ, A. & MELKEMI, M. (2007) Discrete regularization on weighted graphs for image and mesh filtering. In: F. Sgallari, A. Murli & N. Paragios (editors), *Scale Space and Variational Methods in Computer Vision*, Springer, pp. 128–139.
- [9] BOUGLEUX, S., ELMOATAZ, A. & MELKEMI, M. (2009) Local and nonlocal discrete regularization on weighted graphs for image and mesh processing. *Int. J. Comput. Vis.* **84**(2), 220–236.
- [10] BRESSON, X., THOMAS, L., UMINSKY, D. & VON BRECHT, J. H. (2013) Multiclass total variation clustering. In: L. Bottou, Z. Ghahramani & K. Q. Weinberger (editors), *Advances in Neural Information Processing Systems*, Lake Tahoe, Nevada, USA, pp. 1421–1429.
- [11] BUADES, A., COLL, B. & MOREL, J.-M. (2008) Nonlocal image and movie denoising. *Int. J. Comput. Vis.* **76**(2), 123–139.
- [12] BÜHLER, T. & HEIN, M. (2009) Spectral clustering based on the graph  $p$ -laplacian. In: *Proceedings of the 26th Annual International Conference on Machine Learning*, Montreal, Quebec, Canada, ACM, pp. 81–88.
- [13] CHAMBOLLE, A., LINDGREN, E. & MONNEAU, R. (2012) A hölder infinity laplacian. *ESAIM: Control, Optimisation Calculus Variations* **18**(03), 799–835.
- [14] COIFMAN, R. R. & LAFON, S. (2006) Diffusion maps. *Appl. Comput. Harmon. Anal.* **21**(1), 5–30.
- [15] COUPRIE, C., GRADY, L., NAJMAN, L. & TALBOT, H. (2011) Power watershed: A unifying graph-based optimization framework. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(7), 1384–1399.

- [16] DESQUESNES, X., ELMOATAZ, A. & LÉZORAY, O. (2013) Eikonal equation adaptation on weighted graphs: Fast geometric diffusion process for local and non-local image and data processing. *J. Math. Imaging Vis.* **46**(2), 238–257.
- [17] DRÁBEK, P. (2007) The  $p$ -laplacian–mascot of nonlinear analysis. *Acta Math. Univ. Comenianae* **76**(1), 85–98.
- [18] ELMOATAZ, A., DESQUESNES, X., LAKHDARI, Z. & LÉZORAY, O. (2014) Nonlocal infinity laplacian equation on graphs with applications in image processing and machine learning. *Math. Comput. Simul.* **102**, 153–163.
- [19] ELMOATAZ, A., DESQUESNES, X. & LÉZORAY, O. (2012) Non-local morphological pdes and-laplacian equation on graphs with applications in image processing and machine learning. *IEEE J. Sel. Topics Signal Process.* **6**(7), 764–779.
- [20] ELMOATAZ, A., LEZORAY, O. & BOUGLEUX, S. (2008) Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing. *IEEE Trans. Image Process.* **17**(7), 1047–1060.
- [21] ELMOATAZ, A., LÉZORAY, O., BOUGLEUX, S. & TA, V. T. (2008) Unifying local and nonlocal processing with partial difference operators on weighted graphs. In: *Proceedings of the International Workshop Local and Non-Local Approximation in Image Processing*, Switzerland, pp. 11–26.
- [22] ELMOATAZ, A., LOZES, F. & TOUTAIN, M. (2016) Nonlocal pdes on graphs: From tug-of-war games to unified interpolation on images and point clouds. *J. Math. Imaging Vis.* **57**(3), 381–401.
- [23] ELMOATAZ, A., TOUTAIN, M. & TENBRINCK, D. (2015) On the  $p$ -laplacian and  $\infty$ -laplacian on graphs with applications in image and data processing. *SIAM J. Imaging Sci.* **8**(4), 2412–2451.
- [24] FALCONE, M., VITA, S. F., GIORGI, T. & SMITS, R. (2013) A semi-lagrangian scheme for the game  $p$ -laplacian via  $p$ -averaging. *Appl. Numer. Math.* **73**, 63–80.
- [25] GHONIEM, M., ELMOATAZ, A. & LEZORAY, O. (2011) Discrete infinity harmonic functions: Towards a unified interpolation framework on graphs. In: *Proceedings of the 18th IEEE International Conference Image Processing (ICIP)*, Brussels, Belgium, IEEE, pp. 1361–1364.
- [26] GILBOA, G. & OSHER, S. (2007) Nonlocal linear image regularization and supervised segmentation. *Multiscale Model. Simul.* **6**(2), 595–630.
- [27] GILBOA, G. & OSHER, S. (2008) Nonlocal operators with applications to image processing. *Multiscale Model. Simul.* **7**(3), 1005–1028.
- [28] HEIN, M. & BÜHLER, T. (2010) An inverse power method for nonlinear eigenproblems with applications in 1-spectral clustering and sparse pca. In: *Advances in Neural Information Processing Systems*, Curran Associates Inc., pp. 847–855.
- [29] KAWOHL, B. (2011) Variations on the  $p$ -laplacian. In: D. Bonheure, P. Takac *et al.* (editors), *Nonlinear Elliptic Partial Differential Equations*, Contemporary Mathematics, Vol. 540, pp. 35–46.
- [30] KERVRANN, C. (2004) An adaptive window approach for image smoothing and structures preserving. In: T. Pajdla and J. Matas (editors), *ECCV*, Lecture Notes in Computer Science, Vol. 3023, Prague, Czech Republic, Springer, pp. 132–144.
- [31] KEYSERS, D., DAHMEN, J., THEINER, T. & NEY, H. (2000) Experiments with an extended tangent distance. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, Vol. 2, IEEE, pp. 38–42.
- [32] LECUN, Y. & CORTES, C. (2010) MNIST handwritten digit database. URL: <http://yann.lecun.com/exdb/mnist/>. (last access end 2016).
- [33] MANFREDI, J. J., PARVIAINEN, M. & ROSSI, J. D. (2012) Dynamic programming principle for tug-of-war games with noise. *ESAIM: Control, Optimisation Calculus Variations* **18**(01), 81–90.
- [34] OBERMAN, A. (2005) A convergent difference scheme for the infinity laplacian: Construction of absolutely minimizing lipschitz extensions. *Math. Comput.* **74**(251), 1217–1230.
- [35] OBERMAN, A. M. (2013) Finite difference methods for the infinity laplace and  $p$ -laplace equations. *J. Comput. Appl. Math.* **254**, 65–80.

- [36] PERES, Y., SCHRAMM, O., SHEFFIELD, S. & WILSON, D. (2009) Tug-of-war and the infinity laplacian. *J. Am. Math. Soc.* **22**(1), 167–210.
- [37] PERES, Y. & SHEFFIELD, S. (2008) Tug-of-war with noise: A game-theoretic view of the  $p$ -laplacian. *Duke Math. J.* **145**(1), 91–120.
- [38] RUDD, M. (2014) Statistical exponential formulas for homogeneous diffusion. *Comm. Pure Appl. Anal.* **1**, 269–284.
- [39] SCHÖNLIEB, C.-B. & BERTOZZI, A. (2011) Unconditionally stable schemes for higher order inpainting. *Commun. Math. Sci.* **9**(2), 413–457.
- [40] TA, V.-T., ELMOATAZ, A. & LÉZORAY, O. (2011) Nonlocal pdes-based morphology on weighted graphs for image and data processing. *IEEE Trans. Image Process.* **20**(6), 1504–1516.
- [41] VAN GENNIP, Y., GUILLEN, N., OSTING, B. & BERTOZZI, A. L. (2014) Mean curvature, threshold dynamics, and phase field theory on finite graphs. *Milan J. Math.* **82**(1), 3–65.
- [42] ZELNIK-MANOR, L. & PERONA, P. (2004) Self-tuning spectral clustering. In: L. K. Saul, Y. Weiss & L. Bottou (editors), *Advances in Neural Information Processing Systems*, Vol. 17, MIT Press, pp. 1601–1608.