Journal of Computational and Applied Mathematics 236 (2012) 3186–3197

Contents lists available at SciVerse ScienceDirect

Journal of Computational and Applied Mathematics

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

Finite element simulations of window Josephson junctions

Manolis Vavalis [a,b,*], Mo Mu [c], Giorgos Sarailidis [a,b]

[a] Department of Computer & Communication Engineering, University of Thessaly, Glavani Str 37, GR 382 21 Volos, Greece

[b] Mechatronics Institute, CERETETH, Technology Park of Thessaly, 1st Industrial Area GR 385 00, Volos, Greece

[c] Department of Mathematics,The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong

ARTICLE INFO

Article history:
Received 7 August 2009
Received in revised form 6 February 2012

MSC:
65N35
65N05
65F10

Keywords:
Window Josephson junctions
Superconductivity
Sine-Gordon
Composite PDE
Finite element
Interface relaxation

ABSTRACT

This paper deals with the numerical simulation of the steady state two dimensional window Josephson junctions by finite element method. The model is represented by a sine-Gordon type composite PDE problem. Convergence and error analysis of the finite element approximation for this semilinear problem are presented. An efficient and reliable Newton-preconditioned conjugate gradient algorithm is proposed to solve the resulting nonlinear discrete system. Regular solution branches are computed using a simple continuation scheme. Numerical results associated with interesting physical phenomena are reported. Interface relaxation methods, which by taking advantage of special properties of the composite PDE, can further reduce the overall computational cost are proposed. The implementation and the associated numerical experiments of a particular interface relaxation scheme are also presented and discussed.

1. Introduction

A Josephson junction [1] consists of two thin superconducting films separated by an isolating oxide barrier which allows quantum superconducting tunneling. Josephson junctions have many important electro-magnetic properties. The applications, modeling and simulation of Josephson junctions have been extensively studied (see [1–4] and the references cited therein). The dynamical behavior of Josephson junctions can be effectively modeled by a hyperbolic Partial Differential Equation (PDE) which admits soliton solutions. The proper initial state for the time dependent problem relies on the study of the associated steady state problem which is commonly modeled as a boundary value problem of the two-dimensional elliptic equation given below, for any real parameter c, in dimensionless variables.

$$\frac{\theta^2 u}{\partial x^2} + \frac{\theta^2 u}{\partial y^2} = c \sin u. \tag{1}$$

Although this equation, which is known as the sine-Gordon equation, is not a soliton equation, it has many similar properties and plays an important role in problems in statistical physics, condensed matter physics and elsewhere. Besides superconductors it also models other real life problems like spin waves in ferromagnetism, etc. In addition to its practical importance the above equation is also considered very interesting from the purely mathematical view point [5].

This paper considers the numerical simulation of a new type of Josephson junction design, called the *two dimensional window junction* [6]. As depicted in Fig. 1, such a device consists of two nested regions: a *window region* as a usual Josephson junction and a surrounding *idle region* where superconducting tunneling is not allowed and which is made by means of

* Corresponding author at: Department of Computer & Communication Engineering, University of Thessaly, Glavani Str 37, GR 382 21 Volos, Greece.
E-mail addresses: mav@inf.uth.gr (M. Vavalis), mamu@ust.hk (M. Mu), gesareli@inf.uth.gr (G. Sarailidis).

0377-0427/$ – see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2012.02.017

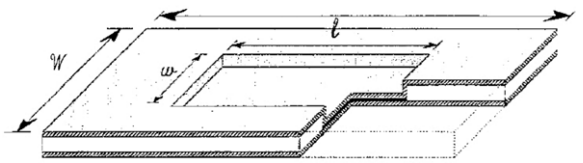

Fig. 1. A window Josephson junction.

a much thicker isolating barrier. By making the idle region much larger than the window region, one can overcome the strong impedance mismatch between the junction and the vacuum for an isolated junction device, and thus improve certain important physical properties.

Now let us denote by Ω and Ω_w the horizontal two-dimensional spatial domains for the global device and the window region, respectively. The mathematical model for the steady state recently proposed and analyzed in [6–9] is described by a composite PDE problem, where the window region is governed by the well known semi-linear sine-Gordon equation while the idle region is modeled by the Poisson equation, and the global solution satisfies certain continuity conditions on the interface of two regions, i.e., $\partial\Omega_w$. More specifically, the phase difference $u(x, y)$ of the order parameters in the two superconductor films is governed by:

$$\begin{cases} \nabla\left(\dfrac{1}{L(x,y)}\nabla u\right) = \dfrac{1}{\lambda^2 L(x,y)}\sin u, \quad \text{in } \Omega_w, \\ \nabla\left(\dfrac{1}{L(x,y)}\nabla u\right) = 0, \quad \text{in } \Omega/\Omega_w, \\ \dfrac{\partial u}{\partial \nu} = g \quad \text{on } \partial\Omega, \\ u \text{ and } \dfrac{1}{L(x,y)}\dfrac{\partial u}{\partial \nu} \text{ are continuous on } \partial\Omega_w, \end{cases} \tag{2}$$

where $L(x, y)$ represents the surface inductance such that $L(x, y) \geq c > 0$, λ is the Josephson penetration length, and $\frac{\partial u}{\partial \nu}$ denotes the outer normal derivative. Usually, $L(x, y)$ assumes different constants inside and outside Ω_w. This model can also be generalized to devices of different geometric window shapes and of more than one windows (e.g. [10]). To simplify the discussion we may consider that our model consists of $\Omega = (-M, M) \times (-N, N)$ and $\Omega_w = (-m, m) \times (-n, n)$ where M, N, m, n are positive numbers such that $m < M$ and $n < N$.

Readers are referred to [6,8] for details of the modeling and the PDE analysis. It is shown that the model properly captures most of the important physical properties of the window junctions. Specifically, in [8] we establish the existence of solutions and we obtain regularity and a priori estimates for the derivatives of the solution. These estimates are independent of the window geometric characteristics and as such they provide the necessary background for the finite element analysis and at the same time they validate several practical observations.

Finite difference methods has already been applied for the numerical solution of (2) in [6,10,7] where difficulties in handling the interface and boundary conditions properly have been observed. The convergence and error analysis of the finite difference discretizations are also challenging. The finite element method has been applied to solve the time dependent hyperbolic problem without the window junction [11] and the numerical results were promising. The nonlinearity in the time dependent problem was well treated, and in fact easily, by using proper time discretization techniques. It is, therefore, numerically more difficult for the steady state problem.

In this paper we propose in Section 2 a finite element method for the steady state model of window Josephson junctions, that is suitable for handling the interface and boundary conditions. Convergence and error analysis are also presented for the finite element approximation applied to this nonlinear problem. Issues in solving the resulting nonlinear discrete system are addressed in detail in Section 3 where a Newton-Preconditioned Conjugate Gradient (PCG) algorithm is proposed too. Numerical results are presented for the computation of nonsingular solution branches and interesting physical phenomena, as well as the convergence behavior of the algorithm are easily observed. In Section 4, interface relaxation methods are proposed to further reduce the overall computational cost. Specifically, we take advantage of the special form of the PDE in the large idle region to avoid the unnecessary extra computational cost due to non-linearity.

A description of our implementation of an interface relaxation method, which we consider the most appropriate for our model, is also presented in this section. It is coupled with an associated set of results obtained through numerical experimentation. Our concluding remarks are given in Section 5.

2. Finite element approximation

From the numerical computation point of view, it is more convenient to use the weak form of (2) and apply the finite element discretization. The interface and boundary conditions as well as the possibly discontinuous coefficients are all treated elegantly by the weak formulation.

Denote by $H^m(\Omega)$ the conventional Sobolev spaces. Let $X = H^1(\Omega)$ and $Y = [H^1(\Omega)]'$, and denote by $B(Y; X)$ the Banach space of bounded linear operators from Y to X. The weak formulation of (2) reads as:

Find $u \in X$, such that

$$a(u, v) = F_g(v), \quad \forall v \in X, \tag{3}$$

where

$$a(u, v) = \int_\Omega \frac{1}{L(x, y)} \left\{ \nabla u \cdot \nabla v + \frac{1}{\lambda^2} \mathcal{I}(x, y) \sin(u) v \right\} dxdy \tag{4}$$

and

$$F_g(v) = \oint_{\partial\Omega} g(x, y) v ds. \tag{5}$$

In (4), $\mathcal{I}(x, y)$ is a characteristic function being 1 in Ω_w and 0 otherwise.

The solution of (3)–(5) is not unique. If u^* is a solution, $u^* + 2k\pi$ is also a solution for any integer k because $\nabla(u^* + 2k\pi) = \nabla(u^*)$ and $\sin(u^* + 2k\pi) = \sin(u^*)$. The non-uniqueness is not simply up to a constant, as this will be seen later in (31).

It is also important to mention that if u_1 and u_2 are two solutions, we have

$$\int_\Omega \frac{1}{L(x, y)} \left(\nabla(u_1 - u_2) \cdot \nabla v + \frac{1}{\lambda^2} \mathcal{I}(x, y)(\sin u_1 - \sin u_2) v \right) dxdy = 0.$$

Taking $v = u_1 - u_2$ and noticing that $|\sin u_1 - \sin u_2| \le |u_1 - u_2|$, we have the (weighted) L_2-estimate:

$$\| \nabla (u_1 - u_2) \|_\Omega \le \frac{1}{\lambda^2} \| (u_1 - u_2) \|_{\Omega_w},$$

where $\|u\|_\Omega \equiv (\int_\Omega \frac{1}{L(x,y)} u^2 dxdy)^{1/2}$. This indicates that the difference between two solutions is close to a constant, specially when λ is large or the window region is small as in the physically interesting regime.

From the computational point of view, we focus in this paper on computing solution branches $u(\lambda)$ away from bifurcation points, the so called branches of nonsingular solutions, for the nonlinear problem (3)–(5). Namely, we assume that, for a certain interval $\lambda \in [\lambda_0, \lambda_1]$ with $\lambda_0 > 0$ and starting with a solution $u(\lambda_0)$, there exists a unique solution branch such that $u(\lambda)$ moves along this branch continuously with the parameter λ. The precise definition will be given later on. We now outline in the operator form the basic idea of a general approach of discretization and error analysis for nonlinear problems which we will utilize later in our study.

Consider a type of nonlinear problems of the form:

$$(\mathcal{F} + \mathcal{G})u = 0, \tag{6}$$

where the linear operator $\mathcal{F}$ corresponds to a uniquely solvable linear problem, and $\mathcal{G}$ denotes the remaining nonlinear part. Let $\mathcal{T}$ denote the inverse operator $\mathcal{F}^{-1}$, then (6) can be written as

$$u + \mathcal{T}\mathcal{G}u = 0. \tag{7}$$

It is natural to apply existing and well understood knowledge of linear problems to the generalized nonlinear case. Let $\mathcal{F}_h$ denote the standard finite element discretization of $\mathcal{F}$ and $\mathcal{T}_h$ denote $\mathcal{F}_h^{-1}$. One can formally discretize (7) by

$$u_h + \mathcal{T}_h \mathcal{G} u_h = 0. \tag{8}$$

The error estimates can then be derived from the linear part $\|\mathcal{T}_h - \mathcal{T}\|$ and certain smoothness properties of the nonlinear operator $\mathcal{G}$. For this purpose, a general convergence theory is available in [12] and has been applied to the Navier–Stokes equations, the Von Karman equations [12], and the Ginzburg–Landau equations [13]. Here we show how to apply this framework to the sine-Gordon type equation (3)–(5).

Introduce a bilinear form $b(u, v)$ defined by

$$b(u, v) = \int_\Omega \left(\frac{1}{L(x, y)} \nabla u \cdot \nabla v + uv \right) dxdy. \tag{9}$$

The linear operator $\mathcal{T} \in B(Y; X)$ is defined as: for $\rho \in Y$, $\eta \equiv \mathcal{T}\rho \in X$ such that

$$b(\eta, v) = \rho(v) + F_g(v), \quad \forall v \in X. \tag{10}$$

It is easy to see that the solution operator $\mathcal{T}$ is well defined and (10) corresponds to the linear problem

$$\begin{cases} -\nabla \left(\dfrac{1}{L(x, y)} \nabla \eta \right) + \eta = \rho, & \text{in } \Omega \\ \dfrac{\partial \eta}{\partial \nu} = g(x, y), & \text{on } \partial\Omega, \end{cases} \tag{11}$$

with continuity conditions on the interface $\partial\Omega_w$.

Let $V_h \subset X$ be a finite element subspace, from the standard finite element analysis for the linear operator $\mathcal{T}$, we have

Lemma 1. *For any $\rho \in Y$,*

$$\|(\mathcal{T} - \mathcal{T}_h)\rho\|_X \to 0, \quad \text{as } h \to 0. \tag{12}$$

Furthermore, if $\eta = \mathcal{T}\rho \in H^{m+1}(\Omega)$ and if the finite element spaces V_h satisfy the standard approximation condition:

$$\|v - \Pi_h v\|_X \le Ch^m \|v\|_{H^{m+1}(\Omega)}, \quad \forall v \in H^{m+1}(\Omega),$$

where Π_h denotes the interpolation operator in V_h, then we have the error estimate:

$$\|(\mathcal{T} - \mathcal{T}_h)\rho\|_X \le Ch^m \|\eta\|_{H^{m+1}(\Omega)}. \tag{13}$$

Corresponding to the splitting of (6), we rewrite (3) as

$$b(u, v) = b(u, v) - a(u, v) + F_g(v), \quad \forall v \in X. \tag{14}$$

Define the nonlinear operator $\mathcal{G}[\lambda, u] : \Lambda \times X \to Y$ as

$$\begin{aligned}\mathcal{G}[\lambda, u](v) &= a(u, v) - b(u, v)\\ &= \int_\Omega \left(\frac{\mathcal{I}(x, y)}{\lambda^2 L(x, y)} \sin u - u\right) v dx dy, \quad \forall v \in X.\end{aligned} \tag{15}$$

Therefore, (14) can be written in operator form as

$$u = -\mathcal{T}\mathcal{G}[\lambda, u]. \tag{16}$$

In other words, Eq. (3) is equivalent, in the operator form, to the following fixed-point problem

$$F(\lambda, u) \equiv u + \mathcal{T}\mathcal{G}[\lambda, u] = 0. \tag{17}$$

It is easy to verify that the discretization (8) is equivalent to the formal Galerkin approximation procedure defined as: find $u_h \in V_h$, such that

$$a(u_h, v_h) = F_g(v_h), \quad \forall v_h \in V_h. \tag{18}$$

A branch of nonsingular solutions $u(\lambda)$ of (17) is defined as satisfying the following properties:

$$\lambda \to u(\lambda) \text{ is a continuous function from } \Lambda \text{ into } X; \tag{19}$$

$$F(\lambda, u(\lambda)) = 0; \tag{20}$$

$$D_u F(\lambda, u(\lambda)) \text{ is an isomorphism of } X, \tag{21}$$

where $D_u F$ is the Frechet derivative with respect to the second argument of $F(\lambda, u)$. The above discussion leads us to the following main theorem.

Theorem 2. *There exist a neighborhood $\mathcal{O}$ of the origin in X and, for $h \le h_0$ sufficiently small, a unique C^2 function $\lambda \in \Lambda \to u_h(\lambda) \in V_h$ such that for all $\lambda \in \Lambda$, where $\Lambda = [\lambda_0, \lambda_1]$ and $\lambda_1 > \lambda_0 > 0$,*

$$u_h + \mathcal{T}_h \mathcal{G}[\lambda, u_h(\lambda)] = 0, \quad u_h(\lambda) - u(\lambda) \in \mathcal{O}. \tag{22}$$

Furthermore, we have for some constant C independent of h and λ, that

$$\|u_h(\lambda) - u(\lambda)\|_{H^1(\Omega)} \le Ch^m \|u(\lambda)\|_{H^{m+1}(\Omega)}. \tag{23}$$

Proof. From (15), we see that the nonlinear operator $\mathcal{G}[\lambda, u]$ can be expressed as

$$\mathcal{G}[\lambda, u] = \frac{\mathcal{I}(x, y)}{\lambda^2 L(x, y)} \sin u - u. \tag{24}$$

Thus, the first and second Frechet derivatives of $\mathcal{G}[\lambda, u]$ with respect to the second argument u can be calculated by

$$D_u \mathcal{G}[\lambda, u] = \frac{\mathcal{I}(x, y)}{\lambda^2 L(x, y)} \cos u - 1, \tag{25}$$

and

$$D_{uu} \mathcal{G}[\lambda, u] = \frac{\mathcal{I}(x, y)}{\lambda^2 L(x, y)} (-\sin u). \tag{26}$$

By observing that $L(x, y) \ge c > 0$, $|\mathcal{I}(x, y)| \le 1$, $|\sin u| \le 1$, and $|\cos u| \le 1$, it follows that both $D_u \mathcal{G}$ and $D_{uu} \mathcal{G}$ are bounded operators on all bounded subsets of $\Lambda \times H^1(\Omega)$. Therefore, applying a general convergence theorem for nonlinear operators in Banach spaces [12, Theorem 6], we obtain (22) and the following estimate

$$\|u_h(\lambda) - u(\lambda)\|_X \le C \left(\|u(\lambda) - \Pi_h u(\lambda)\|_X + \|(\mathcal{T}_h - \mathcal{T})\mathcal{G}(\lambda, u(\lambda))\|_X\right). \tag{27}$$

(23) then follows from (27) and Lemma 1. □

3. Solution of the nonlinear system

As it is common to nonlinear elliptic problems, there are several computational issues to be addressed with the ones related to path following and the analysis of bifurcation points being the most important from both the computational cost and the practical impact view points. Initial efforts to trace all parts of the bifurcation diagram of the window Josephson junction model considered in the present study using legacy libraries [14] show promising prospects [8,15]. A comprehensive bifurcation analysis using state of the art numerical algorithms and software (e.g. [16]) is currently under development. Here we apply the Newton's method to solve (18), which converges extremely rapidly provided that a good initial approximation $u^{(0)}$ is available [17]. Noticing that $D_u \sin u = \cos u$, we have the following iterative procedure:

For a given approximation solution $u_h^{(m)} \in V_h$, the next iterate $u_h^{(m+1)} \in V_h$ is defined by

$$A_m(u_h^{(m+1)}, v_h) = R(u_h^{(m)}, v_h) + F_g(v_h), \quad \forall v_h \in V_h, \tag{28}$$

where the bilinear form $A_m(u, v)$ is defined as

$$A_m(u, v) = \int_\Omega \left(\frac{1}{L(x,y)} \bigtriangledown u \bigtriangledown v + \frac{\mathcal{L}(x,y)}{\lambda^2 L(x,y)} \cos(u_h^{(m)}) uv \right) dxdy, \tag{29}$$

and

$$R(u, v) = \int_\Omega \left(\frac{\mathcal{L}(x,y)}{\lambda^2 L(x,y)} (\cos(u)u - \sin(u))v \right) dxdy. \tag{30}$$

The solvability of (28) depends on the sign of $\cos(u_h^{(m)})$. If for a given spacing h and at some step (28) becomes singular or too ill-conditioned, Newton iteration may break down. In this case, we propose to replace $\cos(u_h^{(m)})$ on both sides of (28) by a cut-off function $c(x, y)$:

$$c(x,y) \equiv \begin{cases} \cos(u_h^{(m)}) & \text{if } \cos(u_h^{(m)}) > 0 \\ 0 & \text{otherwise.} \end{cases}$$

This modified Newton's scheme, on one hand, retains the information of the Jacobian as much as possible, and on the other hand, imposes the positiveness on the linear system.

The efficiency for solving the Jacobian problem (28) at each iteration is also crucial. For this purpose, we apply the substructuring preconditioning [18] technique with some modification to the conjugate gradient linear solver. Notice that $L(x, y)$ usually takes different constants in the window and idle regions and the fact that $0 \le c(x, y) \le 1$. From Poincare's inequality, we have

$$\begin{aligned} c_1 \| \bigtriangledown u \|^2_{L_2(\Omega_w)} &\le \int_{\Omega_w} \left(\frac{1}{L(x,y)} \bigtriangledown u \bigtriangledown v + \frac{\mathcal{L}(x,y)}{\lambda^2 L(x,y)} c(x,y) uv \right) dxdy \\ &\le c_2 \| \bigtriangledown u \|^2_{L_2(\Omega_w)}, \quad \forall u \in H_0^1(\Omega_w). \end{aligned}$$

Thus, we can simply use a fast Poisson solver for both subdomain solvers of the preconditioner for the window and idle regions, while still retain the optimal condition number as that shown in [18]. Numerical evidence shows that the convergence behavior remains as good even when the full Newton algorithm is used, i.e., $\cos(u_h^{(m)})$ is not replaced by the cut-off function $c(x, y)$. Because the subdomain preconditioners are stationary in the Newton iteration, they can be factorized only once and used repeatedly. We also note that in the case of multiple windows, the algorithm leads to natural parallelism.

As already mentioned, for the Newton iteration to converge, a good initial guess is required. For the one-dimensional version of the sine-Gordon equation over the infinity domain, there exist a family of analytical solutions [1]

$$u = 4 \arctan(e^{\mp x/d}). \tag{31}$$

These can be used along the longer dimension of the window region as initial guesses for the Newton iteration. Once it converges to a solution $u(\lambda_0)$, the continuation technique is then applied to trace the regular branch by varying λ gradually with the previously computed solution as the initial guess for the current Newton iteration. The process breaks down when a bifurcation point is nearly reached. Then another process may also be used to compute regular branches for large value of λ. Notice that in the limit case when λ approaches infinity, the nonlinear term $\sin u$ drops off. One can compute a unique solution, up to a constant, of the corresponding linear problem. Starting with any such a solution and applying the continuation, we trace the branch by varying λ.

We now present the numerical results of solving (3)–(5) by the finite element approximation using bilinear elements. For the purpose of testing the algorithm, we consider a domain $\Omega = (-20, 20) \times (-10, 10)$ with a narrow window region $\Omega_w = (-10, 10) \times (-2, 2)$ and a homogeneous boundary condition $g = 0$. Most of solutions are computed by using a 30×30 rectangular mesh. Fig. 2 shows surface plots of a solution branch for $\lambda \in [1, 1.8]$. In this computation, we start with computing a solution for $\lambda = 1$. Based on (31) we select as initial guess the function $u^{(0)}(x, y) = 4 \arctan(e^x)$ because the window is long and narrow along the x-direction. From this initial value the Newton iteration converges rapidly in two

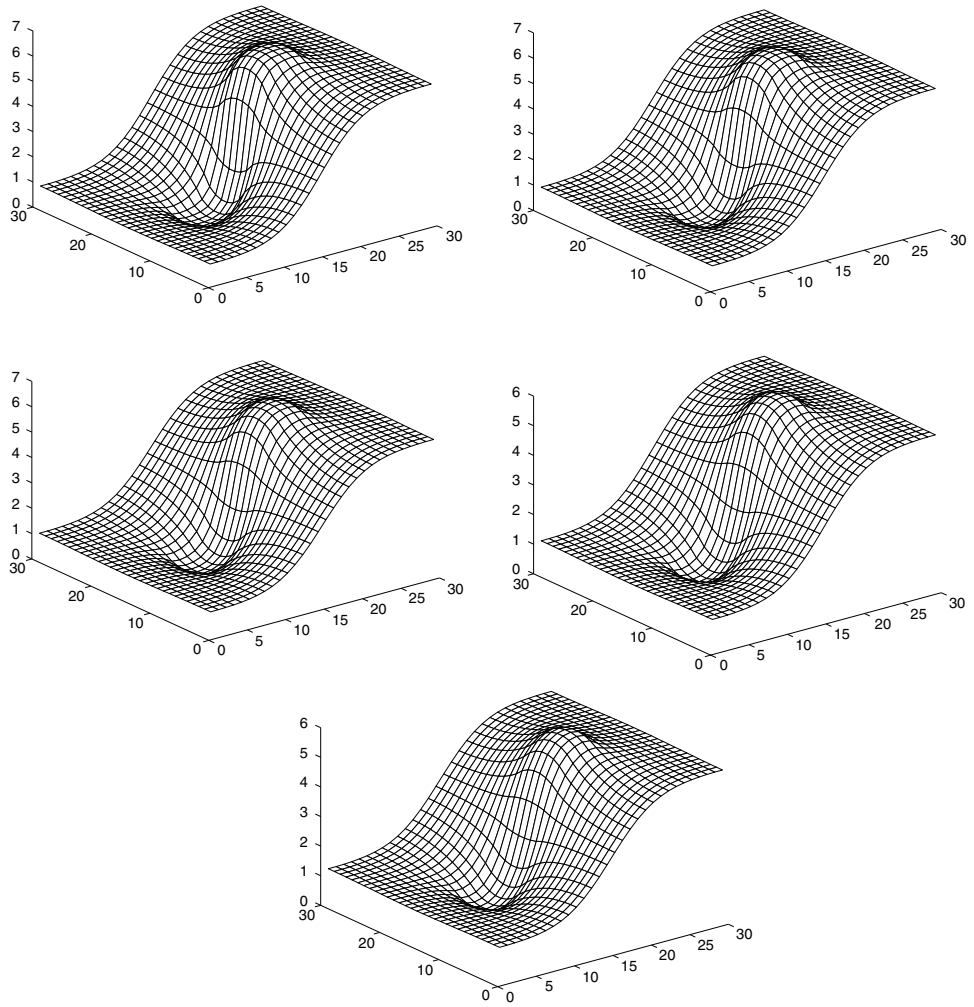

Fig. 2. Surface plots of $u(\lambda)$ for a branch of solutions with $\lambda \in [1, 1.8]$, where the subplots from top down and left right correspond to $\lambda = 1$, 1.2, 1.4, 1.6 and 1.8, respectively.

Table 1
The error history of $\|u^{(m+1)} - u^{(m)}\|_{\max}$ of the Newton iterations associated with the solutions considered in Fig. 2. m is the iteration step and "–" stands for convergence.

m	$\lambda = 1$	$\lambda = 1.2$	$\lambda = 1.4$	$\lambda = 1.6$	$\lambda = 1.8$
1	2.1308E−02	2.8257E−01	2.4793E−01	2.2753E−01	2.0543E−01
2	8.2016E−05	2.6093E−02	2.7220E−02	1.7741E−02	1.5133E−02
3	–	2.1076E−04	1.5862E−02	1.0161E−03	4.9028E−03
4		8.1062E−06	1.5187E−04	4.7445E−05	5.6167E−03
5		–	1.9288E−04	–	5.0986E−03
6			2.7537E−04		1.2714E−02
7			2.0552E−04		6.2289E−03
8			5.7220E−05		4.6082E−03
9			–		1.2372E−02

steps. Then we march along λ with step size 0.2 by continuation. When λ reaches 1.8, the Newton iteration oscillates, and for the next step $\lambda = 2.0$, the Newton iteration blows up. This indicates that a bifurcation point may be reached or a smaller marching step is needed to provide a better initial guess for the next step Newton process. The error history of the Newton iterations for this continuation process is given in Table 1. The contour plots of the solution branch are shown in Fig. 3 for better visualizing the smooth change of u as λ increases.

When the above continuation breaks down, we restart the process with $\lambda = 2$ and look for another solution branch. In this case, we use the same initial guess $u^{(0)}(x, y) = 4 \arctan(e^x)$ as before. The Newton iteration converges quickly again.

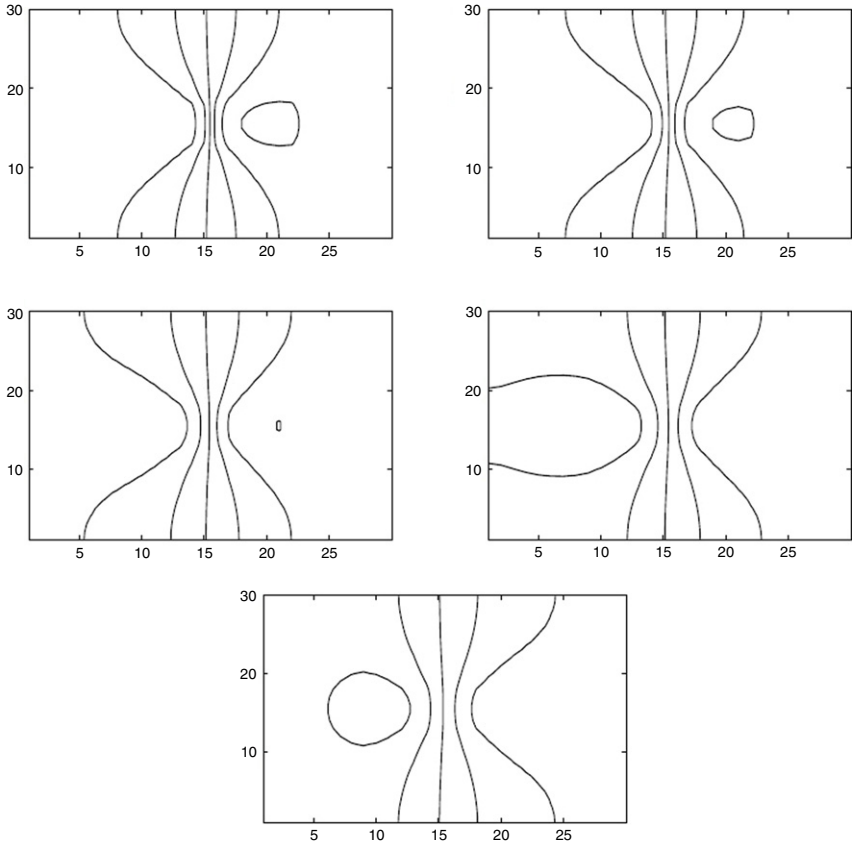

Fig. 3. Contour plots of u corresponding to Fig. 2.

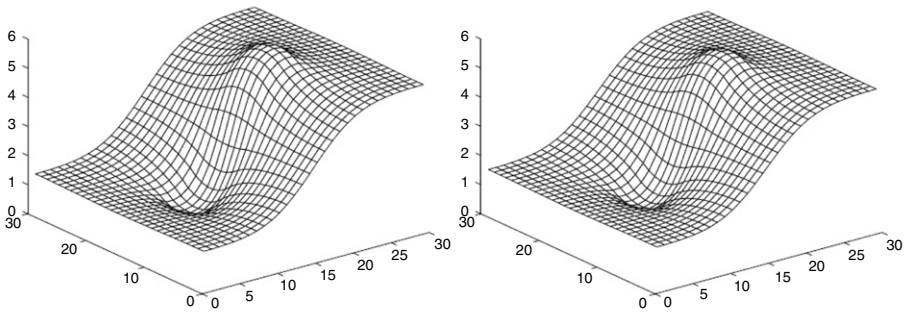

Fig. 4. Surface plots of u for a solution branch computed by restarting the continuation process with $\lambda = 2$ (left) and 2.2 (right).

For the illustration of this branch, we show in Figs. 4 and 5 the surface and contour plots for the solutions corresponding to $\lambda = 2$ and 2.2, respectively.

The model problem considered is of particular physical interest. For this we also plot in Fig. 6 the magnetic energy distribution $e_{mag}(x, y) = \frac{1}{2}((\partial u/\partial x)^2 + (\partial u/\partial y)^2)$ (left figure) and the supercurrent (right figure) which is proportional to $\mathcal{J}(x, y) \sin u(x, y)$. The separation of the maximum and minimum of the supercurrent along the x-axis is of special physical interest and importance. It represents the effective fluxon width (with a factor of 0.5) inside the junction [13,8].

In the above experiments, the surface inductance is constant with $L(x, y) = 1$ in Ω. We now consider the jump case with $L(x, y) = 2.2$ inside the window region and $L(x, y) = 1$ in the outside idle region. Specifically, Fig. 7 shows the surface plots for u (left figure) and the corresponding energy (right figure) with $\lambda = 1$. A sharper energy peak is observed inside the junction region. In addition, it is worth to point out that, as expected, the jump in $L(x, y)$ does not affect the convergence rate of the Newton iteration and the inside preconditioned conjugate gradient iteration.

Finally, we illustrate the effects of mesh refinement on the numerical behavior of the algorithm. Fig. 8 compares the computation of the same problem as for Fig. 7, but with a finer 50×50 mesh. As shown in Table 2, the Newton iteration takes more steps to converge as the mesh resolution is increased, still exhibiting rapid convergence though. In other words

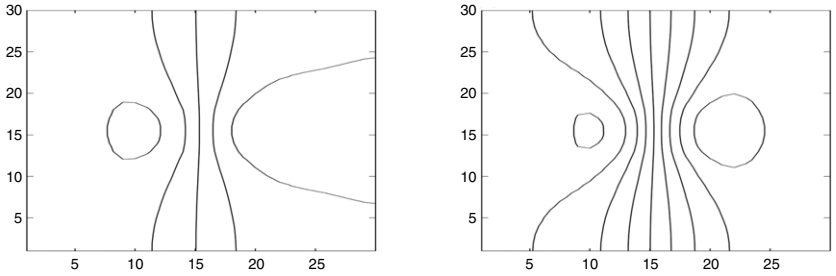

Fig. 5. Contour plots of u corresponding to Fig. 4.

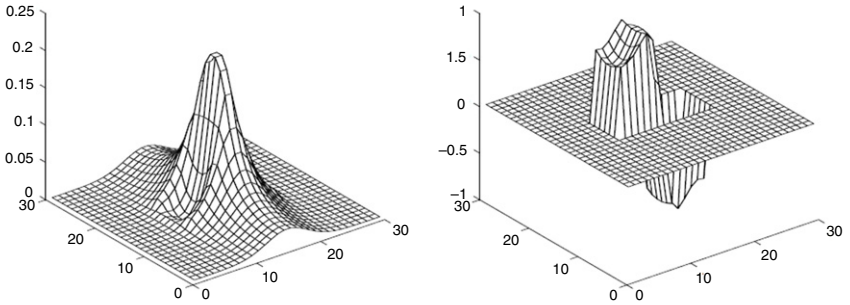

Fig. 6. Surface plots of the energy and supercurrent with $\lambda = 2.0$.

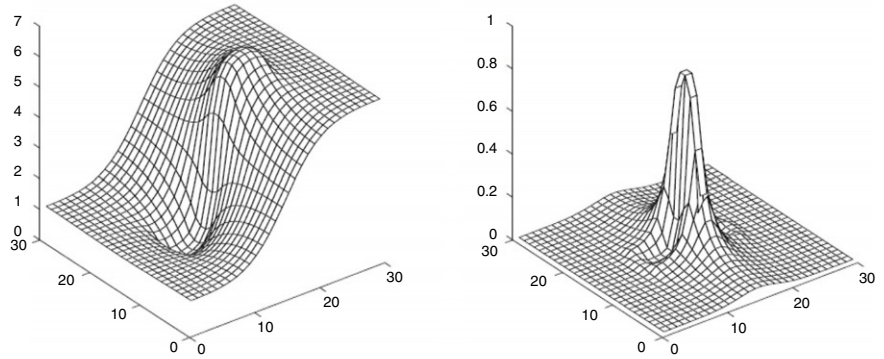

Fig. 7. Surface plots of u and the energy with $\lambda = 1$, $L(x, y) = 2.2$ in Ω_w and $L(x, y) = 1.0$ in Ω/Ω_w.

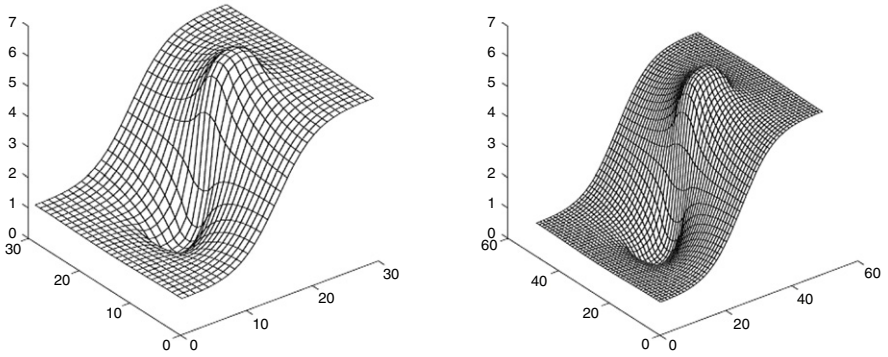

Fig. 8. Surface plots of u with a 30×30 mesh (left figure) and a 50×50 mesh (right figure).

the convergence rate of the preconditioned conjugate gradient iteration changes very little, which verifies our theoretical prediction that the condition number is almost independent of the spacing h.

Table 2
The error history of $\|u^{(m+1)} - u^{(m)}\|_{\max}$ of the Newton iterations in the case of discontinuous surface inductance. m is the iteration step and "–" stands for convergence.

m	30×30 mesh	50×50 mesh
1	7.7123E−02	8.2891E−02
2	1.2503E−03	1.0778E−02
3	2.7657E−05	1.5490E−03
4	–	2.1768E−04
5		1.1539E−04
6		1.6880E−04
7		9.5129E−05
8		3.3379E−05

For associated finite difference experiments on interesting physical aspects, such as non-homogeneous boundary conditions with external magnetic field and current, geometric effects, application to soliton solutions of the time dependent model, the reader is referred to [7].

We should mention here that in order to avoid difficulties in the integration to construct the stiffness matrix in our implementation the computational mesh lines up with the interface of Ω and Ω_w.

4. Domain decomposition algorithms

As we can easily observe, the nonlinear problem in our model is only imposed on a relatively very small window region. The simple Poisson equation is imposed in a much larger idle region. It is best to make use of this special property in order to avoid significant unnecessary cost due to the nonlinear iteration for the window region. We propose to use the interface relaxation approach [19,20] which is already proved to be convergent for several linear cases.

The basic idea is to solve the linear problem in the large idle region and solve the nonlinear sine-Gordon equation only in the window region by Newton iteration, and then relax the interface condition by an outer loop of relaxation iteration. Notice that at each relaxation step, we solve a Poisson equation only once in the large idle region. The Newton iteration carried out is restricted to a relatively much smaller region. Thus, if the relaxation converges fast enough, the overall computational cost could be significantly reduced.

There are several possible ways to relax the interface conditions. One is to use the alternating Dirichlet–Neumann interface conditions [19]. Namely, starting with an initial guess of u on $\partial\Omega_w$, we compute the local solution for one region, called Ω_D, with the Dirichlet data on $\partial\Omega_w$. Then the outer normal derivative of the computed local solution is calculated and passed to the other region, called Ω_N. This outer normal derivative together with the interface conditions will, in turn, be used to determine the Neumann data on $\partial\Omega_w$ to compute the local solution for Ω_N. This updates the value of u on $\partial\Omega_w$ so that the above process can be iterated. To apply this approach, we have to decide which region is taken to be Ω_N because of the non-uniqueness caused by the Neumann boundary condition. We choose the window region Ω_w as Ω_N. By doing so, the solution in the idle region is always unique because of the Dirichlet condition imposed on part of its boundary. For fixing the solution in Ω_w, we can use the boundary value of Ω_N to select a proper initial guess for the Newton iteration of the nonlinear solver of Ω_w.

An alternative relaxation scheme is to use the Robin transmission condition on $\partial\Omega_w$ for both window and idle regions [20]. In this case, the interface conditions will impose uniqueness on local solutions. The above two relaxation schemes both have fast convergence rate. However, they may diverge for problems of negative eigenvalues, which is the case when the term $\sin u$ is present.

One possibility is to propose a third approach which is based on the residual relaxation. Notice that the interface conditions in (2) can be expressed in an implicit form:

$$\alpha(u_{\text{in}} - u_{\text{out}})^2 + \beta\left(\frac{1}{L_{\text{in}}(x,y)}\frac{\partial u_{\text{in}}}{\partial \nu} + \frac{1}{L_{\text{out}}(x,y)}\frac{\partial u_{\text{in}}}{\partial \nu}\right)^2 = 0, \quad \text{on } \partial\Omega_w, \tag{32}$$

where *in* and *out* stand for inside and outside the window region, respectively, α and β are Lagrange multiplier like parameters. For linear problems, it is shown in [19,21,22] that the relaxation converges even when negative eigenvalues are present. The idea is to guess an approximation for u on $\partial\Omega_w$, solve the local PDEs in each region, update u on $\partial\Omega_w$ by the "interface" residual of (32) and then iterate until convergent. Notice that we always have $u_{\text{in}} = u_{\text{out}}$ because we impose the Dirichlet condition on $\partial\Omega_w$ for both regions. However, it seems that we cannot use non-homogeneous Dirichlet conditions on $\partial\Omega_w$ for the sine-Gordon problem. Our numerical experiments show that this leads to non-convergent Newton iterations. Unfortunately, we cannot impose non-homogeneous Neumann conditions alone on the Laplace problem since this, in general, corresponds to an ill-posed physical problem.

We therefore formulate a similar to the above interface relaxation scheme that imposes on $\partial\Omega_w$ Dirichlet boundary conditions for the Laplace problem and Neumann boundary conditions for the sine-Gordon problem. More specifically, we describe this new algorithm as follows:

Interface relaxation algorithm:

(i) Start with an initial guess $u^{(0)}$ in Ω and set $U_\Gamma^{(0)} = u^{(0)}|_{\partial\Omega_w}$ (for the Laplace problem), $\frac{\partial U_\Gamma^{(0)}}{\partial \nu} = \frac{\partial u^{(0)}}{\partial \nu}|_{\partial\Omega_w}$ (for the sine-Gordon problem) and $u_{\text{in}}^{(0)} = u^{(0)}|_{\Omega_w}$ and $k = 0$.

(ii) Set $k = k + 1$ and solve the sine-Gordon problem

$$\begin{cases} \nabla\left(\dfrac{1}{L(x,y)} \nabla u_{\text{in}}^{(k)}\right) = \dfrac{1}{\lambda^2 L(x,y)} \sin u_{\text{in}}^{(k)}, & \text{in } \Omega_w, \\ \dfrac{\partial u_{\text{in}}^{(k)}}{\partial \nu} = \dfrac{\partial U_\Gamma^{(k-1)}}{\partial \nu}, & \text{on } \partial\Omega_w. \end{cases} \tag{33}$$

By the Newton iteration with $u_{\text{in}}^{(k-1)}$ as the initial guess.

(iii) Solve the Laplace problem

$$\begin{cases} \nabla\left(\dfrac{1}{L(x,y)} \nabla u_{\text{out}}^{(k)}\right) = 0, & \text{in } \Omega/\Omega_w, \\ \dfrac{\partial u_{\text{out}}^{(k)}}{\partial \nu} = g \quad \text{on } \partial\Omega, \\ u_{\text{out}}^{(k)} = U_\Gamma^{(k-1)}, & \text{on } \partial\Omega_w. \end{cases} \tag{34}$$

(iv) Update the Neumann and Dirichlet data on the interface, used by the sine-Gordon and Laplace problems respectively, by residual relaxation as follows:

$$U_\Gamma^{(k)} = U_\Gamma^{(k-1)} + \alpha(u_{\text{in}}^{(k)} - u_{\text{out}}^{(k)}) \tag{35}$$

$$\frac{\partial U_\Gamma^{(k)}}{\partial \nu} = \frac{\partial U_\Gamma^{(k-1)}}{\partial \nu} + \beta\left(\frac{1}{L_{\text{out}(x,y)}} \frac{\partial u_{\text{out}}^{(k)}}{\partial \nu} - \frac{1}{L_{\text{in}}(x,y)} \frac{\partial u_{\text{in}}^{(k)}}{\partial \nu}\right). \tag{36}$$

(v) Iterate (ii)–(iv) until convergent.

The parameters α and β can be properly chosen to accelerate the convergence rate of the relaxation procedure.

The formal convergence analysis of the above relaxation schemes is rather complicated and is beyond the scope of this paper. Nevertheless, and in order to numerically confirm our above given claims we have implemented the above proposed interface relaxation scheme utilizing a state-of-the-art and widely used finite element library [23]. Specifically, we have implemented our algorithm in C++ fully utilizing the deal.II library by creating two separate solvers corresponding to the Laplace and sine-Gordon problems, respectively. Those were called repeatedly and alternately, by a main function, and were fed with the updated boundary conditions.

All the developed software that realizes the above described interface relaxation method, and was used to produce the results presented next, is publicly available at https://github.com/mvavalis.

4.1. Numerical experiments

For the numerical verification of the above proposed interface relaxation scheme we select the same problem we have considered in Section 3. Specifically we solve (3)–(5) with $\lambda = 1, L(x,y) = 1, \Omega = (-20, 20) \times (-10, 10)$ and $\Omega_w = (-10, 10) \times (-2, 2)$. We start our interface iterations with $u^{(0)}(x,y) = 4\arctan(e^x)$ as an initial guess.

In Fig. 9 we present the convergence history of the interface relaxation assuming no current and no magnetic field applied. Specifically on the x-axis we have the iteration (interface relaxation iteration) number k and on the y-axis the L_2 norm of the difference of two successive iterates, i.e. $\|u^{(k)} - u^{(k-1)}\|_\Omega$.

To further elucidate some of the convergence characteristics of the proposed methods, snapshots of the detailed history of convergence for the case $\alpha = 0.001$ in Fig. 9 are graphically depicted in Fig. 10 where we plot the initial guess and the computed solutions at iterations $k = 1, 2, 7, 20$ and 60. As it can be clearly seen, our interface relaxation method first cuts rapidly (in just 2–5 iterations) the high frequencies terms of the error and then spends the rest of the iterations further smoothing the solution by cutting the remaining low frequency terms. We should note here that the synthesis of the two computed solutions in each subdomain into the integrated 3D graphs found in this figure can be created in a very straight forward and natural way using any of the scientific visualization software systems available.[1]

Finally it is worth to point out that we may further reduce the cost by making use of the analytical expression for the solution of the Poisson equation over the large idle region to construct a solution method for the window region. This is currently under investigation.

[1] In our case we have used Tecplot http://www.tecplot.com.

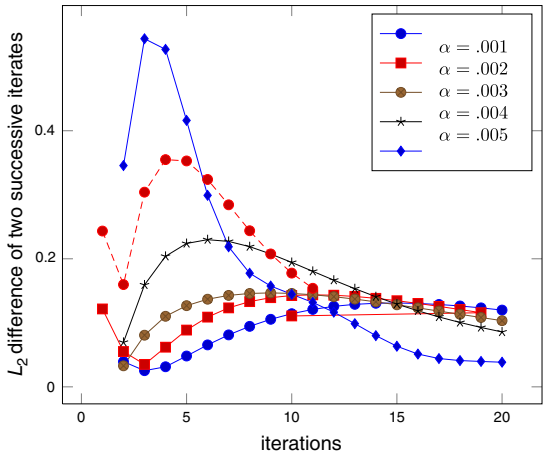

Fig. 9. Graph of the norm of the difference of two successive iterates versus the interface relaxation iterations.

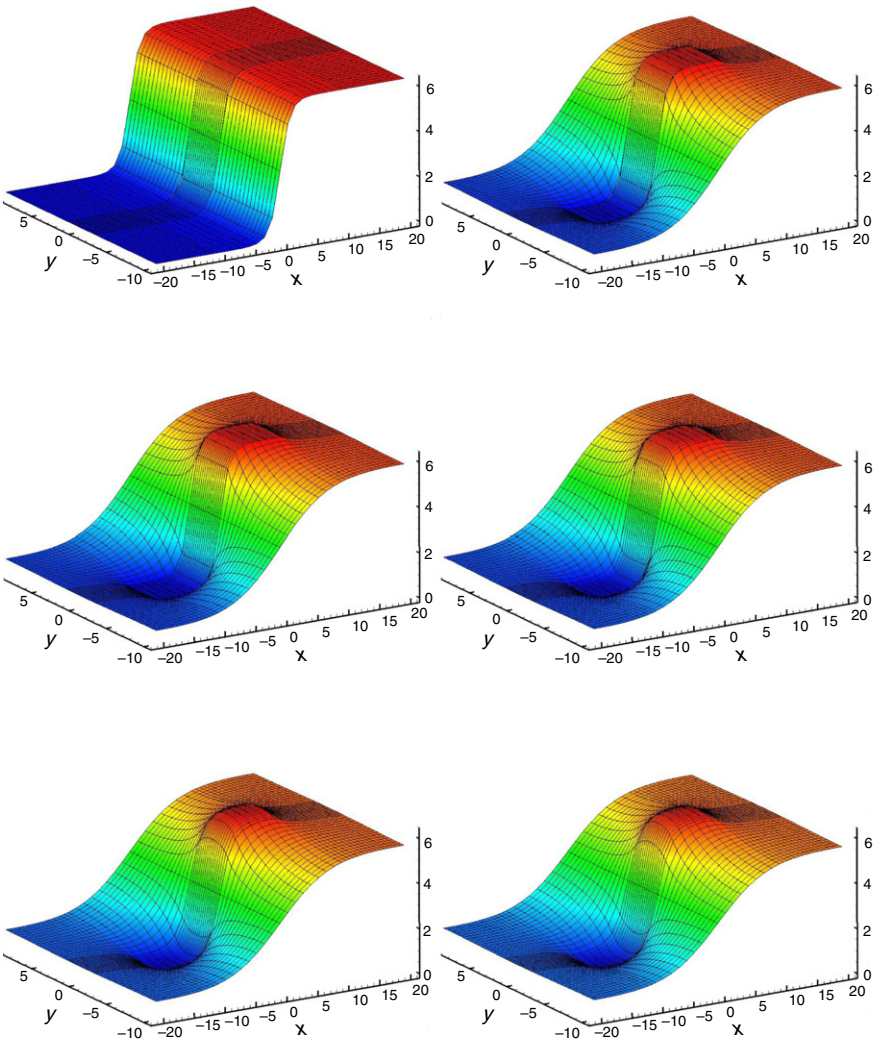

Fig. 10. 3D plots of the initial guess and the numerical solutions computed by the proposed interface relaxation method at iterations $k = 1, 2, 7, 20$ and 60 (subplots from left to right, top to bottom respectively).

5. Synopsis

We apply finite element approximation to numerical simulation of the steady state window Josephson junctions. The suitability of the proposed method for this composite type PDE problem is clearly depicted. The convergence and error analysis are presented. A combination of the Newton iteration and the preconditioned linear solver is proposed to successfully and efficiently compute solution branches of the sine-Gordon type nonlinear problem. Numerical results are promising, exhibiting the effectiveness and the efficiency of the proposed method and verifying our theoretical results. Interface relaxation methods are proposed to reduce the extra computational cost in the large idle region. The most promising of these proposed methods has been implemented using state-of-the-art numerical software and its behavior is rather promising.

Acknowledgments

The authors would like to thank Steffen Härting, Luca Heltai, Wolfgang Bangerth and Ivan Christov for their advice on using deal.II in Section 4. We would also like to thank Tecplot Inc. for the trial license provided.

The second author's work is supported in part by Hong Kong Competitive Earnmarked Grant HKUST593/94E. The first author was supported in part by MMSPMM (http://mmspmm.wordpress.com/) Thales GSRT grant.

The first and second authors would like to thank the Department of Computer Sciences, Purdue University for supporting his sabbatical visit during which part of the presented work has been carried out.

References

[1] A. Barone, G. Paterno, Physics and Applications of the Josephson Effect, John Wiley, New York, 1982.
[2] J.-G. Caputo, L. Loukitch, Designing arrays of Josephson junctions for specific static responses, Inverse Problems 24 (2008) 1–18.
[3] G. Chen, Z. Ding, C.R. Hu, W.M. Ni, A note on the elliptic sine-Gordon equation, Contemp. Math. 357 (2004) 49–67.
[4] D. Brown, M. Forest, B. Miller, N. Petersson, Computation and stability of fluxons in a singular perturbed sine–Gordon model of the Josephson junction, SIAM J. Appl. Math. 54 (1994) 1048–1066.
[5] B. Pelloni, Spectral analysis of the elliptic sine-Gordon equation in the quarter plane, Theoret. Math. Phys. 160 (1) (2009) 1031–1041.
[6] J.G. Caputo, N. Flytzanis, E. Vavalis, A semi–linear elliptic PDE model for static solution of Josephson junctions, Internat. J. Modern Phys. C 6 (1995) 241–262.
[7] J.G. Caputo, N. Flytzanis, Y. Gaididei, E. Vavalis, Two-dimensional effects in josephson junctions: I static properties, Phys. Rev. (1996) 2092–2101.
[8] J.G. Caputo, N. Flytzanis, A. Tersenov, E. Vavalis, Analysis of a semi-linear PDE for modelling stationary solutions of Josephson junctions, SIAM J. Math. Anal. 34 (2003) 1355–1378.
[9] J.-G. Caputo, L. Loukitch, Statics of point Josephson junctions in a micro strip line, SIAM J. Appl. Math. 67 (2007) 810–836.
[10] J.G. Caputo, N. Flytzanis, E. Vavalis, Effect of geometry on fluxon width in a Josephson junction, Internat. J. Modern Phys. C 7 (1996) 191–216.
[11] J. Argyris, M. Haase, J.C. Heinrich, Finite element approximations to two-dimensional sine-Gordon solitons, Comput. Methods Appl. Mech. Eng. 86 (1991) 1–26.
[12] F. Brezzi, J. Rappaz, P.A. Raviart, Finite dimensional approximation of nonlinear problems, Numer. Math. 36 (1980) 1–25.
[13] Q. Du, M.D. Gunzburger, J.S. Peterson, Analysis and approximation of the GL model of superconductivity, SIAM Rev. 34 (1992) 54–81.
[14] J.R. Rice, R.F. Boisvert, Solving Elliptic Problems Using ELLPACK, Springer-Verlag, New York, NY, 1985.
[15] J.G. Caputo, N. Flytzanis, Y. Gaididei, N. Stefanakis, E. Vavalis, Stbility analysis of static solutions in a josephson junction, Supercond. Sci. Technol. 13 (2000) 423–438.
[16] R. Bank, PLTMG: A Software Package for Solving Elliptic Partial Differential Equations. Users' Guide 8.0, SIAM, Philadelphia, PA, 1998.
[17] G. Birkoff, R.E. Lynch, Numerical Solution of Elliptic Problems, SIAM, Philadelphia, PA, 1984.
[18] J. Bramble, J. Pasciak, A. Schatz, The construction of preconditioners for elliptic problems by substracturing, I–III, Math. Comp. 47 (1986) 103–134.
[19] M. Mu, Solving composite problems with interface relaxation, SIAM J. Sci. Comput. 20 (4) (1999) 1394–1416.
[20] J.R. Rice, P. Tsompanopoulou, E. Vavalis, Interface relaxation methods for elliptic differential equations, Appl. Numer. Math. 32 (2000) 219–245.
[21] J.R. Rice, P. Tsompanopoulou, E. Vavalis, Fine tuning interface relaxation methods for elliptic differential equations, Appl. Numer. Math. 43 (2002) 459–481.
[22] P. Tsompanopoulou, E. Vavalis, Analysis of an interface relaxation method for composite elliptic differential equations, J. Comput. Appl. Math. 226 (2009) 370–387.
[23] W. Bangerth, R. Hartmann, G. Kanschat, deal.II—a general purpose object oriented finite element library, ACM Trans. Math. Software 33 (4) (2007) 24/1–24/27.