PHYSICAL REVIEW E **95**, 063304 (2017)

Auxiliary variables for numerically solving nonlinear equations with softly broken symmetries

Ken D. Olum* and Ali Masoumi†

Institute of Cosmology, Department of Physics and Astronomy, Tufts University, Medford, Massachusetts 02155, USA

(Received 1 February 2017; revised manuscript received 21 April 2017; published 12 June 2017)

General methods for solving simultaneous nonlinear equations work by generating a sequence of approximate solutions that successively improve a measure of the total error. However, if the total error function has a narrow curved valley, the available techniques tend to find the solution after a very large number of steps, if ever. The solver first converges rapidly to the valley, but once there it converges extremely slowly to the solution. In this paper we show that in the specific physically important case where these valleys are the result of a softly broken symmetry, the solution can often be found much more quickly by adding the generators of the softly broken symmetry as auxiliary variables. This makes the number of variables more than the equations and hence there will be a family of solutions, any one of which would be acceptable. We present a procedure for finding solutions in this case and apply it to several simple examples and an important problem in the physics of false vacuum decay. We also provide a *Mathematica* package that implements Powell's hybrid method with the generalization to allow more variables than equations.

DOI: 10.1103/PhysRevE.95.063304

I. INTRODUCTION

Numerical methods of solving equations are one of the most important topics in numerical analysis. There is a plethora of techniques of which each is adequate for specific sets of problems. For examples refer to Ref. [1] and references therein. All techniques attempt to successively improve some guess for the variable values and eventually to converge to an adequate solution.

In order to know whether the new guess is an improvement, we need a single measure of the total error in all the equations. Successive steps must then improve the total error, and the ability of the technique to make adequate progress depends on the shape of this function. In certain cases, the correct solution lies in a narrow curved valley in the total error. Once the solver finds the valley, it will be difficult to improve matters further, because there is no direction in which one can move any substantial distance (in a straight line) without making the total error worse. Any technique using such a function will be forced to take a very large number of tiny steps, sometimes so tiny as to make it impractical to reach the solution.

To improve the speed of convergence and the likelihood of finding the answer, we would like to reformulate such problems using new variables to accomplish two goals. First, we would like the valley to be straight, as much as possible, in the new variables. Then a solution procedure that moves in straight lines can take long steps toward the solution. Second, we would like to encourage the solver to move along the valley, which we expect to be productive, rather than in other directions. This can be done by rescaling the new variables, as we discuss below.

However, if one does not know the solution to a set of equations, how could one design such a reformulation? A general method seems impossible, but if the valley results from a softly broken symmetry, the generators of the symmetry tell us how to move along the valley floor. In this case, we propose adding auxiliary variables, which are the symmetry generators, without adding new equations. The idea of adding variables for solving equations is not new, but it is usually accompanied by adding an equal number of equations. In our method we add new variables only.

This paper is organized as follows. In Sec. II we review some of the procedures for solving simultaneous nonlinear equations. In Sec. III we describe the problem and lay out the formalism for adding new variables. In Sec. IV we present two simple examples with softly broken symmetries and the reason for adding new unknowns. In Sec. V we present a specific theoretical physics problem that can be solved this way. In Sec. VI we briefly introduce a *Mathematica* package for Powell's hybrid method. We conclude in Sec. VII.

II. SOLUTION OF A SET OF NONLINEAR EQUATIONS

In this section we review techniques for solving simultaneous nonlinear equations. Let us write the equations in the form

$$f_i(\mathbf{x}) = 0, \quad i = 1,\ldots,N, \tag{2.1}$$

where $\mathbf{x}$ is an N-element vector. We define the total error in the usual way,

$$\mathrm{error}(\mathbf{x}) = \left(\sum_{i=1}^{N} f_i(\mathbf{x})^2\right)^{1/2}, \tag{2.2}$$

and demand that successive steps reduce error(x). If a trial step does not decrease the error, then we reject it and try again with a shorter step.

We will try to find the solution by starting from some initial guess $\mathbf{x}_0$ and taking steps according to a (predetermined) rule for approaching the solution. At each step we will generate a new trial value, so we have a sequence $\mathbf{x}_1, \mathbf{x}_2, \ldots$, which we hope will eventually reach some $\mathbf{x}$ for which the $f_i(\mathbf{x})$ vanish to some desired tolerance.

The best known procedure for solving a set of equations is Newton's method (also called Newton-Raphson), which works by linearizing the equations around the current guess and using

*kdo@cosmos.tufts.edu

†ali@cosmos.phy.tufts.edu

the solution to the linearized problem as the next guess. It converges very rapidly when it gets near the solution, but if it is not close to the solution it can take large steps in unhelpful directions that make the next guess worse than the previous one. We can modify Newton's method to fit into the above framework by trying shorter and shorter steps in the same direction until we find one that reduces the error.

An alternative, which works better when one is far from the solution, is just to descend the gradient of the total error. This method improves the solution at each step, but can be very slow. We can use the linearized equations to determine the direction of steepest descent and the distance we can travel, assuming the linearized approximation is still good, before the error starts to become worse. We try this distance and direction and then back off as above if that makes the total error worse.

Powell's hybrid (or "dogleg") method [2] combines the two techniques. It maintains a step size that, based on its performance so far, is likely to lead to an improvement. It then tries a step of this length, which is, in general, a combination of the step recommended by Newton's method and a step in the direction of steepest descent of the total error. If a step does a good job of improving the error, the method increases the step size for future steps. If the step makes things worse, it tries again with a shorter step size and takes shorter steps in the future. This method gives many of the advantages of Newton's method and of gradient descent. It is the one that we usually use to solve sets of nonlinear equations.

III. PROBLEM AND SOLUTION TECHNIQUE

We now explain the difficulties encountered by algorithms such as those above, when the total error has narrow valleys, as shown in Fig. 1. Consider first a system that works purely by gradient descent. It starts by descending the total error surface until it reaches the valley. It will in general not be exactly at the bottom, but somewhere on the valley wall. At the bottom of the valley, the gradient would point down the valley. However, when the valley floor is fairly level, while the walls are steep, if we are not exactly at the bottom the gradient points mostly down the valley wall toward the bottom with a small admixture of the direction that the valley floor descends.

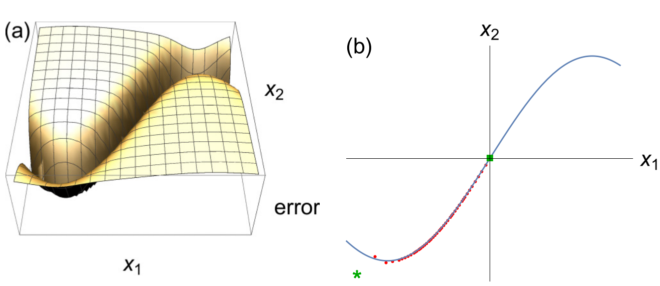

FIG. 1. (a) Error surface for a system of two equations that exhibits a narrow valley. (b) Trajectory that Powell's hybrid method took to find the solution. The initial guess is shown by the green asterisk. The solver first quickly converged to the valley, but then it moved very slowly along the valley (solid line) to the solution (green square) at the origin. If the valley has very sharp curves, it will take many more steps to find the solution.

Thus the next step will be primarily toward the valley floor. We will generally fall short of the floor and it will take a number of steps to get down to the point where we move along the valley to any significant degree. In a straight valley, we would move closer and closer to the floor and take larger and larger steps in the direction of the valley and soon approach the solution. However, if the valley is curved, steps initially in the right direction are likely to take us up the wall again after a short distance and so we can never travel rapidly along the valley floor. Hence, instead of moving towards the solution, the solver "bounces" very frequently from the walls of the valley and achieves little improvement. Thus reaching the solution is very laborious and in sufficiently extreme cases impractical. When there are more than two equations, several directions may be "along the valley," while several other directions are "descending the wall." There are then many more ways for the valley to curve.

It is important to understand that the above description is the "bird's eye" description of what is happening. The solver itself does not know that it is stuck in a curving valley. All it knows is that it is analyzing local conditions, choosing what appears to be a good direction in which to move, discovering that only a very small step yields any improvement, and repeating these steps, leading to very slow progress.

We know of no general solution to this problem, but in the special case where the valley results from an approximate symmetry we can resolve it. The idea here is that if the symmetry were exact, there would be a valley at the bottom of which every point is a solution. The breaking of the symmetry leads to only one such point being a solution, but because the symmetry is only softly broken, the function values vary slowly along the valley. Thus the valley floor is only slightly slanted and we have the troublesome situation above. Several examples are given in the following sections.

To make faster progress, we would like to formulate the problem so that some of the variables generate motion along the valley floor. In these variables the valley will be straight, which is the first goal discussed in the Introduction. The needed variables are just the generators of the symmetry and are easily chosen. In some cases, it is then possible to choose other variables to fill out the degrees of freedom of the problem and the particular techniques described below are not necessary. However, in many other cases there is no simple formulation of this kind. In such cases, we propose to keep all the original variables and simply add the symmetry generators as extra variables $\mathbf{y} = \{y_1, y_2, \ldots, y_K\}$, without adding extra equations.

We then extend our functions $f_i(\mathbf{x})$ to some $g_i(\mathbf{x},\mathbf{y})$ such that $g_i(\mathbf{x},0) = f_i(\mathbf{x})$. It is now possible to move in new directions given by $\mathbf{y}$. In these directions, the valley will be straight, so the solver will be able to take large steps that quickly reduce the error and lead to a solution. We show examples of this phenomenon in Secs. IV and V.

Of course, since we add variables without adding constraints, the solution is not unique. For each solution of $f_i(\mathbf{x}) = 0$ there will be a K-dimensional family of solutions to $g_i(\mathbf{x},\mathbf{y}) = 0$. When the process succeeds, we have some solution $\{\mathbf{x},\mathbf{y}\}$ and we need to get from there to the solution of the original problem, $f_i(\mathbf{x}) = 0$. However, this is straightforward if the $\mathbf{y}$ are the generators of a symmetry. We can solve the original problem by simply applying the symmetry transformation to the $\mathbf{x}$ that we found.

We now show explicitly how to find a solution of N equations of $N+K$ variables using Powell's hybrid method, discussed in Sec. II. This method takes steps that are a combination of the step recommended by Newton's method and a step in the direction that most rapidly reduces error($\mathbf{x}$), i.e., the negative of the gradient of error($\mathbf{x}$). In the gradient case, the procedure is unaffected by additional variables. We simply have a scalar function of $N+K$ variables whose gradient we descend.

The application of Newton's method is slightly more complicated. Newton's method consists of linearizing the equations around some point $\mathbf{x}$ and solving the linearized equations. With N equations in $N+K$ variables, there will be a K-dimensional subspace of solutions. In this case, we choose the solution that is nearest to the current guess, in the Euclidean metric on the $N+K$ variables. This is straightforwardly determined by singular value decomposition of the rectangular Jacobian.

By introducing new coordinates we have made the valley straight, so the solution procedure can in principle take large steps along the valley. However, in general it will do so only to a limited degree. Changing each y_i is equivalent to moving the point $\mathbf{x}$. Let $d\mathbf{x}$ be the change equivalent to a displacement of dy_i. Suppose that we have chosen the arbitrary scale of y_i so that magnitude of the vector $d\mathbf{x}/dy_i$ is 1. Then in the linearized equations, the direction of y_i and the direction $d\mathbf{x}/dy_i$ are equivalent. (They are very different beyond linearized order: The y_i direction leads down the valley, while the $d\mathbf{x}/dy_i$ direction is a straight line in $\mathbf{x}$ that does not follow the valley as it curves.) As a result, the step taken by the solver will move equally in these two directions.

This is an improvement over moving only in the $d\mathbf{x}/dy_i$ direction, but only a mild one. The step size is limited by climbing up the valley wall, so we must still take short steps, though each one is more effective because in addition to moving in $\mathbf{x}$ we move in y_i, which does not take us off the valley floor.

Thus we still need to accomplish the second goal in the Introduction, which is to get the solution procedure to make good use of the additional directions. We can do this by multiplying the additional variables y_i by some factors $s_i > 1$. Gradient methods try to find small steps in the space of variables that will lead to large improvements in the error. Thus, if an additional variable is multiplied by some factor $s > 1$ before being used in the function evaluation, the solver will give higher weight by factor s to changes produced by this variable and will be thus more likely to exploit that additional variable to simplify the problem.

The unmodified Newton method is invariant under rescaling of the variables (and the functions), because that does not change the solution to the linearized problem. However, the modified method here chooses the closest point in the subspace of solutions. The factor s increases the effect of the additional variable, so the change in that variable to achieve the same result is smaller. Thus increasing s causes the modified Newton method to choose a solution whose differences are due more to the affected variable and less to the others.

The best choice of s depends on the problem at hand. Larger s cause the solver to move more in the direction of the broken symmetry and so make faster progress, but too large a value may cause the process to fail completely. We have generally found good results with $s = 10$, relative to the parametrization of y_i that makes $|d\mathbf{x}/dy_i| = 1$.

IV. SIMPLE EXAMPLES

In this section we present two problems with a softly broken symmetry and show the improvement made by adding a new variable. The problems in this section can straightforwardly be solved analytically, but we show these cases as a proof of concept in a simple setup where the basic ideas can be seen most vividly. Thus we assume that we have been given these functions in a "black box" and do not know their analytic forms. We will try to solve them numerically, meaning that all we can do is evaluate the functions (and evaluate or approximate their first derivatives) and search for the solution. In Sec. V we present a case where there is no analytic solution.

A. Softly broken rotational symmetry

We start from a very easy problem to demonstrate the main idea behind our proposal. Consider the functions

$$f_1(x,y) = x^2 + y^2 - 1, \quad f_2(x,y) = \epsilon x \tag{4.1}$$

for some parameter ϵ. We want to find x and y such that $f_1 = f_2 = 0$. The solution is $y = \pm 1$ and $x = 0$, but we suppose we do not know that and are trying to find the solution by purely numerical methods. In Fig. 2(a) we plot the error for this set of equations with $\epsilon = 0.1$, as a function of the variables x and y. Figure 2(b) shows a magnified picture of the curving and slanted valley. In Fig. 2(c) we show the 14 steps taken by Powell's method to find the solution.

Now let us introduce an extra variable that allows rotation, the broken symmetry. Instead of using x and y and variables, we use x', y', and θ, with x and y given by

$$\begin{aligned} x &= x' \cos\theta + y' \sin\theta, \\ y &= -x' \sin\theta + y' \cos\theta. \end{aligned} \tag{4.2}$$

Our functions thus become

$$\begin{aligned} f_1(x,y) &= x'^2 + y'^2 - 1, \\ f_2(x,y) &= \epsilon(x' \cos\theta + y' \sin\theta). \end{aligned} \tag{4.3}$$

Now there will be an infinite number of solutions. For each value of θ, there will be a solution x' and y'. We can recover the values of x and y using (4.2). With the extra variable, the solution takes only eight steps.

To improve matters further, we can use a rescaled variable. Let the variables be (x', y', ϕ), with $\theta = s\phi$ in (4.3). With $s = 10$, this choice reduces the number of steps to six, as shown in Fig. 2(d).

In Fig. 3 we show the path our solver took for $\epsilon = 10^{-4}$. The symmetry here is broken softly, so there is not much slant in the valley. Powell's method took 535 steps to solve this problem, mostly creeping slowly along the valley. Introducing

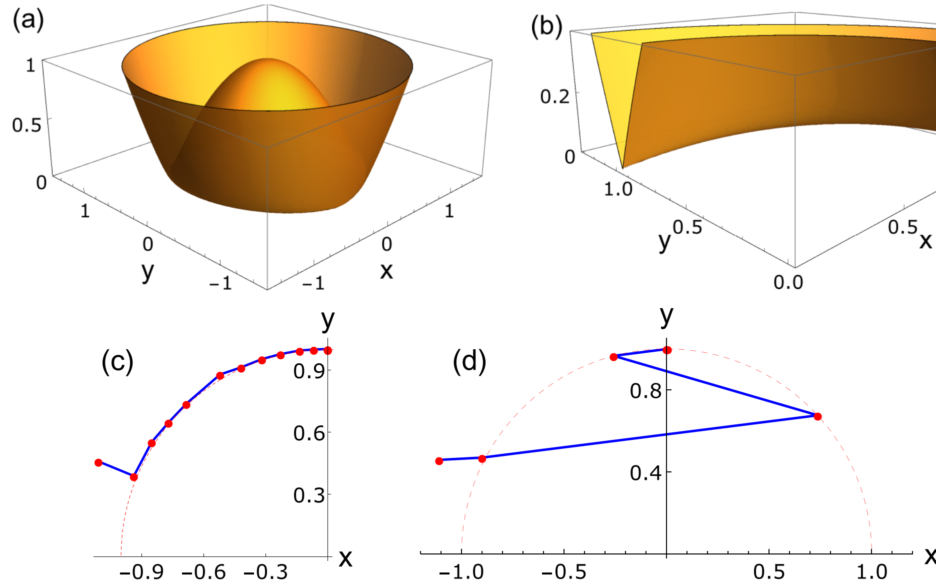

FIG. 2. (a) Error as a function of x and y for the functions defined in (4.1) with $\epsilon = 0.1$. The problem has an approximate rotational symmetry. (b) To see the breakdown of the symmetry we zoomed in on the valley, which is slanted. (c) Path that the solver took without an extra variable. (d) Path in x and y taken when we use the extra variable ϕ and choose $s = 10$. In this case, the solver takes steps in x', y', and ϕ, but we graph the points x and y given by (4.2) with $\theta = s\phi$. In both cases the starting guess is $[-1.2\cos(\pi/8), 1.2\sin(\pi/8)]$. The left panel took 14 steps and the right only 6 steps. We see in Fig. 3 that smaller ϵ increases the number of steps rapidly.

θ gives only a modest improvement, down to 151 steps, but with $\theta = s\phi$ and $s = 10$, the solution can again be found in six steps.

This particular problem is rather trivial. Instead of adding a variable one could simply work in polar coordinates, which manifest the symmetry, and get the solution right away. This is possible because one can easily parametrize the remaining degree of freedom after the symmetry has been factored out.

One also can solve this problem by using the pure Newton method. Since one of the equations is linear, Newton's method will solve it exactly in every step by jumping to some point on the y axis. Newton's method is not affected by rescaling the equations and hence the smallness of ϵ is not important and the symmetry is not broken softly.

In the following sections we present two other problems where there is no obvious change of coordinate and the Newton technique does not necessarily work well.

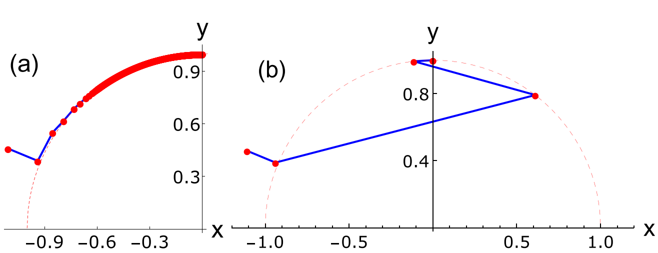

FIG. 3. Steps taken by the solver for the example presented in (4.1) for $\epsilon = 10^{-4}$. Now the rotational symmetry is broken very softly. (a) Steps taken without the extra variable and (b) steps taken with an extra variable ϕ that generates the broken symmetry and $s = 10$. In both cases the starting guess is $[-1.2\cos(\pi/8), 1.2\sin(\pi/8)]$. The left panel took 535 steps and the right only 6 steps.

B. Softly broken translational symmetry

In this section we present a problem where the correct parametrization is not as trivial as the previous one. Suppose we have some functions $f(x)$ and $g(x)$, chosen from families of similar functions, and we consider the function given by

$$F(x) = \begin{cases} f(x), & f(x) < 0 \\ g(x), & g(x) \geqslant 0, \end{cases} \tag{4.4}$$

where $f(x)$ and $g(x)$ are monotonically increasing functions. That is to say that F follows f from $x = -\infty$ until $f(x)$ reaches 0, and g from $x = \infty$ toward smaller x until $g(x)$ reaches 0. To have a well-defined function, we would like $f(x)$ and $g(x)$ to reach 0 at the same point, and to have a C^1 function we would like the derivatives of f and g to match also at this point. Our f and g will be chosen from the following classes of functions:

$$\begin{aligned} f(x) &= ae^x + \epsilon e^{2x} - 1, \\ g(x) &= 1 - be^{-(1+\epsilon)x}, \end{aligned} \tag{4.5}$$

where a and b specify which functions we chose and ϵ is a fixed small parameter.

To have a problem where the symmetry is easily shown, rather than specifying a and b we will specify that F takes on a given value $F_1 < 0$ at a given point x_1 and similarly a given value $F_2 > 0$ at a given point x_2. We will then attempt to vary F_1 and F_2 to find a well-defined C^1 function F.

First we explain the broken symmetry. If $\epsilon = 0$, we have

$$f(x) = ae^x - 1, \tag{4.6}$$

$$g(x) = 1 - be^{-x}. \tag{4.7}$$

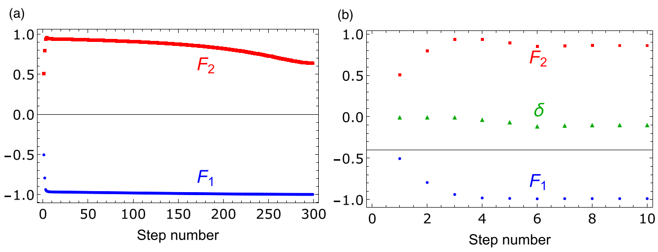

FIG. 4. (a) Values of F_1 (blue circles) and F_2 (red squares) at steps taken by the solver before finding the solution. (b) Same as in (a) but with an extra variable δ (green triangles) added. The solution was found in ten steps.

We want the two functions to join (vanish) at some x_0. For each value x_0 in the interval (x_1,x_2) there will be a solution in the form

$$f(x_1) = e^{x_1 - x_0} - 1, \tag{4.8}$$

$$g(x_2) = 1 - e^{x_0 - x_2}. \tag{4.9}$$

Hence the values of F_1 and F_2 depend only on $x_1 - x_2$ and are invariant under a shift in these two numbers.

The terms that include ϵ break this translational symmetry softly and there will be unique values for F_1 and F_2 that make the function smooth. However, because ϵ is small, it is difficult to find the correct values once we find some that solve the $\epsilon = 0$ problem.

We now choose $x_1 = -5$, $x_2 = 1$, and $\epsilon = 10^{-3}$. Without adding an extra variable it took 299 steps for the solver to find F_1 and F_2. However, because we know the broken symmetry is translational, we simply allow for a shift in the values of x by changing the equations to

$$f(x) = ae^{x - s\delta} + \epsilon e^{2(x - s\delta)} - 1, \tag{4.10}$$

$$g(x) = 1 - be^{-(1+\epsilon)(x - s\delta)}. \tag{4.11}$$

We shifted here by $s\delta$ with $s = 10$ to encourage the solver to make use of δ. Now there will be an infinite set of solutions $\{F_1,F_2,\delta\}$. After finding one such solution, we recover the original values of F_1 and F_2, by evaluating (4.10) at $x_1 + s\delta$ and (4.11) at $x_2 + s\delta$. With δ added, it took only ten steps for the solver to find the solution. We show the steps that the solver took for this problem in Fig. 4.

In the next section we present an important physics problem that we solved in much more generality in Ref. [3].

V. TUNNELING IN FIELD THEORIES

Problems of differential equations with boundary conditions at two points are commonplace in physics. One important example is the equations used for calculation of cosmological phase transitions. Here we explain the problem briefly. The details can be found in Ref. [4]. In Fig. 5 we present a potential that has a metastable (false) minimum at ϕ_f and a stable (true) minimum at $\phi_t = 0$. Finding the lifetime of this metastable minimum is tantamount to finding the solution of

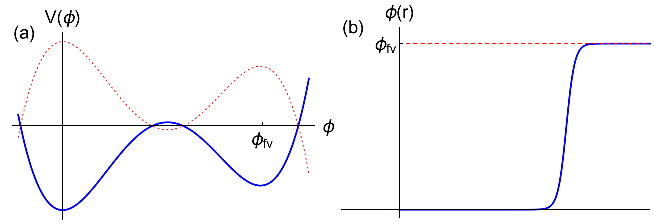

FIG. 5. (a) Potential with two minima at $\phi_t = 0$ and ϕ_f in solid blue and the upside down potential as the dotted red graph. The minimum at ϕ_f is a metastable minimum and can tunnel quantum mechanically to the other minimum. (b) Solution to the field equation.

the differential equation

$$\phi''(r) + \frac{3}{r}\phi'(r) = \frac{\partial U}{\partial \phi}, \tag{5.1}$$

with two boundary conditions

$$\phi'(0) = 0, \quad \phi(\infty) = \phi_f. \tag{5.2}$$

This is the same as the motion of a particle in the upside down potential shown in the red dotted graph in Fig. 5, under the influence of a velocity-dependent friction given by $-3\phi'(r)/r$. The field profile that solves this equation is shown in Fig. 5(b).

The standard technique to solve such a problem is shooting. We try to find the correct value of $\phi(0)$ such that evolving the fields from this point leads ϕ to approach ϕ_f as $r \to \infty$. It is easy to see that this problem always allows a solution.[1] If one chooses the value $\phi(0)$ very close to ϕ_t the field does not have any significant change until r gets large. However, then the friction term is negligible and the "energy" is conserved and it passes ϕ_f. On the other hand, if one chooses $\phi(0)$ such that $U[\phi(0)] < \phi_f$, the particle can never reach ϕ_f. The boundary between these two is the correct solution that asymptotes to ϕ_f at infinite r.

Unfortunately, when we integrate the differential equation up the hill (in the inverted potential) toward the false vacuum, we encounter an instability. Numerical errors grow because of the growing mode in which ϕ accelerates down the hill. Thus it is better to take a second differential equation with an initial condition near the false vacuum at large r and integrate toward smaller r. Then we require that ϕ and ϕ' must agree at a junction in the middle.

In Ref. [3], we further generalized this method to use more than two shooting regions, which is especially important in problems with more than one field dimension. By using sufficiently short shooting regions we can overcome any instabilities in integration of Eq. (5.1). We also use analytic solutions near the true and false vacua to avoid extreme sensitivity on boundary conditions. We describe our method much more fully in Ref. [3].

This method with three intervals is shown in Fig. 6. Our variables are the value of the field at r_1 and r_4 and the value of the field and its derivative at r_2, in short $\{\phi_1,\phi_2,\phi_2',\phi_4\}$. With these data we can evolve the field equation (5.1) along the blue curves. The goal is finding the correct values of $\{\phi_1,\phi_2,\phi_2',\phi_4\}$

[1] This solution is not necessarily unique, as explained in Ref. [3].

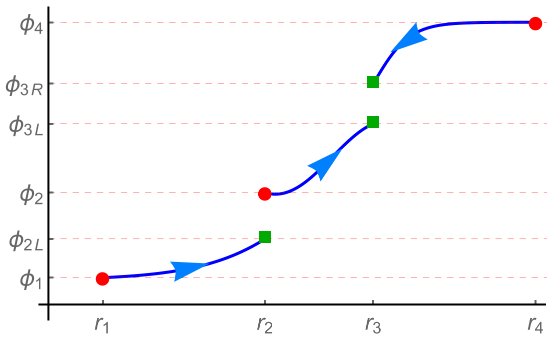

FIG. 6. Illustration of multishooting method. One has to adjust the values of $\{\phi_1,\phi_2,\phi_2',\phi_4\}$ and evolve the field equations (5.1) along the blue curves such that the field and its derivative are continuous at r_2 and r_3. This produces a system of four equations of four variables.

such that the field and its derivative are continuous at r_2 and r_3. This creates a system of four equations of four variables.

This technique rapidly finds accurate solutions for many potentials. However, in many other cases it makes very slow progress or is never able to reach the solution. The problem is one of an approximate symmetry. If not for the middle (friction) term in (5.1) and the fact that $\phi(0)$ is not exactly zero, Eq. (5.1) would possess a translational symmetry. If $\phi(r)$ satisfied the equations of motion and the boundary conditions, $\phi(r+\delta)$ would satisfy them also. Translations in r make only small differences to the equations that we are trying to solve, but in terms of our variables $\{\phi_1,\phi_2,\phi_2',\phi_4\}$, such a translation requires a complicated change that is not a straight line in the space of variable values and is thus difficult for the solver to follow.[2] As an example, for the simple potential

$$U(\phi) = 0.2\phi - 2\phi^2 + \phi^4, \quad \phi_t \approx -1.024, \quad \phi_f \approx 0.9740, \tag{5.3}$$

it took the hybrid solver 851 steps to find the solution. (We chose the values of $\{r_1,r_2,r_3,r_4\} = \{12.82,14.03,15.23,16.43\}$ using an analogy with the thin-wall solution. The details can be found in Ref. [3].)

To make use of the approximate translation symmetry, we introduce an additional variable δ, whose meaning is that the variables $\{\phi_1,\phi_2,\phi_2',\phi_4\}$ refer to these quantities at $r_1+\delta$, $r_2+\delta$, and $r_4+\delta$ (and the point at which the middle and the final regions must agree is $r_3+\delta$). With this additional variable, the hybrid solver could find the solution in 12 steps. Most of the steps were taken in the valley, which corresponds to a rigid shift of the profile $\phi(r)$. In Fig. 7 we show the change of error with and without adding the auxiliary variable δ.

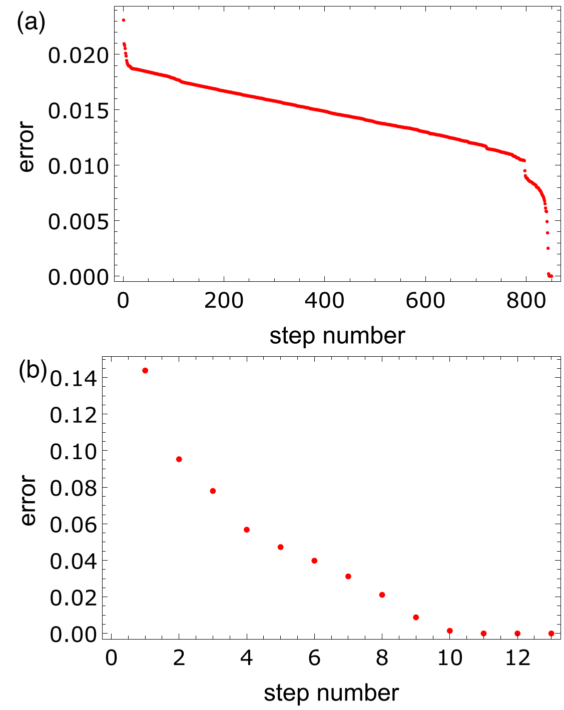

FIG. 7. Change of error after each step taken (a) without adding an auxiliary variable and (b) when including δ, for the potential in (5.3) in order to solve (5.1).

VI. POWELL-HYBRID PACKAGE

We include with this paper a *Mathematica* code developed by Olum [5], for the solution of simultaneous nonlinear equations using Powell's hybrid method [2]. This code includes the extension described here for solving $N+K$ equations in N variables. However, it handles only problems where the Jacobian can be computed analytically. Powell [2] also includes a method for approximating the Jacobian using, mainly, the successive function evaluations, but we did not implement that. The code and a manual for using it can be downloaded from [5]. This code is the same one used in Ref. [3].

VII. CONCLUSION

In this paper we described a method for solving equations with a softly broken symmetry. The broken symmetry makes it very difficult for the solver to make progress towards minimizing the error in the desired equations. The error as a function of the variables usually has narrow valleys where moving along these valleys does not make large improvement. As a result, the steps that the solver takes are small and it takes a very large number of steps for the solver to converge on the solution.

We introduced a method where one adds a set of auxiliary variables that are the generators of the softly broken symmetry. Adding these variables makes the solver take large steps in the direction along the valley and hence converges very quickly. Our method may be applicable beyond the theories with softly broken symmetries. We believe that whenever such a valley is present, adding variables that make the solver move in the proper direction makes converging much faster and the basin of attraction larger. However, for general valleys that are not the results of broken symmetries it may not be easy to identify the correct auxiliary variables to be added. For this reason we only mentioned these cases for which we know the auxiliary variables must generate the broken symmetry. These

[2]One might think that this problem could be solved by fixing ϕ and using the corresponding values of r as variables, but this brings a host of other problems, including difficulty generalizing to more field dimensions.

techniques will be much more powerful and versatile if one can find a systematic way to determine the needed auxiliary variables.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation (Grant No. PHY-1518742).

[1] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes: The Art of Scientific Computing*, 3rd ed. (Cambridge University Press, New York, 2007).

[2] M. J. D. Powell, in *Numerical Methods for Nonlinear Algebraic Equations*, edited by P. Rabinowitz (Gordon and Breach, New York, 1970), Chap. 6, pp. 87–114.

[3] A. Masoumi, K. D. Olum, and B. Shlaer, Efficient numerical solution to vacuum decay with many fields, J. Cosmol. Astropart. Phys. 01 (2017) 051.

[4] S. Coleman, Fate of the false vacuum: Semiclassical theory, Phys. Rev. D **15**, 2929 (1977); Erratum: Fate of the false vacuum: Semiclassical theory, **16**, 1248(E) (1977).

[5] K. D. Olum, *Powell's Hybrid Method in Mathematica*, available at http://cosmos.phy.tufts.edu/Powell.