A scheme for the game p-Laplacian and its application to image inpainting

Published Version:
Carlini E., Tozza S. (2024). A scheme for the game p-Laplacian and its application to image inpainting. APPLIED MATHEMATICS AND COMPUTATION, 461, 1-10 [10.1016/j.amc.2023.128299].

Availability:
This version is available at: https://hdl.handle.net/11585/943916 since: 2023-10-05

Published:
DOI: http://doi.org/10.1016/j.amc.2023.128299

Elisabetta Carlini, Silvia Tozza, A scheme for the game p-Laplacian and its application to image inpainting, Applied Mathematics and Computation, Volume 461, 2024, 128299

The final published version is available online at: ***https://doi.org/10.1016/j.amc.2023.128299***

A scheme for the game p-Laplacian and its application to Image Inpainting

Elisabetta Carlini[a,*], Silvia Tozza[b]

[a] *Dept. of Mathematics, Sapienza University of Rome, P.le Aldo Moro 5, Rome, 00185, Italy*

[b] *Dept. of Mathematics, Alma Mater Studiorum University of Bologna, Piazza di Porta S. Donato 5, Bologna, 40126, Italy*

Abstract

We propose a new numerical scheme for the game p-Laplacian, based on a semi-Lagrangian approximation. We focus on the 2D version of the game p-Laplacian, with the aim to apply the new scheme in the context of image processing. Specifically, we want to solve the so-called inpainting problem, which consists in reconstructing one or more missing parts of an image using information taken from the known part. The numerical tests show the reliability of the proposed method and the advantages of taking a $p > 1$ in terms of execution time and accuracy.

Keywords: game p-Laplacian, Semi-Lagrangian scheme, Inpainting, Viscosity solution

---

1. Introduction

In this paper we address the following problem: Given two bounded open domains $D, \Omega \subset \mathbb{R}^2$ such that $D \subset \Omega$ with $\overline{\Omega} \cap \overline{D} = \emptyset$, and two functions $f : D \to \mathbb{R}$ and $F : \overline{\Omega} \setminus D \to \mathbb{R}$, the goal is to find a function $u : \overline{\Omega} \to \mathbb{R}$ solution of the following problem

$$\begin{cases} -\Delta_p^G u = f & \text{in } D, \\ u = F & \text{in } \overline{\Omega} \setminus D. \end{cases} \tag{1}$$

*Corresponding author

Email addresses: elisabetta.carlini@uniroma1.it (Elisabetta Carlini), silvia.tozza@unibo.it (Silvia Tozza)

Preprint submitted to Applied Mathematics and Computation *January 25, 2024*

The operator Δ_p^G denotes the game p-Laplacian and it has been introduced for the first time in [25] to model a stochastic game. The interest in such an operator derives from the observation that it can include in itself various operators as particular cases (i.e. the operator in the Aronsson equation [3], the infinity Laplacian [26], the mean curvature motion operator [20], or, in the case $p = 2$, a multiple of the ordinary Laplacian).

Our final aim is to deal with problem (1) with $f \equiv 0$ in the context of the inpainting problem. Inpainting in image processing consists in reconstructing one or more missing or damaged parts of an image using information taken from the known part. A grayscale image is interpreted as a bounded function $u : \overline{\Omega} \to [0, +\infty)$. Typically, Ω is a rectangular domain and $u(x)$ represents the intensity of the gray level at the point x. Usually, the missing part is called *inpainting domain* and it is denoted by D. Recently, the image inpainting has been also applied as a decoding step for image compression, see e.g. [19] and references therein for more details.

The numerical image inpainting methods can be roughly separated into two categories: variational and non-variational ones. In both methods the reconstructed image is obtained as solution of a partial differential equation (PDE). The first approach is based on a energy minimization model and the PDE is the Euler-Lagrange equation associated to the optimality condition. In the non-variational approach, the PDE does not necessarily derives from a variational principle. For a complete survey, we refer the interested reader to the books by Aubert and Kornprobst and by Schönlieb [2, 27].

In the context of differential approaches for facing the inpainting problem, the 1-Laplacian is used in order to propagate information in the directions of the isophotes, i.e. the direction orthogonal to the gradient of the image, $\nabla u^{\perp}$, see [5]. Anisotropic diffusion processes have been also considered. Among these we mention [6], where the anisotropic process is applied to the whole original image, with the purpose of minimizing the influence of noise on the estimation of the direction of the isophotes arriving at the damaged portion of the image. More recently, fourth order anisotropic diffusion processes have been proposed, in order to achieve better accuracy with respect to second order operator, see [21] and references therein.

Image inpainting was also addressed through the p-Laplacian in [28], showing better results than the performance of the 1-Laplacian. Here, we want to follow this idea by using the game p-Laplacian and explore the behavior for different values of p in terms of accuracy and execution time, as we will see later in the section related to numerical tests. Note that the game

p-Laplacian has been already used in image processing (see e.g. [15] for its application on weighted graphs with applications in image inpainting and data clustering).

With the aim to present some recent advances in the numerical discretization of (1), here we propose a new scheme, applying it to the inpainting problem for improving results.

Numerical schemes for second order possibly degenerate equations have been presented by several authors. We focus our attention on semi-Lagrangian (SL) discretization, which have been shown to be particularly suitable for the numerical treatment of degenerate diffusions, see [10]. For a comprehensive introduction to SL scheme, we refer to the book by Falcone and Ferretti [17]. A SL scheme for the mean curvature motion has been proposed in [11]. High-order SL techniques to treat possibly degenerate advection-diffusion equations are analyzed in [18], whereas SL methods for diffusion equation with non linear reaction term are addressed in [7]. A SL scheme for the game p-Laplacian has been proposed and analyzed in [16]. Compared to this scheme, we propose a method that does not require a minimization procedure to find the direction of diffusion, which is instead obtained explicitly using the numerical gradient. For this reason, the scheme here proposed results to have a lower computational cost with respect to the SL scheme proposed in [16]. The scheme we propose is much more in the spirit of [11, 13], where the second order operator is approximated by means of directional second finite differences.

The theoretical convergence of the scheme is beyond the scope of this work and will be addressed in the future.

The paper is organized as follows: in Section 2 we recall the definition of the game p-Laplacian and how we can rewrite it as a convex combination of the ∞-Laplacian and 1-Laplacian operators. In Section 3 we propose a new SL scheme, showing its performance in Section 4. In Section 5 we apply the SL scheme to the inpainting problem. The paper ends with final comments and future perspectives contained in Section 6.

2. Game p-Laplacian operator

Let us recall the definition of the so-called game p-Laplacian operator introduced by Peres and Sheffield [25], that is

$$\Delta_p^G u := \frac{1}{p}|\nabla u|^{2-p}\mathrm{div}(|\nabla u|^{p-2}\nabla u) \quad \text{for} \quad 1 < p < \infty. \tag{2}$$

With respect to the variational p-Laplacian operator, in the game version appears the multiplicative term $\frac{1}{p}|\nabla u|^{2-p}$ before the divergence. This term causes the game p-Laplacian to be singular at critical points ($\nabla u = 0$) for every $p \neq 2$, whereas the variational p-Laplacian is singular for any $1 < p < 2$. Both are degenerate for $p > 2$.

Well-posedness for this kind of problem should be understood in a weak sense, we refer to [24, 12, 4, 23] for the weak and viscosity theory framework. In [26], authors shows that problem (1), for the case of the infinity Laplacian, has a unique viscosity solution when F, f are uniformly continuous and f does not change sign, i.e. $\inf_{\overline{\Omega}} f > 0$ or $\sup_{\overline{\Omega}} f > 0$. See also [22] for an overview on the basic results of Tug-of-War games.

If u is a smooth function, by expanding the derivative in (2), we obtain

$$\Delta_p^G u = \frac{1}{p}\Delta_2 u + \frac{p-2}{p}|\nabla u|^{-2}\sum_{i,j}\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\frac{\partial^2 u}{\partial x_i \partial x_j}. \tag{3}$$

At that point, if we formally take the limit for $p \to \infty$, one can define the game ∞-Laplacian operator as

$$\Delta_\infty^G u := |\nabla u|^{-2}\sum_{i,j}\frac{\partial u}{\partial x_i}\frac{\partial u}{\partial x_j}\frac{\partial^2 u}{\partial x_i \partial x_j}. \tag{4}$$

Finally, at the points where $|\nabla u| \neq 0$, the game ∞-Laplacian can be viewed as the second derivative in the direction of ∇u, that is

$$\Delta_\infty^G u = \sigma_\infty(\nabla u)^T D^2 u \, \sigma_\infty(\nabla u), \tag{5}$$

where $D^2 u$ denotes the Hessian matrix, and $\sigma_\infty : \mathbb{R}^2 \to \mathbb{R}^2$ is defined as

$$\sigma_\infty(a) := \frac{1}{|a|}\begin{pmatrix} a_1 \\ a_2 \end{pmatrix}. \tag{6}$$

The game 1-Laplacian is defined as

$$\Delta_1^G u := \Delta_2 u - \Delta_\infty^G u. \tag{7}$$

If u is a smooth function, by using (3) and the definition (7), the game p-Laplacian can be expressed as a convex combination of the ∞-Laplacian and 1-Laplacian as

$$\Delta_p^G u = \frac{1}{p}\Delta_1^G u + \frac{1}{q}\Delta_\infty^G u, \tag{8}$$

with q such that $\frac{1}{p}+\frac{1}{q}=1$. Note that the game 1-Laplacian, at the points where $|\nabla u|\neq 0$, can be viewed as the second derivative in the direction of $(\nabla u)^{\perp}$, that is

$$\Delta_1^G u := \sigma_1(\nabla u)^T D^2 u\, \sigma_1(\nabla u), \tag{9}$$

where $\sigma_1 : \mathbb{R}^2 \to \mathbb{R}^2$ is defined as

$$\sigma_1(a) = \frac{1}{|a|}\begin{pmatrix} -a_2 \\ a_1 \end{pmatrix}. \tag{10}$$

3. A new semi-Lagrangian scheme for the game p-Laplacian

A simple way to construct a semi-Lagrangian scheme for (1) consists in discretizing the second order operators by a directional second finite difference. Supposing $\sigma \in \mathbb{R}^2$ is given, let us introduce a discretization parameter $\delta > 0$ and let us consider the following approximation:

$$\sigma^T D^2 u(x)\, \sigma \approx \frac{1}{\delta^2}\left(u(x+\delta\sigma)+u(x-\delta\sigma)-2u(x)\right). \tag{11}$$

A similar approximation with σ replaced by $\sigma_\infty(\nabla u)$ and $\sigma_1(\nabla u)$ can be considered in order to approximate (5) and (9), respectively. Such approximations are valid only in the non-singular case, i.e. when $\nabla u \neq 0$. By using (8) we derive, for x such that $\nabla u(x) \neq 0$, an approximation for (2) given by

$$\begin{aligned}\Delta u_p^G(x) \approx\ & \frac{1}{p\delta^2}\left(u(x+\delta\sigma_1(\nabla u(x)))+u(x-\delta\sigma_1(\nabla u(x)))-2u(x)\right) + \\ & \frac{1}{q\delta^2}\left(u(x+\delta\sigma_\infty(\nabla u(x)))+u(x-\delta\sigma_\infty(\nabla u(x)))-2u(x)\right).\end{aligned} \tag{12}$$

For simplicity, let us suppose Ω be the unit square, $\Omega := (0,1)\times(0,1)$. Given a integer N_h, we define a space discretization step $h = \frac{1}{N_h}$. Let us introduce a uniform grid $\mathcal{G}_h(\overline{\Omega}) = \{x_j = jh, j \in \{0,\dots,N_h\}^2\}$ and let us consider the following sets of indexes:

$$Q = \{j \in \mathbb{Z}^2 \text{ such that } x_j \in D\}, \quad Q_b = \{j \in \mathbb{Z}^2 \text{ such that } x_j \in \overline{\Omega}\setminus D\}. \tag{13}$$

For any $j \in Q$, we denote by $D_j[u]$ a centered finite difference approximation of $\nabla u(x_j)$ and we define two couples of discrete characteristics as

$$y_\infty^{\pm}(x_j) := x_j \pm \delta\sigma_\infty(D_j[u]), \qquad y_1^{\pm}(x_j) := x_j \pm \delta\sigma_1(D_j[u]). \tag{14}$$

Since $|\sigma_\infty| = |\sigma_1| = 1$, for δ small enough, characteristics never leave from $\overline{\Omega}$. In the case the characteristics are points which belong to D, we need to introduce an interpolation operator in order to reconstruct the value of u at these points. Given a grid function $v : \mathcal{G}_h(\overline{\Omega}) \to \mathbb{R}$, we denote by $I[v] : \Omega \to R$ a piecewise polynomial interpolation of v in Ω. Note that, in the case $y_1^\pm(x_j)$ or $y_\infty^\pm(x_j)$ lies in $\overline{\Omega}\backslash D$, there is no need to interpolate since the data F is given. The knowledge of F in $\overline{\Omega} \setminus D$ avoids extrapolation techniques, truncation or reflection of the characteristics, as done, respectively, in [7, 17, 9] in order to numerically treat boundary conditions in the context of SL schemes. Then, we define

$$\widetilde{I}[v](x) := \begin{cases} I[v](x) & x \in D, \\ F(x) & x \in \overline{\Omega} \setminus D. \end{cases} \tag{15}$$

In order to deal with the singular case, we consider a finite difference approximation of $\frac{1}{2}\Delta u = \Delta_2^G u(x)$ at the points x where $\nabla u(x) \approx 0$. This is in agreement with the definition of viscosity solution at points where the gradient vanishes, see Remark 2.1 in [16].
Summing up, we propose the following scheme to approximate (1). Find $v : \mathcal{G}_h(\overline{\Omega}) \to \mathbb{R}$ such that, for any $x_j \in \mathcal{G}_h(\overline{\Omega})$,

$$G_\rho(x_j, v) = 0, \tag{16}$$

where

$$G_\rho(x_j, v) := \begin{cases} S_\rho(x_j, v) - f_j & j \in Q, \\ v(x_j) - F(x_j) & j \in Q_b, \end{cases} \tag{17}$$

with $\rho := (h, \delta)$, and S_ρ defined as

$$\begin{aligned} S_\rho(x_j, v) := \quad & \frac{1}{p\delta^2}\left(\widetilde{I}[v](y_1^+(x_j)) + \widetilde{I}[v](y_1^-(x_j))\right) \\ & + \frac{1}{q\delta^2}\left(\widetilde{I}[v](y_\infty^+(x_j)) + \widetilde{I}[v](y_\infty^-(x_j))\right) - \frac{2}{\delta^2} v_j, \end{aligned}$$

for $j \in Q$ such that $|D_j[u]| > Ch^s$, where $s > 0$ is a fixed parameter, and

$$S_\rho(x_j, v) := \frac{1}{2\delta^2}\left(\sum_{i \in \mathcal{D}(j)} v_i - 4v_j\right),$$

for $j \in Q$ such that $|D_j[v]| \le Ch^s$, with $\mathcal{D}(j) = \{i \in Q \text{ such that } |i - j| = 1\}$.

4. Numerical Results

In this section we show the performance of the scheme in solving two problems for which the exact solution is known. We approximate heuristically the solution v of (16) by a fixed point iteration method based on a time marching approximation. Given $\Delta t > 0$ and an initial condition $v^0 : \mathcal{G}_h(\overline{\Omega}) \to \mathbb{R}$, we compute the sequence $(v^n)_{n\in\mathbb{N}}$ with $v^n : \mathcal{G}_h(\overline{\Omega}) \to \mathbb{R}$ by the following iterative scheme

$$\begin{cases} v_j^n = v_j^{n-1} + \Delta t S_\rho(x_j, v^{n-1}), & j \in Q, \\ v_j^n = F(x_j) & j \in Q_b. \end{cases} \tag{18}$$

Then the solution v of (16) is approximated as $v_j \simeq \lim_{n\to\infty} v_j^n$ for any $j \in Q \cup Q_b$.

The errors are obtained by comparing the numerical solution v^n with the exact solution u on the grid nodes using the following discrete norms

$$\|u(\cdot) - v^n(\cdot)\|_\infty := \max_{j\in Q} |u(x_j) - v_j^n|,$$

$$\|u(\cdot) - v^n(\cdot)\|_1 := h^2 \sum_{j\in Q} |u(x_j) - v_j^n|.$$

We denote by r_∞, r_1 the corresponding rates. For the fictitious time iteration, we consider the following stopping criterion

$$\|v^{n+1} - v^n\|_1 \le \varepsilon,$$

where $\varepsilon > 0$ is a given tolerance. Since we take a time step of size Δt and δ represents the size of a diffusion (mean squared displacement of Brownian walk in the direction of the gradient), δ should be proportional with $\sqrt{2\Delta t}$. We expect rate one of convergence, which will be confirmed by the simulations in the following two tests. The first problem is related to the case $p = \infty$ and the solution is not smooth. The second problem has smooth solution and it is tested with $p = 1.2$ and $p = \infty$. We observe that in Test 1 a smaller time step have to be considered, in addition to a regularization of the discrete gradient, in order to obtain the expected convergence rate. The restriction on the time step is due to accuracy reasons and not to stability reasons, as point out also in [18]. Heuristically, supposing that the direction σ is computed exactly, for smooth solution and for the choice $\delta = O(\sqrt{\Delta t})$

the truncation error has order given by $\Delta t + \frac{h^2}{\Delta t}$ (the first term is due to the remainder of the Taylor expansion and the second term is due to the linear interpolation). Then, the choice $\Delta t = h$ would optimize the consistency order, but, at the same time, would produce a stencil of size 2δ which can lead to low accuracy, especially when the solution to be reconstructed has a high gradient or is, in general, not smooth. This motivates the choice of $\Delta t = h^2$ in Test 1, since the solution is non-differentiable, and $\Delta t = h$ in Test 2, where the solution is smooth. In Test 2, scheme (18) is therefore implemented without the usual parabolic CFL condition. This remark seem to confirm that the proposed scheme retains a main advantage of SL schemes, with respect to explicit difference schemes, [14]. In particular, the results in the next tests indicate that the scheme (18) does not need to be implicit in order to be absolute stable.

4.1. Test 1

We compute the solution of problem (1) on $D = (-1,1) \times (-1,1)$ with $f = 0$, $\overline{\Omega} = [-1.5, 1.5] \times [-1.5, 1.5]$, $p = \infty$ and

$$F(x) = |x_1|^{4/3} - |x_2|^{4/3}.$$

We recall that this classical benchmark has an explicit solution, which is $u(x) = |x_1|^{\frac{4}{3}} - |x_2|^{\frac{4}{3}}$, known as *Aronsson function*. The solution is only continuous and is not differentiable in $x_1 = 0$ and $x_2 = 0$. For this reason, we need to regularize the gradient as following

$$\widetilde{D}_j[u] = \frac{1}{9} \left(\sum_{i \in \mathcal{D}^9(j)} D_i[u] \right)$$

with

$$\mathcal{D}^9(j) = \{ i \in Q \text{ such that } \|i - j\|_\infty \leq 1, j \in Q \}$$

and $D_i[u]$ is computed by centered finite difference with step $2h$. We compute the errors in the ∞ and 1 norm, with parameters $\delta = 2\sqrt{2\Delta t}$, $\Delta t = h^2$, $C = 1$, $s = 0.5$, $v^0 = 0$, and $\varepsilon = 1e - 8$. In Table 1, we show errors, convergence rates, and number of iterations needed to verify the stopping criterion. Rates greater than one are obtained in both norms, except in the last refinement where a degradation of the rate with respect to the 1 norm is shown, compared to an almost two order visible for the previous refinements.

Table 1: Test1. Errors and rates for $p = \infty$.

h	$\|\cdot\|_\infty$	$\|\cdot\|_1$	r_∞	r_1	n
$1.00 \cdot 10^{-1}$	$7.65 \cdot 10^{-2}$	$3.35 \cdot 10^{-2}$			115
$5.00 \cdot 10^{-2}$	$3.06 \cdot 10^{-2}$	$1.12 \cdot 10^{-2}$	1.32	1.58	311
$2.50 \cdot 10^{-2}$	$1.21 \cdot 10^{-2}$	$3.08 \cdot 10^{-3}$	1.33	1.82	1055
$1.25 \cdot 10^{-2}$	$4.72 \cdot 10^{-3}$	$2.27 \cdot 10^{-3}$	1.35	0.44	3630

4.2. Test 2

We approximate the solution of problem (1) on $D = (-1, 1) \times (-1, 1)$ with $f = 1$, $\overline{\Omega} = [-1.5, 1.5] \times [-1.5, 1.5]$ and

$$F(x_1, x_2) = \frac{1 - x_1^2 - x_2^2}{2}.$$

This problem has an exact solution given by $u = F$ for any $p > 1$. We consider $p = 1.2$ and we compute the errors in the ∞ and 1 norm, with parameters $\delta = \sqrt{2\Delta t}$, $\Delta t = h$, $C = 0.1$, $s = 1$, $v^0 = 0$, and $\varepsilon = 1e - 5$. In Tables 2 and 3, we report errors, convergence rates, and number of iterations needed to verify the stopping criterion, for the choices $p = 1.2$ and $p = \infty$, respectively. Both tables show a rate of convergence mostly close to 1 in both norms. Note that, since this test has a smooth exact solution, much larger time step Δt is allowed.

Table 2: Test2. Errors and rates for $p = 1.2$.

h	$\|\cdot\|_\infty$	$\|\cdot\|_1$	r_∞	r_1	n
$1.00 \cdot 10^{-1}$	$1.03 \cdot 10^{-2}$	$1.89 \cdot 10^{-2}$			33
$5.00 \cdot 10^{-2}$	$5.28 \cdot 10^{-3}$	$9.69 \cdot 10^{-3}$	0.96	0.96	66
$2.50 \cdot 10^{-2}$	$2.75 \cdot 10^{-3}$	$5.17 \cdot 10^{-3}$	0.94	0.90	125
$1.25 \cdot 10^{-2}$	$1.35 \cdot 10^{-3}$	$2.58 \cdot 10^{-3}$	1.022	1.00	323

5. Application to the inpainting problem

In this Section we apply the new proposed SL scheme to the inpainting problem. The purpose is also to compare different choices of values for p, considering both, a qualitative and a quantitative analysis of the results. As

Table 3: Test2. Errors and rates for $p = \infty$.

h	$\|\cdot\|_\infty$	$\|\cdot\|_1$	r_∞	r_1	n
$1.00 \cdot 10^{-1}$	$1.08 \cdot 10^{-2}$	$2.53 \cdot 10^{-2}$			78
$5.00 \cdot 10^{-2}$	$7.98 \cdot 10^{-3}$	$1.36 \cdot 10^{-2}$	0.43	0.89	139
$2.50 \cdot 10^{-2}$	$4.56 \cdot 10^{-3}$	$8.97 \cdot 10^{-3}$	0.80	0.60	262
$1.25 \cdot 10^{-2}$	$2.11 \cdot 10^{-3}$	$4.18 \cdot 10^{-3}$	1.11	1.10	498

Image Quality Metrics for a quantitative evaluation, we consider the following:

- The Mean Squared Error (MSE), defined as

$$MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} (I_{true}(i,j) - I_{approx}(i,j))^2 \tag{19}$$

  where I_{true} represents the original image not damaged (if given), I_{approx} denotes the image resulting after the iterative process of our scheme, $M \times N$ indicates the size of the image.

- Peak Signal-to-Noise Ratio (PSNR) computed via the Matlab routine $peaksnr = psnr(I\_approx, I\_true)$.
  A greater PSNR value indicates better image quality. PSNR is the ratio between the maximum possible power of an image and the power of corrupting noise that affects the quality of its representation. In formula, we can define it as

$$PSNR = 10 \log_{10}(\frac{R^2}{MSE}) \tag{20}$$

  where R is the maximum fluctuation in the input image data type (if the input image has a double-precision floating-point data type, then $R = 1$. If it has an 8-bit unsigned integer data type, $R = 255$, etc.).

- Structural Similarity Index Measure (SSIM) defined as follows:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{21}$$

where μ_x is the average of I_{true}, μ_y is the average of I_{approx}, σ_x^2 is the variance of I_{true}, σ_y^2 is the variance of I_{approx}, σ_{xy} is the covariance of I_{true} and I_{approx}, c_1 and c_2 are constants that are proportional to the dynamic range of the pixel values. SSIM is a perception-based model which takes into account inter-dependencies between pixels as index of information on the structure of the objects under observation. Greater value of SSIM, closer to 1, corresponds to a better quality of the image in terms of similarity with the reference image I_{true}. ($SSIM = 1$ means that the two considered images are equal).

These Full-Reference Quality Metrics are the most common used in inpainting papers [1].
For all the tests illustrated in this section, we consider $f = 0$, and we fix the following parameters: $h = \Delta t = 1, \delta = \sqrt{2\Delta t}, s = 1$.

5.1. Test 1: Robin Hood with lettering

We start the numerical experiments related to the inpainting problem considering a real image representing a statue of Robin Hood (size 500×667), visible in Figure 1. The missing part is behind the writings and the purpose of this test is to compare the results obtained from the new proposed scheme with respect to different choices of the parameter p. The results obtained choosing $p = 1$ and $p = 1.2$ are visible in Figure 2, first row. The qualitative comparison is enough in this case to conclude that the case with a bigger value of p is better, as also stressed in the second row of the same figure, where artifacts are visible in the results related to the case with $p = 1$ which are not present in the case $p = 1.2$. Some of these artifacts are highlighted inside the red circles and compared with the absence of them in the case with $p = 1.2$ (zooming in for a better comparison is recommended). As stopping rule, we fix a common number of iterations to be done, equal to $itermax = 1000$.

5.2. Test 2: Sleeping dog

For this second test, we consider a real image of a sleeping dog (size 601×421), for which we know the original image (see Figure 3). We damaged the original image I_{true} in two different ways, visible in Figure 4: adding three small ellipses or three larger hearts. The results related to the first damaged image, i.e. the sleeping dog with three ellipses, are visible in Figure 5. For this numerical test, one needs a quantitative evaluation in addition to the qualitative comparison done by the human visual perception. In fact, looking at the results in Figure 5, it is almost impossible to distinguish which value of p leads to the best performance. For this reason, we compute the Image Quality Metrics introduced at the beginning of Subsect. 5.2. The errors are reported into Table 4. Looking at Table 4, we can observe that comparable results with a fixed common tolerance ($\varepsilon = 0.001$) are achieved with a much smaller number of iterations choosing a $p > 1$ (193 using $p = 1.2$, and 85 with

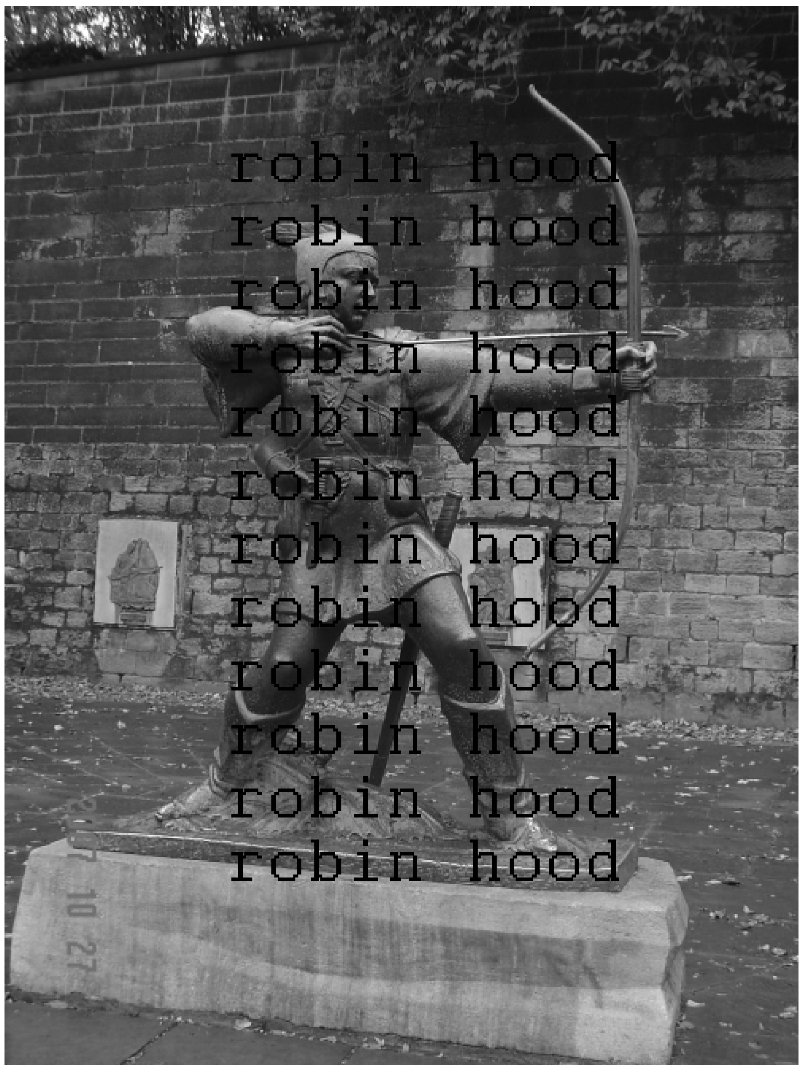

Figure 1: Test 1: Damaged real image of a statue of Robin Hood.

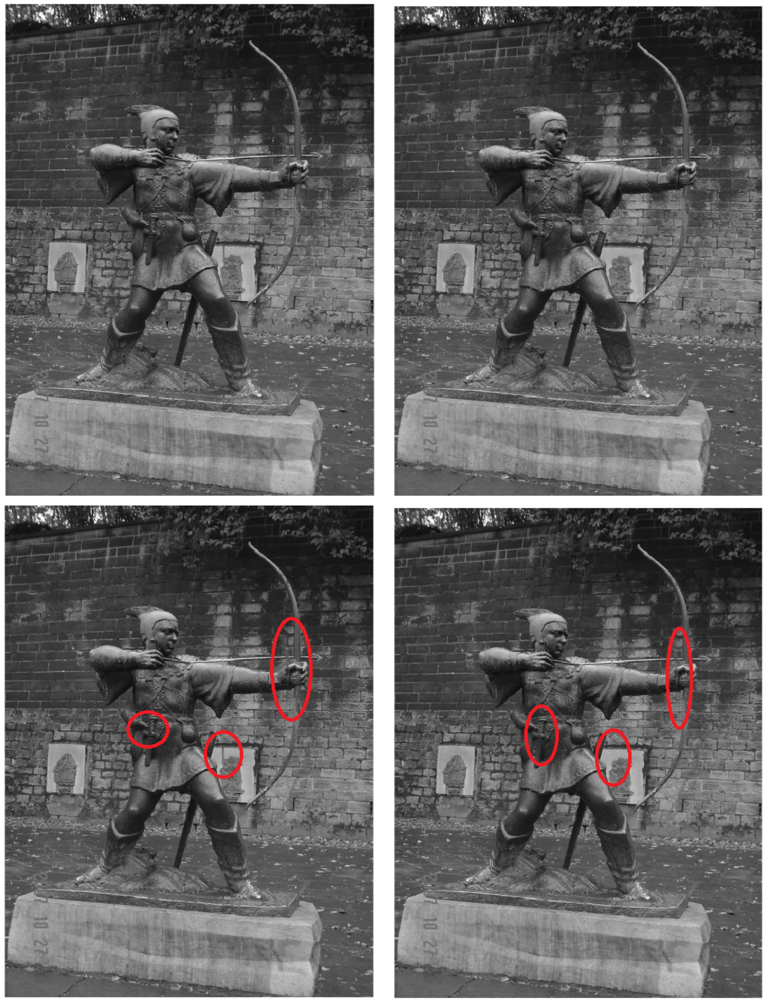

Figure 2: Test 1: First row: Restored images obtained with $p = 1$ (on the left) and $p = 1.2$ (on the right), and $itermax = 1000$. Second row: The same results visible in the first row with some artifacts highlighted in red related to the results with $p = 1$, which disappear with $p = 1.2$. Zooming in for a better visualization.

$p = 2.0$). Analyzing the errors reported in Table 4, we see that best results are achieved choosing $p = 1.2$, but all the errors related to the three cases are of the same order of magnitude. The big difference lies in the number of iterations.

Starting from the second damaged image, i.e. the sleeping dog with three hearts visible in Figure 4 on the right, we want to illustrate the behavior

Figure 3: Test 2: Original real image of a sleeping dog.

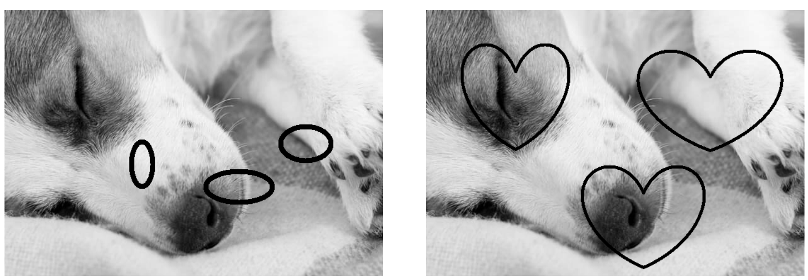

Figure 4: Test 2: Damaged images. On the left, three ellipses representing the missing part of the sleeping dog image. On the right, three hearts are the missing part.

Table 4: Test 2: Image Quality Metrics MSE, PSNR, and SSIM related to the results visible in Figure 5.

Game p-Laplacian	iter	**MSE**	**PSNR**	**SSIM**
$p = 1.0$	1000	3.34E-05	4.48E+01	9.95E-01
$p = 1.2$	193	2.56E-05	4.59E+01	9.95E-01
$p = 2.0$	85	2.89E-05	4.54E+01	9.95E-01

of the proposed scheme varying much more the value of p. We considered $p = 1$, $p = 1.2$, $p = 2$, $p = 5$, $p = 10$. The achieved results obtained

Figure 5: Test 2. From left to right: Restored images obtained with $p = 1$, $p = 1.2$, $p = 2$, respectively, starting from the sleeping dog with three ellipses. Tollerance $\varepsilon = 0.001$, $C = 0.01$. Zooming in for a better visualization.

with the chosen values of p are visible in Figure 6. The associated quality metrics are reported in Table 5. Looking at the table, we can observe that as p value increases, comparable accuracy is obtained with fewer iterations, which decreasing by increasing p. The MSE error decreases by increasing the value of p, whereas the PSNR and SSIM metrics are almost the same for all the p. The big advantage is, hence, in terms of the execution time.

Table 5: Test 2: Image Quality Metrics MSE, PSNR, and SSIM related to the results visible in Figure 6.

Game p-Laplacian	iter	**MSE**	**PSNR**	**SSIM**
$p = 1.0$	191	8.37E-05	4.08E+01	9.91E-01
$p = 1.2$	77	8.23E-05	4.08E+01	9.91E-01
$p = 2.0$	66	8.09E-05	4.09E+01	9.91E-01
$p = 5.0$	62	8.06E-05	4.09E+01	9.91E-01
$p = 10.0$	62	8.05E-05	4.09E+01	9.91E-01

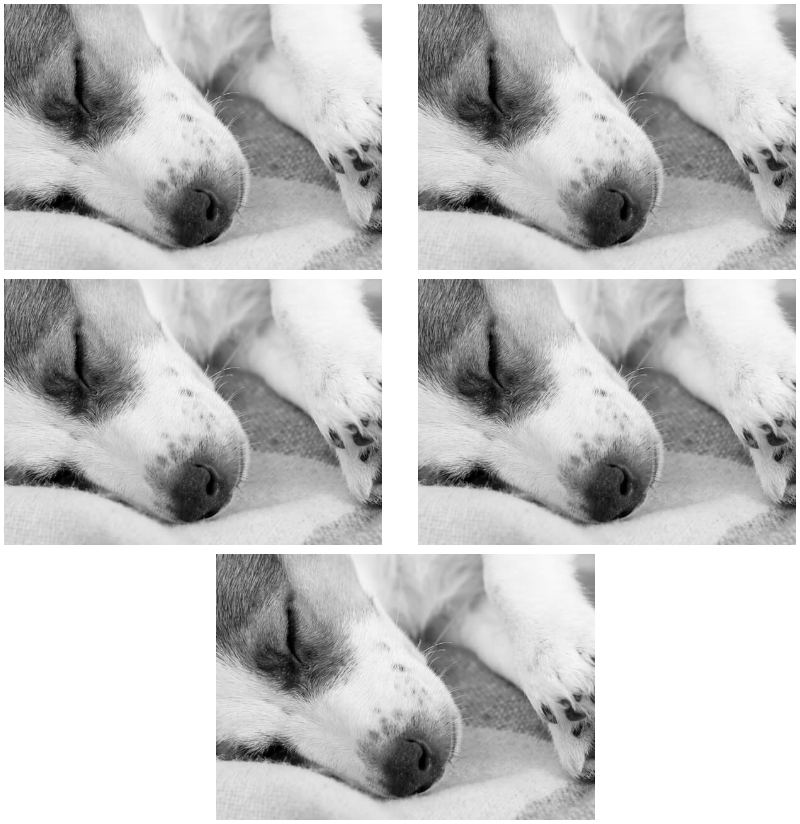

Figure 6: Test 2. From left to right: Restored images obtained with $p = 1$, $p = 1.2$, $p = 2$, $p = 5$, $p = 10$, respectively, starting from the sleeping dog with three hearts. $\varepsilon = 0.001$, $C = 0.1$. Zooming in for a better visualization.

6. Conclusions and future perspectives

In this work we have presented a new semi-Lagrangian scheme for the game p-Laplacian equation, by observing that the game p-Laplacian operator can be expressed as a convex combination of the ∞-Laplacian and 1-Laplacian. We have applied the new proposed scheme to the inpainting image problem, analyzing the results in terms of both qualitative and quantitative accuracy. The numerical simulations have showed the behavior of the proposed scheme, with the advantage of considering larger values of p, which allows one to obtain good accuracy in fewer iterations. In the future, we would like to investigate theoretically the numerical scheme introduced in this work and its properties, such as stability and convergence. Moreover, a high order extension of the scheme could be addressed, in particular implicit-explicit Runge-Kutta schemes for the fictitious time marching (see for instance [8]) coupled with a high order approximation of the p-Laplacian may be considered.

Acknowledgments

The authors are members of the INdAM Research Group GNCS. This work was carried out within the INdAM-GNCS project "Metodi numerici per l'imaging: dal 2D al 3D", Code CUP_E55F22000270001.

References

[1] M. Ali Qureshi, M. Deriche, A. Beghdadi, A. Amin, A critical survey of state-of-the-art image inpainting quality assessment metrics, *Journal of Visual Communication and Image Representation*, **49**:177–191, 2017.

[2] G. Aubert, P. Kornprobst, *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*, Springer, 2nd Edition, 2010

[3] E.N. Barron, L.C. Evans, R. Jensen, The infinity Laplacian, Aronsson's equation and their generalizations, *Trans. Amer. Math. Soc.*, **360**(1):77–101, 2008.

[4] G. Barles, Solutions de viscosité des équations de Hamilton-Jacobi, *Mathématiques & Applications* 17, Springer-Verlag, Paris, 1994.

[5] M. Bertalmío, V. Caselles, G. Haro, G. Sapiro, *PDE-Based Image and Surface Inpainting*. In: Paragios, N., Chen, Y., Faugeras, O. (Eds.) Handbook of Mathematical Models in Computer Vision. Springer, Boston, MA, 2006

[6] M. Bertalmío, G. Sapiro, V. Caselles, C. Ballester, *Image Inpainting.* In: Kurt Akeley (Ed.) Proceeding of the SIGGRAPH, pp 417.424. ACM Press, ACM SIGGRAPH, Addison Wesley Longman, 2000.

[7] L. Bonaventura, E. Carlini, E. Calzola, R.Ferretti, Second Order Fully Semi-Lagrangian Discretizations of Advection-Diffusion-Reaction Systems, *Journal of Scientific Computing*, **88**(23):1–29, 2021.

[8] S. Boscarino, F. Filbet, G. Russo, High order semi-implicit schemes for time dependent partial differential equations. *Journal of Scientific Computing*, **68**(3):975–1001, 2016.

[9] E. Calzola, E. Carlini, D. Xavier, F.J. Silva, A semi-Lagrangian scheme for Hamilton-Jacobi-Bellman equations with oblique derivatives boundary conditions *Numer. Math.*, **153**(1):49–84, 2023.

[10] F. Camilli, M. Falcone, An approximation scheme for the optimal control of diffusion processes, *Mathematical Modelling and Numerical Analysis*, **29**(1):97–122, 1995.

[11] E. Carlini, M. Falcone, R. Ferretti, Convergence of a large time step scheme for Mean Curvature Motion, *Interfaces and Free Boundaries*, **12**:409–441, 2011.

[12] M.G. Crandall, H. Ishii, P.L. Lions, User's guide to viscosity solutions of second order partial differential equations, *Bull. Amer. Math. Soc.*, **27**:1–67, 1992.

[13] M.G. Crandall, P.L. Lions, Convergent difference schemes for nonlinear parabolic equations and mean curvature motion, *Numer. Math.*, **75**:17–41, 1996.

[14] F. Del Teso, E. Lindgren, A finite difference method for the variational p-Laplacian, *Journal of Scientific Computing*, **90**(67):1–31, 2022.

[15] A. Elmoataz, X. Desquesnes, M. Toutain, On the game p-Laplacian on weighted graphs with applications in image processing and data clustering, *European Journal of Applied Mathematics*, **28**(6):922–948, 2017.

[16] M. Falcone, S. Finzi Vita, T. Giorgi, R.G. Smits, A semi-Lagrangian scheme for the game p-Laplacian via p-averaging, *Applied Numerical Mathematics*, **73**:63–80, 2013.

[17] M. Falcone, R. Ferretti, *Semi-Lagrangian Approximation Schemes for Linear and Hamilton–Jacobi Equations*, SIAM, 2013.

[18] R. Ferretti, A technique for high-order treatment of diffusion terms in semi-lagrangian schemes, *Commun. Comput.*, **8**(2):445–470, 2010.

[19] I. Galić, J. Weickert, M. Welk, A. Bruhn, A. Belyaev, H.-P. Seidel, Image compression with anisotropic diffusion, *J Math Imaging Vis*, **31**(2–3):255–269, 2008.

[20] Y. Giga, *Surface Evolution Equations – a level set method*, Hokkaido University Technical Report Series in Mathematics, n. 71, 2002.

[21] A. Halim, B.V. Rathish Kumar, *An anisotropic PDE model for image inpainting*, Computers & Mathematics with Applications, **79**(9):2701–2721, Elsevier, 2020.

[22] M. Lewicka, *A Course on Tug-of-War Games with Random Noise: Introduction and Basic Constructions*, 1st ed., Cham, Springer, 2020.

[23] P.Lindqvist, *Notes on the Stationary p-Laplace Equation.* Springer Briefs in Mathematics. Springer, Cham (2019)

[24] P. Juutinen, P. Lindqvist, J. Manfredi, On the equivalence of viscosity solutions and weak solutions for a quasi-linear equation, *SIAM J. Math. Anal.*, **33**:699–717, 2001.

[25] Y. Peres, S. Sheffield, Tug-of-war with noise: a game-theoretic view of the p-Laplacian, *Duke Math. J.*, **145**:91–120, 2008.

[26] Y. Peres, O. Schramm, S. Sheffield, D.B. Wilson, Tug-of-war and the infinity Laplacian, *J. Amer. Math. Soc.*, **22**:167–210, 2009.

[27] C.-B. Schönlieb, *Partial Differential Equation Methods for Image Inpainting*, Cambridge University Press, 2015.

[28] H.Y. Zhang, Q.C. Peng, Y.D. Wu, Wavelet Inpainting Based on p -Laplace Operator, *Acta Automatica Sinica*, **33**(5):546–549, 2007.