Generalised diffusion based regularization for inverse problem in image processing

Innocent Souopgui, Arthur Vidard, François-Xavier Le Dimet

2nd International Conference on Engineering Optimization
September 6-9, 2010, Lisbon, Portugal

Generalised diffusion based regularization for inverse problems in image processing

Innocent Souopgui, François-Xavier Le Dimet, Arthur Vidard

INRIA and Laboratoire Jean-Kuntzmann
Université Joseph Fourier
38041 Grenoble
FRANCE
innocent.souopgui@imag.fr, ledimet@imag.fr, arthur.vidard@imag.fr

Abstract
Due to the ill-posedness of inverse problems, all known a priori informations on the solution must be taken into account while solving such a problem. These informations are generally used as constraints to get the appropriate solution. In usual cases, constrains are turned into penalization of some characteristics of the solution. A common constraint is the regularity of the solution leading to regularization techniques for inverse problems. Regularization by penalization is affected by three principal problems: - as the cost function is composite, the convergence rate of minimization algorithms decreases - when adequate regularization functions are defined, one has to define weighting parameters between regularization functions and the objective function to minimize. It is very difficult to get optimal weighting parameters since they are strongly dependant on the observed data and the truth solution of the problem. The third problem affects regularization based on the penalization of spatial variation. Although the penalization of spatial variation is known to give best results (gradient penalization and second order regularization), there is no physical underlying foundation. Penalization of spatial variations lead to smooth solution that is an equilibrium between good and bad characteristics. Here, we introduce a new approach for regularization of ill-posed inverse problems. Penalization of spatial variations is weighted by an observation based trust function. The result is a generalized diffusion operator that turns regularization into pseudo covariance operators. All the regularization informations are then embedded into a preconditioning operator while solving the problem. On one hand, this method do not need any extra terms in the cost function, and of course is affected neither by the ill-convergence due to composite cost function, nor by the choice of weighting parameters. On the other hand, The trust function introduced here allows to take into account the observation based a priori knowledges on the problem. We suggest a simple definition of the trust function for inverse problems in image processing. Preliminary results show a promising method for regularization of inverse problems.
Keywords: regularization, inverse problems, image processing.

1. Introduction

Motion estimation is an example of inverse problem in computer vision and images processing. The expression inverse problem is used as opposite to direct problem. Given a complete description of the behavior of a physical system in terms of mathematical models and physical parameters, the state of the system can be computed using the mathematical model; this is known as the forward (direct, modeling or simulation) problem. The inverse problem consists in using given measurements of the system's state to infer the values of the parameters characterizing the model. In motion estimation the inverse problem consists in determining motion vectors that describe the transformation from one 2D image to another. Motion estimation is affected by ill-posedness as general inverse problem. Due to the ill-posedness, one has recourse to a priori informations on the solution while solving inverse problems. A priori informations include but are not limited to

- background and background errors covariance
- regularity of the solution

These informations are generally used as constraints to get the appropriate solution when optimization techniques are used to solve an inverse problem. In usual cases, constrains are turned into penalization of some characteristics of the solution. A common constraint is the regularity of the solution leading to regularization techniques for inverse problems. Until now, regularization is generally used as penalization while solving inverse problems. This practice is affected by two principal problems: - as the cost function is composite, the convergence rate of optimization algorithms decreases - when adequate regularization functions are defined, one have to define balance parameters between regularization functions and the objective function to minimize. The determination of the optimal weighting parameter requires second order analysis. Here, we suggest a new approach for regularization of ill-posed inverse problems. We introduce an observation based trust function that is used to define an appropriate norm for the cost function. This approach does not need extra terms in the cost function, and of course is not affected nor by the ill-convergence due to composite cost function, nor by the choice of weighting parameters. The present document is organized as followed : in section (2), we present inverse problems in a general framework, the use of a priori informations while solving inverse problems. In section (3), we present regularization methods for inverse problems; we emphasize on vector fields regularization. In section (4), we present the derivation of the new approach and comparisons with classical methods.

2. Inverse problems

2.1. Definition of inverse problems

Direct problem. Given a physical system whose the state $\mathbf{y} \in \mathcal{Y}$ can be defined as a function of a so called control variable $\mathbf{v} \in \mathcal{V}$

$$\begin{array}{lccc} \mathcal{M}: & \mathcal{V} & \rightarrow & \mathcal{Y} \\ & \mathbf{v} & \mapsto & \mathbf{y} = \mathcal{M}(\mathbf{v}) \end{array} \tag{1}$$

The model $\mathcal{M}$ (that link the control space $\mathcal{V}$ to the state space $\mathcal{Y}$) defines the direct problem. Given a realization of the control variable $\mathbf{v}$, this problem has a unique solution in the deterministic case. It is common to have not a realization of the control variable, but observations of the system state. The problem of inferring the control variable from observations is known as an inverse problem.

Inverse problem. The inverse problem associated to the direct problem (equation 1) is defined in term of optimization problem as followed :

$$\text{find } \mathbf{v}^* = \text{ArgMin}(J(\mathbf{v})), \mathbf{v} \in \mathcal{V} \tag{2}$$

where

$$J(\mathbf{v}) = J_o(\mathbf{v}) = \frac{1}{2}\|\mathcal{M}(\mathbf{v}) - \mathbf{y^o}\|^2_{\mathcal{O}} \tag{3}$$

$\|.\|_{\mathcal{O}}$ is the appropriate norm (taking into account observations covariance errors) in the observation space $\mathcal{O}$

The problem defined by (equation 2) is known as the unconstrained inverse problem.

The existence and uniqueness of the solution to the unconstrained problem (equation 2) is guaranteed if J is strictly convex and lower semi continuous with

$$\lim_{\|\mathbf{v}\| \rightarrow +\infty} J(\mathbf{v}) \rightarrow +\infty$$

under these conditions, if J is differentiable, then the solution to the unconstrained inverse problem (equation 2) is also the solution of the Euler-Lagrange equation

$$\nabla J(\mathbf{v}) = \mathbf{0} \tag{4}$$

To address the ill-posedness, one uses of all a priori knowledge of the properties of the solution.

2.2. Use of a priori knowledges in solving inverse problemsTitle

A priori knowledges are a set of constraints on the solution of the inverse problems. These constraints define a subset $\mathcal{W} \subset \mathcal{V}$ of admissible candidates leading to a constraint problem defined as

$$\begin{array}{ll} \text{find} & \mathbf{v}^* = \text{ArgMin}(J(\mathbf{v})) \\ & \mathbf{v} \in \mathcal{W} \end{array} \tag{5}$$

Here, we are interrested in cases where the set of admissible solutions can mathematically be defined as $\mathcal{W} = \{\mathbf{v} \in \mathcal{V} / g(\mathbf{v}) = 0\}$, the function g being to define. In this case, the constraint problem can be reduced to the unconstrained penalized problem

$$\text{find } \mathbf{v}^*_\epsilon = \text{ArgMin}(J_o(\mathbf{v}) + \frac{1}{\epsilon_c} J_c(\mathbf{v})), \mathbf{v} \in \mathcal{V} \tag{6}$$

where J_o is the observation cost function defined by (equation 3) and the constraint cost function J_c is defined as

$$J_c(\mathbf{v}) = \frac{1}{2}\|g(\mathbf{v})\|^2 \tag{7}$$

The solution $\mathbf{v}^*_{\epsilon_\mathbf{c}} \to \mathbf{v}^*$ when $\epsilon_c \to 0$. Instead of using parameter ϵ_c and let it go to zero, one can use a multiplicative parameter α_c and let it go to infinity. We are going to consider this case in the remainder part of the document. *It is known that pure penalization as defined above is not numerically efficient; it is better to used augmented Lagrangian algorithms see Glowinski et Le Tallec [2]*

Developpement here will be limited to background informations and the regularity of the solution. In this cases, the goal is usually not to find the exact solution $\mathbf{v} \in \mathcal{W}$, but to find the solution that realizes the best fit between the observation cost function and the constraint cost function. This is choosing the best parameter ϵ_c or α_c.

Background and background errors covariance. If one gets from some previous process an approximation of the control state and the associate covariance error also known as background and background covariance errors, one may asks to the computed solution to be closed to this background. This can be defined in term of penalization as

$$\text{find } \mathbf{v}^*_{\alpha_\mathbf{b}} = \text{ArgMin}(J_o(\mathbf{v}) + \alpha_b J_b(\mathbf{v})), \mathbf{v} \in \mathcal{V} \tag{8}$$

where α_b is the weighting parameter associated to the background part of the cost function defined as

$$J_b(\mathbf{v}) = \frac{1}{2}\|\mathbf{v} - \mathbf{v^b}\|^2_\mathcal{V} \tag{9}$$

well known as Tikhonov regularization [1]. $\mathbf{v^b}$ is the background knowledge of the solution, and $\|\cdot\|_\mathcal{V}$ the appropriate norm defined in term of the background covariance errors.The norm $\|\cdot\|_\mathcal{V}$ is defined in terms of the covariance errors matrix $\mathbf{B}$ as $\|\mathbf{x}\|^2_\mathcal{V} = \|\mathbf{x}\|^2_{\mathbf{B}^{-1}} = \langle \mathbf{x}, \mathbf{B^{-1}x}\rangle$ with $\langle \mathbf{x}, \mathbf{y}\rangle$ the dot product in the appropriate space. The definition of this norm is based on the property that $\langle \mathbf{x}, \mathbf{Ay}\rangle$ defined a dot product if $\mathbf{A}$ is symmetric and positive definite; this is the case for covariance matrices and their inverse.

Background informations are very important in solving inverse problems; this is a simple way to address the ill-posedness of the problem. Even in the case where there is no background information, it is a usual practice to consider the zero background constraining the solution to have small norm. In real live applications, background comes from previous analysis; this is the case of forecast centers.

It is common to define the cost function J in term of the increment $\delta\mathbf{v} = \mathbf{v} - \mathbf{v^b}$ leading to incremental problem,

$$J(\delta\mathbf{v}) = \mathbf{\frac{1}{2}}\|\mathcal{M}(\mathbf{v^b} + \delta\mathbf{v}) - \mathbf{y^o}\|^\mathbf{2}_\mathcal{O} + \mathbf{\frac{1}{2}}\alpha_\mathbf{b}\|\delta\mathbf{v}\|^\mathbf{2}_\mathcal{V} \tag{10}$$

Regularity of the solution. Sometime, the physics of the problem defines the regularity of admissible solutions (eg. irrotational or divergence free flow.) These are constraints defines as functions of the derivatives of the control variable. In these case, one defines the penalized problem

$$\text{find } \delta\mathbf{v}^*_{\alpha_\mathbf{r}} = \text{ArgMin}(J_o(\delta\mathbf{v}) + \alpha_b J_b(\delta\mathbf{v}) + \alpha_r J_r(\delta\mathbf{v})), \mathbf{v} \in \mathcal{V} \tag{11}$$

where α_r is the weighting parameter associated to the regularization part of the cost function defined in terms of the derivatives of the control variable. Regularization will be explored in more details in section (3.)

3. Vector fields regularization

As we said previously, regularization is a class of a priori knowledges used to address the ill-posedness while solving inverse problem. One adds regularization terms J_r to the cost function. The function J_r is based on the derivatives of $\mathbf{v}$. The order of the derivatives used in the definition of J_r defines the order of the regularization. We will name m-order regularization those involving up to $m-$order derivatives. It is useful to give some specifications of the notations defined in section (2.), especially for the control space.

3.1. Notations Let Ω be an open subset of R^m ($\Omega \subset R^m$), this is the physical space of the system, we are interested in control spaces defined as $\mathcal{V} = (L^2(\Omega))^n$. control states are then defined as $\mathbf{v} \in \mathcal{V} = (L^2(\Omega))^n$, $\mathbf{v}(\mathbf{x}) = (v_i(\mathbf{x}))_{1\leq i\leq n}$ and $\mathbf{x} = (x_i)_{1\leq j\leq m} \in \Omega$

3.2. First order methods The first order regularization methods define J_r as a function of the first order derivatives of $\mathbf{v}$:

$$J_r(\mathbf{v}) = J_r\left(\frac{\partial v_i}{\partial x_j}\right)_{1\leq i,j\leq n,} \tag{12}$$

The most used of first order regularization methods is the gradient penalization. It has been used by Horn and Schunck in the formulation of optical flow [3] for motion estimation. The regularization function of Horn and Schunck is defined as follow:

$$J_{grad}(\mathbf{v}) = \frac{1}{2}\int_\Omega \sum_{i=1}^{n} \|\nabla v_i\|^2 d\mathbf{x} \tag{13}$$

For incompressible fluid or irrotational flow, it is common to penalize the divergence or the curl of the vector field leading to divergence penalization

$$J_{div}(\mathbf{v}) = \frac{1}{2}\int_\Omega \|div(\mathbf{v})\|^2 d\mathbf{x} \tag{14}$$

for incompressible fluid flow and curl penalization

$$J_{curl}(\mathbf{v}) = \frac{1}{2}\int_\Omega \|curl(\mathbf{v})\|^2 d\mathbf{x} \tag{15}$$

for irrotational flow

3.3. Second order methods The second order regularization methods are based on the second order derivatives of $\mathbf{v}$.

$$J_r(\mathbf{v}) = J_r\left(\frac{\partial^2 v_i}{\partial x_j \partial x_k}\right)_{1\leq i,j,k\leq n,} \tag{16}$$

An example based on the first order derivatives of *div* and *curl* is the regularization of Suter [8] defined as followed :

$$J_{suter}(\mathbf{v}) = \frac{1}{2}\int_\Omega \alpha\|\nabla div(\mathbf{v})\|^2 + \beta\|\nabla curl(\mathbf{v})\|^2 d\mathbf{x} \tag{17}$$

Higher order derivatives of v can also be used for regularization; for example (17) has been generalized by Chen and Suter [7] using m-order derivatives of *div* and *curl*.

$$J_m(\mathbf{v}) = \frac{1}{2}\int_\Omega \alpha\|\nabla^m div(\mathbf{v})\|^2 + \beta\|\nabla^m curl(\mathbf{v})\|^2 d\mathbf{x} \tag{18}$$

4. Turning regularization functions into covariance operators

4.1. Regularization operator out of an optimization process : case of gradient penalization

Definition. Let :

- $\mathbf{v}(\mathbf{x})$ be an incomplete/inconsistent state of the studied system with $\mathbf{x} \in \mathbf{\Omega}$ the space on which the system is defined
- $\varphi(\mathbf{x})$ a scalar positive trust function given the quality of the state $\mathbf{v}$ at $\mathbf{x}$

$$\begin{cases} \text{small value meaning bad/lack/inconsistent state} \\ \text{big value for good quality state} \end{cases}$$

we define a restoration of v as the minimum argument of the function

$$\varepsilon(\mathbf{u}) = \frac{1}{2}\int_{\Omega}\sum_{i=1}^{n}\|\nabla u_i(x)\|^2 + \varphi(\mathbf{x})\|\mathbf{u}(\mathbf{x}) - \mathbf{v}(\mathbf{x})\|^2 d\mathbf{x} \tag{19}$$

The minimization of ε is achieved by setting $\mathbf{u}$ to be closed to $\mathbf{v}$ when φ is large ($\mathbf{v}$ is of good quality) and smooth (small gradient norm) when φ is small ($\mathbf{v}$ is not of good quality)

Practical use. Under the conditions given in section (), MinArg(ε) equation (19) is defined by the Euler-Lagrange condition

$$\nabla_{\mathbf{u}}\varepsilon(\mathbf{u}) = 0 \tag{20}$$

The difficulty with nonlinear problems is to express $\nabla\varepsilon$. When $\nabla\varepsilon$ is expressed, it can be used in descent type algorithms to solve the minimization problem. $\nabla\varepsilon$ can be obtained by making explicit the linear dependency of the Gateaux derivatives $\hat{\varepsilon}$ with respect to the gradient. Development based on vector calculus leads to

$$\nabla\varepsilon^{grad}(\mathbf{u}) = -\Delta\mathbf{u}(\mathbf{x}) + \varphi(\mathbf{x})(\mathbf{u}(\mathbf{x}) - \mathbf{v}(\mathbf{x})), 1 < i < n \tag{21}$$

Instead of using classical descent type algorithm to get the solution of the problem, u_i can be considered as a function of time and the solution obtained by solving (22) according to development in [6],

$$\frac{\partial}{\partial t}u_i(\mathbf{x},t) \quad = \quad \Delta u_i(\mathbf{x},t) - \varphi(\mathbf{x})(u_i(\mathbf{x},t) - v_i(\mathbf{x}))), \quad 1 \le i \le n \tag{22}$$

the set of equations (22) are known as the generalized diffusion equations. The diffusion operator $\mathcal{L}$ giving the solution $\mathbf{u}^* = \mathcal{L}(\mathbf{v}) = \text{minArg}(\varepsilon)$ can then used as a covariance operator to define the appropriate norm for the cost function of the inverse problem. In fact, an appropriate choice of the trust function makes the discrete operator symmetric and positive definite. This is a good candidate for the definition of an appropriate norm form the regularization terms of the cost function. We called the such a method generalized diffusion regularization.

Generalization. If the regularization term is defined as $J_r(\mathbf{v}) = \|\mathbf{\Phi}(\mathbf{v})\|$, the function ε can be generalized as

$$\varepsilon(\mathbf{u}) = \frac{1}{2}\int_{\Omega}\|\Phi(\mathbf{u}(\mathbf{x}))\|^2 + \varphi(\mathbf{x})\|\mathbf{u}(\mathbf{x}) - \mathbf{v}(\mathbf{x})\|^2 d\mathbf{x} \tag{23}$$

The minimum of ε is achieved by setting $\mathbf{u}$ to be closed to $\mathbf{v}$ when φ is large ($\mathbf{v}$ is of good quality) and Φ-smooth when φ is small ($\mathbf{v}$ is not of good quality)

4.2. Application to motion estimation

Trust function for motion estimation. From the previous paragraph, one have an idea on the definition of appropriate trust function φ. A basic idea is to set large values on discontinuities (contours) for motion component along the normal to the contour, and small values in homogeneous areas.
Following [6], let defines contour or edge map $c(\mathbf{x})$ of luminance function $f(\mathbf{x})$ as followed:

$$c^1(\mathbf{x}, f) \quad = \quad \|\nabla_{\mathbf{x}} f(\mathbf{x})\|^2 \tag{24}$$

$$c^2(\mathbf{x}, f) \quad = \quad \|\nabla_{\mathbf{x}}(\mathbf{G}_\sigma(\mathbf{x}) * f(\mathbf{x}))\|^2 \tag{25}$$

where $\mathbf{G}_\sigma(\mathbf{x})$ is a Gaussian function with standard deviation σ and $*$ the convolution operator. The gaussian convolution is used for denoising purpose and can be replaced by any other denoising operator. A simple choice of the trust function is the edge map : $\varphi(\mathbf{x}) = c^i(\mathbf{x}, f^0)_{i=1,2}$. When this trust function is used to find MinArg(ε) (with ε defined by equation ...), it has the effect of keeping the solution $\mathbf{u}^*$ closed to $\mathbf{v}$ on edges, but forces it to be $\Phi-$smooth in homogeneous areas. This is the desired effect in motion estimation. For experimental results in this paper, we used the trust function defined by c^1.

Experimental results. We performed a set of twin experiments in order to analyze the behavior of the approach we introduced here. We use images from experimental study of the drift of a vortex on a turntable. The evolution of a vortex in the atmosphere is simulated at the CORIOLIS experimental turntable (Grenoble, France). The vortex is created by stirring the water and made visible by addition of a passive tracer (fluorescein). Photographs of the vortex are taken from above the turntable, we will call the images sequence from this experiment coriolis-sequence. For more details on the statements and the original motivations of this experiment, see [4]. The original images resolution were 1024×1024 pixels for a spatial domain of $2.525m \times 2.525m$. This is 4 $pixels/cm$. The resolution has been reduced to $0.5 pixels/cm$ for 128×128 pixels images on the same spatial domain.

The zonal component $v_1 = v_1(x, y)$ and meridional component $v_2 = v_2(x, y)$ of the current velocity are computed by direct image sequence assimilation (DISA) [5], (x, y) are the physical two dimensional space coordinates. This velocity field is used as true control state ($\mathbf{v^t}$) for error analysis in our experiments. From the first image of coriolis-sequence (f^0), we create an observed image (f^1, the image at the experimental time t) using the conservation equation (26) and the velocity field computed by DISA. We defined the ME-sequence (Motion Estimation sequence) as the sequence $\{f^0, f^1\}$

Figure (1) shows 2 images from coriolis-experiment and the velocity field by direct image sequence assimilation.

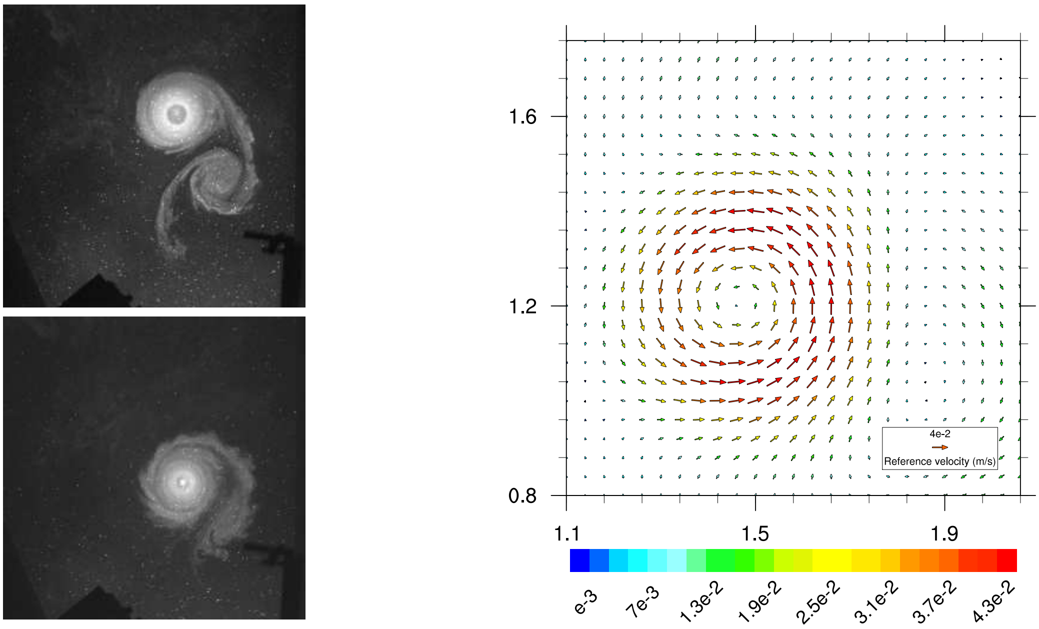

Figure 1: True current velocity field for twin experiments : on the left, 2 images of the coriolis-sequence; on the right, current velocity field computed by direct image assimilation.

$$\partial_t f + v_1 \partial_x f + v_2 \partial_y f = 0, \tag{26}$$

where $f = f(t, x, y)$ is the passive tracer concentration. Given the tracer concentration $f(t = 0, x, y) = f^0(x, y) = f^0$ at time 0 and the velocity field $\mathbf{v}(x, y)$ the tracer concentration $f(t) = f(t, x, y)$ at time t can be obtained by integration of the conservation equation (26.) This integration defines the motion estimation model $\mathcal{M}$ as

$$\begin{array}{rccl} \mathcal{M}: & \mathcal{V} & \rightarrow & \mathcal{F} \\ & \mathbf{v} & \mapsto & f(t) = \mathcal{M}(\mathbf{v}) \end{array} \tag{27}$$

for the experimental time t. $\mathcal{V}$, (resp. $\mathbf{v} = (v1, v2)$) is the control space (resp. control variable) as defined in section (3.) The physical domain Ω is the image domain, this is the subset of $\mathbb{R}^2$ defining the image area. The state space $\mathcal{F} \subset C^1(\mathcal{L}^2(\Omega))$. The observation space is the same as the state space, we do not need an observation operator. from the ME-sequence, the observation and the background terms of the cost function are defined as

$$J_o(\delta\mathbf{v}) = \frac{1}{2}\|\mathcal{M}(\mathbf{v^b} + \delta\mathbf{v}) - f^1\|^2_{\mathcal{F}} \tag{28}$$

$$J_b(\delta\mathbf{v}) = \frac{1}{2}\|\delta\mathbf{v}\|^{\mathbf{2}}_{\mathcal{V}} \tag{29}$$

$\|.\|_{\mathcal{F}}$ and $\|.\|_{\mathcal{V}}$ are simple $\mathcal{L}^2 - norm$

The background is set to zero, $\mathbf{v^b} = \mathbf{0}$. For the minimization of the cost function, we use the M1QN3 algorithm of the LIBOPT library [9]. The result of the minimization is the analyzed state $\mathbf{v^a}$. For error analysis, we define velocity error (error on the analyzed control state, equation 30) and vorticity error (this is a diagnostic for the analyzed sate, equation 31.)

$$e^{\mathbf{v}}(x,y) = \|\mathbf{v^a}(x,y) - \mathbf{v^t}(x,y)\| \tag{30}$$

$$e^{\mathbf{vr}}(x,y) = \|\nabla_x \times \mathbf{v^a}(x,y) - \nabla_x \times \mathbf{v^t}(x,y)\| \tag{31}$$

The mean error $\bar{e}$ is defined as the mean of the error over the physical domain

$$\bar{e} = \frac{\displaystyle\int_\Omega e(\mathbf{x})d\mathbf{x}}{\displaystyle\int_\Omega 1d\mathbf{x}} \tag{32}$$

and the normalized mean error as

$$\bar{\bar{e}} = \frac{\bar{e}}{\bar{e}_b} \tag{33}$$

where $\bar{e}_b$ is the mean error on the background. We also defined the normalized cost function as

$$\bar{\bar{J}} = \frac{\bar{J}}{\bar{J}_b} \tag{34}$$

where J_b is the cost function value associated to the background.

We use second order analysis to define optimal parameter for gradient, first order div-curl and second order div-curl (Suter) regulization. These optimal parameters are then used to make a set of experiments. Figure () shows the evolution of the normalized root mean square error (RMSE, log coordinates) on velocity and vorticity with the minimization iterations. The graphic clearly shows that the approach introduced here is better than the others and their combination. With this new approach, velocity error decreases from 100% to 10% after 40 iterations while classical regularization need more than 200 iteration to get the same result. Furthermore, with the new approach, velocity error can be reduced to less than 1% while for the other methods, the best result is affected by about 10% of error. Vorticity error is reduce to 4% with the new approach and only to 40% with classical regularization methods.

5. Conclusion

We introduced here a new formalism for taking into account a priori knowledges on the regularity of the solution while solving inverse problems. This new formalism is based on the generalized diffusion equations. Preliminary results shows the superiority of this formalism over classical methods that include regularity informations as penalization in the cost function.

References

[1] A. N. Tikhonov, Regularization of incorrectly posed problems, *Soviet Math*, 4, 1963.

[2] R. Glowinski and P. Le Tallec, Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics, *Journal*, (999), 1987.

[3] Berthold K.P. Horn and Brian G. Schunck, Determining Optical Flow, *Artificial Intelligence*, (999), 1981.

[4] J.-B. Flór, I. Eames, Dynamcis of monopolar vortices on a topographic beta-plane, *J. Fluid Mech*, 456, 2002.

[5] O. Titaud, A. Vidard, I. Souopgui and F.-X. Le Dimet, Assimilation of image sequences in numerical models, *Tellus A*, 62, 2009.

[6] C. Xu and J. L. Prince, Snakes, Shapes, and Gradient Vector Flow, *TIP*, 7, (2), 1998.

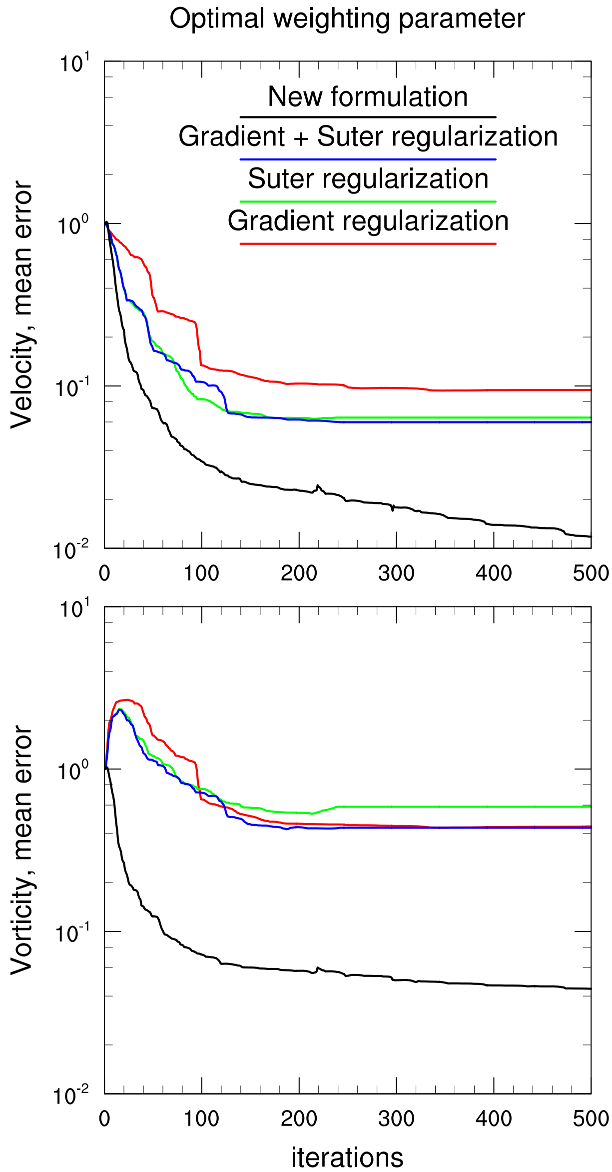

Figure 2: Comparison of normalized RMSE on velocity and vorticity, classical regularization (optimal parameters) and new formulation; evolution with the minimization's iterations

[7] F. Chen and D. Suter, Image Coordinate Transformation Based on DIV-CURL Vector Splines, *Proceedings of the 14th International Conference on Pattern Recognition*, 1, 1998.

[8] D. Suter, Motion Estimation and Vector Splines, *CVPR94*, (1), 1994.

[9] Jean Charles Gilbert and Xavier Jonsson, LIBOPT - An environment for testing solvers on heterogeneous collections of problems - Version 1.0, *INRIA Technical report*, RT-0331, 2007, http://hal.inria.fr/inria-00135013/fr/.