Journal of Global Optimization **21:** 289–316, 2001.

A Class of Global Optimization Problems as Models of the Phase Unwrapping Problem

PIERLUIGI MAPONI[1] and FRANCESCO ZIRILLI[2]
[1]*Dipartimento di Matematica e Fisica, Università di Camerino, 62032 Camerino, Italy;*
[2]*Dipartimento di Matematica 'G. Castelnuovo', Università di Roma 'La Sapienza', 00185 Roma, Italy*

Abstract. Let $I = \{(i, j) \mid i = 1, 2, \ldots, N_1,\ j = 1, 2, \ldots, N_2\}$ and let $U = U_{i,j},\ (i, j) \in I$ be a discrete real function defined on I. Let $[\cdot]_{2\pi}$ be $\cdot$ modulus 2π, we define $W : I \to [-\pi, \pi)$ as follows $W = [U]_{2\pi}$. The function U will be called phase function and the function W will be called wrapped phase function. The phase unwrapping problem consists in recovering U from some knowledge of W. This problem is not well defined, that is infinitely many functions U correspond to the same function W, and must be 'regularized' to be satisfactorily solvable. We propose several formulations of the phase unwrapping problem as an integer nonlinear minimum cost flow problem on a network. Numerical algorithms to solve the minimum cost flow problems obtained are proposed. The phase unwrapping problem is the key problem in interferometry, we restrict our attention to the SAR (Synthetic Aperture Radar) interferometry problem. We compare the different formulations of the phase unwrapping problem proposed starting from the analysis of the numerical experience obtained with the numerical algorithms proposed on synthetic and real SAR interferometry data. The real data are taken from the ERS missions of the European Space Agency (ESA).

Key words: SAR interferometry problem, phase unwrapping problem, nonlinear minimum cost flow, integer programming

1. Introduction

The phase unwrapping problem occurs in several application fields being a key problem in interferometry, (for a review see Oppenheim and Lim, 1981 and Zebker and Goldstein, 1986). Recently phase unwrapping has been used in remote sensing to recover the digital elevation map of the Earth surface from SAR (Synthetic Aperture Radar) interferograms obtained from data measured with SAR systems travelling on board of satellites or airplanes, (see, for example Zebker and Goldstein, 1986 and Goldstein et al., 1988).

Let us begin introducing some notation necessary to define the phase unwrapping problem. Let $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{R}$ be the sets of natural, integer and real numbers respectively. Let $N \in \mathbb{N}$, we denote with $\mathbb{Z}^N$ the set of N-tuples on integers, with $\mathbb{R}^N$ the N-dimensional real euclidean space. Let $\underline{x} = (x_1, x_2, \ldots, x_N)^t \in \mathbb{R}^N$ be a generic vector, where the superscript t denotes the transposition operation, for $\underline{x}$,

$\underline{y} \in I\!R^N$ we denote with $\underline{x}^t \underline{y}$ the euclidean scalar product of $\underline{x}$ and $\underline{y}$. Let $p \geqslant 1$, the p-norm in $I\!R^N$ is defined as follows:

$$\left\| \underline{x} \right\|_p = \Big(\sum_{n=1}^{N} |x_n|^p \Big)^{1/p},$$

when $p = \infty$ the ∞-norm is defined as follows: $\left\| \underline{x} \right\|_\infty = \max\{|x_1|, |x_2|, \ldots, |x_N|\}$. Note that $p = 2$ gives the euclidean vector norm induced by the euclidean scalar product.

The problem addressed in this paper consists in the computation of a real valued function (the unwrapped phase function) defined on a discrete set (i.e. rectangular grid) from the knowledge of its values modulus 2π (the wrapped phase function). What follows is the mathematical formulation of this problem. Let $N_1, N_2 \in I\!N$, we define $I = \{(i, j) \mid i = 1, 2, \ldots, N_1,\ j = 1, 2, \ldots, N_2\}$. We consider the functions $U : I \to I\!R$ that we call *(unwrapped) phase functions*, the $[\cdot]_{2\pi}$ operation that consists in taking the modulus 2π of $\cdot$, note that we choose $[\cdot]_{2\pi} \in [-\pi, \pi)$, so that for example we have $\left[\frac{3}{2}\pi\right]_{2\pi} = -\frac{\pi}{2}$, moreover we consider the functions $W = [U]_{2\pi}$ that we call *wrapped phase functions*. With $[-\pi, \pi)$ we denote the bounded interval $-\pi, \pi$ closed to the right end point and open to the left end point. Let $U = U_{i,j}$, $(i, j) \in I$, the function $W = [U]_{2\pi} : I \to [-\pi, \pi)$ is defined as follows: for $(i, j) \in I$, $W = W_{i,j} = [U]_{2\pi\ i,j} = [U_{i,j}]_{2\pi}$. The *phase unwrapping problem* in its simplest form can be stated as follows: given the function W or more in general some information about W find U such that $W = [U]_{2\pi}$.

The phase unwrapping problem as stated is not well defined. In fact let $\mathcal{N} = \{u : I \to I\!R |\ u = u_{i,j} = 2\pi k_{i,j}, k_{i,j} \in \mathbb{Z}\ , i = 1, 2, \ldots, N_1,\ j = 1, 2, \ldots, N_2\}$ it is easy to see that when $u \in \mathcal{N}$ we have $[u]_{2\pi} = 0$. That is if $[U]_{2\pi} = W$ then $u + U$ with $u \in \mathcal{N}$ satifies $[u + U]_{2\pi} = [U]_{2\pi} = W$, so that there are infinitely many different functions U such that $[U]_{2\pi} = W$. Note that $W = [W]_{2\pi}$. This means that the previous phase unwrapping problem is stated too loosely. In fact between the functions:

$$U = W + u\ , \quad u \in \mathcal{N}\ , \tag{1}$$

that are solutions of the phase unwrapping problem stated above we must introduce a merit function to be able to choose the ‘best solution’.

In this paper we reformulate the phase unwrapping problem as a global optimization problem. Of course this can be done in many different ways. We choose to model the phase unwrapping problem as an integer minimum cost flow problem on a network. This choice generalizes the choice made in (Costantini, 1998). In (Costantini, 1998) the objective function was chosen to be the 1-norm of the unknown vector, in this paper we consider as objective function the p-norm of the unknown vector with $p > 1$ or $p = \infty$. The case of the 1-norm is a very special one since in that case the fact that we work with integer variables, that is $k_{i,j} \in \mathbb{Z}$, $(i, j) \in I$, can be overcome easily using the fact that integer linear programming on a network can be reformulated as a problem with real variables (Bertsekas, 1991, p. 10, Cook et al., 1998, p. 94), and solved using the simplex method. When the p-norm with $p > 1$ is considered the problem becomes an integer nonlinear programming problem where the integer character of the variables cannot be avoided with simple transformations. Moreover if we have in mind applications to SAR interferometry in remote sensing we must remember that problems involving 10 000-100 000 independent variables are the rule. We propose two numerical algorithms to solve the integer programming problems that model the phase unwrapping problem. The first algorithm is based on an approximation of the p-norm of a vector, when $p > 1$ or $p = \infty$, with an expression that is linear with respect to the components of the vector. The second algorithm solves the integer programming problem for $p = \infty$ with a sequence of integer linear programming problems on a network of the same nature of the $p = 1$ problems, these problems can be solved easily.

Finally we explain briefly the basic facts about a SAR interferometry experiment and we test the mathematical models and the numerical algorithms proposed to solve them on synthetic and real SAR interferometry data. The real data are obtained from the ERS missions of the European Space Agency. The quality of the reconstructed phase function depends on the mathematical model used and on the character of the data, i.e., in the SAR interferometry experiment the content of the scene observed. We propose a heuristic analysis to choose the most appropriate mathematical model given the character of the data.

In Section 2 we propose the mathematical models of the phase unwrapping problem considered and the numerical algorithms to solve them. In Section 3 a SAR interferometry experiment is explained and the models and algorithms of Section 2 are tested on synthetic and real SAR interferometry data. Finally we comment on the problem of choosing the appropriate model to process a given set of data.

2. The mathematical formulation of the phase unwrapping problem and its numerical solution

From the knowledge of $W_{i,j} \in [-\pi, \pi)$, $(i, j) \in I$, we have to reconstruct $U_{i,j}$, $(i, j) \in I$, such that $W_{i,j} = [U_{i,j}]_{2\pi}$, $(i, j) \in I$.

To solve this problem we begin with the following observation. Let $I_1 = \{(i, j), i = 1, 2, \ldots, N_1 - 1, j = 1, 2, \ldots, N_2\}$, $I_2 = \{(i, j), i = 1, 2, \ldots, N_1, j = 1, 2, \ldots, N_2 - 1\}$ and let $F : I \to I\!R$ be a function, we define the operators Δ_1, Δ_2 acting on the function F as follows:

$$F_1 = F_{1,i,j} = \Delta_1 F_{i,j} = F_{i+1,j} - F_{i,j}\ , \quad (i, j) \in I_1, \tag{2}$$

$$F_2 = F_{2,i,j} = \Delta_2 F_{i,j} = F_{i,j+1} - F_{i,j}\ , \quad (i, j) \in I_2. \tag{3}$$

In the following we call the operators Δ_1, Δ_2 *discrete partial derivatives* of the function F. The vector field $(F_1, F_2)^t = (\Delta_1 F, \Delta_2 F)^t$ defined on $I_1 \cap I_2$ will be called *discrete gradient vector field* of the function F.

It is easy to see that a vector field $(F_1, F_2)^t$ defined on $I_1 \cap I_2$ is the gradient of a function F defined on D if and only if:

$$\Delta_2 F_{1,i,j} = \Delta_1 F_{2,i,j}\ , \quad (i, j) \in I_1 \cap I_2. \tag{4}$$

In the following condition (4) will be called *irrotational property*, moreover when condition (4) is satisfied we have:

$$F_{i,j} = F_{1,1} + \sum_{l=1}^{i-1} F_{1,l,1} + \sum_{l=1}^{j-1} F_{2,i,l}\ , \quad (i, j) \in I. \tag{5}$$

In (5) when i or j are equal to 1, that is when the upper index of the sums $i-1$ or $j-1$ is equal to zero, we define the corresponding sum to be equal to zero. Note that formulae (4), (5) are discrete translations of well known results of elementary calculus in the study of integrable first order differential forms. We have:

$$[\Delta_1 U]_{2\pi\ i,j} = \Big[[U]_{2\pi\ i+1,j} - [U]_{2\pi\ i,j}\Big]_{2\pi}\ , \quad (i, j) \in I_1, \tag{6}$$

and when $\Delta_1 U_{i,j} \in [-\pi, \pi)$, $(i, j) \in I_1$ we have:

$$[\Delta_1 U]_{2\pi\ i,j} = \Delta_1 U_{i,j}\ , \quad (i, j) \in I_1. \tag{7}$$

Formulae similar to (6), (7) hold for $\Delta_2 U$.

We note that when $\Delta_1 U$ and $\Delta_2 U$ take values in $[-\pi, \pi)$ the phase unwrapping problem can be solved as follows: since $[U]_{2\pi} = W$ from the knowledge of W and from (6) we obtain $[\Delta_1 U]_{2\pi}$ so that from (7) we obtain $\Delta_1 U$. Similarly we can derive $\Delta_2 U$. Finally from the knowledge of the vector field $(\Delta_1 U, \Delta_2 U)^t$ using formula (5) we obtain U up to an undetermined additive constant, i.e. $U_{1,1}$. For slowly varying U, that is when $\Delta_\nu U \in [-\pi, \pi)$, $\nu = 1, 2$, the previous observation is a satisfactory solution of the phase unwrapping problem.

In the general case we define:

$$G_{\nu,i,j} = [\Delta_\nu W]_{2\pi\ i,j} + 2\pi k_{\nu,i,j}\ , \quad (i, j) \in I_\nu\ , \quad \nu = 1, 2, \tag{8}$$

where $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$, are $N_1(N_2 - 1) + N_2(N_1 - 1)$ integer variables to be determined. Note that we can define $N_1(N_2 - 1) + N_2(N_1 - 1)$ integers $b_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ such that:

$$[\Delta_\nu W]_{2\pi\ i,j} = \Delta_\nu W_{i,j} + 2\pi b_{\nu,i,j}\ , \quad (i, j) \in I_\nu\ , \quad \nu = 1, 2. \tag{9}$$

Moreover it is easy to see that the choice of $b_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ made in (9) is unique when we require that $b_{\nu,i,j} \in \{-1, 0, 1\}$, $(i, j) \in I_\nu$, $\nu = 1, 2$. Note that the requirement $b_{\nu,i,j} \in \{-1, 0, 1\}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ is implied by the fact that $W_{i,j} \in [-\pi, \pi)$, $(i, j) \in I$. Equation (4) must hold in order for $(G_1, G_2)^t$ given by (8) to be the gradient vector field of a function U, that is:

$$\Delta_2 G_{1,i,j} = \Delta_1 G_{2,i,j} \ , \quad (i, j) \in I_1 \cap I_2. \tag{10}$$

We note that the discrete partial derivatives introduced above are always used with fixed 'stepsize'. There is no reason in considering the behaviour of the discrete partial derivatives as a function of the 'stepsize' since in the applications we have in mind the 'stepsize' is fixed. Condition (10) translates into the following constraints on the integer variables $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$:

$$\begin{aligned} k_{1,i,j+1} - k_{1,i,j} - k_{2,i+1,j} + k_{2,i,j} &= \\ -(b_{1,i,j+1} - b_{1,i,j} - b_{2,i+1,j} + b_{2,i,j}&), \\ (i, j) \in I_1 \cap I_2&. \end{aligned} \tag{11}$$

The equations (11) are $(N_1 - 1)(N_2 - 1)$ linear constraints for the $N_1(N_2 - 1) + N_2(N_1 - 1)$ integer variables $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$. Moreover it is easy to see that these constraints are always feasible. In fact $k_{\nu,i,j} = -b_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ is always a feasible point of the constraints (11).

For any choice of the variables $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ that satisfies the constraints (11) the vector field $(G_1, G_2)^t$ given by (8) is the gradient vector field of a function U. This function, the unwrapped phase, can be recovered up to an additive constant using formula (5). Infinitely many different reconstructions are possible one for each feasible point of the constraints (11). Moreover we know that when $\Delta_1 U, \Delta_2 U \in [-\pi, \pi)$ the choice of the feasible point $k_{\nu,i,j} = 0$, $(i, j) \in I_\nu$, $\nu = 1, 2$ gives the correct reconstruction. We must keep this in mind when we introduce a merit function to distinguish the 'best' reconstruction between all the possible ones.

Let $N = N_1(N_2 - 1) + N_2(N_1 - 1)$, $\underline{k} \in \mathbb{Z}^N$, be the vector whose components are given by $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$, then we require the $\left\|\underline{k}\right\|_p$ to be as small as possible. Let $A\underline{k} = \underline{b}$ be the system of linear equations (11) written in matrix–vector notation. The matrix A is the matrix of the coefficients of the unknowns, it is easy to see that A has $(N_1 - 1)(N_2 - 1)$ rows and N columns; $\underline{b} \in \mathbb{Z}^{(N_1-1)(N_2-1)}$ is the vector containing the terms on the right hand side of equations (11). Then we consider the following optimization problem:

$$\begin{aligned} &\min f_p(\underline{k}) \\ &\text{s.t. : } A\underline{k} = \underline{b}, \\ &\qquad \underline{k} \in \mathbb{Z}^N, \end{aligned} \tag{12}$$

where for $\underline{k} \in \mathbb{Z}^N$ we define:

$$f_p(\underline{k}) = \begin{cases} \left\|\underline{k}\right\|_p^p, & 1 \leqslant p < \infty, \\ \left\|\underline{k}\right\|_\infty, & p = \infty. \end{cases} \tag{13}$$

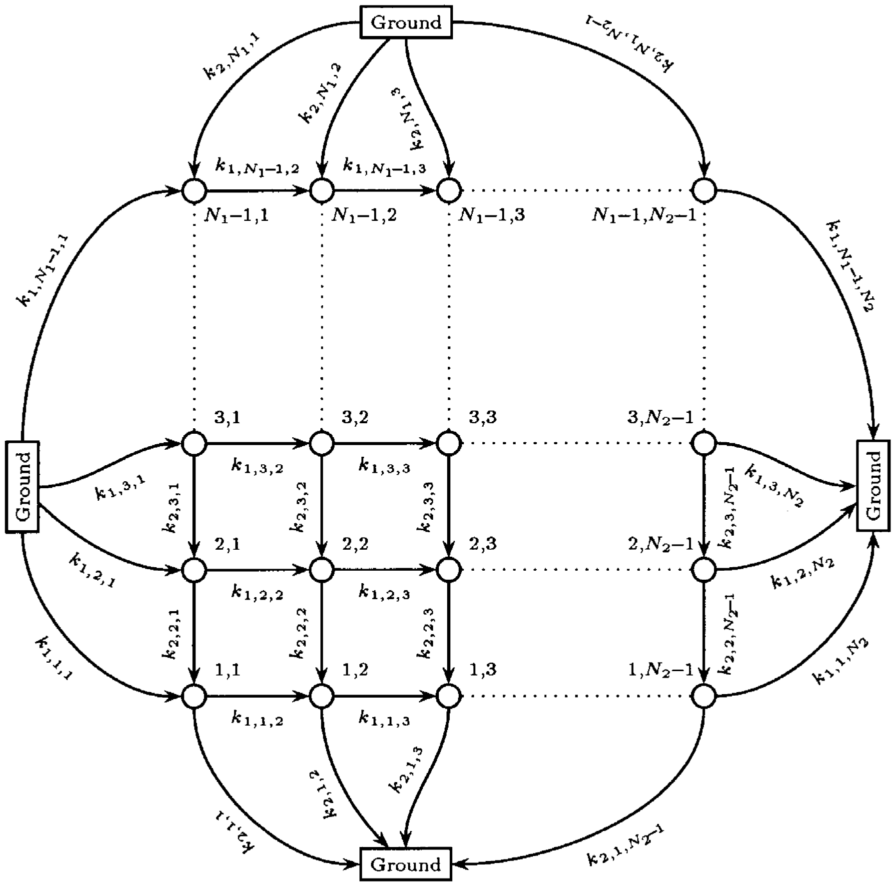

Figure 1. The graph $\mathcal{G}$ associated to problem (12): $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ gives the flow on each arc of $\mathcal{G}$, conditions (11) give the flow balance on the nodes of $\mathcal{G}$, 'ground nodes' are auxiliary nodes.

We consider the class of problems (12), for $p \geqslant 1$ or $p = \infty$ as mathematical models of the phase unwrapping problem. We note that for increasing values of p we expect a decreasing value of the infinity norm of the minimizer of problem (12). The formulation chosen in (12) has the advantage that the solution of problem (12) can be avoided when G_ν, $\nu = 1, 2$ as defined in (8) verify the irrotational property (10) that is when $k_{\nu,i,j} = 0$, $(i, j) \in I_\nu$, $\nu = 1, 2$ satisfy (11). This occurs for example when $\underline{b} = (0, 0, \ldots, 0)^t \in \mathbb{Z}^{(N_1-1)(N_2-1)}$. Many other mathematical formulations have been proposed for the phase unwrapping problem, (see, for example Goldstein et al., 1988; Costantini, 1998; Costantini et al., 1999; Fried, 1977).

Note that the minimizer of problem (12) may be not unique, so that further conditions depending on a priori information about the data to be treated may be useful. We note that the matrix A is a submatrix of the node-arc incidence matrix associated to the graph $\mathcal{G}$ shown in Figure 1. In particular the constraints $A\underline{k} = \underline{b}$ of problem (12) can be seen as flow conservation conditions on the nodes of the graph $\mathcal{G}$ that are labeled by $(i, j) \in I_1 \cap I_2$, where the term $\underline{b}$ can be seen as the exogenous supplies at the nodes of the graph $\mathcal{G}$ that are labeled by $(i, j) \in I_1 \cap I_2$. The objective function of problem (12) can be interpreted as the function that gives the cost of the flow on graph $\mathcal{G}$.

We note that (12) is not a network problem because the constraints $A\underline{k} = \underline{b}$ do not give flow conservation conditions on 'ground nodes'. However we can consider a network problem equivalent to problem (12), i.e., having the same solution of problem (12). This can be done introducing an artificial node in graph $\mathcal{G}$ and connecting this node to each ground node. Moreover we require that each ground node has exogenous supply equal to zero and that the artificial node has exogenous supply equal to $-\sigma$, where σ is the sum of the components of vector $\underline{b}$. It is easy to see that such a network problem and problem (12) are equivalent, in virtue of this equivalence we can consider also problem (12) as a network problem. Note that the particular requirement on the exogenous supply of the artificial node assures that the sum of the exogenous supplies of all the nodes in the network are equal to zero. This is a necessary condition for the solvability of a generic minimum cost flow problem on a network (for details, see Bertsekas, 1991, p. 11).

We note that problem (12) for $p = 1$ can be rewritten as an integer linear minimum cost flow problem on a network, so that the integer constraints on the variables cause no further difficulty, (see, for example, Bertsekas, 1991 p. 10; Cook et al., 1998, p. 94). For $p \in I\!R$, with $p > 1$, or $p = \infty$ the objective function of problem (12) is convex, so that problem (12) without the integer constraints on the variables can be solved easily; however, the integer constraints on the variables make the problem difficult to solve because the problem becomes a integer convex nonlinear optimization problem. The following example shows that the convexity of this problem is not sufficient to guarantee a satisfactory solution of it via a relaxion of the integer constraints.

EXAMPLE 1. Let us consider the following problem:

$$\begin{aligned}
&\min\{x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2 + x_6^2\},\\
&\text{s.t.}: x_1 + x_2 = 5,\\
&\qquad -x_1 + x_3 + x_4 = 0,\\
&\qquad -x_2 - x_3 + x_5 = 0,\\
&\qquad -x_4 - x_6 = -5,\\
&\qquad -x_5 + x_6 = 0.
\end{aligned} \tag{14}$$

We note that problem (14) is a nonlinear minimum cost flow problem and in Figure 2 is shown the related graph. If we consider $\underline{x} = (x_1, x_2, x_3, x_4, x_5, x_6)^t \in I\!R^6$ the minimizer of problem (14) is $\underline{x}^* = (\frac{30}{11}, \frac{25}{11}, -\frac{5}{11}, \frac{35}{11}, \frac{20}{11}, \frac{20}{11})^t$, where the objective function is equal to $\frac{325}{11}$. Moreover when we consider problem (14) with the additional constraint $\underline{x} \in \mathbb{Z}^6$, then the minimizer is $\underline{y}^* = (3, 2, 0, 3, 2, 2)^t$ and

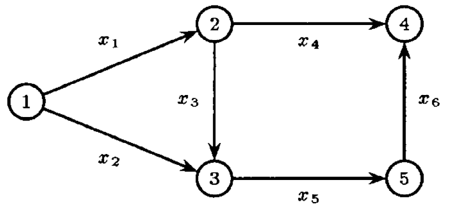

Figure 2. The graph 'associated' to problem (14): the variables of problem (14) give the flow on the arcs of this graph, the constraints appearing in problem (14) give the flow balance on the nodes of this graph, note that node 1 is a source and node 4 is a sink.

$\left\| \underline{y}^* \right\|_2^2 = 30$. Let C be the set of vectors of $\mathbb{Z}^6$ whose components satisfy the constraints of problem (14). It is easy to see that $\underline{y} = (5, 0, 1, 4, 1, 1)^t \in C$ is the closest point to $\underline{x}^*$ belonging to C, but we have $\left\| \underline{y} \right\|_2^2 = 44$.

We propose two algorithms to approximate the solution of problem (12). These algorithms are based on the fact that the solution of problem (12) is easily computable when the objective function is a linear function of the unknown vector $\underline{k}$ or can be reduced to a linear function as in the case $p = 1$. In the following we assume to have an algorithm to solve the integer linear minimum cost flow problem on a network or equivalently to solve problem (12) for $p = 1$ (see, for example, Bertsekas, 1991, p. 279; Moré and Wright, 1993, p. 53). We note that many algorithms to solve problems similar to problem (12) are proposed in the scientific literature, (see, for example Fletcher and Leyffer, 1994; Toint and Tuyttens, 1992).

METHOD 1

The method is based on a classical approximation of a convex function by piecewise linear functions (see, for example, Dantzing, 1963, p. 484). Let us fix $\delta > 0$, $\overline{n} \in \mathbb{N}$ and let $p \in \mathbb{R}$, with $p > 1$. For $(i, j) \in I_\nu$, $\nu = 1, 2$ we define the following linear function of the variables $h^{\pm}_{\nu,i,j,n}$, $n = 1, 2, \ldots, \overline{n}$:

$$\begin{aligned} &K_{\nu,i,j}(h^+_{\nu,i,j,1}, h^-_{\nu,i,j,1}, h^+_{\nu,i,j,2}, h^-_{\nu,i,j,2}, \ldots, h^+_{\nu,i,j,\overline{n}}, h^-_{\nu,i,j,\overline{n}}) = \\ &\quad \sum_{n=1}^{\overline{n}} \Big(n^p - (n-1)^p\Big)\delta^{p-1}\Big(h^+_{\nu,i,j,n} + h^-_{\nu,i,j,n}\Big), \\ &\quad h^{\pm}_{\nu,i,j,n} \in [0, \delta], \; n = 1, 2, \ldots, \overline{n}, \end{aligned} \tag{15}$$

where $[0, \delta]$ denotes the bounded closed interval of end points 0 and δ. For every $(i, j) \in I_\nu$, $\nu = 1, 2$ the function $K_{\nu,i,j}(h^+_{\nu,i,j,1}, h^-_{\nu,i,j,1}, h^+_{\nu,i,j,2}, h^-_{\nu,i,j,2}, \ldots, h^+_{\nu,i,j,\overline{n}}, h^-_{\nu,i,j,\overline{n}})$ can be seen as the piecewise linear interpolation of the function $\left|k_{\nu,i,j}\right|^p$ in the interval $[-\overline{n}\delta, \overline{n}\delta]$ corresponding to the nodes $k_{\nu,i,j} = -\overline{n}\delta, -(\overline{n}-1)\delta, \ldots, -\delta, 0, \delta, \ldots, (\overline{n}-1)\delta, \overline{n}\delta$ when the following relation between $k_{\nu,i,j}$ and $h^+_{\nu,i,j,n}$,

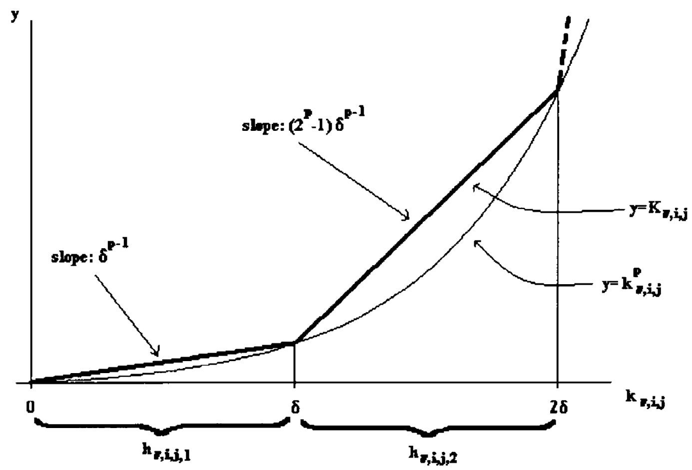

Figure 3. The function $k^p_{\nu,i,j}$, $k_{\nu,i,j} > 0$, $p > 1$, and its piecewise linear interpolation $K_{\nu,i,j}$.

$h^-_{\nu,i,j,n}$, $n = 1, 2, \ldots, \overline{n}$, holds: when $k_{\nu,i,j} \in [(m-1)\delta, m\delta]$, $1 \leqslant m \leqslant \overline{n}$, we must have $h^+_{\nu,i,j,n} = \delta$ for $n < m$, $h^+_{\nu,i,j,m} = k_{\nu,i,j} - (m-1)\delta$, $h^+_{\nu,i,j,n} = 0$ for $m < n \leqslant \overline{n}$, $h^-_{\nu,i,j,n} = 0$ for $1 \leqslant n \leqslant \overline{n}$; when $k_{\nu,i,j} \in [m\delta, (m+1)\delta]$, $-\overline{n} \leqslant m \leqslant -1$, we must have $h^-_{\nu,i,j,-n} = \delta$ for $n > m$, $h^-_{\nu,i,j,-m} = (m+1)\delta - k_{\nu,i,j}$, $h^-_{\nu,i,j,-n} = 0$ for $-\overline{n} \leqslant n < m$, $h^+_{\nu,i,j,n} = 0$ for $1 \leqslant n \leqslant \overline{n}$, see Figure 3. Let $\underline{h} \in I\!R^{2\overline{n}N}$ be the vector obtained substituting each component $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ of the vector $\underline{k}$ with the corresponding $2\overline{n}$ components $h^+_{\nu,i,j,n}$, $h^-_{\nu,i,j,n}$, $n = 1, 2, \ldots, \overline{n}$. We denote $\mathcal{L}(N, \delta, \overline{n})$ the space of the vectors $(\underline{k}^t, \underline{h}^t)^t$ whose components are in the previously mentioned relations. Using the previous approximation when $(\underline{k}^t, \underline{h}^t)^t \in \mathcal{L}(N, \delta, \overline{n})$ we can substitute the objective function of problem (12), $\|\underline{k}\|_p^p$, $p > 1$, with the following function:

$$l_p(\underline{h}) = \sum_{\nu=1,2} \sum_{(i,j)\in I_\nu} K_{\nu,i,j}(h^+_{\nu,i,j,1}, h^-_{\nu,i,j,1}, h^+_{\nu,i,j,2}, h^-_{\nu,i,j,2}, \ldots, h^+_{\nu,i,j,\overline{n}}, h^-_{\nu,i,j,\overline{n}}). \tag{16}$$

Note that when $p = 1$ we can choose $\overline{n} = 1$, so that for δ large enough no approximation is involved, that is: $\|\underline{k}\|_1 = l_1(\underline{h})$ for every $(\underline{k}^t, \underline{h}^t)^t \in \mathcal{L}(N, \delta, \overline{n})$. Moreover we can rewrite the constraints (11) as follows:

$$\begin{aligned}
&\sum_{n=1}^{\overline{n}} \Big(h^+_{1,i,j+1,n} - h^-_{1,i,j+1,n} - h^+_{1,i,j,n} + h^-_{1,i,j,n} - \\
&\quad h^+_{2,i+1,j,n} + h^-_{2,i+1,j,n} + h^+_{2,i,j,n} - h^-_{2,i,j,n} \Big) = \\
&\quad -(b_{1,i,j+1} - b_{1,i,j} - b_{2,i+1,j} + b_{2,i,j}), \\
&\quad (i, j) \in I_1 \cap I_2.
\end{aligned} \tag{17}$$

The constraints (17) can be rewritten in matrix-vector notation as $\tilde{A}\underline{h} = \underline{b}$, it is easy to see that $\tilde{A}$ has $(N_1 - 1)(N_2 - 1)$ rows and $2\overline{n}N$ columns, and that $\underline{b} \in \mathbb{Z}^{(N_1-1)(N_2-1)}$ is the vector appearing in problem (12). We consider the following problem:

$$\begin{aligned} \min \quad & l_p(\underline{h}) \\ \text{s.t.}: \quad & \tilde{A}\underline{h} = \underline{b}, \\ & \underline{h} \in \mathbb{Z}^{2\overline{n}N}, \\ & \underline{0} \leqslant \underline{h} \leqslant \underline{\delta}, \end{aligned} \tag{18}$$

where $\underline{0} = (0, 0, \ldots, 0)^t \in \mathbb{Z}^{2\overline{n}N}$, $\underline{\delta} = (\delta, \delta, \ldots, \delta)^t \in \mathbb{Z}^{2\overline{n}N}$.

Problem (18) is an approximation of problem (12) when the variables $k_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$ belong to the interval $[-\overline{n}\delta, \overline{n}\delta]$. In fact let $\underline{h}^* \in \mathbb{Z}^{2\overline{n}N}$ be a minimizer of problem (18) and $h^{+^*}_{\nu,i,j,n}$, $h^{-^*}_{\nu,i,j,n}$, $(i, j) \in I_\nu$, $\nu = 1, 2$, $n = 1, 2, \ldots, \overline{n}$ be the components of $\underline{h}^*$. It can be shown that the vector $\underline{h}^*$ has the following property: for $\nu = 1, 2$, $(i, j) \in I_\nu$, given n_1, with $1 \leqslant n_1 \leqslant \overline{n}$, such that $h^{+^*}_{\nu,i,j,n_1} \in (0, \delta]$ (or $h^{-^*}_{\nu,i,j,n_1} \in (0, \delta]$) then we have $h^{+^*}_{\nu,i,j,n} = \delta$ (or $h^{-^*}_{\nu,i,j,n} = \delta$) for $n = 1, 2, \ldots, n_1 - 1$ and $h^{-^*}_{\nu,i,j,n} = 0$ (or $h^{+^*}_{\nu,i,j,n} = 0$) for $n = 1, 2, \ldots, \overline{n}$. Note that $(0, \delta]$ denotes the bounded interval of end points 0, δ which is open to the left. This property is a consequence of the convexity of the objective function of problem (12), that makes the coefficients in (15) increasing when n increases, that is we have $\left(n_1^p - (n_1-1)^p\right)\delta^{p-1} < \left(n_2^p - (n_2-1)^p\right)\delta^{p-1}$ when $1 \leqslant n_1 < n_2 \leqslant \overline{n}$, $p > 1$. Let $\underline{k}^* \in \mathbb{Z}^N$ be the vector having components $k^*_{\nu,i,j} = \sum_{n=1}^{\overline{n}} \left(h^{+^*}_{\nu,i,j,n} - h^{-^*}_{\nu,i,j,n}\right)$, $(i, j) \in I_\nu$, $\nu = 1, 2$, then $\underline{k}^*$ is the approximation of the minimizer of problem (12) obtained substituting $\left\|\underline{k}\right\|_p^p$ with $l_p(\underline{h})$. We note that the property of $\underline{h}^*$ introduced previously guarantees that $l_p(\underline{h}^*)$ is an approximation of $\left\|\underline{k}^*\right\|_p^p$, for every $\underline{h}^*$ minimizer of problem (18), in fact we have that $(\underline{k}^{*t}, \underline{h}^{*t})^t \in \mathcal{L}(N, \delta, \overline{n})$. This does not guarantee that in general the solution of problem (12) is exactly the solution of problem (18). More precisely we have that $\left\|\underline{k}^*\right\|_p^p$ is an upper bound of the solution of problem (12), moreover the difference between $\left\|\underline{k}^*\right\|_p^p$ and the minimum attained at the solution of problem (12) is bounded by a quantity increasing with δ. When $\delta = 1$ we can state the following convergence result:

THEOREM 1. *Given $N_1, N_2, \overline{n} \in \mathbb{N}$, let A be the coefficient matrix of the linear system (11), $\tilde{A}$ be the coefficient matrix of the linear system (17), $\underline{b}$ be the vector on the right hand side of the linear systems (11) and (17). Let $\underline{h}^* \in \mathbb{Z}^{2\overline{n}N}$ be a solution of problem (18) where $\delta = 1$ such that $h^{\pm^*}_{\nu,i,j,\overline{n}} = 0$, $(i, j) \in I_\nu$, $\nu = 1, 2$; then $k^*_{\nu,i,j} = \sum_{n=1}^{\overline{n}} \left(h^{+^*}_{\nu,i,j,n} - h^{-^*}_{\nu,i,j,n}\right)$ is a minimizer of problem (12).*

Proof. It is a straightforward consequence of the fact that when $\delta = 1$ and $\overline{n}$ is large enough we have $\left\|\underline{k}\right\|_p^p = l_p(\underline{h})$, $\underline{k} \in \mathbb{Z}^N$, $\underline{h} \in \mathbb{Z}^{2\overline{n}N}$ and $(\underline{k}^t, \underline{h}^t)^t \in \mathcal{L}(N, \delta, \overline{n})$.

We note that Theorem 1 can be immediately extended to integer convex nonlinear programming problems more general than problem (12).

ALGORITHM 1. Read the parameters $N_1, N_2, p, \delta, \overline{n}$ and the data $W_{i,j}$, $(i, j) \in I$; compose the matrix $\tilde{A}$ and the vector $\underline{b}$ of problem (18); perform the following steps:

1. compute $\underline{h}^*$ as a minimizer of problem (18);
2. compute $\underline{k}^*$ as the vector whose components are given by:

$$k^*_{\nu,i,j} = \sum_{n=1}^{\overline{n}} \left(h^{+^*}_{\nu,i,j,n} - h^{-^*}_{\nu,i,j,n} \right), \quad (i, j) \in I_\nu , \quad \nu = 1, 2; \tag{19}$$

3. if $\left| k^*_{\nu,i,j} \right| = \overline{n}\delta$ for some $(i, j) \in I_\nu$, $\nu = 1, 2$ then stop and run again the algorithm with a larger value for the parameters $\overline{n}$, δ; otherwise compute the gradient vector field $(G_1, G_2)^t$ using formula (8), with $k_{\nu,i,j} = k^*_{\nu,i,j}$ $(i, j) \in I_\nu$, $\nu = 1, 2$ and compute $U_{i,j}$, $(i, j) \in I$ using formula (5);
4. stop.

Problem (18) is an integer linear minimum cost flow problem on a network corresponding to a graph $\tilde{\mathcal{G}}$, which is a slight variation of graph $\mathcal{G}$ shown in Figure 1. So that the integer constraints in problem (18) can be handled easily.

METHOD 2

This algorithm computes a solution of problem (12) when $p = \infty$. It is an algorithm more efficient than the previous one but it can deal only with problem (12) when $p = \infty$ because it is based on a simple remark about the solutions of problem (12) when $p = \infty$ and the solution of the same problem with an arbitrary objective function. That is we can consider problem (12) for an arbitrary objective function and with the additional bound constraints: $-\underline{e} \leqslant \underline{k} \leqslant \underline{e}$, where $\underline{e} = (e, e, \ldots, e)^t \in \mathbb{N}^N \subset \mathbb{R}^N$, where e is the smallest positive integer that makes the problem feasible. Then every solution of this last problem is a solution of problem (12) when $p = \infty$. This remark is a convergence result for the following algorithm as a method to solve the problem (12) when $p = \infty$.

ALGORITHM 2. Read the parameters N_1, N_2, and the data $W_{i,j}$, $(i, j) \in I$; compose the matrix A and the vector $\underline{b}$ of problem (12); perform the following steps:

1. compute $\underline{k}^{(0)}$ as a feasible point of problem (12);
2. set c equal to 1;
3. set the components of the vector $\underline{e} \in \mathbb{Z}^N$ equal to $\left\| \underline{k}^{(0)} \right\|_\infty$;

4. consider the following problem:

$$\begin{array}{ll} \min & \underline{0}^t \underline{k} \\ \text{s.t.}: & A\underline{k} = \underline{b} \\ & \underline{k} \in \mathbb{Z}^N, \\ & -\underline{e} \leqslant \underline{k} \leqslant \underline{e}, \end{array} \tag{20}$$

   where $\underline{0} = (0, 0, \ldots, 0)^t \in \mathbb{Z}^N$;
5. if problem (20) is feasible then compute $\underline{k}^{(c)}$ as the minimizer of problem (20); otherwise go to step 8;
6. increase c by 1;
7. if $\|\underline{k}^{(c-1)}\|_\infty > 1$ set the components of the vector $\underline{e} \in \mathbb{Z}^N$ equal to $\|\underline{k}^{(c-1)}\|_\infty -$ 1 and go to step 4;
8. compute the gradient vector field $(G_1, G_2)^t$ using formula (8), where $k_{\nu,i,j} = k^{(c-1)}_{\nu,i,j}$ $(i, j) \in I_\nu$, $\nu = 1, 2$ are the components of the vector $\underline{k}^{(c-1)}$ and compute $U_{i,j}$, $(i, j) \in I$ using formula (5);
9. stop.

Note that in Algorithm 1 and 2 the function $U = U_{i,j}$, $(i, j) \in I$ reconstructed through formula (5) is determined up to an additive constant. In Section 3 we consider the problem of determining how the choice of the parameter p affects the reconstruction of the function U. In the following we consider problem (12) for $p = 1$, $p = 2$ and $p = \infty$. Problem (12) for $p = \infty$ can be solved using Algorithm 1 when we consider a large value for p, such as for example $p = 10$, or by Algorithm 2. The two solutions obtained are in general different even if they have the same ∞-norm due to nonuniqueness of the minimizer. In the experiment shown in section 3 we comment on this fact.

3. Numerical experiments using SAR interferometry data

We present the SAR interferometry problem. This problem contains a phase unwrapping problem, which can be solved using the methods proposed in Section 2. Let $(x, y, z)^t$ be the cartesian coordinates system of $\mathbb{R}^3$ having the z-axis along the vertical direction and oriented downward, let $(r, y, \theta)^t$ be the cylindrical coordinates system of $\mathbb{R}^3$, given by the following change of variables:

$$r = \sqrt{x^2 + z^2}, \tag{21}$$

$$\theta = \begin{cases} \arctan \frac{z}{x}, & z > 0, x > 0, \\ \frac{\pi}{2}, & z > 0, x = 0. \end{cases} \tag{22}$$

We note that in the radar literature the cylindrical coordinate r is usually called *slant range coordinate* and the y coordinate is usually called *azimuth coordinate*. The Earth surface will be described by a surface of the type $z = f(x, y)$, $(x, y)^t \in \mathbb{R}^2$ for some given function f. The SAR system usually travels on board

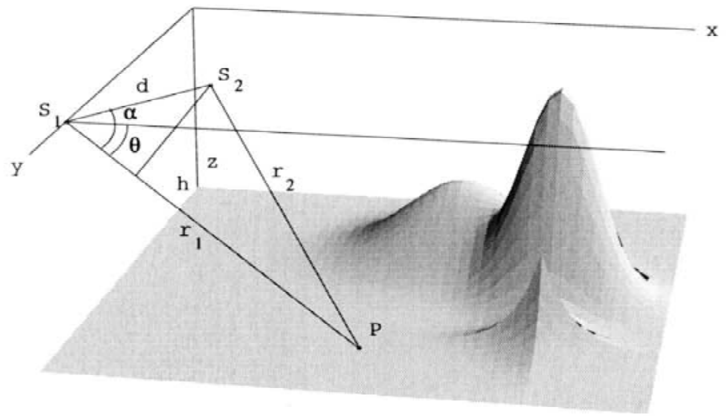

Figure 4. The SAR systems.

of satellites or airplanes. It is made by an antenna that measures information about the electromagnetic field backscattered by the observed scene on the Earth surface. This backscattering phenomenon is due to the presence of a known electromagnetic incident field emitted by the SAR system. These measurements are processed by the SAR system so that they can be interpreted as the measurements obtained with an antenna whose aperture is larger than the aperture of the physical antenna, that is a synthetic aperture antenna, so that the processed information has a higher resolution than the unprocessed one. Let us consider two SAR systems S_1, S_2 travelling on parallel trajectories along the direction of the y-axis (see Figure 4). We assume that S_1, S_2 have the same y coordinate, so that their reciprocal position can be determined by the distance d and the angle α that is the angle between the segment joining S_1, S_2 and the horizontal plane passing through S_1 (see Figure 4).

Note that in Figure 4 h is a reference distance, that is roughly speaking the distance of the SAR systems from the Earth. Each SAR system measures the phase of the electromagnetic radiation received modulus 2π as a function of the point P on the observed scene. Note that S_1, S_2, P belong to a plane y=constant. The observed scene is covered since the SAR systems S_1, S_2 travel along the y direction. The difference modulus 2π of the phases measured by S_1 and S_2 is called the *wrapped interferometric phase*. We note that in principle we can compute the z coordinate of the points of the observed scene from the knowledge of their $(x, y)^t$ coordinates, of the unwrapped interferometric phase and from the knowledge of the positions of S_1 and S_2. The *SAR interferometry phase unwrapping problem* can be stated as follows: from the knowledge of a wrapped interferometric phase compute the corresponding (unwrapped) interferometric phase. We can conclude that SAR interferometry when used to reconstruct the elevation map of the observed scene contains as a key subproblem a phase unwrapping problem.

Let $\Omega_1 \subset \{(x, y)^t \in I\!R^2\colon x \geqslant 0\}$ be a bounded open set, let $Z : \Omega_1 \to I\!R$ be a given function, then we assume that the observed scene is described by the surface $z = h - Z(x, y)$, $(x, y)^t \in \Omega_1$, in cartesian coordinates and by the function $\theta = \Theta(r, y)$, $(r, y)^t \in \Omega_2$, in cylindrical coordinates, where $\Omega_2 \subset I\!R^2$ is the image of Ω_1 through the change of variables (21), (22). We note that Θ is in general a multivalued function and/or a non injective function. For example let us fix a value y_0 for the coordinate y; it can occur that the curve in the x, z plane $z = h - Z(x, y_0)$, $(x, y_0)^t \in \Omega_1$, contains various points having the same value of the cylindrical coordinate r and distinct values of the cylindrical coordinate θ. This occurence generates the so called *layover phenomenon* in the SAR measurements. Moreover it can occur that the curve in the x, z plane $z = h - Z(x, y_0)$, $(x, y_0)^t \in \Omega_1$, contains various points having the same value of the cylindrical coordinate θ and distinct values of the cylindrical coordinate r. This occurence generates the so called *shadow phenomenon* in the SAR measurements. Layover phenomena and shadow phenomena usually produce inaccurate solution of the phase unwrapping problem. These phenomena correspond to real losses of information in the wrapped interferometric phase. Let us suppose that layover phenomena and shadow phenomena do not occur, then it is easy to see that the exact unwrapped interferometric phase is given by:

$$u_e(r, y) = \frac{4\pi}{\lambda} d \cos(\Theta(r, y) - \alpha) , \quad (r, y)^t \in \Omega_2, \tag{23}$$

where λ is the wavelength of the electromagnetic fields used to perform the measurements, (for details, see Costantini et al, 1999). For $(r, y)^t \in \Omega_2$ the quantity $u_e(r, y)$ is given by (23) in the points where there is no layover or shadow phenomenon and is given by the formulae explained in (Costantini et al, 1999) where there is layover or shadow phenomenon. To take care of the experimental errors in the measurements the unwrapped interferometric phase is perturbed with a random error term, that is:

$$u(r, y) = u_e(r, y) + \epsilon(r, y) , \quad (r, y)^t \in \Omega_2, \tag{24}$$

where $\epsilon(r, y)$ is the random error. The wrapped interferometric phase is given by:

$$w(r, y) = [u(r, y)]_{2\pi} , \quad (r, y)^t \in \Omega_2. \tag{25}$$

We assume that the measurement of w is made by a sampling operation; we define:

$$W_{i,j} = w(r_0 + j\delta_2, y_0 + i\delta_1) , \quad (i, j) \in I, \tag{26}$$

where the set I is defined as done in the previous sections and r_0, y_0, $\delta_1 > 0$, $\delta_2 > 0$ are parameters depending on the sampling operation. Of course we have to compute the interferometric phase u_e according to the same sampling parameters, that is we have to compute:

$$U_{e,i,j} = u_e(r_0 + j\delta_2, y_0 + i\delta_1) , \quad (i, j) \in I, \tag{27}$$

such that $W_{i,j} = [U_{e,i,j} + \epsilon(r_0 + j\delta_2, y_0 + i\delta_1)]_{2\pi}$, $(i, j) \in I$.

This is a phase unwrapping problem of the form stated in the introduction, so that we can compute its solution using the methods proposed in Section 2.

In the numerical simulation we consider synthetic SAR interferometry data and real SAR interferometry data. We note that the numerical results of this section are obtained solving problem (18) when Algorithm 1 is used and problems (12), (20) when Algorithm 2 is used with RELAX4.⋆ We note that in the reconstruction procedure of the unwrapped interferometric phase we must perform two different steps. In the first one, analyzing the wrapped interferometric phase we must choose the parameter p to use in the unwrapping procedure. In the second step we must solve the phase unwrapping problem, with the parameter p chosen in the first step, using Algorithm 1, or Algorithm 2. We have not jet a satisfactory method to perform the first step of this reconstruction procedure. However for a more complete information of the reader about this step we show, for the synthetic data, the graphical representation of the wrapped interferometric phase and of the minimizer $\underline{k}^*$ of problem (12) for $p = 1$ and $p = \infty$.

SYNTHETIC DATA

In the following m stands for meter, the usual unit of length. Three different examples are considered. We fix the following parameters: $h = 782563$m, $d = 143.13$m, $\alpha = -1.44$radiants, $\lambda = 0.056415$m. This choice corresponds to the choice made in the experiment carried out in the ERS-1 mission of the European Space Agency. In the following examples we consider a grid of points on Ω_2, and from relations (21), (22), (23), (24), (27) we compute the interferometric phase $U_{i,j}$, $(i, j) \in I$ corresponding to a given surface $z = h - Z(x, y)$ defined for $(x, y)^t \in \Omega_1$. Moreover, with the previous notation, the grid on Ω_2 is defined by the following parameters: $r_0 = 846645$m, $y_0 = -797$m, $\delta_1 = 15.944$m, $\delta_2 = 7.905$m, $N_1 = 100$, $N_2 = 100$.

EXAMPLE 2. Let us consider the scene given by the following surface:

$$\begin{aligned} Z(x, y) &= \max\left\{0, 100 - \left((x - x_c)^2 + (y - y_c)^2\right)^{1/2}\right\}, \\ & x \in [323115m, 325179m], \quad y \in [-797m, 797m], \end{aligned} \tag{28}$$

where $x_c = 324148$m, $y_c = 0$m (see Figure 5). We assume no noise in the measurements of the interferometric phase, i.e. in formula (24) we take $\epsilon(r, y) = 0$, $(r, y)^t \in \Omega_2$.

⋆ RELAX4 is a minimum cost flow problems solver designed by D.P. Bertsekas and P. Tseng (for details see Bertsekas, 1991, p. 279). This software package is available free of charge in the Web site `http://www.mit.edu/people/dimitrib/RELAX4.txt`

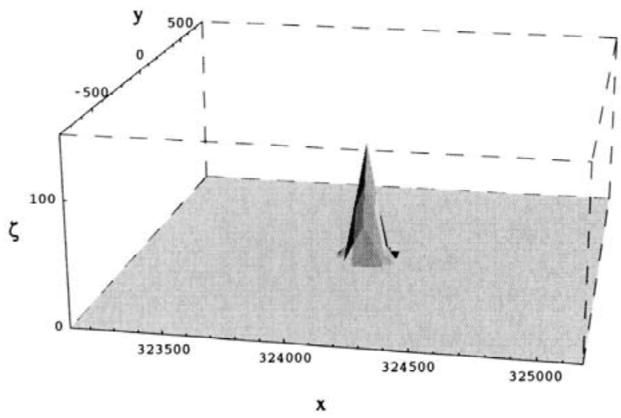

Figure 5. The scene of Example 2. In the figure we show $\zeta = Z(x, y)$.

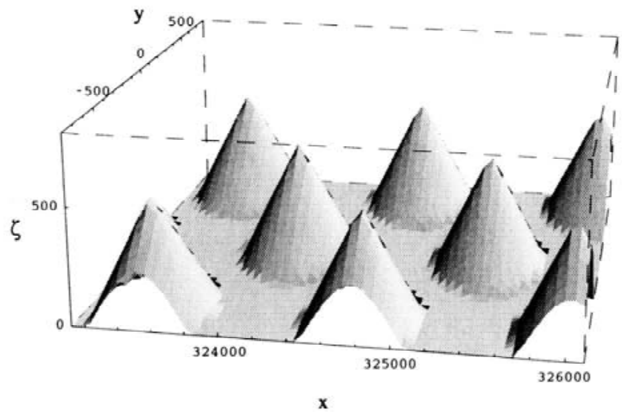

Figure 6. The scene of Examples 3, 4. In the figure we show $\zeta = Z(x, y)$.

EXAMPLE 3. Let us consider the scene given by the following surface:

$$\begin{aligned} Z(x, y) = \max \Big\{ & 0, 500 - \big([(x - x_c) - (y - y_c)]^2_{1250} + \\ & [(x - x_c) + (y - y_c)]^2_{1250}\big)^{1/2}\Big\}, \\ & x \in [323115m, 326104m], \;\; y \in [-797m, 797m], \end{aligned} \tag{29}$$

where $x_c = 324148$m, $y_c = 0$m (see Figure 6). Note that in (29) $[\cdot]_{1250}$ is the operation that consists in taking the modulus 1250 of $\cdot$ and we choose $[\cdot]_{1250} \in [-625, 625)$. We assume no noise in the measurements of the interferometric phase, i.e. in formula (24) we take $\epsilon(r, y) = 0$, $(r, y)^t \in \Omega_2$.

EXAMPLE 4. Let us consider the scene of Example 3 (see Figure 6). We assume some noise in the measurements of the interferometric phase, i.e. in formula (24) we compute $\epsilon(r, y)$, $(r, y)^t \in \Omega_2$ in the grid points $(r_0 + j\delta_2, y_0 + i\delta_1)^t$, $(i, j) \in I$ sampling a random variable uniformly distributed in the interval $\left[-\frac{\pi}{3}, \frac{\pi}{3}\right]$.

We note that the scene in Example 2 is given by a horizontal plane at level $z = h$ and by a cone having height equal to 100m and slope equal to $\frac{\pi}{4}$ radiants. The scene in Examples 3 and 4 is given by a horizontal plane at level $z = h$ and by several cones of height equal to 500m and slope equal to $\frac{\pi}{4}$radiants. Note that in figures 5 and 6 different scales are used.

In Table 1 we report the results obtained in the Examples 2, 3, 4 using Algorithm 1 for $p = 1$, $p = 2$, $p = \infty$ and using Algorithm 2. In Algorithm 1 we have chosen $\delta = 1$, $\overline{n} = 3$. The results obtained with Algorithm 2 are shown to evaluate the quality of the results obtained with Algorithm 1 for $p = \infty$. In particular from the solution $\underline{k}^*$ of problem (12) we compute the reconstructed interferometric phase U using formula (5) and choosing $U_{1,1} = U_{e,1,1}$. Two performance indices defined:

$$E_{l^2} = \left(\frac{\sum_{(i,j)\in I} (U_{e,i,j} - U_{i,j})^2}{\sum_{(i,j)\in I} (U_{e,i,j} - U_{e,1,1})^2} \right)^{1/2}, \tag{30}$$

$$K_{l^p} = \left(\frac{\sum_{\nu=1,2} \sum_{(i,j)\in I_\nu} \left| k^*_{\nu,i,j} + b_{\nu,i,j} \right|^p}{\sum_{\nu=1,2} \sum_{(i,j)\in I_\nu} \left| b_{\nu,i,j} \right|^p} \right)^{1/p}, \quad p \geqslant 1, \tag{31}$$

$$K_{l^p} = \frac{\max\left\{ \left| k^*_{\nu,i,j} + b_{\nu,i,j} \right|,\ (i, j) \in I_\nu,\ \nu = 1, 2 \right\}}{\max\left\{ \left| b_{\nu,i,j} \right|,\ (i, j) \in I_\nu,\ \nu = 1, 2 \right\}}, \quad p = \infty, \tag{32}$$

are shown in Table 1. Note that the performance index E_{l^2} is independent from the additive constant $U_{1,1}$ used to reconstruct the unwrapped phase function U. Index E_{l^2} is a measure of the distance between U_e and U, when $E_{l^2} = 0$ we have $U_e = U$, when E_{l^2} is 'small' we have that U is close to U_e. Index K_{l^p} is a measure of the distance between the vector $\underline{k}^*$ and the vector having components $-b_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$. Note that this last vector is always a feasible point for problem (12), and it can be considered a trivial solution of problem (12). Finally in Table 1 we report the number of variables N and the number of constraints M of problem (12) and the time T, measured in seconds(s), necessary to solve the corresponding problem on the following computer: Digital Ultimate Workstation 533au, CPU: DEC 21164A Alpha AXP533MHz.

Let us make some comments on the properties of the solutions of problem (12) depending on the value of the parameter p, (for details, see Aluffi-Pentini et al., 1999; Björck, 1996). It can be seen that any non-trivial solution of problem (12) for p large or $p = \infty$ tends to have more non-null components than the corresponding solutions for p small or $p = 1$. Moreover the components of the solutions for p large or $p = \infty$ tend to assume smaller values than the corresponding components of the solutions for p small or $p = 1$. We note that a non-null correction $k_{\nu,i,j}$,

Table 1. The numerical results obtained on the synthetic data.

	$p=1$	$p=2$	$p=\infty$	
	Algorithm 1	Algorithm 1	Algorithm 1	Algorithm 2
Example 2	$\|\underline{k}^*\|_1 = 2$	$\|\underline{k}^*\|_1 = 2$	$\|\underline{k}^*\|_1 = 2$	$\|\underline{k}^*\|_1 = 2$
	$\|\underline{k}^*\|_2^2 = 2$	$\|\underline{k}^*\|_2^2 = 2$	$\|\underline{k}^*\|_2^2 = 2$	$\|\underline{k}^*\|_2^2 = 2$
	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$
	$E_{l^2} = 0$	$E_{l^2} = 0$	$E_{l^2} = 0$	$E_{l^2} = 0$
	$K_{l^1} = 0.98$	$K_{l^2} = 0.99$	$K_{l^\infty} = 1$	$K_{l^\infty} = 1$
	$N = 39608$	$N = 158020$	$N = 158020$	$N = 39608$
	$M = 9806$	$M = 29412$	$M = 29412$	$M = 9806$
	$T = 1.37s$	$T = 8.36s$	$T = 1.37s$	$T = 2.85s$
Example 3	$\|\underline{k}^*\|_1 = 510$	$\|\underline{k}^*\|_1 = 510$	$\|\underline{k}^*\|_1 = 510$	$\|\underline{k}^*\|_1 = 721$
	$\|\underline{k}^*\|_2^2 = 518$	$\|\underline{k}^*\|_2^2 = 510$	$\|\underline{k}^*\|_2^2 = 510$	$\|\underline{k}^*\|_2^2 = 721$
	$\|\underline{k}^*\|_\infty = 2$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$
	$E_{l^2} = 0.80$	$E_{l^2} = 0.80$	$E_{l^2} = 0.80$	$E_{l^2} = 0.70$
	$K_{l^1} = 0.61$	$K_{l^2} = 0.77$	$K_{l^\infty} = 2$	$K_{l^\infty} = 2$
	$N = 39608$	$N = 158020$	$N = 158020$	$N = 39608$
	$M = 9806$	$M = 29412$	$M = 29412$	$M = 9806$
	$T = 1.60s$	$T = 8.09s$	$T = 8.65s$	$T = 1.83s$
Example 4	$\|\underline{k}^*\|_1 = 569$	$\|\underline{k}^*\|_1 = 570$	$\|\underline{k}^*\|_1 = 570$	$\|\underline{k}^*\|_1 = 824$
	$\|\underline{k}^*\|_2^2 = 579$	$\|\underline{k}^*\|_2^2 = 570$	$\|\underline{k}^*\|_2^2 = 570$	$\|\underline{k}^*\|_2^2 = 824$
	$\|\underline{k}^*\|_\infty = 2$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$	$\|\underline{k}^*\|_\infty = 1$
	$E_{l^2} = 0.92$	$E_{l^2} = 0.92$	$E_{l^2} = 0.92$	$E_{l^2} = 0.87$
	$K_{l^1} = 0.51$	$K_{l^2} = 0.73$	$K_{l^\infty} = 2$	$K_{l^\infty} = 2$
	$N = 39608$	$N = 158020$	$N = 158020$	$N = 39608$
	$M = 9806$	$M = 29412$	$M = 29412$	$M = 9806$
	$T = 1.44s$	$T = 10.39s$	$T = 2.54s$	$T = 1.59s$

$(i, j) \in I_\nu$, $\nu = 1, 2$ in (8) is expected where the interferometric phase comes from points of the scene where the irrotational property is violated. We note that in Example 2 the irrotational property is violated at 0.04% of the points of $I_1 \cap I_2$; in Example 3 the irrotational property is violated at 3.37% of the points of $I_1 \cap I_2$; in Example 4 the irrotational property is violated at 4.76% of the points of $I_1 \cap I_2$.

From these remarks we expect that in Example 2 problem (12) for p small gives a better solution than the solution obtained solving the same problem for p large. On the contrary we expect that in Examples 3, 4 problem (12) for p small gives a worse solution than the solution obtained solving the same problem for p large. These opinions are confirmed in Table 1 when the solution of problem (12) with $p = \infty$ obtained with Algorithm 2 is considered. Moreover we note that in

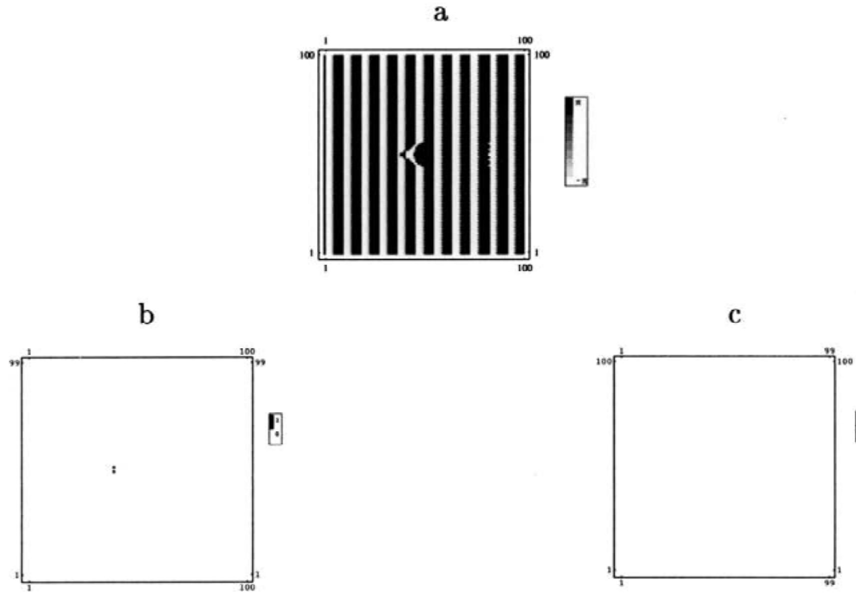

Figure 7. Example 2:(a) the wrapped interferometric phase W, (b) the exact unwrapping integers $|k_{1,i,j}|$, $(i, j) \in I_1$, (c) the exact unwrapping integers $|k_{2,i,j}|$, $(i, j) \in I_2$.

all the examples Algorithms 1, 2 have provided a solution different fron the trivial solution $-b_{\nu,i,j}$, $(i, j) \in I_\nu$, $\nu = 1, 2$, in fact we have always $K_{l^p} \neq 0$. For Example 3 we have large E_{l^2} errors in the reconstructed phase U. These errors are due to the fact that the scene considered induces a difficult phase unwrapping problem. In Example 4 the random perturbation ϵ makes the situation worse. In Table 1 can be observed that there are several solutions of problem (12) for $p = \infty$, in fact different algorithms find different solutions. Furthermore one of the solutions corresponding to $p = \infty$ is better than the other solutions. This is a promising fact that should be exploited. We hope to be able to take advantage of the degeneracy of problem (12) when $p = \infty$ to find a satisfactory solution of the phase unwrapping problem when difficult scenes are considered.

In Figure 7 we have shown for Example 2 the unwrapped interferometric phase W and the exact unwrapping integers k_ν, $\nu = 1, 2$, that is the integers satisfying relation (8) where G_ν, $\nu = 1, 2$ are substituted with $\Delta_\nu U_e$, $\nu = 1, 2$. In Figures 9 and 11 it is shown tha same information for Example 3, 4 respectively. In Figures 8, 10 and 12 we have shown for Example 2, 3, 4 respectively the results obtained with Algorithm 1 for $p = 1$ and with Algorithm 2 ($p = \infty$).

REAL DATA

We consider an example where the interferometric data are obtained by a pair of ERS-1 SAR measurements of a region of Sardinia, Italy. These SAR measurements

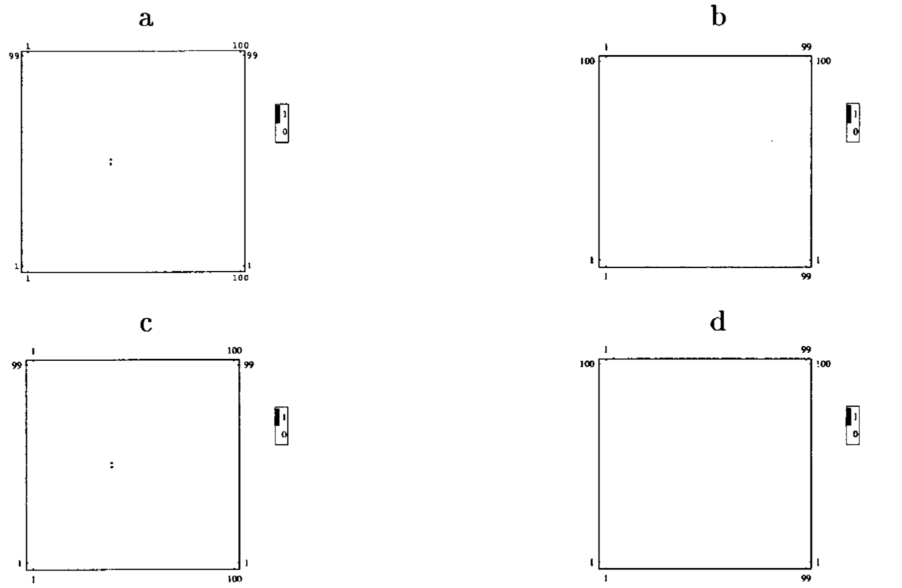

Figure 8. Example 2: (a) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$ computed with Algorithm 1 for $p = 1$, (b) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$ computed with Algorithm 1 for $p = 1$, (c) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$ computed with Algorithm 2 ($p = \infty$), (d) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$ computed with Algorithm 2 ($p = \infty$).

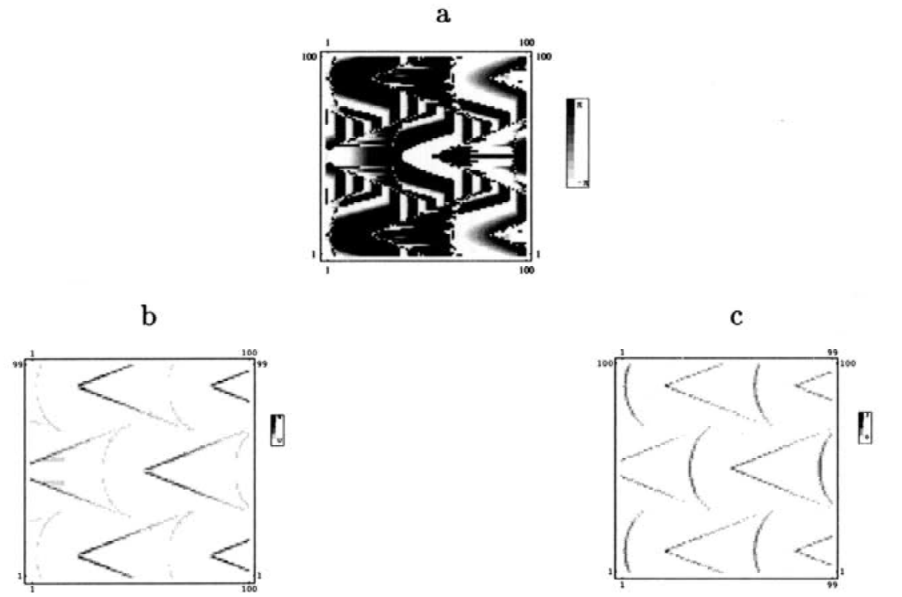

Figure 9. Example 3: (a) the wrapped interferometric phase W, (b) the exact unwrapping integers $\left|k_{1,i,j}\right|$, $(i, j) \in I_1$, (c) the exact unwrapping integers $\left|k_{2,i,j}\right|$, $(i, j) \in I_2$.

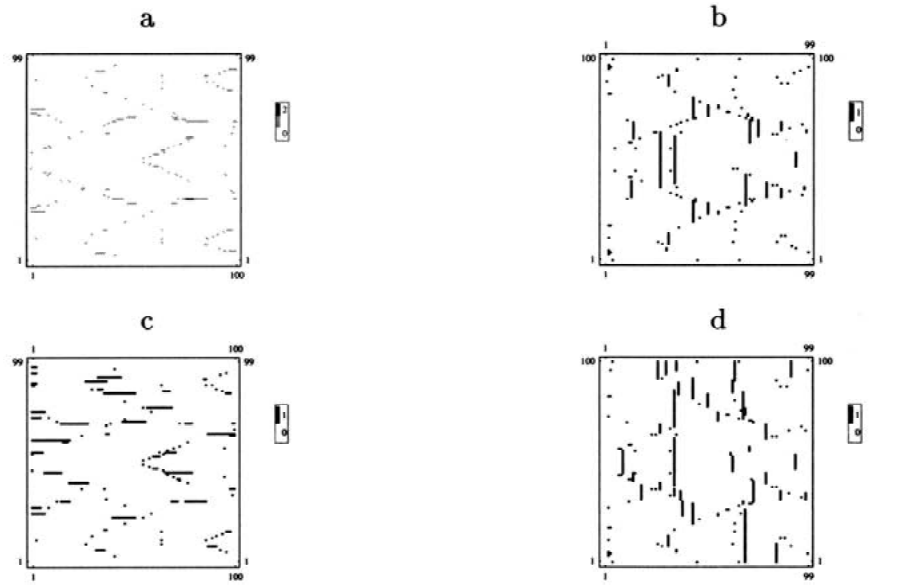

Figure 10. Example 3: (a) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 1 for $p = 1$, (b) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 1 for $p = 1$, (c) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 2 ($p = \infty$), (d) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 2 ($p = \infty$).

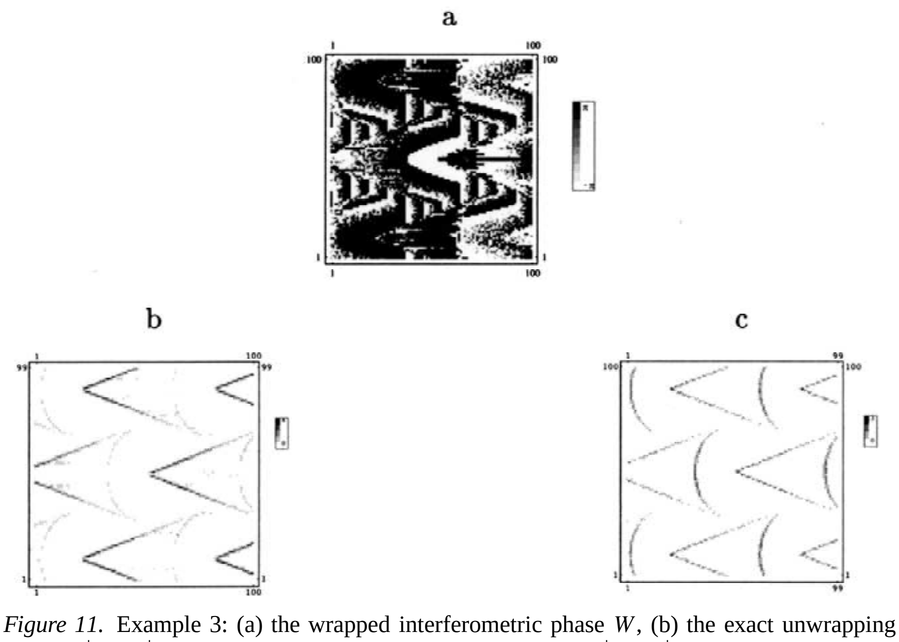

Figure 11. Example 3: (a) the wrapped interferometric phase W, (b) the exact unwrapping integers $\left|k_{1,i,j}\right|$, $(i, j) \in I_1$, (c) the exact unwrapping integers $\left|k_{2,i,j}\right|$, $(i, j) \in I_2$.

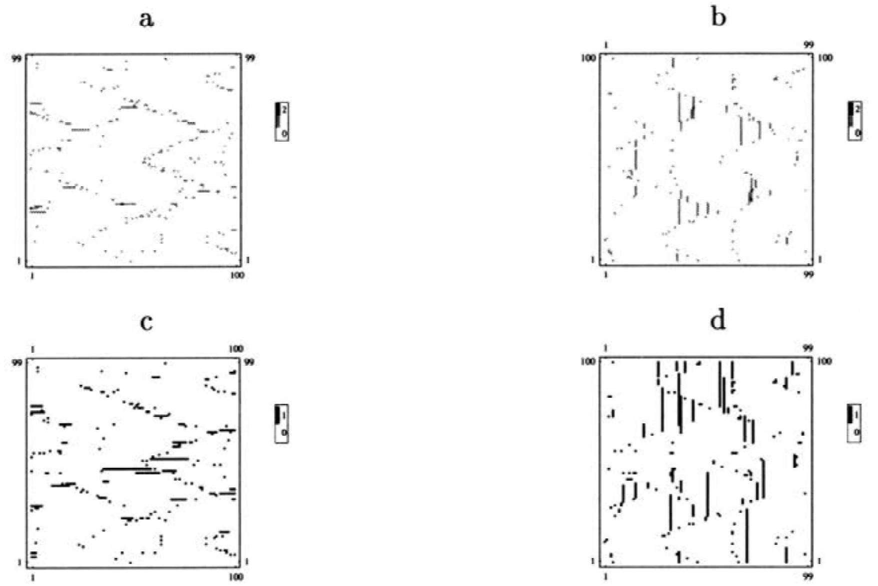

Figure 12. Example 4: (a) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 1 for $p = 1$, (b) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 1 for $p = 1$, (c) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 2 ($p = \infty$), (d) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 2 ($p = \infty$).

are relative to $N_1 = 641$ samples along the azimuth direction and $N_2 = 329$ samples along the slant range direction.

Figure 13 reports the wrapped interferometric phase $W = W_{i,j}$, $(i, j) \in I$ measured by the ERS-1 satellite. Moreover we note that for these data the irrotational property is violated at 5.28% of the points of $I_1 \cap I_2$.

Table 2 is similar to Table 1. Of course in Table 2 does not appear the performance index E_{l^2} defined in (30) since we do not know the exact unwrapped interferometric phase U_e. A satisfactory analysis of the solution obtained with real data can be done comparing the true digital elevation map of the region of Sardinia related to the data considered with the digital elevation map obtained from the unwrapped interferometry phase computed with the different methods proposed here. However at this time we can not report this comparison, because is not available to us sufficiently accurate information about the digital elevation map of Sardinia.

Figure 14 reports the solution computed with Algorithm 1 for $p = 1$ and the solution computed with Algorithm 2. Figures 15–18 report the corresponding unwrapped phase functions obtained for several choices of the parameter p that defines the mathematical model used. Moreover in Algorithm 1 we have chosen $\delta = 1$, $\overline{n} = 3$ All the reconstructions of unwrapped phase functions U are computed assuming $U_{1,1} = 0$.

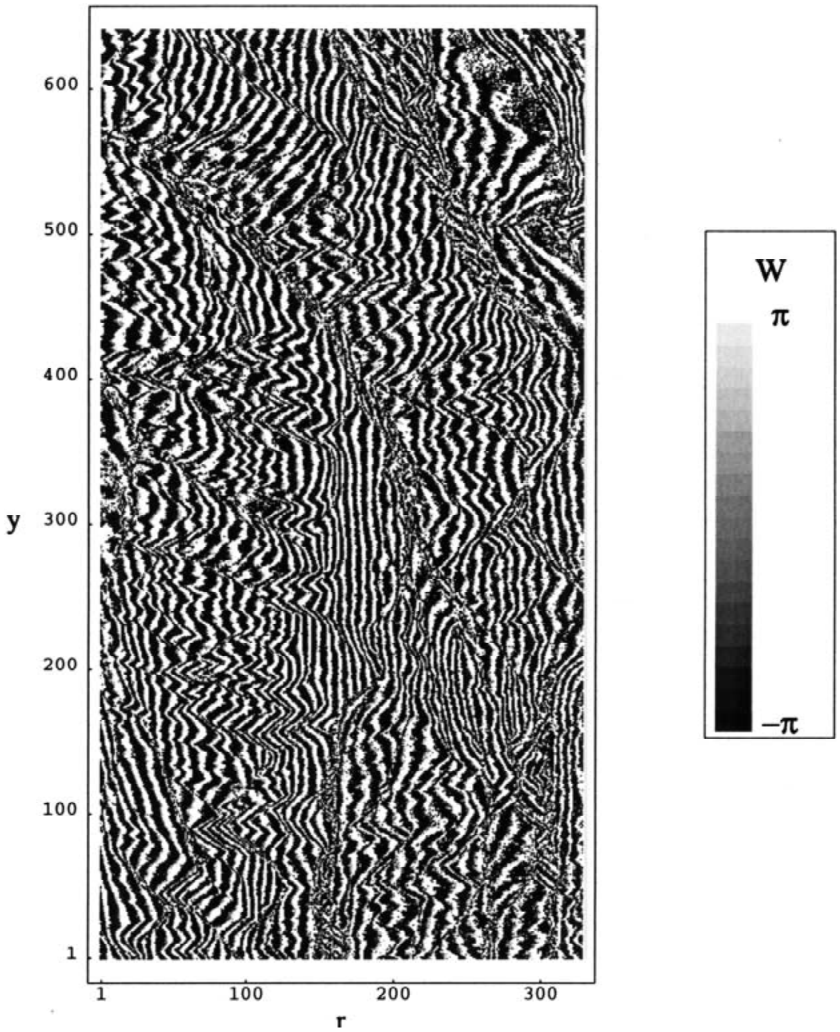

Figure 13. The wrapped interferometric phase of a region of Sardinia, Italy. Measurements made by the ERS-1 SAR system.

We can see that the unwrapped phase functions depend on the parameter p for local details. Further research in this direction to characterize features of the data useful to choose p will be reported elsewhere.

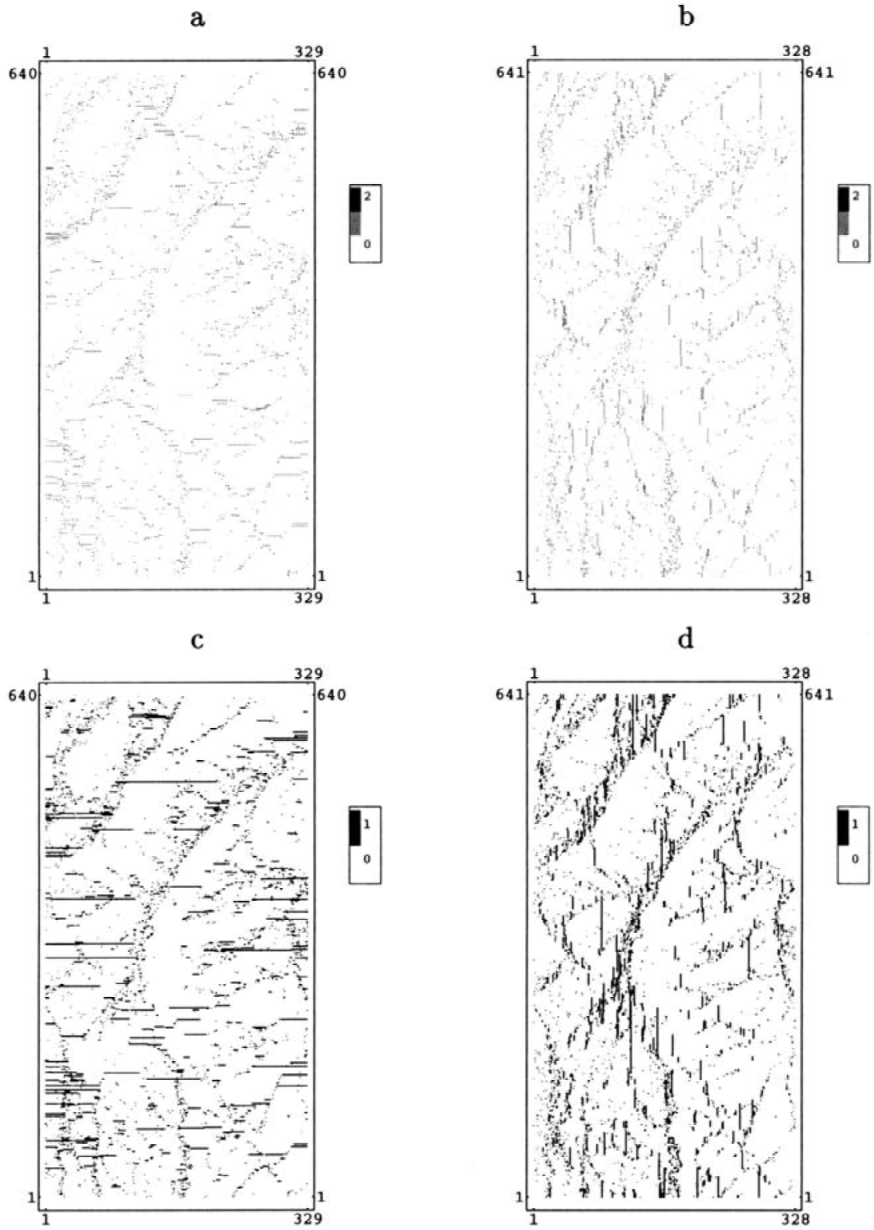

Figure 14. Real data: (a) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 1 for $p = 1$, (b) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 1 for $p = 1$, (c) $\left|k^*_{1,i,j}\right|$, $(i, j) \in I_1$, computed with Algorithm 2 ($p = \infty$), (d) $\left|k^*_{2,i,j}\right|$, $(i, j) \in I_2$, computed with Algorithm 2 ($p = \infty$).

Table 2. The numerical results obtained on the real data.

$p = 1$	$p = 2$	$p = \infty$	
Algorithm 1	Algorithm 1	Algorithm 1	Algorithm 2
$\lVert \underline{k}^* \rVert_1 = 12810$	$\lVert \underline{k}^* \rVert_1 = 12817$	$\lVert \underline{k}^* \rVert_1 = 12817$	$\lVert \underline{k}^* \rVert_1 = 18661$
$\lVert \underline{k}^* \rVert_2^2 = 12924$	$\lVert \underline{k}^* \rVert_2^2 = 12817$	$\lVert \underline{k}^* \rVert_2^2 = 12817$	$\lVert \underline{k}^* \rVert_2^2 = 18661$
$\lVert \underline{k}^* \rVert_\infty = 2$	$\lVert \underline{k}^* \rVert_\infty = 1$	$\lVert \underline{k}^* \rVert_\infty = 1$	$\lVert \underline{k}^* \rVert_\infty = 1$
$K_{l^1} = 0.61$	$K_{l^2} = 0.78$	$K_{l^\infty} = 2$	$K_{l^\infty} = 2$
$N = 841624$	$N = 3364544$	$N = 3364544$	$N = 841624$
$M = 209925$	$M = 629769$	$M = 629769$	$M = 209925$
$T = 17.26s$	$T = 74.02s$	$T = 62.99s$	$T = 12.85s$

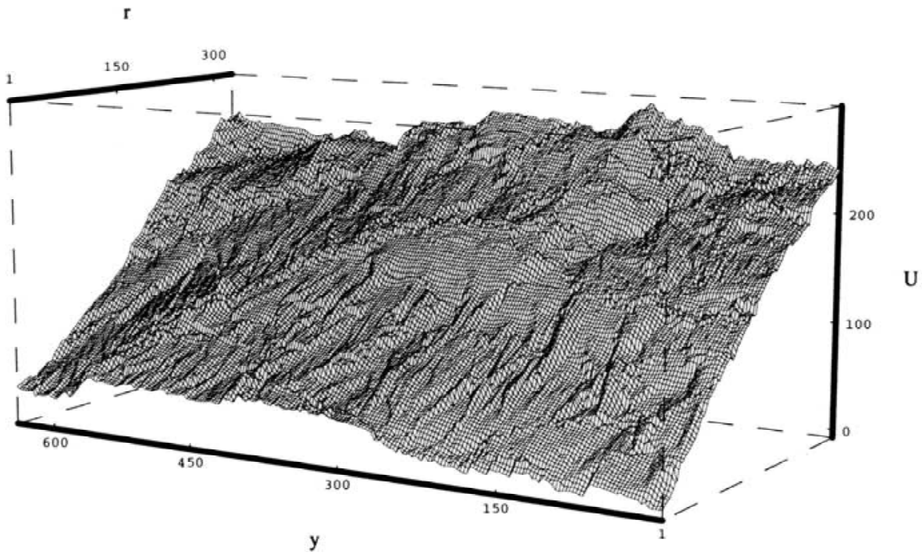

Figure 15. The unwrapped interferometric phase computed with Algorithm 1 choosing $p = 1$.

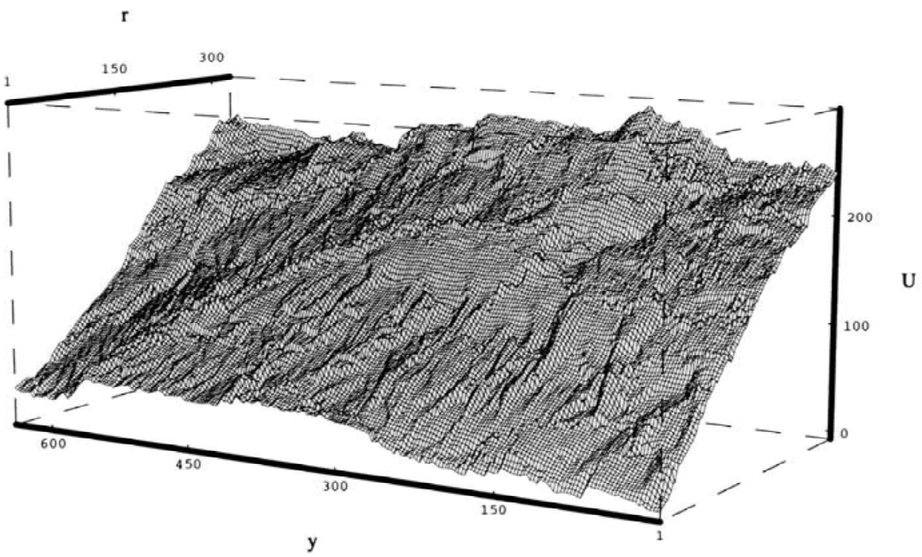

Figure 16. The unwrapped interferometric phase computed with Algorithm 1 choosing $p = 2$.

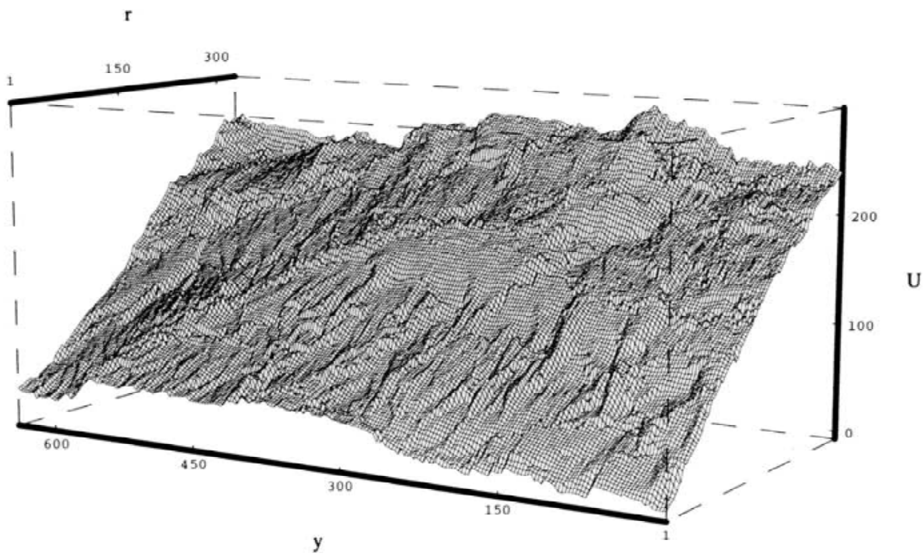

Figure 17. The unwrapped interferometric phase computed with Algorithm 1 choosing $p = \infty$.

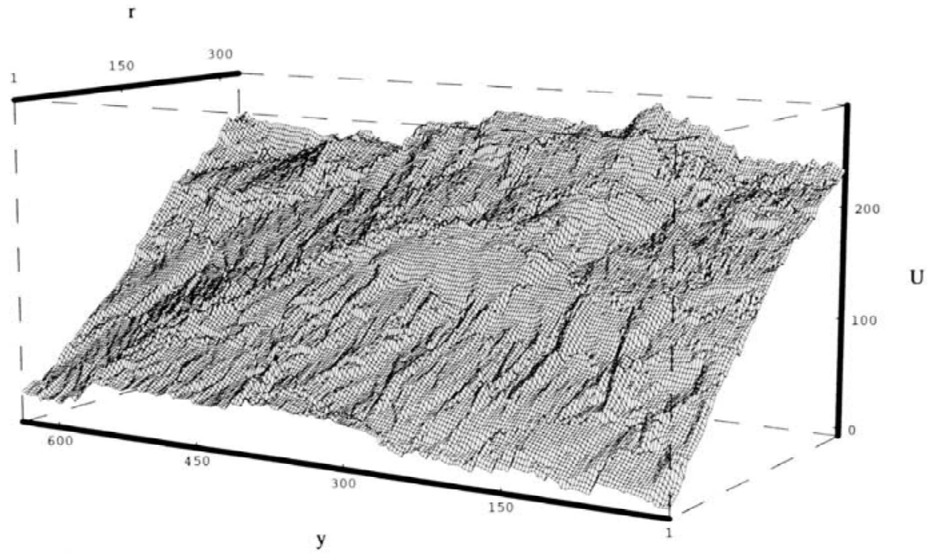

Figure 18. The unwrapped interferometric phase computed with Algorithm 2.

Acknowledgement

This research is partially supported by ASI-Agenzia Spaziale Italiana under the grant ASI-ARS-99-50.

References

Oppenheim, A.V. and Lim, J.S. (1981). The importance of phase in signals. *Proceedings of the IEEE* 69, 529–541.

Zebker, H.A. and Goldstein, R.M. (1986). Topographic mapping from interferometric synthetic aperture radar observations. *Journal of Geophysical Research* 91, 4993–4999.

Goldstein, R.M., Zebker, H.A. and Werner, C.L. (1988). Satellite radar interferometry: two-dimensional phase unwrapping, *Radio Science* 23, 713–720.

Costantini, M. (1998). A novel phase unwrapping method based on network programming. *IEEE Transactions on Geoscience and Remote Sensing* 36, 813–821.

Bertsekas, D.P. (1991). *Linear Network Optimization*. MIT Press, Cambridge, MA.

Cook, W.J., Cunningham, W.H., Pulleyblank, W.R. and Schrijver, A. (1998). *Combinatorial Optimization*. John Wiley & Sons, New York.

Costantini, M., Farina, A. and Zirilli, F. (1999). A fast phase unwrapping algorithm for SAR interferometry. *IEEE Transactions on Geoscience and Remote Sensing* 37, 452–460.

Fried, D.L. (1977). Least-squares fitting a wave-front distorsion estimated to an array of phase-difference measurements. *Journal of the Optical Society of America* 67, 370–375.

Moré, J.J. and Wright, S.J. (1993). *Optimization Software Guide*. SIAM, Philadelphia, PA.

Fletcher, R. and Leyffer, S. (1994). Solving mixed integer nonlinear programs by outer approximation. *Mathematical Programming* 66, 327–349.

Toint, Ph.L. and Tuyttens, D. (1992). LSSNO: a Fortran subroutine for solving large-scale nonlinear network optimization problems. *ACM Transactions on Mathematical Software* 18, 308–328.

Dantzing, G.B. (1963). *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ.

Aluffi-Pentini, F., Castrignanò, T., Maponi, P., Parisi, V. and Zirilli, F. (1999). Generalized solution of linear systems and image restoration. *Journal on Optimization Theory and Applications* 103, 45–64.

Björck, Å. *Numerical Methods for Least Squares Problems*, SIAM, (1996) Philadelphia, PA.