Abstract: The paper presents an algorithm that solves two-point boundary-value problems that arise in economic control models in continuous time with an infinite time horizon and several state variables. The algorithm can determine optimal trajectories that converge to an isolated equilibrium point. It therefore provides a numerical solution to a large class of problems for which no solvers were yet available.

Key words: Two-point boundary-value problem, infinite time horizon, dynamical optimization, numerical solution.

JEL classification: C63, C61, Q2.

1 Introduction

"Pure economic theory has undergone a revolution of thought – from statical to dynamic modes", wrote Samuelson in [1947]. The analysis of economic dynamics has since become a main pillar of economic theory and economic education. The problems to be modelled have become more complex and many can only be solved numerically.

It is then surprising that for a large class of problems, namely infinite-horizon optimal-control models, there exist no algorithms or programs that allow to solve them numerically. In a recent article in this journal, Goffe (1993) presented a "user's guide to the numerical solution of two-point boundary value problems arising in continuous time dynamic economic models" but, apparently, was unable to present algorithms or programs that would deal with an infinite time horizon.

The aim of the present note is to (partially) fill this gap and to present an algorithm suitable for the solution of a large class of infinite-horizon problems that are common in economics. To put it more precisely: we consider problems where the solution of the optimal-control problem approaches an isolated equilibrium point. We present a `FORTRAN` subroutine `OPTTRJ`, which solves boundary value problems on $[0, \infty)$, including the determination of an equilibrium x_∞ and the eigenvalues of the linearization at the equilibrium. It serves as an easy-to-use interface to sophisticated public-domain software. The complete software package can be obtained on request from the authors.

The paper is organized as follows: we present the general structure of the problems to be solved in section 2. In section 3 the algorithm is discussed, and in section 4 we present a number of examples together with the results obtained with the

†Department of Mathematics, University of Oldenburg, D-26111 Oldenburg, Germany, (kunkel@uni-oldenburg.de)

‡Department of Economics, University of Oldenburg, D-26111 Oldenburg, Germany (vdhagen@uni-oldenburg.de)

implementation of our algorithm. Appendix A finally contains a brief manual of the code.

2 Optimal-control modelling

The focus of this paper are dynamical optimization problems satisfying the following assumptions:

- an infinite time horizon,
- continuous time,
- no (relevant) constraints on controls or state variables,
- an asymptotic isolated equilibrium point, implying nonlinearity with respect to the controls,[1] and
- a hyperbolic equilibrium (where the Jacobian has nonzero determinant and has no eigenvalues with a vanishing real part).

Even though it may appear ludicrous for mortals to consider an infinite time horizon, there are two important arguments that support such a modelling assumption. First, it makes perfect sense to maximize the present value of a project (e.g. a firm) that may exceed the owner's lifespan because the project can be sold in the market at any time. The second argument relates to the fact that the aspect of reality that such an approach tries to capture is the absence of a terminal time. In discounted problems the faraway future is of no importance, but the absence of a terminal time is. It is, therefore, common in economics to consider infinite horizon models.

[1]If the problem is linear in u, impulse, bang-bang or singular controls usually allow an equilibrium to be reached in finite time. A solution to such a problem is possible with the algorithms presented in Goffe (1993) as well as the `FORTRAN` subroutine `COLCON` (see Bader and Kunkel, 1989), which is part of the package described below. It will therefore not be dealt within this paper.

The general structure of economic optimization problems in this context consists of an n-vector of state variables $x[t]$, an m-vector of control variables $u[t]$, an objective functional and differential equations that govern the changes of x over time. Formally, the optimization problem considered can be written as

$$\max_{u[t]} \int_0^\infty e^{-\delta t} F[x,u]\, dt \tag{1}$$

$$\text{s.t.} \quad \dot{x} = f[x,u], \tag{2}$$

$$x[0] = x_0. \tag{3}$$

The standard tool used today to solve such an optimization problem is the theory of optimal control that became available following Pontryagin's ([1964]) work.[2] Since time t enters the problem directly only in the discount factor, it is useful to derive the optimality condition from the current-value Hamiltonian

$$H = F[x,u] + \lambda f[x,u].$$

For every state variable $x_i[t]$ a costate variable $\lambda_i[t]$ is introduced. The optimality conditions consist of the maximum principle

$$\frac{\partial H}{\partial u} = F_u + \lambda f_u = 0 \tag{4}$$

and the canonical equations

$$\dot{\lambda} = \delta\lambda - \frac{\partial H}{\partial x} = \lambda(\delta - f_x) - F_x. \tag{5}$$

If (4) is used to eliminate u, (2) and (5) represent a system of $2n$ ordinary differential equations with n initial conditions given by (3). The remaining n conditions to solve the system require λ to be chosen such that the dynamical system converges to an equilibrium.

The equilibrium values of control, state and costate variables can be determined by solving $\dot{\lambda} = 0$, $\dot{x} = 0$ and (4). Then the final values of all variables are determined.

[2]Standard textbooks include Kamien and Schwartz (1991) and Feichtinger and Hartl (1986).

However, this is not sufficient to integrate (2) and (5) backwards because $\dot{K}$ and $\dot{\lambda}$ are both zero at the equilibrium. On the other hand, if the equilibrium is hyperbolic, the dynamic behaviour of the nonlinear system can be approximated by a linearized system around the equilibrium point. A time path that converges to the equilibrium must lie in the stable manifold whose linear approximation around the equilibrium is spanned by the eigenvectors that correspond to the stable eigenvalues with negative real parts. The n initial conditions for the state variables require n eigenvalues to have negative real parts for the optimization problem to have a solution.

3 The algorithm

On a more formal level, we want to solve boundary value problems

$$\begin{aligned} \dot{x} &= f[x] \\ Ax[0] &= Ax_0, \ x[\infty] = x_\infty, \end{aligned} \tag{6}$$

where x_∞ is an equilibrium, i.e. $f[x_\infty] = 0$, and A controls that part of x_0 that can be prescribed. Note that in this form the boundary value problem is overdetermined since we have more boundary conditions than differential equations.

There are two basic approaches to solving boundary value problems over an infinite interval: either an approximation of the true infinite interval by a finite one (cf. Steindl, 1984) or the transformation of the infinite interval into a finite one. The latter approach is used in this paper because there are no additional truncation errors to be looked at.

There are several possibilities for such a transformation. In view of (1), a natural choice of transformation would be to change the independent variable t to τ according to

$$\tau = e^{\rho t}, \quad \text{or } t = \ln[\tau]/\rho$$

with some $\rho < 0$. This maps the interval $[0, \infty)$ onto $[0, 1]$. Because of

$$x'[\tau] = \frac{\dot{x}[t]}{\rho\tau},$$

where the prime denotes differentiation with respect to τ, the corresponding transformation of the differential equation yields

$$x' = \frac{1}{\rho\tau} f[x].$$

The transformation thus generates a singularity at $\tau = 0$ (corresponding to $t = \infty$). To study the effect of this singularity, we may linearize the differential equation at $\tau = 0$ and (assuming $f_x[x_\infty]$ is diagonalizable) transform onto a basis of eigenvectors. In this case, the differential equation decomposes into scalar problems of the form

$$x' = \frac{1}{\rho\tau} \lambda x,$$

where λ is an eigenvalue of $f_x[x_\infty]$. By assumption we have $Re[\lambda] \neq 0$. The general solution of this equation is given by

$$x[\tau] = c\tau^{\lambda/\rho}, \quad c \in \Re.$$

Since the imaginary part of λ only gives rise to a bounded factor, we are allowed to restrict ourselves to real eigenvalues. If $\lambda < 0$ (corresponding to a stable mode), we have $\lambda/\rho > 0$. In this case, the solution automatically satisfies $x[0] = 0$. Thus, for this part we need not require a boundary condition at $\tau = 0$. However, we must observe that a numerical procedure requires the solution to be k-times continuously differentiable for some known k (depending on the specific method). In order to achieve this (and taking into account that we must choose the same ρ for several different eigenvalues), we choose ρ sufficiently small in modulus.

If $\lambda > 0$ (corresponding to an unstable mode), we have $\lambda/\rho < 0$. Thus, if this mode were present, we would have a pole at $\tau = 0$. Consequently, this mode must be excluded by the boundary condition. An appropriate choice of the boundary condition at $\tau = 0$ (or $t = \infty$), so-called asymptotic boundary conditions, can be found in Steindl (1984).

The above discussion leads to the following formulation of the boundary-value problem (6)

$$\begin{aligned} x' &= \frac{1}{\rho\tau} f[x] \\ Bx[0] = Bx_\infty, &\quad Ax[1] = Ax_0. \end{aligned} \tag{7}$$

If n_s is the number of stable eigenvalues of $f_x[x_\infty]$ and $n_u = n - n_s$, the matrix $A \in \Re^{n_s,n}$ selects n_s initial conditions (with respect to $t = 0$) while $B \in \Re^{n_u,n}$ represents the asymptotic boundary conditions mentioned above. Following Steindl (1984) we can choose B in the following way. Let

$$T^{-1} f_x[x_\infty] T = \begin{pmatrix} \Lambda_u & 0 \\ 0 & \Lambda_s \end{pmatrix}$$

such that Λ_s has only stable eigenvalues and Λ_u only unstable ones: Then B is given by

$$T^{-1} = \begin{pmatrix} B \\ C \end{pmatrix}.$$

A main problem that arises when solving a nonlinear problem like (7) is the choice of an initial guess for the solution. In the present context, however, we can embed (7) into the problem

$$\begin{aligned} x' &= \frac{1}{\rho\tau} f[x] \\ Bx[0] = Bx_\infty, &\quad Ax[1] = A(sx_0 + (1 - s)x_\infty) \end{aligned} \tag{8}$$

which for $s = 0$ has the solution $x[\tau] = x_\infty$, whereas it coincides with (7) for $s = 1$. Thus, we can apply homotopy techniques starting from a known solution (see e.g. Bader and Kunkel, 1989).

In view of the above discussion, the numerical treatment of (6) consists of the following parts:

a) Determination of an equilibrium x_∞

Since we must solve the nonlinear problem $f[x] = 0$ possibly starting with a poor initial guess, we use an implemention of the damped Newton method (see Nowak and Weimann, 1990).

b) Determination of the stable and unstable parts of $f_x[x_\infty]$, including the matrix B of the asymptotic boundary condition and ρ for the transformation Here we use standard software from the `EISPACK` library. If $\lambda_1, \ldots, \lambda_{n_s}$ are the stable eigenvalues, a possible choice for ρ is given by

$$\rho = \frac{1}{5} \max_{i=1,\ldots,n_s} Re[\lambda_i].$$

c) Treatment of the parametrized boundary value problem (8)

An appropriate software package for this problem class is given by `COLCON` due to Bader and Kunkel (1989). Since it uses collocation at Gaussian points, the right-hand side of the differential equation in (8) is not evaluated at $\tau = 0$. Thus there is no need for additional considerations regarding the treatment of the singularity in the right-hand side.

Remark:

In the case $n_s = 1$, (7) can be solved by simply integrating the (one-dimensional) manifold from $\tau = 0$ until the desired initial value x_0 is reached. See de Hoog and Weiß (1980) for more details on such (singular) initial value problems.

4 Examples

4.1 Capital accumulation

The first example has only a single state variable. For such a problem, solution procedures have been available (cf. Conrad and Clark, 1987). It is included here to show the solution of a simple problem and to illustrate the solution to a control problem graphically.

Consider the optimal investment problem: A single state variable, the capital stock K, decays at a constant rate α and is built up by investment I, which is the control variable. The capital stock and other fixed factors are used to produce some output, whose units of measurement are chosen such that the price is unity. The technology is given by the production function $F[K]$, where the fixed arguments are dropped. The cost of investment measured in terms of the output is given by $C[I]$. The problem of maximizing the present value of a firm then leads to the current-value Hamiltonian:

$$H = F[K] - C[I] + \lambda(I - \alpha K),$$

the maximum principle:

$$\frac{\partial H}{\partial I} = -C' + \lambda = 0,$$

and the canonical equation:

$$\dot{\lambda} = -F' + \lambda(\delta + \alpha),$$

where a prime denotes the derivative with respect to a single argument. Since the maximum principle has to hold at each point in time, it can be used to derive a differential equation for the control variable I. In principle it is possible to solve for the optimal trajectory in K-I space or in K-λ space. In the present context the choice does not matter. However, in more general problems with bounded controls the costate variable will still be continuous (cf. Intriligator, 1971) while the time path of the control variable may include jumps. It is, therefore, preferable to solve the problem in K-λ space.

The maximum principle and the differential equations for K and λ define equilibrium values K^*, λ^* and I^*. Graphically the equilibrium lies at the intersection of the *isoclines* where $\dot{\lambda} = 0$ or $\dot{K} = 0$, respectively. This intersection generates four isosectors in the neighbourhood of the equilibrium. The behaviour of the trajectories

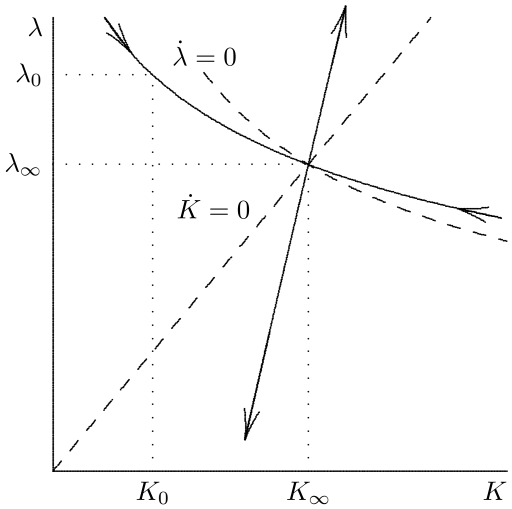

Figure 1: Phase plane diagram – Capital accumulation

in these isosectors is governed by the eigenvalues and corresponding eigenvectors of the Jacobian of the linearized system at the equilibrium.

Model specifications	
$F[K]$	$\sqrt{K}$
$C[I]$	I^2
α	0.03
δ	0.07
$K[0]$	1
Equilibrium values	
$K[\infty]$	19.0786
$\lambda[\infty]$	1.14471
Jacobian at equilibrium	
Eigenvalues	$\{0.110664, -0.0406637\}$
corresponding Eigenvectors	$(-0.962632, -0.270815)$ $(-0.999773, 0.0213226)$
Solution	
$\lambda[0]$	2.21972
Control variable	
$I[0]$	1.10987
$I[\infty]$	0.572357

Table 1: Capital accumulation

In figure 1 the optimal trajectories are plotted and the initial value of λ can then be derived from the initial value K_0. The numerical solution given by `OPTTRJ` is reported in table 1.

4.2 Harvesting in a multi-species environment

Consider an environment that supports two species: X and Y. These species interact with each other which makes optimal harvesting the solution of a complicated problem. A typical objective would be

$$\max \int_0^\infty e^{-\delta t}(B[X, Y, H_X, H_Y] - C[H_X, H_Y])\, dt,$$

where B is a concave benefit (utility) and C a convex cost component. Let the

α_X	β_X	γ_X	α_Y	β_Y	γ_Y	δ	$X[0]$	$Y[0]$	B	C
2	5	-2	1	8	20	5%	7/40	9/16	$\ln[X\, H_X] + aY$	$H_X^2 + H_Y^2$

Table 2: Predator-prey harvesting: Parameters and functions

change in the resource stocks be described by

$$\dot{X} = X(\alpha_X - \beta_X X + \gamma_X Y) - H_X \tag{9}$$

$$\dot{Y} = Y(\alpha_Y - \beta_Y Y + \gamma_Y X) - H_Y. \tag{10}$$

H_X and H_Y represent harvesting of the respective resource (the control variables). The α-parameters describe a linear birth process and the β-parameters a quadratic death process. Without harvesting and without interaction ($\gamma_i = 0,\ i = X, Y$) each resource would follow a logistic growth path. Three interactive cases are usually distinguished:

Predator-prey systems require γ_A and γ_B to be of opposite sign. The species with a positive γ is the predator population, it thrives with a large prey population. The prey population with a negative γ obviously suffers if the predator population is large.

Competition occurs if both γs are negative. It can describe a situation where both species compete for the same resource.

Mutualism describes a situation where both γs are positive, i.e., where the species thrive on each other.

From the many possible situations we will compare three predator-prey environments where the prey population has a negative, a positive or no impact on the benefit B.

	$a = -1/2$	$a = 0$	$a = 1/2$
Equilibrium values			
$X[\infty]$	0.160586	0.167910	0.173274
$Y[\infty]$	0.436187	0.493066	0.525142
$\lambda_X[\infty]$	7.939787	8.699643	9.273645
$\lambda_Y[\infty]$	-0.630049	-0.407935	-0.277638
Jacobian at equilibrium			
Eigenvalues	$-1.35544 + 1.034165I$	$-2.08062 + 0.79346I$	$-2.37904 + 0.36952I$
	$-1.35544 - 1.034165I$	$-2.08062 - 0.79346I$	$-2.37904 - 0.36952I$
	$1.40544 + 1.034165I$	$2.13062 + 0.79346I$	$2.42904 + 0.36952I$
	$1.40544 - 1.034165I$	$2.13062 - 0.79346I$	$2.42904 - 0.36952I$
Solution			
$\lambda_X[0]$	7.237390	8.258903	9.173408
$\lambda_Y[0]$	-0.439305	-0.346543	-0.257470
Control variables			
$H_X[0]$	0.133264	0.117725	0.106536
$H_Y[0]$	0.219652	0.173271	0.128735
$H_X[\infty]$	0.122187	0.112060	0.105435
$H_Y[\infty]$	0.315025	0.203968	0.138819

Table 3: Predator-prey harvesting

Given the functional forms of B and C as given in table 2, we can formulate the current-value Hamiltonian:

$$H = \ln X + \ln H_X + aY - H_X^2 - H_Y^2 + \lambda_X \dot{X} + \lambda_Y \dot{Y}.$$

It leads to the maximum principle

$$1/H_X - 2H_X - \lambda_X = 0$$

$$-2H_Y - \lambda_Y = 0$$

and to the canonical equations

$$\dot{\lambda}_X = \delta\lambda_X - (1/X + \lambda_X(\alpha_X - 2\beta_X X + \gamma_X X) + \lambda_Y \gamma_Y Y) \tag{11}$$

$$\dot{\lambda}_Y = \delta\lambda_Y - (a + \lambda_X \gamma_X X + \lambda_Y(\alpha_Y - 2\beta_Y Y + \gamma_Y Y)) \,. \tag{12}$$

The parameters of the model are given by table 2 and $a \in \{-1/2, 0, 1/2\}$. The initial values of the state variable correspond to the natural equilibrium without harvesting.[3] The results of `OPTTRJ` are presented in table 3.

It is not surprising that an increase in the marginal benefit of the predator population Y leads to less harvesting of it (note that the harvesting of the predator population generates no direct benefit). It is interesting to observe, though, that the equilibrium population sizes of the predator and prey population both increase.

4.3 Pollution and economic growth

The following example (including the basic notation) is based on Steindl (1984). A representative individual consumes a consumption good C and pollution W. His objective is

$$\max_{C,A} \int_0^\infty e^{-\rho t} U[C, W]\, dt.$$

The output can be used for consumption, environmental improvement or capital accumulation. All three uses pollute the environment – at different rates, though. The differential equations for the state variables are

$$\dot{K} = F[K] - C - A - \beta K \tag{13}$$

$$\dot{W} = \varepsilon_1 F[K] + \varepsilon_2 C + \varepsilon_3 K - g[A] - \delta W. \tag{14}$$

The notation and the results are presented in table 4. The Hamiltonian

$$H = U[C, W] + p\dot{K} + q\dot{W},$$

[3] There are, of course, also trivial equilibria where either one or both of the species are extinct.

Model specifications		
K	$K[0] = 1000$	capital stock
W	$W[0] = \quad 8$	pollution
C		consumption
A		resource use for cleaning activity
$g[A]$	$A^{0.4}$	effect of cleaning activity
$F[K]$	$1.73K^{0.75}$	production function
β	0.05	capital depreciation rate
ε_1	0.0026	pollution due to production
ε_2	0.0158	pollution due to consumption
ε_3	0.00015	pollution due to capital input
δ	0.04	self-purification rate of the environment
$U[C, W]$	$C^{0.75} - 1.5W^{1.2}$	utility function
ρ	0.08	discount rate
Equilibrium values		
$K[\infty]$	3862.1416	
$W[\infty]$	4.187323	
$p[\infty]$	0.083437	
$q[\infty]$	-4.993655	
Jacobian at equilibrium		
Eigenvalues	$\{0.174026, 0.840253, -0.094026, -0.760253\}$	
Solution		
$p[0]$	0.203554	
$q[0]$	-4.388706	
Control variables		
$A[0]$	36.2685	
$C[0]$	57.0508	
$A[\infty]$	198.853	
$C[\infty]$	455.593	

Table 4: Pollution and economic growth

where p and q are the costate variables for K and W respectively, gives rise to the following optimality conditions:

$$U_C - p + q\varepsilon_2 = 0$$

$$-p - qg' = 0$$

$$\dot{p} = \rho p - (p(F' - \beta) + q(\varepsilon_1 F' + \varepsilon_3))$$

$$\dot{q} = \rho q - (U_W - q\delta).$$

The result is qualitatively as expected: consumption and the capital stock grow over time – as does the expenditure for environmental protection.

References

Bader, G., and Kunkel, P., 1989, Continuation and collocation for parameter-dependent boundary value problems, *SIAM Journal on Scientific and Statistical Computing 10(1)*, January, 72–88. 1, 3, 3

Conrad, J.M., and Clark, C.W., 1987, *Natural resource economics - Notes and problems*, Cambridge University Press, Cambridge. 4.1

de Hoog, F.R., and Weiß, R., 1980, An approximation theory for boundary value problems on finite intervals, *Computing 24*, 227–239. 3

Feichtinger, G., and Hartl, R.F., 1986, *Optimale Kontrolle ökonomischer Prozesse - Anwendungen des Maximumprinzips in den Wirtschaftswissenschaften*, de Gruyter, Berlin. 2

Goffe, W.L., 1993, A user's guide to the numerical solution of two-point boundary value problems arising in continuous time dynamic economic models, *Computational Economics 6(3-4)*, November, 249–255. 1, 1

Intriligator, M.D., 1971, *Mathematical optimization and economic theory*, Prentice-Hall, Englewood Cliffs. 4.1

Kamien, M.I., and Schwartz, N.L., 1991, *Dynamic optimization - The calculus of variations and optimal control in economics and management*, North-Holland, Amsterdam, 2nd edition. 2

Nowak, U. and Weimann, L., 1990, *A family of Newton codes for systems of highly nonlinear equations - Algorithm, implementation, application*, Techn. Rep. TR 90-10, Konrad-Zuse-Zentrum, Berlin. 1

Pontryagin, L.S., Boltyanskii, V.G., Gamkredlidze, R.V., and Mishchenko, E.F., 1964, *The mathematical theory of optimal processes*, Pergamon Press, Oxford. 2

Samuelson, P.A., 1947, *Foundations of economic analysis*, Harvard University Press, Cambridge. 1

Steindl, A., 1984, *Numerische Behandlung von Randwertproblemen auf unendlichen Intervallen*, Dissertation, Technische Universität Wien, Wien. 3, 3, 3, 4.3

A Manual

A.1 Purpose

The program OPTTRJ solves boundary-value problems on $[0, \infty)$ e.g. arising from optimal-control problems with an infinite time horizon. It also determines the equilibrium values and the eigenvalues of the linearization at the equilibrium. It serves as easy-to-use interface to sophisticated public domain software.

A.2 Specification

```
      SUBROUTINE OPTTRJ(N,XEQUI,X0,I0,R,TOL,CUT,KPRINT,IERR,
     1  RWORK,LRWORK,IWORK,LIWORK)
      INTEGER IERR,I0(N),IWORK(LIWORK),KPRINT,LIWORK,LWORK,N
      REAL*8 CUT,R,RWORK(LRWORK),TOL,XEQUI(N),X0(N)
```

A.3 Description

The program OPTTRJ solves a boundary value problem for a system of n autonomous differential equations in the interval $[0, \infty)$ of the form

$$\dot{x} = f[x], \quad Ax[0] = Ax_0, \quad x[\infty] = x_\infty,$$

where x_∞ is a hyperbolic equilibrium of the differential equations. It consists of three parts. In the first part the equilibrium x_∞ is computed by means of a damped Newton method. Then, hyperbolicity of x_∞ is checked by an eigenvalue analysis of $f_x[x_\infty]$ and the correct formulation of the boundary conditions at $t = \infty$ (so-called asymptotic boundary conditions) are determined. In the last part the boundary value problem is solved by homotopy techniques.

A.4 Parameters

N – INTEGER *Input*

number of differential equations

Constraint: $\mathtt{N} \leq 32$

XEQUI(N) – DOUBLE PRECISION array *Input/Output*

On entry: estimate of equilibrium

On exit: computed equilibrium

X0(N) – DOUBLE PRECISION array *Input*

initial values for the differential equation

`IO(N)` – `INTEGER` array *Input*

indexation of the initial values `XO`. The number of possible initial values equals the number of stable eigenvalues of the equilibrium. If the number of stable eigenvalues is n_s, the code uses the initial conditions

$$\mathtt{X}_{\mathtt{IO(I)}}(0) = \mathtt{XO}_{\mathtt{IO(I)}}, \quad \mathtt{I} = 1, ..., n_s.$$

`R` – `DOUBLE PRECISION` *Input*

parameter for the transformation of $[0, \infty)$ to $[0, 1]$ by $\tau = \exp(\mathtt{R}\, t)$. If the value is not negative, the code tries to choose an appropriate value. The modulus of `R` should be small enough to give rise to a sufficiently smooth solution on $[0, 1]$.

`TOL` – `DOUBLE PRECISION` *Input*

tolerance used in the public domain software for solving nonlinear equations and boundary value problems

`CUT` – `DOUBLE PRECISION` *Input*

tolerance used in the classification of the eigenvalues. Numbers smaller than cut in modulus are treated as zero.

`KPRINT` – `INTEGER` *Input*

print parameter (output unit is 6)

0 no output
1 small output
2 large output

`RWORK` – `DOUBLE PRECISION` array *Workspace*

On exit: a successfully computed solution of the boundary value problem

`LRWORK` – `DOUBLE PRECISION` *Input*

length of `RWORK`

`IWORK(LIWORK)` – `INTEGER` array *Workspace*

On exit: a successfully computed solution of the boundary value problem

`LIWORK` – `INTEGER` *Input*

length of `IWORK`

`IERR` – `INTEGER` *Output*

error exit parameter

0 no error occurred
1 invalid input
2 determination of equilibrium failed
3 determination of eigenvalues failed
4 nonhyperbolic equilibrium found
5 inversion of eigenvector matrix failed
6 boundary value problem solver failed

On return the information on a successfully computed solution of the boundary value problem is stored in the workspaces. Function values can be obtained by calling the subroutine `APPSLN` which is part of the `COLCON` package. Since the homotopy parameter is part of this solution, the solution vector must be dimensioned as `X(N1)` where `N1=N+1`. For details see the manual contained in the `COLCON` package.

A.5 Subroutines supplied by the user

The user must supply subroutines for the evaluation of f and its Jacobian at a given point x. They must have the following form.

```
SUBROUTINE FEVAL(N,X,F)
INTEGER N
REAL*8 X(N),F(N)
.
.
.
RETURN
END

SUBROUTINE JEVAL(N,L,X,DFDX)
INTEGER L,N
REAL*8 X(N),DFDX(L,N)
.
.
.
RETURN
END
```

The parameters `N`, `L`, and `X` are input and must not be altered by the subroutines. On return `F` and `DFDX` must contain $f[x]$ and $f_x[x]$ respectively. Note that the subroutines must be named `FEVAL` and `JEVAL`.

A.6 Other subroutines required

`NLEQ1` – solver for systems of nonlinear equations

This package is part of the `CODELIB` library and can be obtained e.g. via

ftp://elib.zib-berlin.de/pub/elib/codelib/nleq1/

Note that this package includes a subroutine `D1MACH` where machine dependent constants must be adapted to the hardware used.

`BALANC, BALBAK, CDIV, HQR2, ORTBAK, ORTHES` – suite of subroutines for computing eigenvalues and eigenvectors of a general real matrix. These subroutines are part of the `EISPACK` library within the `NETLIB` library and can be obtained e.g. via

ftp://elib.zib-berlin.de/pub/netlib/eispack/

`COLCON` – package for solving boundary value problems. This package can be obtained by request from

mailto://kunkel@math.uni-oldenburg.de

The implementation is based on linear algebra subroutines from the `LINPACK` and `BLAS` library. These can be found e.g. at

ftp://elib.zib-berlin.de/pub/netlib/linpack/

and

ftp://elib.zib-berlin.de/pub/netlib/blas/