Open access • Journal Article • DOI:10.1017/S1365100507070034

Multi-Dimensional Transitional Dynamics: A Simple Numerical Procedure

Timo Trimborn, Karl-Josef Koch, Thomas M. Steger

Institutions: Leibniz University of Hanover, University of Siegen, Leipzig University

Published on: 01 Jun 2008 - Macroeconomic Dynamics (Cambridge University Press)

Topics: Stiff equation, Optimal control and Dynamic programming

Macroeconomic Dynamics, **12**, 2008, 301–319. Printed in the United States of America.
doi: 10.1017/S1365100507070034

ARTICLES

MULTIDIMENSIONAL TRANSITIONAL DYNAMICS: A SIMPLE NUMERICAL PROCEDURE

TIMO TRIMBORN
University of Hannover

KARL-JOSEF KOCH
University of Siegen

THOMAS M. STEGER
University of Leipzig

We propose the relaxation algorithm as a simple and powerful method for determining the transition process in growth models numerically. This method has a number of important advantages: (1) It can easily deal with a wide range of dynamic systems including stiff differential equations and systems giving rise to a continuum of stationary equilibria. (2) The application of the procedure is fairly user-friendly. The only input required consists of the dynamic system. (3) The variant of the relaxation algorithm we propose exploits in a natural manner the infinite time horizon, which usually underlies optimal control problems in economics. As an illustrative application, we compute the transition process of the models of Jones [Jones, C.I. (1995) R&D-based models of economic growth. *Journal of Political Economy* 103 (3), 759–784] and Lucas [Lucas, R.E., Jr. (1988) On the mechanics of economic development. *Journal of Monetary Economics* 22, 3–42].

Keywords: Transitional Dynamics, Continuous Time Growth Models, Saddle-point Problems, Multidimensional Stable Manifolds

1. INTRODUCTION

Dynamic macroeconomic theory nowadays relies heavily on infinite-horizon optimization models, which usually give rise to systems of nonlinear differential equations. These dynamic systems are then interpreted to describe the evolution of the economy under consideration. Many studies in the field of growth theory have confined their analysis to the balanced-growth path (BGP). A comprehensive understanding of the model under study requires, however, that we investigate in addition the transition process. At least two important arguments support this view:

For helpful comments and suggestions we would like to thank Max Albert, Beatriz Gaitan Soto, Christian Groth, Chad Jones, and Holger Strulik, and two anonymous referees. Address correspondence to: Timo Trimborn, Department of Economics, Institute for Macroeconomics, University of Hannover, Koenigsworther Platz 1, 30167 Hannover, Germany; e-mail: trimborn@vwl.uni-hannover.de.

First, the positive and normative implications might differ drastically depending on whether an economy converges toward its BGP or grows along the BGP [e.g., Jones (1995)]. Second, dynamic macroeconomic models are often employed to conduct comparative welfare investigations of different policy regimes or instruments. In this context, the transition process needs to be taken into account. Linearizing the dynamic system might be appropriate in many cases but can be potentially misleading, especially when the analysis aims at a Pareto ranking of different policy instruments. This overall perspective is nicely summarized by the following statement, due to Jonathan Temple (2003, p. 509): Ultimately, all that a long-run equilibrium of a model denotes is its final resting point, perhaps very distant in the future. We know very little about this destination, and should be paying more attention to the journey.

The models employed in growth theory are often multidimensional in the sense that there is more than one (predetermined) state variable. Examples comprise R&D-based growth models [e.g., Romer (1990); Jones (1995); Eicher and Turnovsky (1999)] as well as human capital–based growth models [e.g., Lucas (1988); Mulligan and Sala-i-Martin (1993); Benhabib and Perli (1994)]. This class of models frequently exhibits characteristics that make the use of standard procedures fairly inconvenient if not impossible. Here we would like to stress two issues: First, assuming usual stability properties in multidimensional models implies that the stable manifold is also multidimensional.[1] Moreover, if the dynamic system is characterized by stable eigenvalues that differ substantially in magnitude (i.e., stiff differential equations), then usual procedures are either not applicable or highly inefficient. This characteristic property is not at all a special (or even pathological) case but instead occurs quite frequently; an example is the well-known Jones (1995) model. Second, most standard solution procedures are not applicable to dynamic systems giving rise to a continuum of saddle-point stable stationary equilibria (i.e., a center manifold). This property arises, for instance, in the popular Lucas (1988) model.

This paper contributes to the literature on dynamic macroeconomic theory by proposing the relaxation algorithm as a powerful method for determining the transition process in growth models numerically. We show that this procedure is in general well-suited and highly efficient. This will be demonstrated by computing the transition processes of two prominent growth models, the Jones (1995) model and the Lucas (1988) model. Despite the fact that these models are widely employed in growth theory, their adjustment processes have hardly been investigated. This is probably due to the characteristics mentioned above, which give rise to serious conceptual difficulties when it comes to computational issues.

In the context of growth theory, the most prominent approaches to computing the transition process comprise shooting [e.g., Judd (1998, Chapter 10)], time elimination [Mulligan and Sala-i-Martin (1991)], backward integration [Brunner and Strulik (2002)], the projection method [Judd (1992)], and the discretization method of Mercenier and Michel (1994). The similarities and differences of the relaxation procedure and the methods mentioned above will be discussed concisely below. The above enumeration shows that there are already some procedures that have been used in economics to solve dynamic systems. Nonetheless, we think that there are a number of good reasons to add the relaxation procedure into the toolbox of dynamic macroeconomic theory.

First, our experiences with the relaxation algorithm are positive throughout. We have applied the procedure to a wide range of dynamic systems, including stiff differential equations and dynamic systems with saddle-point stable center manifolds, as well as highly dimensional computable general equilibrium models. It is remarkable that an increase in the dimension of the model under study does not cause any conceptual problems. The researcher need not take restrictions with respect to the model dimension into account. In addition, the procedure seems to be efficient with respect to computer time.

Second, the application of the procedure is fairly user-friendly. Specifically, the only input that must be provided by the researcher consists of the dynamic system and the set of underlying parameters. No preliminary manipulations of the dynamic system under study need to be conducted before the procedure can be applied; this is different from most other procedures, as described in Section 3.

Third, the variant of the relaxation algorithm that we propose exploits in a natural manner the infinite time horizon that usually underlies standard optimal-control problems. This is achieved by a simple transformation of real calendar time into a transformed time scale (as explained in Section 2.1). For most other procedures, this issue must be dealt with explicitly (explained in Section 3).

Overall, it seems that the relaxation algorithm can easily cope with a large number of problems that arise frequently in the context of multidimensional, infinite-time-horizon optimal control problems. Finally, it should be noticed that the focus here is on continuous-time dynamic models, which have been extensively employed in growth theory. The relaxation procedure has been employed to investigate discrete-time dynamic macroeconomic models [Laffargue (1990); Juillard et al. (1998)]. However, the procedures employed to solve discrete-time models numerically differ from those applied to continuous-time models in that the stage at which the discretization is done can be chosen in the latter case.[2]

The paper is structured as follows: In Section 2, the relaxation procedure is described concisely and then evaluated numerically, employing the Ramsey–Cass–Koopmans model as an example. Section 3 provides a short comparison to alternative methods. In Section 4, we apply the procedure to compute the transition process of the Jones (1995) model and the Lucas (1988) model. Section 5 summarizes and concludes. The Appendix provides a more formal description of the relaxation procedure. Finally, the relaxation algorithm has been programmed in MatLab. This program, together with a concise instruction manual, is available for free download at http://www.relaxation.uni-siegen.de.

2. THE RELAXATION PROCEDURE

2.1. Description of the Relaxation Procedure

Relaxation is a particular type of finite-difference method used to solve a differential equation numerically [e.g., Press et al. (1989), p. 645]. The differential equation is replaced by an approximate finite-difference equation on a mesh of points in time. An initial solution in the form of values associated with the mesh points is guessed. Relaxation brings these values simultaneously into close agreement with the unknown true solution.

Relaxation-type algorithms applied to differential equations have two very useful properties. First of all, they can easily cope with boundary conditions, such as initial conditions for state variables and transversality conditions of optimal growth. Second, additional equations, e.g., equilibrium conditions or feasibility constraints, can be incorporated straight away. Beyond this, by transformation of the (independent) time variable one can solve infinite-horizon problems, as they arise from many dynamic optimization problems in economics.

Suppose we want to compute a numerical solution of a differential equation in terms of a large (finite) sequence of points representing the desired path. To start with, we take an arbitrary trial solution, typically not satisfying the slope conditions implied by the differential equation nor the boundary conditions. We measure the deviation from the true path by a multidimensional error function and use the derivative of the error function to improve the trial solution in a Newton-type iteration. Hence, at each point of the path, the correction is related to the particular inaccuracy in slope and in solving the static equation. The crucial difference to the various shooting methods is the simultaneous adjustment along the path as a whole.

Figure 1 illustrates the adjustment by relaxation of a linear initial guess toward the saddle path in the Ramsey–Cass–Koopmans model. The initial guess starts with a fixed initial value of the state variable k and an arbitrary initial value of the control variable c. It consists of 30 mesh points lined up equidistantly between the starting point and the known steady state of the model. In evaluating the multidimensional error function, the algorithm realizes that the fit to the differential equation can be improved by an upward shift of the curve without jeopardizing the boundary conditions. After a few steps the error is sufficiently small and the algorithm stops.

The outline of the algorithm proposed in this paper leans on Press et al. (1989, pp. 645–672). We have implemented the algorithm in MatLab. The code is published for free download on the Internet[3] and a print version is available on request.[4]

We apply the method to the following kind of problem: Consider a system of $\tilde{N}$ ordinary differential equations together with $N - \tilde{N}$ (static) equations in N real variables. This system describes a vector field on an $\tilde{N}$-dimensional surface in $\mathbf{R}^N$. We impose a list of n_1 boundary conditions at the starting point and n_2 at the end point of a path sufficient to determine a particular trajectory. To meet all dimensional requirements, n_1 and n_2 must add up to $\tilde{N}$.

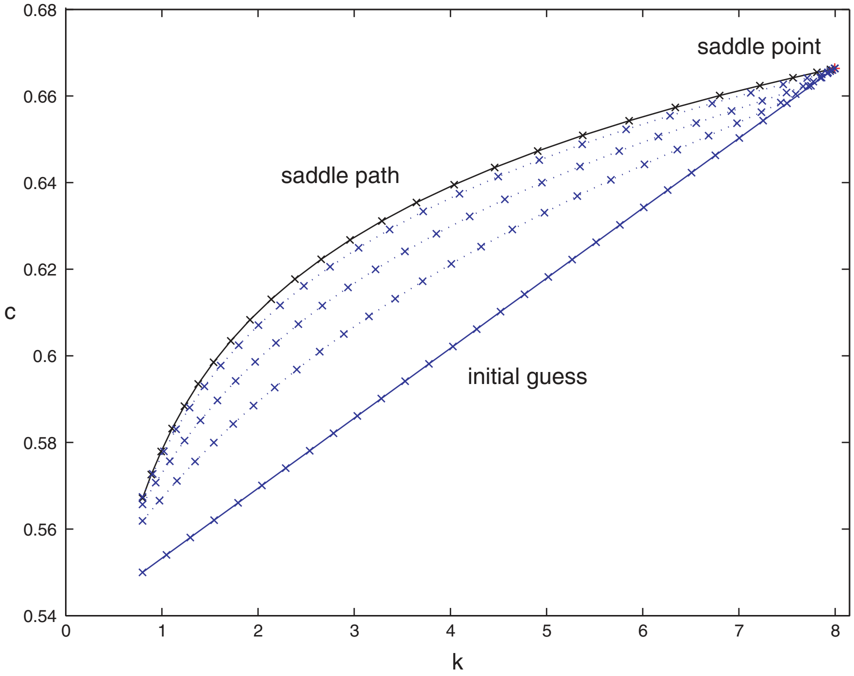

FIGURE 1. Relaxation in the Ramsey–Cass–Koopmans model.

For the finite representation of the problem, we fix a time mesh of M points in time. In the case of an infinite time horizon, we choose a transformation to map the interval $[0, \infty]$ to $[0, 1]$. At each point in time, an N-dimensional vector has to be determined. We approximate the differential equation by $M-1$ systems of equations of dimension $\tilde{N}$ for the slope between neighboring mesh points. Together with $\tilde{N}$ boundary conditions, we have an $M \times \tilde{N}$-dimensional system of equations. After adding the $N - \tilde{N}$ static equations, which have to hold at each of the M mesh points, we have incorporated all restrictions available. The final system of nonlinear equations is of dimension $M \times N$ and involves the same number of unknowns.

We apply a Gauss–Newton procedure to compute a root of this system. Step by step, we adjust the trial solution until the error is sufficiently small. This involves the solution of a linear equation with the Jacobian matrix of the system of nonlinear equations. At first glance, there seems little chance to achieve good solutions, because the complexity of the problem is proportional to the size of the Jacobian matrix, which is quadratic in M.

However, the Jacobian is not an arbitrary matrix of dimension $M \times N$. The Jacobian matrix inherits a specific structure from the approximation of the differential equation. The boundary conditions and the static equations each depend only on one respective vector, and the interior slope is conditioned only on neighboring vectors. Hence the Jacobian matrix shows nonzero entries only close to the diagonal. This can be used to solve the linear system by a special version of a Gauss algorithm carried out recursively on N-dimensional blocks along the diagonal. This recursive procedure makes it possible to increase the number M of mesh points without increasing the dimension of the blocks. Only the number of blocks increases in proportion to M. The complexity of the problem is only linear in the number of mesh points and not quadratic. Hence, a fairly good approximation of the continuous path is possible without using too much computer time.

2.2. Implementation of the Algorithm

To illustrate, we describe the steps that must be taken in implementing the relaxation algorithm using the Ramsey–Cass–Koopmans model [Ramsey (1928); Cass (1965); Koopmans (1965)] as an example. It is important to notice, however, that this description serves as an illustration only. The researcher who intends to solve a specific model numerically using the program (provided as a supplement to this paper) need not follow these steps.

It is well known that this simple growth model exhibits saddle-point stability and hence the determination of the solution is all but trivial.[5] The model gives rise to a system of two differential equations for consumption c and capital per effective labor k [Barro and Sala-i-Martin (2004, Chapter 2)],

$$\dot{c} = \frac{c}{\theta}(\alpha k^{\alpha-1} - (\delta + \rho + x\theta)) \tag{1}$$

$$\dot{k} = k^{\alpha} - c - (n + x + \delta)k, \tag{2}$$

where α denotes the elasticity of capital in production, n the population growth rate, δ the depreciation rate, x the exogenous growth rate of technology, ρ the parameter for time preference, and θ the inverse of the intertemporal elasticity of substitution, respectively. The steady state is $k^* = (\frac{\alpha}{\delta+\rho+x\theta})^{1/(1-\alpha)}$ and $c^* = (k^*)^{\alpha} - (n + x + \delta)k^*$ and is saddle-point stable.

As a first step, one must choose a time mesh, that is, a set of points in time at which the solution should be calculated. We select the time mesh to be uniform in the transformed time scale (as explained in Section 2.1).

Second, the two differential equations have to be transformed into two nonlinear equations that describe the slope between two neighboring mesh points. These equations have to be satisfied between every two mesh points. For M mesh points, this leads to $2 \cdot (M - 1)$ nonlinear equations.

Third, two boundary conditions have to be chosen to complete the set of equations to $2 \cdot M$. In this example, the relaxation algorithm needs one initial boundary condition and one terminal boundary condition. We set the initial value of the state variable (capital) equal to 10% of its steady state value. For the terminal boundary condition there are several possibilities for formulating an equation. It would be possible to choose each of the two equations (1) or (2) and set the RHS

TABLE 1. Accuracy of the relaxation algorithm for the Ramsey–Cass–Koopmans model

Number of mesh points	Max error c	Max error k	Mean error
10	$< 1.3 \times 10^{-2}$	$< 3.4 \times 10^{-2}$	$< 3.0 \times 10^{-3}$
100	$< 1.1 \times 10^{-4}$	$< 8.6 \times 10^{-5}$	$< 2.7 \times 10^{-6}$
1,000	$< 1.1 \times 10^{-6}$	$< 8.5 \times 10^{-7}$	$< 8.2 \times 10^{-9}$
10,000	$< 1.1 \times 10^{-8}$	$< 8.5 \times 10^{-9}$	$< 2.6 \times 10^{-11}$
100,000	$< 1.1 \times 10^{-10}$	$< 8.5 \times 10^{-11}$	$< 8.2 \times 10^{-14}$

equal to zero. However, here the steady-state values for consumption and capital can be computed analytically, and therefore we can set consumption equal to its steady-state value as the terminal boundary condition. It should be noted that only one terminal condition is needed. Thus the algorithm does not make use of the knowledge of the steady-state value of capital. It is reached automatically.

At last, an initial guess for the solution has to be made. For instance, we can choose c and k to be constant at their steady-state values $(c_t, k_t) \equiv (c^*, k^*)$.[6] The Newton procedure always converges quickly, indicating a high degree of robustness with respect to the initial guess.

2.3. Evaluation of the Procedure

For the special parameterization $\theta = \frac{\delta+\rho}{\alpha(\delta+n+x)-x}$, the representative consumer chooses a constant saving rate $s = \frac{1}{\theta}$, and hence the solution can be expressed analytically [Barro and Sala-i-Martin (2004, pp. 106–110)].[7] This allows us to compare the computed results with the analytical solution, which has a precision close to the machine epsilon. The relative error is computed for every mesh point. Table 1 shows the maximum relative error of consumption and capital per unit of effective labor for different numbers of mesh points. In addition, the quadratic mean error of combined c and k provides information about the distribution of the error.[8] Table 1 reveals that multiplying the number of mesh points by x reduces the maximum error of each solution vector by the factor $\frac{1}{x^2}$, which indicates the order 2 of the difference procedure. Even with a moderate number of mesh points and therefore a short computation time, a sufficiently high degree of accuracy can be achieved. Moreover, the accuracy can be improved to a very high degree by increasing the number of mesh points.[9] The treatment of higher-dimensional systems with multidimensional stable manifolds is largely analogous to the example described above. This is why the algorithm performs similarly well for more complicated models.

3. COMPARISON TO OTHER PROCEDURES

The relaxation procedure and similar finite-difference procedures have already been employed in various fields of economics. Prominent examples comprise the solution of two-point boundary-value difference equations [e.g., Laffargue (1990); Juillard et al. (1998)], differential–difference equations [e.g., Boucekkine et al. (1997)], and partial differential equations [e.g., Candler (1999)].

However, to the best of our knowledge, the relaxation algorithm has not yet been exploited systematically to solve deterministic continuous-time two-point boundary-value problems in growth theory. Nonetheless, there are a few applications in the economics literature. For instance, Oulton (1993) and Robertson (1999) employ the relaxation routine provided by Press et al. (1989) to solve a continuous-time deterministic growth model.

We compare the relaxation procedure to the most popular alternative solution methods employed in deterministic growth theory. These comprise backward integration [Brunner and Strulik (2002)], the finite-difference method as proposed by Candler (1999), time elimination [Mulligan and Sala-i-Martin (1991)], projection methods [e.g., Judd (1992); Judd (1998, Chapter 11)], and the method of Mercenier and Michel (1994, 2001). This section is kept brief because most of the procedures and their relative advantages are described in Judd (1998) and Brunner and Strulik (2002).

Finite-difference methods as described by Candler (1999) employ an algorithm similar to the relaxation procedure to solve partial differential equations. In a first step, the Bellman equation associated with the maximization problem under study is derived and stated as a partial differential equation. In a second step, this equation is solved numerically. For the solution of this initial-value problem, the equation is integrated in time until the solution is no longer time-dependent. The strength of this procedure lies in the fact that, by stating the Bellman equation as a partial differential equation, it can easily be extended to solve higher-dimensional problems as well as stochastic models. This approach is conceptually different from our procedure, because we apply the relaxation algorithm to the set of ordinary differential equations derived from Pontryagin's maximum principle, which has been predominantly employed in the analysis of continuous-time deterministic growth models. Although it is straightforward to extend the finite-difference method to higher-dimensional systems, computational costs grow substantially with additional state variables.[10]

Backward integration, as suggested by Brunner and Strulik (2002), exploits the numerical stability of the backward-looking system by inverting time. By starting near the steady state of the transformed system, the resulting initial-value problem is stable and the solution converges toward the stable manifold of the forward-looking system quickly. This method can solve systems with one-dimensional stable manifolds efficiently and conveniently. Moreover, it is very intuitive and can also be applied to solve discrete-time models [Strulik (2004)]. For multi-dimensional manifolds Brunner and Strulik (2002) suggest generating starting values on an orbit around the steady state. To pass through a prespecified point (determined by the specific shock under study), it is necessary to iterate until the trajectory hits this point. However, if the real parts of the stable eigenvalues differ substantially, the problem of stiff differential equations occurs. It is well known that these problems are very hard to handle numerically. For large differences between the stable eigenvalues, it is impossible to meet the prespecified point, because the backward-directed trajectories will be attracted by the submanifold, which is associated with the eigenvalue with the largest, in absolute terms, real part. The resulting trajectories hence cannot represent a specified shock and potentially have no economic meaning. Furthermore, if there exists a continuum of steady states represented by a (saddle-point stable) center manifold, then the specific steady state to which the economy converges depends on the initial boundary conditions.[11] If one particular steady state is chosen for backward integration, then only one initial condition can be satisfied. To find a trajectory that fulfills all initial conditions, an iteration process has to be applied. This procedure typically gives rise to problems of convergence.

Mercenier and Michel (1994, 2001) propose to transform the continuous-time, infinite-horizon problem into a finite-horizon maximization problem in discrete time with the same steady state or balanced-growth rate. The transformed problem can be solved with a static optimization procedure. Applying the necessary and sufficient conditions stated by Mercenier and Michel (1994, 2001) yields a considerable improvement in the numerical accuracy of the discrete-time model. Moreover, Alemdar et al. (2006) show that the overall optimization performance can be improved substantially if an optimal allocation of the time mesh is chosen for the transition. Our approach is to solve the system of differential equations directly. Here the discretization is done at a later stage. To apply the relaxation algorithm the researcher simply has to insert the differential equations into the program code, instead of converting the complete maximization problem. Apart from simplicity, the relaxation algorithm has some further advantages.

First, the proposed version of the relaxation algorithm can deal with a compactification of the time interval. It is not necessary to choose an adequate terminal time where the optimization is truncated. Also, the treatment of a post-terminal stationary phase does not apply. Second, the relaxation algorithm leaves room for selecting different discretization rules, also of higher order. This leads to a higher level of accuracy with the same number of mesh points. The discretization rule of the method of Mercenier and Michel is a first-order rule, whereas the relaxation procedure uses a second-order rule.[12]

Projection methods, introduced in Judd (1992) and Judd (1998, Chapter 11), cover a wide range of algorithms. Therefore, they can be applied to a large number of numerical problems. For many applications, they prove to be fast and accurate, but also require high programming effort. Moreover, they are usually applied to solve for the policy function. However, if the model exhibits nonmonotonic adjustments, the policy function cannot be computed at the extremal points. Furthermore, if there exists a continuum of steady states represented by a center manifold, the interval of integration is not known in advance, because it depends on the final steady state to which the economy converges. In this case, projection methods appear to be inappropriate. In addition, the polynomial bases and therefore the computation costs grow exponentially when the dimension of the problem increases. To avoid this “curse of dimensionality,” a special complete polynomial basis is chosen, but still the computation costs grow considerably.

Time elimination is easy and intuitive. However, it is also plagued by some of the disadvantages already mentioned. First, in the case of nonmonotonic adjustments, the policy function cannot be computed at the extremal points and, second, if there exists a continuum of steady states, the interval of integration is unknown.

4. TWO ILLUSTRATIVE APPLICATIONS

The relaxation procedure is employed to investigate the transition process of two prominent growth models. As a first example, we consider the Jones (1995) model. For usual calibrations this model gives rise to a system of stiff differential equations. The four-dimensional transition toward the unique steady state appears to be nonmonotonic. The second example, the Lucas (1988) model, implies a saddle-point stable center manifold. The different points on this curve reflect level effects of transition toward long-run growth.

It should be noted that the transition process of these popular growth models has hardly been investigated in detail so far, which is probably due to the conceptual problems mentioned above. Moreover, the numerical analysis of the Lucas model yields a number of interesting insights, which have not yet been discussed in the literature. For this reason we devote more space to the discussion of this model.

4.1. The Jones (1995) Model

The technology for final output Y is given by $Y = \alpha_F(\phi L)^{\sigma_L} \int_0^A x(i)^{1-\sigma_L} di$, where ϕ denotes the share of labor allocated to final-output production, $x(i)$ the amount of differentiated capital goods of type i, A the number of differentiated capital goods, α_{F} a constant overall productivity parameter, and σ_L the elasticity of labor in final-output production. Noting the general symmetry among $x(i)$ and using the definition of aggregate capital $K := Ax$, the final-output technology can be written as $Y = \alpha_{\mathrm{F}}(A\phi\mathrm{L})^{\sigma_L} K^{1-\sigma_{\mathrm{L}}}$. The R&D technology is $\dot{A} = J = \alpha_J A^{\eta_A}[(1-\phi)L]^{\eta_L}$ with $\eta_L := \eta_L^p + \eta_L^e, \eta_L^p = 1, -1 < \eta_L^e < 0$, where $\dot{A} := dA/dt, \alpha_J$ denotes a constant overall productivity parameter, η_A the elasticity of technology in R&D, and η_L the elasticity of labor in R&D.

The dynamic system that governs the evolution of the economy under study can be summarized as[13]

$$\dot{k} = y - c - \delta k - \beta_K n k \tag{3}$$

$$\dot{a} = j - \beta_A n a \tag{4}$$

$$\dot{c} = \frac{c}{\gamma}[r - \delta - \rho - (1-\gamma)n] - \beta_K n c \tag{5}$$

$$\dot{v}_a = v_a[r - (\beta_K - \beta_A)n] - \pi \tag{6}$$

$$\frac{\sigma_L y}{\phi} = v_a \frac{\eta_L^p j}{1-\phi}, \tag{7}$$

where $y = \alpha_F (a\phi)^{\sigma_L} k^{1-\sigma_L}$, $j = \alpha_J a^{\eta_A} (1 - \phi)^{\eta_L}$, $r = \frac{(1-\sigma_L)^2 y}{k}$, $\pi = \frac{\sigma_L(1-\sigma_L)y}{a}$, $\beta_K = \frac{1-\eta_A+\eta_L}{1-\eta_A}$, and $\beta_A = \frac{\eta_L}{1-\eta_A}$. Note that the dynamic system is expressed in scale-adjusted variables, which are defined by $y := Y/L^{\beta_K}$, $k := K/L^{\beta_K}$, $c := C/L^{\beta_K}$, $a := A/L^{\beta_A}$, $j := J/L^{\beta_A}$, and $v_a := v/L^{\beta_K - \beta_A}$. The (unique) stationary solution of this dynamic system corresponds to the (unique) BGP of the economy expressed in original variables.

Equations (3) and (4) are the equations of motion of (scale-adjusted) capital and technology, (5) is the Keynes–Ramsey rule of optimal consumption c, (6) shows capital market equilibrium, with v_a denoting the (scale-adjusted) price of blueprints, and (7) determines the privately efficient allocation of labor across final-output production and R&D.[14]

The objective is to solve the four-dimensional system of differential equations (3)–(6), taking into account the static equation (7), which must hold at all points in time. The steady state is a saddle point with a two-dimensional stable manifold. Because the steady state can be determined numerically only, the algorithm computes the steady state of the system first by applying a Newton algorithm. The choice of $k(0) = k_0$ and $a(0) = a_0$ as initial boundary conditions is obvious because k and a are the state variables. Again, there is some freedom when it comes to the determination of boundary conditions. We have set the RHS of equations (5) and (6) equal to zero. Moreover, we choose once more, as an initial guess, all variables to be constant at their steady-state values. This always lead to quick convergence, indicating that the procedure is relatively robust with respect to the initial guess.

The transition process considered below results from a combination of two simultaneous shocks. Specifically, it is assumed that the overall productivity parameter in the production function for final output α_F increases from 1.0 to 1.3, whereas the overall productivity parameter in the production function for new ideas α_J decreases from 1.0 to 0.9. This shock was chosen to demonstrate that the adjustment can be nonmonotonic [as can be recognized by inspecting Figure 2 (vi)] and therefore the policy functions at certain points cannot be computed with conventional methods.[15]

Figure 2 gives a summary of the adjustment process. The plots (i) to (iii) show the time path of the jump variables c, ϕ, and v_a, plots (iv) and (v) display the time path of the state variables k and a, and plot (vi) gives the projection of the adjustment trajectory into the (k, a)-plane.

4.2. The Lucas (1988) Model

The second example is the model of Lucas (1988), as discussed by Mulligan and Sala-i-Martin (1993), Caballe and Santos (1993), Benhabib and Perli (1994), and others. The long-run equilibria of the scale-adjusted version of this endogenous growth model form a center manifold. The different points on this curve reflect different levels of long-run growth. From the numerical solution converging to such a point we can read all relevant information for the comparison of different

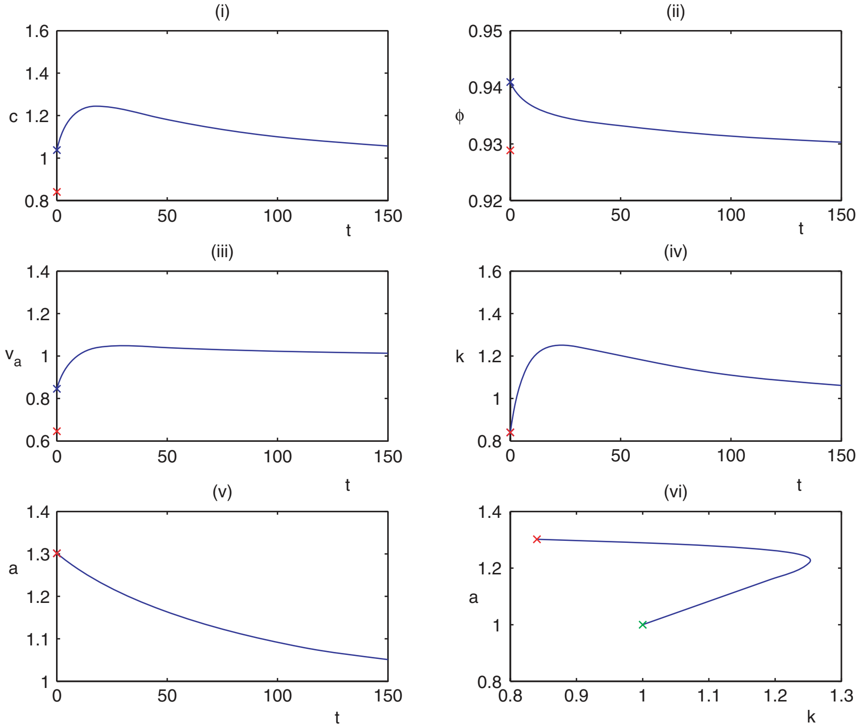

FIGURE 2. Summary of the transition of the Jones (1995) model.

initial states. For example, we obtain initial optimal consumption levels right away and can easily compute utility integrals.

Assume that final output is produced from physical and human capital, k and h. The stock of human capital can be split into a share u used for final output production and a share $1 - u$ employed to increase human capital. Because of human capital spillover effects, there are increasing returns to scale in the production sector. Intertemporal utility of consumption c with constant elasticity of intertemporal substitution σ^{-1} and discount rate ρ is to be maximized. First-order conditions for optimal solutions can be computed in the usual way. In terms of growth rates (denoted by a circum flex), the system is

$$\hat{k} = APK - c/k \tag{8}$$

$$\hat{h} = \delta(1 - u) \tag{9}$$

$$\hat{c} = \sigma^{-1}(\alpha APK - \rho) \tag{10}$$

$$\hat{u} = \frac{(\gamma - \alpha)\delta}{\alpha}(1 - u) + \frac{\delta}{\alpha} - \frac{c}{k}, \tag{11}$$

where $APK := A\,k^{\alpha-1}h^{1-\alpha+\gamma}u^{1-\alpha}$ denotes the average productivity of capital.

Balanced growth requires that u and c/k as well as APK be constant. The latter requirement in turn demands that $(1-\alpha)\hat{k} = (1-\alpha+\gamma)\hat{h}$.

The common balanced-growth rate μ of k and c can be computed by solving the system under balanced-growth assumptions:

$$\mu = \frac{1-\alpha+\gamma}{(1-\alpha+\gamma)\sigma-\gamma}(\delta-\rho).$$

Growth is balanced if the four variables of the system satisfy the three equations

$$\begin{aligned} 1-u &= \frac{1-\alpha}{(1-\alpha+\gamma)\sigma-\gamma}(1-\rho/\delta) \\ c/k &= ((\gamma-\alpha)\psi\mu+\delta)/\alpha \\ k^{\alpha-1}h^{1-\alpha+\gamma} &= \frac{\sigma\mu+\rho}{\alpha A}(u^*)^{\alpha-1}, \end{aligned}$$

where $\psi := (1-\alpha)/(1-\alpha+\gamma)$. The question arises of whether other solutions initially suffering from unbalancedness converge to a BGP. One method for checking whether convergence occurs is scale adjustment. Scale adjustment slows the motion of variables according to their respective balanced-growth rates. The transformed variables are

$$ke^{-\mu t}, \quad he^{-\psi\mu t}, \quad ce^{-\mu t}, \quad \text{and} \quad u.$$

To avoid extra notation we continue to use the old designations of variables. The new, adjusted growth rates are reduced by the constants of adjustment μ and $\psi\mu$, respectively. The growth rate of u remains unchanged. Due to scale adjustment, the BGP of the original system [shown in Figure 3 (i)] turns into a curve representing a continuum of stationary equilibria, which is labeled CSE [displayed in Figure 3 (ii)] with the same shape. This curve represents a (saddle-point stable) center manifold of the new system.[16] An optimal solution with unbalanced initial state conditions (k_0, h_0) now approaches a particular point on the curve CSE. Yet there is no way to compute this point analytically.

Numerical computation requires the solution of a differential equation system with two initial conditions and two final conditions.[17] The initial conditions are given by the initial values of the state variables $k(0) = k_0$, $h(0) = h_0$. Final conditions that determine the path, and work well with the relaxation algorithm, are stationarity conditions for the state variables, implicitly defined by $\dot{k}(\infty) = 0$ and $\dot{h}(\infty) = 0$.

By numerical solution of the scale-adjusted model we can now answer the following type of question: Consider two economies (1 and 2) differing in their initial states (k_0^1, h_0^1) and (k_0^2, h_0^2) only. Will they converge to the same point on the CSE? Or will, alternatively, one economy have a permanent advantage in the sense of exhibiting a higher level of consumption along the BGP? Figure 3 illustrates such a situation, where the solid trajectories display a development

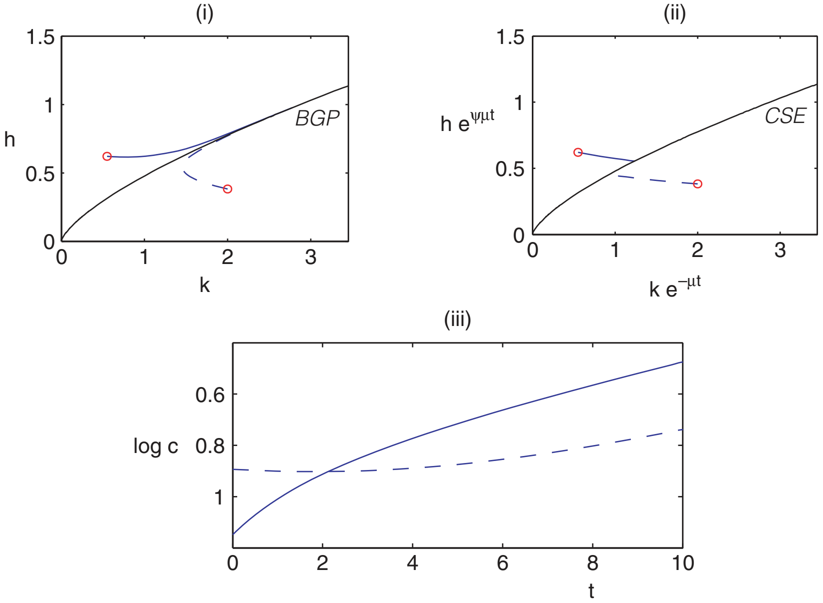

FIGURE 3. Summary of the transition of the Lucas (1988) model.

implying a higher long-run consumption level, as can be recognized by inspecting Figure 3 (iii). The value of the utility integrals amounts to -4.81 for the solid path and -9.22 for the broken one.

5. SUMMARY

We propose the relaxation algorithm as a powerful and efficient procedure for investigating the transition process of continuous-time growth models. At a very general level, this method has two main advantages: First, it is simpler than most other procedures. Second, and more importantly, the relaxation procedure can easily deal with complex dynamic systems for which conventional algorithms appear to be inappropriate. Specifically, the relaxation procedure can easily handle stiff differential equations as well as dynamic systems giving rise to saddle-point stable center manifolds. It has been demonstrated that this type of systems result from basic workhorse models in growth theory. Finally, it is important to notice that the relaxation algorithm can easily deal with highly dimensional dynamic systems. Potential applications comprise models with heterogeneous agents as well as computable general equilibrium (CGE) models. Moreover, the application of the method is restricted neither to growth models nor to infinite-horizon problems. Especially in the field of CGE modeling, it seems that researchers often make extensive use of simplifying assumptions to keep the analysis tractable; for instance, the assumption of a constant saving rate in dynamic models. We think that the relaxation procedure enables researchers to relax most of these simplifying assumptions in a convenient and efficient manner.[18]

NOTES

[1] In the case of saddle-point stability, the dimension of the stable manifold equals the dimension of the state space, whereas indeterminacy implies that the dimension of the stable manifold exceeds the dimension of the state space.

[2] Moreover, for continuous-time models, discretization schemes can be applied that yield a higher order of accuracy.

[3] http://www.relaxation.uni-siegen.de.

[4] In the Appendix we give a detailed description of the algorithm.

[5] Nonetheless, the model is comparably simple in that the stable manifold is one-dimensional. We will turn to a model with a multidimensional stable manifold below.

[6] This is in contrast to Figure 1, where the initial guess is an upward sloping line.

[7] The analytical solution is $k(t) = [\frac{1}{(\delta+n+x)\theta} + (k_0^{1-\alpha} - \frac{1}{(\delta+n+x)\theta})e^{-(1-\alpha)(\delta+n+x)t}]^{1/(1-\alpha)}$ and $c(t) = (1 - \frac{1}{\theta})k(t)^\alpha$.

[8] It is defined as $\varepsilon = \frac{1}{NM}\sqrt{\sum_{i=1}^N \varepsilon_{c_i}^2 + \sum_{i=1}^N \varepsilon_{k_i}^2}$, with ε_{c_i} and ε_{k_i} denoting the relative error of k and c at mesh point i, respectively.

[9] It should be mentioned that the allocation of the mesh was chosen exogenously. The accuracy of the algorithm could be improved with a self-allocating time mesh, as proposed by Press et al. (1989, Chapter 16.5). They suggest automating the allocating of mesh points so that more mesh points are placed in regions in which the variables are changing rapidly.

[10] Press et al. (1989, p. 700) report that computational requirements increase by a factor of at least 100 if one switches from a one-dimensional problem with 100 grid points to a two-dimensional problem with 100 × 100 grid points.

[11] For instance, in the Lucas (1988) model presented below, the actual steady state to which the economy converges depends on the initial levels of human and physical capital, h_0 and k_0.

[12] When the number of mesh points is multiplied by x, a first-order rule leads to a reduction of the global error by $\frac{1}{x}$, whereas a second-order rule reduces the error by $\frac{1}{x^2}$.

[13] The presentation of the Jones (1995) model basically follows Eicher and Turnovsky (1999), who formulated the social planner's solution of the general nonscale R&D-based growth model. For a detailed derivation of the decentralized solution see Steger (2005).

[14] The presence of the static efficiency condition (equation (7)) is due to the fact that labor enters neither final output nor R&D linearly. Hence, it is in general not possible to solve for the optimal amount of labor explicitly. Note that the presence of an additional algebraic condition such as equation (7) does not cause any problems for the relaxation algorithm.

[15] The set of parameters used for the numerical solution is $\sigma_L = 0.6$, $\sigma_K = 0.4$, $\delta = 0.05$, $n = 0.015$, $\eta_A = 0.6$, $\eta_L = 0.5$, $\eta_L^p = 0.6$, $\rho = 0.04$, and $\gamma = 1$. In this case, set the ratio of the stable eigenvalue amounts to 12.8.

[16] The scale-adjusted system has one zero eigenvalue, which gives rise to a continuum of stationary equilibria (i.e., a center manifold). For details on the basic concept of center manifolds see, for instance, Tu (1994, pp. 187–191).

[17] The set of parameters used for the numerical solution of the Lucas model is $A = 1$, $\alpha = 0.3$, $\delta = 0.1$, $\gamma = 0.3$, $\sigma = 1.5$, and $\rho = 0.05$. With this set of parameters indeterminacy is excluded (c.f. Benhabib and Perli, 1994).

[18] Feraboli and Trimborn (2006) employ the relaxation algorithm to solve a highly dimensional CGE model with heterogenous households exhibiting endogenous saving rates. The phase space can be divided into a six-dimensional stable manifold and a six-dimensional unstable manifold.

REFERENCES

Alemdar, N.M., S. Sirakaya, and F. Hüsseinov (2006) Optimal time aggregation of infinite horizon control problems. *Journal of Economic Dynamics and Control* 30, 569–593.

Barro, R.J. and X. Sala-i-Martín (2004) *Economic Growth*. Cambridge, MA: MIT Press.

Benhabib, J. and R. Perli (1994) Uniqueness and indeterminacy: On the dynamics of endogenous growth. *Journal of Economic Theory* 63, 113–142.

Boucekkine, R., O. Licandro, and C. Paul (1997) Differential–difference equations in economics: On the numerical solution of vintage capital growth models. *Journal of Economic Dynamics and Control* 21, 347–362.

Brunner, M. and H. Strulik (2002) Solution of perfect foresight saddlepoint problems: A simple method and applications. *Journal of Economic Dynamics and Control* 26, 737–753.

Caballe, J. and M.S. Santos (1993) On endogenous growth with physical and human capital. *Journal of Political Economy* 101(6), 1042–1067.

Candler, G.V. (1999) Finite-difference methods for continuous-time dynamic programming. In R. Marimon and A. Scott (eds.), *Computational Methods for the Study of Dynamic Economies*. Oxford, UK: Oxford University Press.

Cass, D. (1965) Optimum growth in an aggregate model of capital accumulation. *Review of Economic Studies* 32, 233–240.

Eicher, T.S. and S.J. Turnovsky (1999) Convergence in a two-sector nonscale growth model. *Journal of Economic Growth* 4, 413–428.

Feraboli, O. and T. Trimborn (2006) Trade Liberalization and Income distribution: A CGE Model for Jordan. Discussion Paper, University of Hamburg.

Jones, C.I. (1995) R&D-based models of economic growth. *Journal of Political Economy* 103(3), 759–784.

Judd, K.L. (1992) Projection methods for solving aggregate growth models. *Journal of Economic Theory* 58, 410–452.

Judd, K.L. (1998) *Numerical Methods in Economics*. Cambridge, MA: MIT Press.

Juillard, M., D. Laxton, P. McAdam, and H. Pioro (1998) An algorithm competition: First-order iterations versus Newton-based techniques. *Journal of Economic Dynamics and Control* 22(8/9), 1291–1319.

Koopmans, T.C. (1965) On the concept of optimal economic growth. In *The Econometric Approach to Development Planning*. Amsterdam: North Holland.

Laffargue, J.-P. (1990) Résolution d'un modèle économique avec anticipations rationelles. *Annales d'Economie et Statistique* 17, 97–119.

Lucas, R.E., Jr. (1988) On the mechanics of economic development. *Journal of Monetary Economics* 22, 3–42.

Mercenier, J. and P. Michel (1994) Discrete-time finite horizon approximation of infinite horizon optimization problems with steady-state invariance. *Econometrica* 62(3), 635–656.

Mercenier U. and P. Michel (2001) Temporal aggregation in a multi-sector economy with endogenous growth. *Journal of Economic Dynamics and Control* 25, 1179–1191.

Mulligan, C. and X. Sala-i-Martin (1991) A Note on the Time Elimination Method for Solving Recursive Growth Models. *National Bureau of Economic Research*, Technical Working Paper No. 116.

Mulligan, C. and X. Sala-i-Martin (1993) Transitional dynamics in two-sector models of endogenous growth. *Quarterly Journal of Economics* 108(3), 737–773.

Oulton, N. (1993) Widening the human stomach: The effect of new consumer goods on economic growth and leisure. *Oxford Economic Papers* 51, 577–594.

Press, W., B. Flannery, S. Teukolsky, and W. Vetterling (1989) *Numerical Recipes in Pascal*. Cambridge: Cambridge University Press.

Ramsey, F. (1928) A mathematical theory of saving. *Economic Journal* 38, 543–559.

Robertson, P.E. (1999). Economic growth and the return to capital in developing economies. *Oxford Economic Papers* 51, 269–283.

Romer, P.M. (1990) Endogenous technological change. *Journal of Political Economy* 98, S71–S102.

Steger, T.M. (2005) Welfare implications of non-scale R&D-based growth models. *Scandinavian Journal of Economics* 107(4), 737–757.

Strulik, H. (2004) Solving rational expectations models using excel. *Journal of Economic Education* 35, 269–283.

Temple, J. (2003) The long-run implications of growth theories. *Journal of Economic Surveys* 17(3), 497–510.

Tu, P.N. (1994) *Dynamical Systems, An Introduction with Applications in Economics and Biology*. Berlin: Springer-Verlag.

APPENDIX

In this section we go through some details of the algorithm. Consider a system of $\tilde{N}$ differential equations on an open set in $\mathbf{R}^N$, with $\tilde{N} \leq N$. Let $\tilde{x}$ be the vector of those components of the full vector $x \epsilon \mathbf{R}^N$ affected by f:

$$\frac{d\tilde{x}}{dt} = f(t, x), \quad f : \mathbf{R}_+ \times \mathbf{R}^N \to \mathbf{R}^{\tilde{N}}.$$

If $\tilde{N}$ is strictly smaller than N, the differential equations are to be supplemented by $N - \tilde{N}$ equations that x has to satisfy at any time:

$$0 = g(t, x), \quad g : \mathbf{R}_+ \times \mathbf{R}^N \to \mathbf{R}^{N-\tilde{N}}.$$

Boundary conditions are supposed to be given in the form of n_1 initial conditions and n_2 final conditions. For the solution to be well determined, we need $n_1 + n_2$ to equal $\tilde{N}$. Finally, it is convenient to denote the codimension $N - \tilde{N}$ of the manifold given by $g(t, x) = 0$ by n_3. Summing up, we have

$$\left.\begin{array}{ll} n_1 & \text{initial conditions} \\ n_2 & \text{final conditions} \\ n_3 & \text{running equations} \end{array}\right\} \quad \text{with } n_1 + n_2 + n_3 = \tilde{N} + n_3 = N.$$

For convenience, we rescale the time range $\mathbf{R}_+$ by introducing a new time parameter τ running from 0 to 1:

$$\tau = \nu t/(1 + \nu t) \quad \nu \in (0, \infty).$$

In terms of τ we get an equivalent differential–algebraic system,

$$\begin{aligned} \frac{d\tilde{x}}{d\tau} &= \xi(\tau, x) = f\left(\frac{\tau}{\nu(1-\tau)}, x\right) \Big/ \nu(1-\tau)^2 \\ 0 &= \phi(\tau, x) = g\left(\frac{\tau}{\nu(1-\tau)}, x\right). \end{aligned} \qquad \textbf{(A.1)}$$

Define a mesh of M points in (transformed) time τ by $T = \{\tau_1, \ldots, \tau_M\}$. Along the mesh, the dependent variable x falls into a list of vectors. To avoid confusion, we denote it by

$y = \{y_1, \ldots, y_M\}$, where y_k is the value of x at τ_k. We use the midpoint of each interval (τ_k, τ_{k+1}) for the discretization of the differential equation

$$\tilde{y}_{k+1} - \tilde{y}_k = (\tau_{k+1} - \tau_k)\ \xi(\bar{\tau}_k, \bar{y}_k) \quad \text{for } k = 1, \ldots, M-1, \tag{A.2}$$

where $\bar{\tau}_k = (\tau_k + \tau_{k+1})/2$ and $\bar{y}_k = (y_k + y_{k+1})/2$. An element of this sequence of difference equations yields an $\tilde{N}$-dimensional error function $H : ([0, \ldots, 1] \times \mathbf{R}^N)^2 \to \mathbf{R}^{\tilde{N}}$,

$$H(\tau_k, y_k, \tau_{k+1}, y_{k+1}) = \tilde{y}_{k+1} - \tilde{y}_k - (\tau_{k+1} - \tau_k)\xi(\bar{\tau}_k, \bar{y}_k).$$

Note that the matrix of partial derivatives of H with respect to y_k and y_{k+1} differ only in their derivatives of $\tilde{y}_{k+1}$ and $\tilde{y}_k$, respectively, and this is plus or minus the identity matrix of dimension $\tilde{N}$.

Let B denote the initial conditions

$$B : \mathbf{R}^N \to \mathbf{R}^{n_1},$$

let F denote the final conditions

$$F : \mathbf{R}^N \to \mathbf{R}^{n_2},$$

and let C denote the running conditions

$$C : [0, \ldots, 1] \times \mathbf{R}^N \to \mathbf{R}^{n_3}.$$

All together this defines a system of equations in $y = (y_1, \ldots, y_M)\epsilon\mathbf{R}^{N \cdot M}$ given a mesh $\tau = (\tau_1, \ldots, \tau_M)\epsilon\mathbf{R}^M$, and we are looking for a root of this system.

For the description of the algorithm it is convenient to list the equations according to the unknown vectors y_k involved. We start with the initial conditions, which only involve y_1, and end with the equations that only involve y_M. Ordered this way, the system can be seen as a system of $M+1$ vector equations $E_0(y), \ldots, E_M(y)$. The first subsystem, $E_0(y)$, depends only on y_1 and consists of n_1 initial conditions. The intermediate subsystems $E_k(y)$ for $k = 1, \ldots, M-1$ depend on y_k and y_{k+1} and are of dimension N. Each of these subsystems begins with n_3 running conditions and is completed by $n_1 + n_2$ difference equations. The last subsystem, $E_M(y)$, depends on y_M and consists of n_3 interior conditions together with n_2 final conditions. It has dimension $n_2 + n_3$:

$$E(y) \equiv \begin{pmatrix} E_0(y) \\ \vdots \\ E_k(y) \\ \vdots \\ E_M(y) \end{pmatrix} = \begin{pmatrix} (\,B(y_1)\,) \\ \vdots \\ \begin{pmatrix} C(y_k) \\ H(y_k, y_{k+1}) \end{pmatrix} \\ \vdots \\ \begin{pmatrix} C(y_M) \\ F(y_M) \end{pmatrix} \end{pmatrix}. \tag{A.3}$$

Each step of the Newton algorithm applied to $E(y) = 0$ computes a change Δy by solving the linear equation

$$D_y E(y) \cdot \Delta y = -E(y).$$

Due to the ordering of subsystems E, this equation is of the following form:

$$\begin{pmatrix} S^{0,R} & & & & & \\ S^{1,L} & S^{1,R} & & & 0 & \\ & S^{2,L} & S^{2,R} & & & \\ & & & \ddots & & \\ & 0 & & & S^{M-1,L} & S^{M-1,R} \\ & & & & & S^{M,L} \end{pmatrix} \begin{pmatrix} \Delta y_1 \\ \vdots \\ \Delta y_M \end{pmatrix} = \begin{pmatrix} -E_0(y) \\ \vdots \\ -E_M(y) \end{pmatrix}. \qquad \textbf{(A.4)}$$

All $S^{k,L}$ and $S^{k,R}$ are Jacobian matrices defined by

$$S^{k,L} = \frac{\partial E_k(y)}{\partial y_k}, \quad \text{and} \quad S^{k,R} = \frac{\partial E_k(y)}{\partial y_{k+1}}.$$

The upper left matrix $S^{0,R}$ has n_1 rows and the lower right matrix $S^{M,L}$ only n_3+n_2, whereas all other matrices $S^{k,L}$ and $S^{k,R}$, resp., are $N \times N$. Hence, the system is not overdetermined. The solution Δy can be computed by a specialized Gaussian algorithm. This algorithm starts in the upper left corner of the matrix and works downward block by block to the lower right corner. The result is a system in upper triangular form with a sequence of $N \times (n_2 + n_3)$ nonzero blocks above the diagonal. Finally, the vector Δy can computed from bottom to top. To be more precise:

step 0: Diagonalize the first n_1 columns of $S^{0,R}$.
step k, $k=1,\ldots,M-1$: Eliminate the first n_1 columns of $S^{k,L}$;
diagonalize the remainder of $S^{k,L}$ together
with the first n_1 columns of $S^{k,R}$.
step M: Eliminate the first n_1 columns of $S^{M,L}$;
Diagonalize the remainder of $S^{M,L}$.
step $M+k$, $k=1,\ldots,M$: Solve for Δy_{M+1-k}.

The Newton algorithm refines the current guess of y by adding Δy or a fraction of this vector to y. The algorithm stops if the error E is sufficiently small according to an appropriate norm.