Journal of Computational Physics 228 (2009) 2566–2571

Contents lists available at ScienceDirect

Journal of Computational Physics

journal homepage: www.elsevier.com/locate/jcp

Sobolev gradient approach for the time evolution related to energy minimization of Ginzburg–Landau functionals

Nauman Raza [a,b,*], Sultan Sial [c], Shahid S. Siddiqi [a]

[a] Department of Mathematics, University of the Punjab, House No. 15, Street No. 39, Lahore, Punjab 54840, Pakistan
[b] Lahore University of Management Sciences, Lahore, Pakistan
[c] Department of Mathematics, Lahore University of Management Sciences, Opposite Sector U, DHA, Lahore Cantt. 54792, Pakistan

ARTICLE INFO

Article history:
Received 11 January 2008
Received in revised form 24 October 2008
Accepted 12 December 2008
Available online 25 December 2008

Keywords:
Sobolev gradient
Model A

ABSTRACT

The Sobolev gradient technique has been discussed previously in this journal as an efficient method for finding energy minima of certain Ginzburg–Landau type functionals [S. Sial, J. Neuberger, T. Lookman, A. Saxena, Energy minimization using Sobolev gradients: application to phase separation and ordering, J. Comput. Phys. 189 (2003) 88–97]. In this article a Sobolev gradient method for the related time evolution is discussed.

1. Introduction

An important approach to the solution of PDEs is to seek a critical point of a functional, constructed so that the equation can be considered to be solved when the functional is minimal. The recent theory of Sobolev gradients [2] gives a unified approach for such problems, both in function spaces and in finite-dimensional approximations to such problems. Sobolev gradients have been used for ODE problems [2,3] in a finite-difference setting, PDEs in finite-difference [3] and finite-element settings [4], minimizing energy functionals associated with Ginzburg–Landau models in finite-difference [1] and finite-element [5,6] settings, the electrostatic potential equation [7], nonlinear elliptic problems [8], semilinear elliptic systems [9], simulation of Bose–Einstein condensates [10], inverse problems in elasticity [11] and groundwater modelling [12].

Sobolev gradients [2] have been discussed before in this journal [1] as a method for energy minimization of Ginzburg–Landau functionals related to phase separation and ordering. In this article we would like to extend that approach to the related Ginzburg–Landau time evolution problems.

First, we briefly review the theory of Sobolev gradients in a finite-difference setting applied to minimizing a Ginzburg–Landau free energy functional. This theory leads to a steepest descent method in an appropriate Sobolev space as in [1]. We then consider two possible approaches to the related time-dependent problem. The numerical results for one approach are then presented. A comparison of Sobolev gradient method is made with Newton's method. Results are reported for numerical experiments. All numerical experiments are carried out on an Intel Xeon 3.2 GHZ dual processor machine with 2 GB RAM.

2. Model A time evolution

The minimization of the Model A Ginzburg–Landau free energy functional

* Corresponding author. Address: Department of Mathematics, University of the Punjab, House No. 15, Street No. 39, Lahore, Punjab 54840, Pakistan.
E-mail addresses: raza_nauman@yahoo.com (N. Raza), sultans@lums.edu.pk (S. Sial), shahidsiddiqiprof@yahoo.co.uk (S.S. Siddiqi).

0021-9991/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2008.12.017

$$F(u) = \int_V \frac{u^4}{4} - \frac{u^2}{2} + \frac{\kappa}{2}|\nabla u|^2 \tag{1}$$

has been considered in [1]. The static and dynamical properties of this model have been extensively studied, primarily in numerical work related to coarsening and growth of domains [13–15].

In the continuous case, the related Ginzburg–Landau time evolution is

$$u_t = -\nabla F(u) \tag{2}$$

which on the interior of the system is

$$u_t = u - u^3 + \nabla^2 u. \tag{3}$$

The method suggested for evolving systems from one time to another in the primary reference for Sobolev gradients [2] is as follows. Form a functional

$$\phi(u) = \left\| u - f + \frac{\delta_t}{2}(u - u^3 + \nabla^2 u) + \frac{\delta_t}{2}(f - f^3 + \nabla^2 f) \right\|^2, \tag{4}$$

where f represents the system at a time t and u represents the system at a time $t + \delta_t$. The numerical version of this problem will involve second-order operators acting in H_2^2. When this functional is minimized sufficiently, the system is considered to have evolved to the next time step. We will follow this approach with the modification that we consider the minimization of a functional

$$\phi(u) = ||u - f + \delta_t(u - u^3 + \nabla^2 u)||^2 \tag{5}$$

for a fully implicit scheme.

2.1. Numerical problem in one-dimension and related functional

Consider the problem

$$u_t = u - u^3 + \kappa \nabla^2 u \tag{6}$$

on the interval [0,10] with Dirichlet boundary conditions.

We will first consider a one-dimensional numerical problem, the theory for the two and three-dimensional cases is similar. We work with a finite-dimensional vector $u \in R^M$ on a uniform rectangular grid. We will denote by L_2 or H_0^2 the vector space R^M equipped with the usual inner product $\langle u, v\rangle = \sum_i u(i)v(i)$. The operators $D_0, D_1, D_{11} : R^M \to R^{M-2}$ are defined by

$$D_0(u)(i) = u(i+1), \tag{7}$$

$$D_1(u)(i) = \frac{u(i+2) - u(i)}{2\delta_x}, \tag{8}$$

$$D_{11}(u)(i) = \frac{(u(i+2) - 2u(i+1) + u(i))}{\delta_x^2} \tag{9}$$

for $i = 1, 2, \ldots, M-2$ and where $\delta_x = 10/(M-1)$ is the spacing between the nodes. D_0 just picks up the points in the grid which are not on the endpoints. D_1 and D_{11} are standard central difference formulas for estimating the first and second derivatives. The choice of difference formula is not central to the theoretical development in this paper, other choices would also work.

The numerical version of the problem of evolving from one time t to a time $t + \delta_t$ is to solve

$$D_0((1 - \delta_t)u + \delta_t u^3 - f) - \delta_t D_{11}(u) = 0, \tag{10}$$

where f in the equation is u at the previous time and u is the u desired at the next time level. We can put the solution of this problem another in terms of minimizing a functional via steepest descent. Define $L \in R^{M-2}$ by

$$L(u) = D_0((1 - \delta_t)u + \delta_t u^3 - f) - \delta_t D_{11}(u) \tag{11}$$

which is zero when the we have the desired u. The functional

$$F(u) = \langle L(u), L(u)\rangle/2 \tag{12}$$

has a minimum of zero when $L(u)$ is zero so we will look for the minimum of this functional.

2.2. Gradients and minimization

The gradient $\nabla F(u) \in R^M$ of a functional $F(u)$ in L_2 is found by solving

$$F(u + h) = F(u) + \langle \nabla F(u), h\rangle + O(h^2) \tag{13}$$

for test functions h. The gradient points in the direction of greatest increase of the functional. The direction of greatest decrease of the functional is $-\nabla F(u)$. This is the basis of steepest descent algorithms.

One can reduce $F(u)$ by replacing an initial u with $u - \lambda\nabla F(u)$ where the step size λ is a positive number. This can be done repeatedly until either $F(u)$ or $\nabla F(u)$ is less than some specified tolerance. We desire a finite-dimensional analogue to the original problem in which there are fixed values of u on the endpoints of the interval. So, we use a projection $\pi : R^M \to R^M$ which projects vectors in R^M onto the subspace in which the first and last entries of vectors are zero. Rather than using $\nabla F(u)$, we will use $\pi\nabla F(u)$. In this particular case,

$$\pi\nabla F(u) = \pi[(1 - \delta_t + 3\delta_t u^2)D_0^t L(u) - \delta_t D_{11}^t L(u)] \tag{14}$$

gives the desired gradient for steepest descent in L_2. The CFL condition [16] implies a problem with the steepest descent approach in L_2. When the grid is made finer, or if we go from a one-dimensional to a two-dimensional or three-dimensional version of the problem, the step size λ will have to be reduced. The Sobolev gradient technique avoids these problems.

The Sobolev gradient approach to the problem of minimizing functionals is to do the minimization in Sobolev spaces which are appropriate to the problem. We define two such spaces in which the minimization can be compared to minimization in L_2. We are prompted to consider the space H_2^2 which is R^M with the inner product

$$(u, v) = \langle D_0(u), D_0(v)\rangle + \langle D_1(u), D_1(v)\rangle + \langle D_{11}(u), D_{11}(v)\rangle \tag{15}$$

because $L(u)$ and $F(u)$ have D_{11} in them. We also define $\widehat{H}_2^2$ as R^M with the inner product

$$(u, v) = (1 - \delta_t)^2\langle D_0(u), D_0(v)\rangle + \langle D_1(u), D_1(v)\rangle + (\delta_t)^2\langle D_{11}(u), D_{11}(v)\rangle \tag{16}$$

because this takes into account the coefficients of D_1 and D_0 in $L(u)$ and $F(u)$.

The desired Sobolev gradients $\pi P\nabla F(u)$ in H_2^2 and $\widehat{H}_2^2$ are found by solving

$$\pi(D_0^t D_0 + D_1^t D_1 + D_{11}^t D_{11})\pi P\nabla F(u) = \pi\nabla F(u), \tag{17}$$

$$\pi((1 - \delta_t)^2 D_0^t D_0 + D_1^t D_1 + (\delta_t)^2 D_{11}^t D_{11})\pi P\nabla F(u) = \pi\nabla F(u), \tag{18}$$

respectively.

For the two-dimensional case the corresponding operator L can be written as

$$L(u) = D_0((1 - \delta_t)u + \delta_t u^3 - f) - \delta_t D_{11}(u) - \delta_t D_{22}(u) \tag{19}$$

and we consider a square domain with edges of length 10 with Dirichlet boundary conditions. The operator $D_0 : R^{M\times N} \to R^{M-2\times N-2}$ picks out the nodes which are not on the edge of the grid. The operators $D_1, D_2, D_{11}, D_{22} : R^{M\times N} \to R^{M-2\times N-2}$ are analogous to the operators in one-dimension and use central differences to estimate derivatives in the x and y directions, replacing the nodal spacing in the x direction δ_x with the nodal spacing in the y direction δ_y when appropriate.

For the three-dimensional case the operator L is

$$L(u) = D_0((1 - \delta_t)u + \delta_t u^3 - f) - \delta_t D_{11}(u) - \delta_t D_{22}(u) - \delta_t D_{33}(u) \tag{20}$$

and we consider a cubic domain with edges of length 10 with Dirichlet boundary conditions. The operator $D_0 : R^{M\times N\times P} \to R^{M-2\times N-2\times P-2}$ picks out the nodes which are not on the edge of the grid. The operators $D_1, D_2, D_{11}, D_{22} : R^{M\times N\times P} \to R^{M-2\times N-2\times P-2}$ are analogous to the operators in one-dimension and use central differences to estimate derivatives in the x, y and z directions, replacing the nodal spacing in the x direction δ_x with δ_y and δ_z when appropriate.

Numerical experiments for solution of time evolution model A were conducted as follows. Systems of M, M^2 and M^3 nodes were set up with $u = 0.0$ on all nodes except at edge nodes where $u = 0.1$ and $u = -0.1$, respectively in the one-dimensional case, $u = 0.1$ on the vertical edges and $u = -0.1$ on the horizontal edges of the plane in the two-dimensional case, and $u = 0.1$ on the top and bottom faces and $u = -0.1$ on the front, back, left and right faces of the cube in the three-dimensional case respectively. The internodal spacing δ was the same in each direction. The value of κ was set to 1 for all the experiments. The function u was then evolved according to the first-order implicit scheme. The updated value of u for a given time step was considered to be correct when the infinity norm of $\pi L(u)$ was less than 10^{-7}. We set $\delta_t = 0.4$ for the time increment. For the gradients in H_2^2 and $\widehat{H}_2^2$ we used the same step size regardless of the nodal spacing. The total number of minimization steps for 15 time steps, the largest value of λ that can be used and CPU time were recorded in Tables 1–3.

Table 1
Numerical results of steepest descent in $L_2, H_2^2, \widehat{H}_2^2$ using $\delta_t = 0.4$ over 15 time steps using second-order operators in the one-dimensional case.

λ			Iterations			CPUs			M
L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	
0.00000018	0.9	0.7	>261264100	754	284	>64003.4	132.9	61.35	801
–	0.9	0.7	–	765	282	–	1118.7	503.2	1601
–	0.9	0.7	–	772	281	–	10100.5	4660.7	3201
–	0.9	0.7	–	778	280	–	92207.6	32941.2	6401

Table 2
Numerical results of steepest descent in $L_2, H_2^2, \widehat{H}_2^2$ using $\delta_t = 0.4$ over 15 time steps using second-order operators in two-dimensional case.

λ			Iterations			CPUs			M
L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	
0.023	0.9	0.7	8132	1450	791	0.137	0.215	0.114	16
0.0018	0.9	0.7	104085	1628	1596	13.3	4.66	2.79	32
0.00012	0.9	0.7	1560337	789	271	940.9	112.53	50.48	64
0.000007	0.9	0.7	>1739560	764	256	>4971.4	2307.3	994.7	128

Table 3
Numerical results of steepest descent in $L_2, H_2^2, \widehat{H}_2^2$ using $\delta_t = 0.4$ over 15 time steps using second-order operators in three-dimensional case.

λ			Iterations			CPUs			M
L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	L_2	H_2^2	$\widehat{H}_2^2$	
0.22	0.9	0.4	1034	507	356	0.0254	0.174	0.105	8
0.014	0.9	0.4	16984	3227	1768	13.3	18.92	10.4	16
0.0009	0.9	0.4	264612	3958	2118	2234.14	863.37	462.2	32

From the tables we see that the results in H_2^2 are far better than L_2 and results in the space $\widehat{H}_2^2$ are the best. It is not clear to us why the number of iterations in 2 for the space $\widehat{H}_2^2$ are not monotone.

2.3. Using the associated functional

Here we suggest another approach, in order to avoid second-order operators. Once again consider the problem

$$u_t = u - u^3 + \kappa\nabla^2 u \tag{21}$$

with Dirichlet boundary conditions. The associated functional for a finite-dimensional version of the problem with discrete time steps is given by

$$G(u) = \langle D_0(u^2/2 - fu), 1\rangle + \delta_t F(u) \tag{22}$$

where

$$F(u) = \langle D_0(u^4/4 - u^2/2), 1\rangle + \frac{\kappa}{2}\sum_i \langle D_i(u), D_i(u)\rangle \tag{23}$$

and we wish to minimize the functional $G(u)$ until $\pi\nabla G(u)$ is smaller than some set tolerance. $G(u)$ has a minimum when the gradient

$$\pi\nabla G(u) = \pi\left(u - f + \delta_t u^3 - \delta_t u - \delta_t \sum_i D_i^t D_i u\right) \tag{24}$$

is equal to zero. And this might be considered the condition for finding u at the next time step.

In one-dimension, this functional becomes

$$G(u) = \langle D_0(u^2/2 - fu), 1\rangle + \delta_t\langle D_0(u^4/4 - u^2/2), 1\rangle + \delta_t\frac{\kappa}{2}\langle D_1(u), D_1(u)\rangle. \tag{25}$$

We want to minimize this functional in L_2, H_1^2, in $\widehat{H}_1^2$ and also in a new inner product space $\dot{H}_1^2$, defined via

$$(u, v) = (1 - \delta_t)\langle u, v\rangle + \kappa\delta_t\langle D_1 u, D_1 v\rangle. \tag{26}$$

We think of the M nodes as dividing up $[0,10]$ into $M-1$ subintervals. $D_0 : R^M \to R^{M-1}$ estimates u on the intervals by

$$D_0(u)(i) = \frac{1}{2}(u(i) + u(i+1)) \tag{27}$$

for $i = 1, 2, \ldots, M-1$. $D_1 : R^M \to R^{M-1}$ estimates a first derivative on the intervals by

$$D_1(u)(i) = \frac{1}{\delta_x}(u(i+1) - u(i)) \tag{28}$$

for $i = 1, 2, \ldots, M-1$ and where δ_x is the internodal spacing.

In two-dimensions the functional is

$$G(u) = \langle D_0(u^2/2 - fu), 1\rangle + \langle D_0(u^4/4 - u^2/2), 1\rangle + \frac{\kappa}{2}\langle D_1(u), D_1(u)\rangle + \frac{\kappa}{2}\langle D_2(u), D_2(u)\rangle. \tag{29}$$

We think of the nodes as dividing the grid into $(M-1)\times(N-1)$ rectangles. $D_0 : R^{M,N} \to R^{M-1,N-1}$ estimates the value of u on a rectangle by averaging the value of u on the four corners of the rectangle. $D_1 : R^{M,N} \to R^{M-1,N-1}$ estimates the derivative of u in the x direction on a rectangle by taking a first-order difference in the x direction along two edges of the rectangle (analogous to the one-dimensional case) and averaging. $D_2 : R^{M,N} \to R^{M-1,N-1}$ does the same in the y direction.

The three-dimensional version of the problem is given by

$$G(u) = \langle D_0(u^2/2 - fu), 1\rangle + \langle D_0(u^4/4 - u^2/2), 1\rangle + \frac{\kappa}{2}\langle D_1(u), D_1(u)\rangle + \frac{\kappa}{2}\langle D_2(u), D_2(u)\rangle + \frac{\kappa}{2}\langle D_3(u), D_3(u)\rangle. \tag{30}$$

We think of the $M\times N\times P$ nodes as dividing the grid into $(M-1)\times(N-1)\times(P-1)$ cubes. $D_0 : R^{M,N,P} \to R^{M-1,N-1,P-1}$ estimates the value of u on a cube by averaging the value of u on the eight corners of the cube. $D_1 : R^{M,N,P} \to R^{M-1,N-1,P-1}$ estimates the derivative of u in the x direction on a cube by taking a first-order difference in the x direction along four edges of the cube (analogous to the one-dimensional case) and averaging. $D_2, D_3 : R^{M,N,P} \to R^{M-1,N-1,P-1}$ do the same in the y and z directions, respectively.

Numerical experiments for solution of time evolution model A were conducted as follows. Systems of M, M^2 and M^3 nodes were set up with $u = 0.0$ on all nodes except at edge nodes where $u = 0.1$ and $u = -0.1$ respectively in the one-dimensional case, $u = 0.1$ on the vertical edges and $u = -0.1$ on the horizontal edges of the plane in the two-dimensional case, and $u = 0.1$ on the top and bottom faces and $u = -0.1$ on the front, back, left and right faces of the cube in the three-dimensional case respectively. And δ was the internodal spacing in each direction. The value of κ was set to 1 for all the experiments. All the systems were evolved according to the first-order implicit scheme. The updated value of u for a given time step was considered to be correct when the infinity norm of $\pi G(u)$ was less than 10^{-7}. We set $\delta_t = 0.4$ for the time increment. For the gradients in $H_1^2, \widehat{H}_1^2$ and also in $\dot{H}_1^2$ we used the same step size regardless of the nodal spacing. But the gradient in the Sobolev space H_1^2 does not work in the three-dimensional case therefore we omit its results. Newton's method for the solution of nonlinear system of equations is used and the results are compared with Sobolev gradient technique in one and two-dimensional cases. The total number of minimization steps for 15 time steps, the largest value of λ that can be used and CPU time are given in Tables 4–6.

We note that the finer the spacing the less CPU time the Sobolev gradient technique uses in comparison to the usual steepest descent method. The step size for minimization in L_2 has to decrease as the spacing is refined. From the tables

Table 4
Numerical results of Newton's method compared with steepest descent in $L_2, H_1^2, \widehat{H}_1^2, \dot{H}_1^2$ using $\delta_t = 0.4$ over 15 time steps using the associated functional in the one-dimensional case.

λ				Iterations					CPUs					M
L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	Newton	L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	Newton	
0.00000075	0.9	0.9	0.78	>24700873	277	277	141	557	>50880	529.9	427.17	228.89	176.337	12801
–	0.9	0.9	0.78	–	278	278	141	1037	–	1717.03	1235.3	639.48	497.13	19201
–	0.9	0.9	0.78	–	278	278	142	1711	–	3804.7	2505.8	1286.14	1230.7	25601
–	0.9	0.9	0.78	–	280	280	142	3847	–	12241.1	7990.76	4155.47	6581.56	38401
–	0.9	0.9	0.78	–	279	279	142	6789	–	27314.8	20696.4	10189.6	21261.3	51201

Table 5
Numerical results of Newton's method compared with steepest descent in $L_2, H_1^2, \widehat{H}_1^2, \dot{H}_1^2$ using $\delta_t = 0.4$ over 15 time steps using the associated functional in the two-dimensional case.

λ				Iterations					CPUs					M
L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	Newton	L_2	H_1^2	$\widehat{H}_1^2$	$\dot{H}_1^2$	Newton	
0.0075	0.35	0.9	0.78	22913	776	268	136	53	22.6	28.13	10.21	5.44	3.34	128
0.0018	0.35	0.9	0.78	95642	781	267	135	53	917.03	641.02	235.7	126.8	96.2	256
0.00085	0.35	0.9	0.78	202593	780	270	134	53	6068.7	3003.9	928.96	478.24	373.28	384
0.00046	0.35	0.9	0.78	374402	784	275	149	53	32073.7	9220.5	2553.4	1517.96	1942.07	512

Table 6
Numerical results of steepest descent in $L_2, \widehat{H}_1^2, \dot{H}_1^2$ using $\delta_t = 0.4$ over 15 time steps using the associated functional in the three-dimensional case.

λ			Iterations			CPUs			M
L_2	$\widehat{H}_1^2$	$\dot{H}_1^2$	L_2	$\widehat{H}_1^2$	$\dot{H}_1^2$	L_2	$\widehat{H}_1^2$	$\dot{H}_1^2$	
0.95	0.9	0.7	266	253	173	0.0062	0.056	0.018	8
0.42	0.9	0.7	411	273	165	0.088	0.34	0.245	16
0.12	0.9	0.7	1563	275	164	6.18	13.34	8.5	32
0.031	0.9	0.7	6308	275	164	565.42	525.7	349.9	64

one can see that the results in $\widehat{H}_1^2$ are far better than L_2 and results in the space $\dot{H}_1^2$ are better than the Newton's method in the one and two-dimensional cases.

3. Summary and conclusions

We have presented minimization schemes for the time evolution model A Ginzberg–Landau functionals based on the Sobolev gradient technique [2,17]. The Sobolev gradient technique is computationally more efficient than the usual steepest descent method as the spacing of the numerical grid is made finer and accordingly, the dimension of the problem is increased. We have also found the Sobolev gradient technique more efficient than Newton's method for this problem. Our results indicate that Sobolev gradient techniques may offer distinct advantages in certain cases. Upon choosing an optimal inner product with respect to which the Sobolev gradient works better, one can improve the performance of Sobolev gradient techniques. What the absolutely optimal inner product might be is still an open question and it is possible that different inner products might not make large differences in computational performance in all cases.

Acknowledgments

We acknowledge the enabling role of the Higher Education Commission Islamabad, Pakistan, and appreciate its financial support through the *Indigenous PhD 5000 Fellowship Program Batch-I*.

References

[1] S. Sial, J. Neuberger, T. Lookman, A. Saxena, Energy minimization using Sobolev gradients: application to phase separation and ordering, J. Comput. Phys. 189 (2003) 88–97.
[2] J.W. Neuberger, Sobolev Gradients and Differential Equations, Springer Lecture Notes in Mathematics, 1670, Springer-Verlag, New York, 1997.
[3] W.T. Mahavier, A numerical method utilizing weighted Sobolev descent to solve singular differential equations, Nonlinear World 4 (1997) 4.
[4] C. Beasley, Finite Element Solution to Nonlinear Partial Differential Equations, Ph.D. Thesis, University of North Texas, Denton, TX, 1981.
[5] S. Sial, J. Neuberger, T. Lookman, A. Saxena, Energy minimization using Sobolev gradients finite element setting, in: Chaudhary, Bhatti (Eds.), Proceedings of the World Conference on 21st Century Mathematics, Lahore, Pakistan, March 2005.
[6] S. Sial, Sobolev gradient algorithm for minimum energy states of s-wave superconductors: finite element setting, Supercond. Sci. Technol. 18 (2005) 675–677.
[7] J. Karatson, L. Loczi, Sobolev gradient preconditioning for the electrostatic potential equation, Comput. Math. Appl. 50 (2005) 1093–1104.
[8] J. Karatson, I. Farago, Preconditioning operators and Sobolev gradients for nonlinear elliptic problems, Comput. Math. Appl. 50 (2005) 1077–1092.
[9] J. Karatson, Constructive Sobolev gradient preconditioning for semilinear elliptic systems, Electron. J. Differ. Equations 75 (2004) 1–26.
[10] J. Garcia-Ripoll, V. Konotop, B. Malomed, V. Perez-Garcia, A quasi-local Gross–Pitaevskii equation for attractive Bose–Einstein condensates, Math. Comput. Simulat. 62 (2003) 21–30.
[11] B. Brown, M. Jais, I. Knowles, A variational approach to an elastic inverse problem, Inverse Probl. 21 (2005) 1953–1973.
[12] I. Knowles, A. Yan, On the recovery of transport parameters in groundwater modelling, J. Comput. Appl. Math. 171 (2004) 277–290.
[13] P.C. Hohenberg, B.I. Halperin, Theory of dynamic critical phenomena, Rev. Mod. Phys. 49 (1977) 435–479.
[14] T.M. Rogers, K.E. Elder, R.C. Desai, Numerical study of the late stages of spinodal decomposition, Phys. Rev. B 37 (1988) 9638–9649.
[15] J.W. Cahn, J.E. Hilliard, Free energy of a nonuniform system. I: Interfacial free energy, J. Chem. Phys. 28 (1958) 258.
[16] R. Courant, K.O. Friedrichs, H. Lewy, Uber die Partiellen Differenzengleichungen der Mathematisches Physik, Math. Ann. 100 (1928) 32–74.
[17] R.A. Adams, Sobolev Spaces, Academic Press, New York, 1975.