Centrum voor Wiskunde en Informatica
REPORT*RAPPORT*

HomCont: an AUTO86 driver for homoclinic bifurcation analysis. Version 2.0

A.R. Champneys, Yu.A. Kuznetsov and B. Sandstede

Department of Analysis, Algebra and Geometry

AM-R9516 1995

Report AM-R9516
ISSN 0924-2953

CWI
P.O. Box 94079
1090 GB Amsterdam
The Netherlands

HomCont: An AUTO86 Driver for Homoclinic Bifurcation Analysis. Version 2.0

A.R. Champneys*, Yu.A. Kuznetsov†, and B. Sandstede‡

* *Department of Engineering Mathematics, University of Bristol,*
Queen's Building, University Walk,
Bristol BS8 1TR, UK
A.R.Champneys@bristol.ac.uk

† *CWI,*
P.O. Box 94079, 1090 GB Amsterdam, The Netherlands
Yu.A.Kuznetsov@cwi.nl
&
Institute of Mathematical Problems of Biology,
Russian Academy of Sciences, Pushchino, Moscow Region,
142292 Russia

‡ *Weierstraß-Institut für Angewandte Analysis und Stochastik,*
Mohrenstraße 39,
10117 Berlin, Germany
Sandstede@iaas-berlin.d400.de

Abstract

HomCont, an auto86-based toolbox for homoclinic bifurcation analysis, is described in detail. The toolbox allows the continuation of codimension-one homoclinic orbits to hyperbolic and non-hyperbolic equilibria, as well as detection and continuation of higher-order homoclinic singularities in more parameters. All known codim 2 cases that involve a unique homoclinic orbit are supported, and certain heteroclinic computations are also possible. The document contains details on the various files supplied with HomCont and how to use them to analyse several tutorial examples.

AMS Subject Classification (1991): 34C37, 34A50, 58F14

Keywords & Phrases: Homoclinic orbits, continuation method, auto86

Note: Financial support for collaboration was received from the UK Engineering and Physical Science Research Council and the Nuffield Foundation.

Table of Contents

Chapter 1
Introduction and installation

1. Purpose of HomCont

HomCont is a suite of routines to accompany auto86 (Doedel & Kernévez (1986), Doedel, Keller & Kernévez (1991*a*, 1991*b*)) in order to perform continuation in two or more parameters of homoclinic solutions to equilibria in ordinary differential equations. In addition, the accurate detection and multi-parameter continuation of certain codimension-two singularities is allowed for, and certain heteroclinic computations are also possible. The theory behind the methods used is explained in Champneys & Kuznetsov (1994), Bai & Champneys (1994), Sandstede (1995*b*, 1995*c*), Champneys, Kuznetsov & Sandstede (1995) and references therein. The final cited paper contains a concise description of the present version.

2. System requirements

HomCont requires that auto86 is installed under unix. Note that the present implementation of HomCont will not run under auto94.

3. How to get HomCont

HomCont is available on the `anonymous` ftp server `ftp.cwi.nl` in the directory `pub/HomCont`. Comments and reports of bugs are very much appreciated.

4. Installation

HomCont comes as the compressed tar-file `HomCont.tar.Z` and can be extracted by running

```
uncompress HomCont.tar.Z
tar -xf HomCont.tar
```

in the directory in which the driver should be installed. Then the directories `circuit, doc, koper, marten, test, shear, src` are created. The manual is contained in `doc`, while the source code is written to `src`. The latter directory also contains the command files as well as the example file `autexample.f` which can be used by the user for setting up new problems. Finally, the files used in the tutorial examples described in Chapter 3 are contained in the remaining directories `test, marten, koper, circuit, shear`. A list of all files can be found in the file `doc/README`.

It is assumed that the user has the standard unix version of the files for auto86 contained in a directory which has been assigned to the environment variable autolibrary and that the various command files, `@svaut`, `@plaut` etc. have been defined (e.g. by executing `source @auto.alias`).

In order to run an HomCont example file from any directory, you first need to set up the command files. To do this, edit the top line of the file `src/@autoh.alias` to set the environment variable homcontlibrary to be the full name of the directory `src` and then execute this file

```
source @autoh.alias
```

Including the above line in your `.login` (or `.cshrc`) file will automatically set up the files for HOMCONT.

If you do not run AUTO86 using the standard UNIX command files, then you will need to adapt the command files `@autoh`, `@adjaut` and `@delhaut` to be similar to the command files used to run your version of AUTO86.

5. NOTES

Because the code is licensed free of charge, there is absolutely no warranty. Please note that the source code for certain NAG library routines for eigenvalues and eigenvectors is included within this package. These routines are not free to be distributed further without the authors' permission and should *not be used for commercial applications.* If you have an access to the standard NAG library on your machine, these source codes are redundant and you can replace `$AUTOLIBRARY/nagroutines.f` in the compilation statements in `@autoh` with the standard option to link to the NAG library.

Acknowledgments: The authors are thankful to E. Doedel (Concordia University, Montreal), W.J. Beyn (Universität Bielefeld), M.J. Friedman (University of Alabama), A. Rucklidge (University of Cambridge), M. Koper (University of Utrecht) and C.J. Budd (University of Bristol) among others for various pieces of help and advice. BS is very grateful to Ingo Bremer for his generous help concerning computers and for installing AUTO86.

Chapter 2
Running HomCont

1. General information

In order to run HomCont on an example, one needs to specify an *example file* called `aut<name>.f` (e.g. look at the file `auttest.f` in Appendix 1). The file `autexample.f` in the directory `src` serves as a sample for new example files. The example files have a similar purpose to the example files of auto86, namely to specify the problem to be solved as well as various parameters and constants defining the computation to be performed. However their structure is different to that of standard auto86 example files. Standard auto86 routines `INIT, FUNC, STPNT, BCND, ICND, USZR` are set up in an additional library `src/autlibh.f` and should not need altering even if, for example, the dimension of the defining differential equations changes (e.g. if computing the orientation of a homoclinic orbit, see below). This library may be adapted by expert users to perform computations not automatically supported by the current version of HomCont. The routines that must be specified by an ordinary user in an example file are `PROBLEM, PF, PDFDU, PEQUIB, PUSZR, PSTPNT` (the 'P' stands for "problem-specific"). These routines may be regarded as higher-level input routines that are called by the standard auto86 routines contained in `autlibh.f`. The purposes of the problem-specific routines are the following.

`PROBLEM` In this routine constants are specified describing the problem and the computations to be performed. It allows the user to choose *some* of the auto86 constants normally set up in `INIT` and additionally, to specify certain other HomCont-specific problem-definition constants. See below for a list of these constants.

`PF` Contains the right-hand-side of the differential equations in the format described in the auto86 manual, see the example files.

`PDFDU` Contains the Jacobian matrix of the differential equation, that is the derivatives with respect to the phase-space variables, in the auto86 format.

`PEQUIB` This routine is only called if `IEQUIB=0`. Then it should contain an analytic expression for the equilibrium towards which the homoclinic orbit converges (or two separate equilibria for which a connecting orbit is sought in the heteroclinic case).

`PUSZR` This routine is similar to the subroutine `USZR` in auto86. `USZR` in HomCont calls `PUSZR` directly, but it is additionally used to define test functions. The number of additional functions at zeros of which output is requested must be assigned to the variable `NPUSZR` in the routine `PROBLEM`. Note that the variable `NUZR` must not be used for this purpose.

`PSTPNT` If an explicit homoclinic solution is known at the start, it can be specified in this routine and will be used as an initial guess for the continuation. This routine is only called if `ISTART=2`. The format in `PSTPNT` is `U=U(T)` where `T` runs in the interval `[-PAR(11)/2, PAR(11)/2]`. The parameter `PAR(11)` is assigned in the routine `PROBLEM`.

The additional constants appearing in the routine `PROBLEM` are as listed below.

NPROB	Problem size (that is, the dimension of phase space).
NSTAB, NUNSTAB	Numbers of stable and unstable eigenvalues.
NFREE,ICP(20)	Number and labels of free parameters.
NPSI,IPSI(20)	Number and labels of activated test functions for detecting homoclinic bifurcations, see below for a list.
NFIXED,IFIXED(20)	Number and labels of fixed test functions. With NPSI=1, IFIXED(1) = i_1 one can continue a zero of IPSI = i_1 in one more parameter, which should be specified in ICP. Bifurcation points of higher codimension can be treated similarly.
NPUSZR	Number of user-defined functions (defined in PUSZR) for output.

IEQUIB = 0 if the equilibrium is specified explicitly in PEQUIB.
= 1 if the saddle equilibrium is to be continued numerically.
= 2 if homoclinic orbits to a saddle-node are followed and the equilibrium is to be continued numerically.

ISTART = 1 if starting data is read from the file pstpnt.dat when IRS=0. This data must be given in the form t,U in multi-column format at each point with t in the interval [0,1].
= 2 if an explicit solution is specified in the subroutine PSTPNT.
= 3 if the "homotopy" approach is used for starting.

ITWIST = 0 the orientation of the homoclinic solution is not computed.
= 1 the orientation of the homoclinic solution is computed. Then the adjoint variational equation is solved for the unique bounded solution. If IRS = 0 an initial guess for this equation must be specified as well. However, the shell routine @adjaut can be used to add an initial guess to the homoclinic solution at a restart label point.

The parameters PAR(1) – PAR(9) can be used freely by the user. The parameter PAR(11) has to be specified by the user, too, while the remaining parameters should not be altered.

PAR(11) The value of PAR(11) equals the length of the time interval over which a homoclinic solution is computed.

PAR(*) The parameters PAR(10) and PAR(12) – PAR(19) are used by HomCont and therefore must not be altered.

The user may in addition change any of the usual auto86 constants *except* for the following ones which must not be altered, because values are assigned to these in the driver HomCont

Constants not allowed to be changed: NDIM, IPS, NUZR, JAC, NBC, NINT.

To run HomCont on an example file aut<name>.f the command file @autoh is used

```
@autoh <name>
```

This operates like the auto86 command @auto and output can be saved, plotted and appended to using the usual auto86 commands @svaut, @plaut and @apaut, see the examples in Chapter 3 below.

To compute the orientation of a homoclinic orbit (i.e. in order to detect inclination-flip bifurcations) it is necessary to compute, in tandem, a solution to the modified adjoint variational equation, by setting ITWIST=1. In order to obtain starting data for such a computation, given auto86 output for just the homoclinic, the following command is used:

```
@adjaut LAB <name1> <name2>
```

which copies the data at the point label LAB from q.<name1> to the file q.<name2> appending at the same time an initial guess for the solution of the adjoint equation. After changing ITWIST=0 to ITWIST=1 a Newton step in the dummy parameter PAR(20) should be performed

```
IRS = LAB   NMX = 2   ICP(1) = 20
```

Then the output contains the homoclinic solution as well as the bounded solution to the adjoint variational equation (see Chapter 3, Section 1.1 or 3.1 for an example).

Another command is used to delete these extra columns of data in the output in the `q.<name>` file when one wants to "switch off" the computation of the orientation (e.g. when the eigenvalues become complex). Calling

```
@delhaut LAB <name1> <name2>
```

stores the data corresponding to the label `LAB` from `q.<name1>` to the file `q.<name2>` removing at the same time the solution data for the adjoint equation. After changing `ITWIST=1` to `ITWIST=0` one can then proceed with computing the homoclinic solution without solving for the adjoint equation.

The output is in an identical format to AUTO86 except that there is additional information at each computed point written in `fort.9`, described as follows. First, the eigenvalues of the lineariazation at the equilibrium are output in the format

```
EIGENVALUES
(  real part,    imaginary part)
(     ...   ,          ...      )
```

Second, if `ITWIST=1` the orientation of the homoclinic solution is indicated below the eigenvalues by either the line

```
ORIENTABLE (value)
```

or

```
NON-ORIENTABLE (value)
```

where the sign of `value` indicates the orientation. Note that the statement about orientability is only meaningful if the leading eigenvalues are not complex and the homoclinic solution is not in a flip configuration, that is, none of the test functions ψ_i for $i = 11, 12, 13, 14$ is zero (or close to zero), see Section 1.1 in Chapter 3. Finally, the values of the `NPSI` activated test functions are written in the format

```
PSI(...)
USZR        FUNCTION =   ...
```

followed by the values of the functions defined by the user in the routine `PUSZR`

```
USER DEFINED FUNCTIONS:
USZR        FUNCTION =  ...
```

2. Test functions

Codimension-two homoclinic orbits are detected along branches of codim 1 homoclinics by locating zeroes of certain test functions ψ_i. The various test functions that are "switched on" during any continuation are given by the choice of the labels i, and are specified by the parameters `IPSI`. The number of activated test functions is defined by in `NPSI`. An example is

```
NPSI = 2   IPSI(1) = 7   IPSI(2) = 4
```

meaning that ψ_4 and ψ_7 are turned on. In general, a list of indices i_1,...,i_n should be specified in the following way

```
NPSI = n   IPSI(1) = i1   ...   IPSI(n) = in
```

activating the test functions ψ_i for $i \in \{i_1,...,i_n\}$.

The following codimension-two homoclinic singularities are given by zeroes of the test function `IPSI(...)=i` for the given value of `i`. The notation

$$\text{Re } \mu_{\texttt{NSTAB}} \leq \cdots \leq \text{Re } \mu_1 \leq 0 \leq \text{Re } \lambda_1 \leq \cdots \leq \text{Re } \lambda_{\texttt{NUNSTAB}},$$

is used for the eigenvalues as in Champneys & Kuznetsov (1994) and Champneys et al. (1995).

`i=1` Resonant eigenvalues (neutral saddle); $\mathrm{Re}(\lambda_1) + \mathrm{Im}(\lambda_1) = -\mathrm{Re}(\mu_1) - \mathrm{Im}(\mu_1)$.

`i=2` Double real leading stable eigenvalues (saddle to saddle-focus transition); $\mu_1 = \mu_2$.

`i=3` Double real leading unstable eigenvalues (saddle to saddle-focus transition); $\lambda_1 = \lambda_2$.

`i=4` Neutral saddle, saddle-focus or bi-focus (includes `i=1`); $\mathrm{Re}(\lambda_1) = -\mathrm{Re}(\mu_1)$.

`i=5` Neutrally-divergent saddle-focus (stable eigenvalues complex); $\mathrm{Re}(\lambda_1) = -\mathrm{Re}(\mu_1) - \mathrm{Re}(\mu_2)$.

`i=6` Neutrally-divergent saddle-focus (unstable eigenvalues complex); $\mathrm{Re}(\mu_1) = -\mathrm{Re}(\lambda_1) - \mathrm{Re}(\lambda_2)$.

`i=7` Three leading eigenvalues (stable); $\mathrm{Re}(\mu_1) = \mathrm{Re}(\mu_2) = \mathrm{Re}(\mu_3)$.

`i=8` Three leading eigenvalues (unstable); $\mathrm{Re}(\lambda_1) = \mathrm{Re}(\lambda_2) = \mathrm{Re}(\lambda_3)$.

`i=9` Local bifurcation (zero eigenvalue or Hopf): number of stable eigenvalues decreases; $\mathrm{Re}(\mu_1) = 0$.

`i=10` Local bifurcation (zero eigenvalue or Hopf): number of unstable eigenvalues decreases; $\mathrm{Re}(\lambda_1) = 0$.

`i=11` Orbit flip with respect to leading stable direction (e.g. 1D unstable manifold).

`i=12` Orbit flip with respect to leading unstable direction, (e.g. 1D stable manifold).

`i=13` Inclination flip with respect to stable manifold (e.g. 1D unstable manifold).

`i=14` Inclination flip with respect to unstable manifold (e.g. 1D stable manifold).

`i=15` Non-central homoclinic to saddle-node (in stable manifold).

`i=16` Non-central homoclinic to saddle-node (in unstable manifold).

Expert users may wish to add their own test functions by editing the function `PSI` in `autlibh.f`.

Information about the values of each activated test function, including possible zeroes which AUTO86 failed to accurately detect, is output to `fort.9` as described in the previous section.

3. Starting strategies

There are four possible starting procedures for continuation.

(i) Data can be read from a previously-obtained output point from AUTO86 (e.g. from continuation of a periodic orbit up to large period; note that the end-point of the data stored must be close to the equilibrium). This data can be read from fort.8 (saved to `q.<name>`) by making `IRS` correspond to the label of the data point in question.

(ii) Data from numerical integration (e.g. computation of a stable periodic orbit, or an approximate homoclinic obtained by shooting) can be read in from a data file called `pstpnt.dat`. This data should be in multi-column format according to the read statement

```
READ(...,*) T(J),(U(I,J),I=1,NPROB)
```

In other words, data should be stored in `pstpnt.dat` in the line format

```
T    U(1)    ...    U(NPROB)
```

where `T` runs in the interval `[0,1]`. Note that this implies that the true time variable must be scaled by a factor of `1/PAR(11)`. In this case `IRS` should be set to zero (see the example on Scheffer's model in Section 2 in Chapter 3 below) and `ISTART=1`.

(iii) By setting `ISTART=2`, an explicit homoclinic solution can be specified in the routine `PSTPNT` in the format `U=..(T)`. Here `T` runs in the interval `[-PAR(11)/2,PAR(11)/2]`.

(iv) The choice `ISTART=3` allows for a homotopy method to be used to approach a homoclinic orbit starting from a small approximation to a solution to the linear problem in the unstable manifold (Doedel, Friedman & Monteiro 1993). For details of implementation, the reader is referred to Section 5.1.2 of Champneys & Kuznetsov (1994), under the simplification that we do not solve for the adjoint $u(t)$ here. The basic idea is to start with a small solution in the unstable manifold, and perform continuation in `PAR(11) = T` and dummy initial-condition parameters ξ_i in order to satisfy the correct right-hand boundary conditions, which are defined by zeroes of other dummy parameters ω_i. See Section 3.1 in Chapter 3 below for an example.

Chapter 3
Tutorial examples

Ordinarily, a user would wish to plot data after each save. In most implementations of AUTO86 this is achieved by using the command

```
@plaut <name>
```

where `<name>` would be `test`, `marten`, `koper`, `circuit` or `shear` in the examples defined below. However, we shall not describe the use of `@plaut` for any example. Certain figures ahead are produced with the help of visualization programs which are independent of AUTO86.

1. GENERAL TEST EXAMPLE
Consider the system (Sandstede 1995*a*)

$$\left\{\begin{array}{rcl} \dot{x} &=& a\,x + b\,y - a\,x^2 + (\tilde{\mu} - \alpha\,z)\,x\,(2 - 3x) \\ \dot{y} &=& b\,x + a\,y - \dfrac{3}{2}\,b\,x^2 - \dfrac{3}{2}\,a\,x\,y - (\tilde{\mu} - \alpha\,z)\,2\,y \\ \dot{z} &=& c\,z + \mu\,x + \gamma\,x\,z + \alpha\,\beta\,(x^2\,(1-x) - y^2). \end{array}\right. \tag{1.1}$$

Choosing the constants appearing in (1.1) appropriately allows one to find inclination and orbit flips as well as non-orientable resonant bifurcations, see (Sandstede 1995*a*) for details and proofs. The starting point for all calculations is $a = 0, b = 1, \alpha\beta = 0, \gamma = \mu = \tilde{\mu} = 0$, where there exists an explicit homoclinic solution given by

$$(x(t), y(t), z(t)) = \left(1 - \left(\frac{1 - e^t}{1 + e^t}\right)^2, 4\,e^t\,\frac{1 - e^t}{(1 + e^t)^3}, 0\right). \tag{1.2}$$

The defining HOMCONT problem is contained in the file `auttest.f` in the directory `test` and is also listed as Appendix 1. Before begining, you should change directory to `test`. The system (1.1) is specified in the subroutine `PF` of `auttest.f` with the following correspondence: x = `U(1)`, y = `U(2)`, z = `U(3)`, a = `PAR(1)`, b = `PAR(2)`, c = `PAR(3)`, α = `PAR(4)`, β = `PAR(5)`, γ = `PAR(6)`, μ = `PAR(7)`, $\tilde{\mu}$ = `PAR(8)`. The subroutine `PDFDU` contains the Jacobian matrix of (1.1).

1.1 Inclination flip
We start with $a = 0$, $b = 1$, $c = -2$, $\alpha = 0$, $\beta = 1$ and $\gamma = \mu = \tilde{\mu} = 0$ as chosen in `auttest.f`. The homoclinic solution is followed in the parameters $(a, \tilde{\mu})$ `=(PAR(1),PAR(8))` up to $a = 0.25$. The following problem-dependent constants are assigned in the subroutine `PROBLEM` of `auttest.f`

```
NPROB = 3   NUNSTAB = 1  NSTAB = 2   IEQUIB = 0   ITWIST = 0   ISTART = 2
```

The constants NPROB, NUNSTAB and NSTAB define the dimension of the system and that of the stable and the unstable invariant manifolds of the saddle at the origin, respectively. The saddle coordinates are explicitly given by the subroutine PEQUIB as indicated by IEQUIB = 0; the orientation of the homoclinic solution is not computed (ITWIST = 0); and the initial homoclinic solution (1.2) is specified in the routine PSTPNT (ISTART = 2). Running AUTO86

```
@autoh test
```

yields the output[1]

```
 BR   PT TY LAB    PAR(1)       ...      PAR(8)
  1    1 EP   1  0.000000E+00   ...   0.000000E+00
  1    5 UZ   2  2.500000E-01   ...  -3.620305E-11
  1   10 EP   3  7.384434E-01   ...  -9.038826E-09
```

which we save in dummy files p.test1, q.test1 and d.test1 using the standard AUTO86 command

```
@svaut test1
```

An initial guess of the adjoint equation is now created in q.test by running

```
@adjaut 2 test1 test
```

Next, we have to perform a Newton step in the dummy parameter PAR(20) in order to obtain the correct solution of the adjoint equation. This can be achieved by making the following changes in auttest.f

```
ITWIST = 1   IRS = 2   NMX = 3   ICP(1) = 20   NPUSZR = 0
```

Note that it is the *first* assignment of ICP(1) (on line 197) that we change. On rerunning AUTO86 and appending the output to p.test, q.test and d.test

```
@autoh test
@apaut test
```

we get the output

```
 BR   PT TY LAB     PAR(20)     ...       PAR(8)           PAR(10)
  1    3 EP  3    5.628636E+00 ...   -3.778346E-11    -4.776378E-09
```

We are now ready to perform continuation of the homoclinic plus adjoint in $(\alpha, \tilde{\mu})$ =(PAR(4), PAR(8)) by changing the constants in the file auttest.f according to

```
IRS = 3   NMX = 50   ICP(1) = 4   NPSI = 2
```

The test functions for detecting resonant bifurcations (ISPI(1)=1) and inclination flips (ISPI(2)=13) are now activated. Running

```
@autoh test
@apaut test
```

yields

```
 BR   PT TY LAB    PAR(4)       ...      PAR(8)         PAR(10)
  1   20      4  7.847220E-01   ...  -3.001077E-11  -4.270131E-09
  1   35 UZ   5  1.230857E+00   ...  -5.782999E-11  -4.552091E-09
  1   40      6  1.383966E+00   ...  -8.165651E-11  -4.665350E-09
  1   50 EP   7  1.695202E+00   ...  -1.386592E-10  -5.096954E-09
```

Note that the artificial parameter ϵ =PAR(10) is zero to within the allowed tolerance, as it is should be according to the theory (Sandstede 1995*c*). The file d.test contains the lines

[1] All the computations in this manual are performed on a SGI Indy under IRIX 5.2 using the standard NAG library.

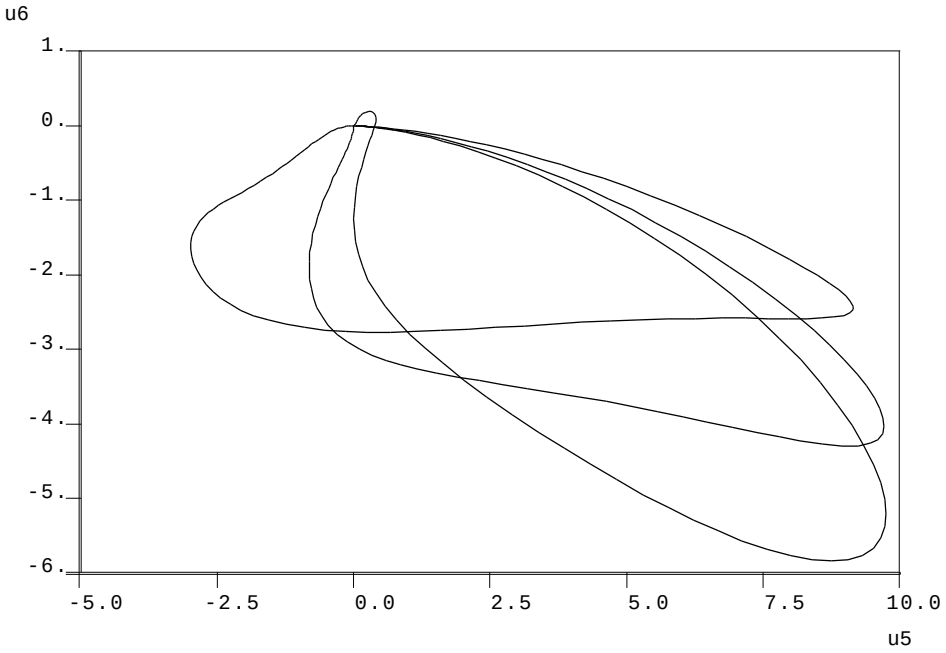

Figure 1.1: Second versus third component of the solution to the adjoint equation at labels `4,5` and `7`

```
 ...

 BRANCH  1 N=  20 IT= 3 ...
 EIGENVALUES
 ...
 ORIENTABLE (    0.1807863751D+06)
 ...
 PSI( 13)
 USZR        FUNCTION =   0.144E+02

 ...
* DETECTION OF SINGULAR POINT : ...
 BRANCH  1 N=  35 IT= 0 ...
 BRANCH  1 N=  35 IT= 1 ...
 EIGENVALUES
 ...
 ORIENTABLE (    0.6491192560D-03)
 PSI( 13)
 USZR        FUNCTION =   0.373E-07

 ...

 BRANCH  1 N=  40 IT= 2 ...
 EIGENVALUES
 ...
 NON-ORIENTABLE (   -0.2617339717D+05)
 ...
 PSI( 13)
```

```
 USZR        FUNCTION =  -0.135E+01

  ...
```

whence we have computed an inclination flip at N=35 corresponding to label 5. Indeed, the corresponding test function is zero. Data for the adjoint equation at LAB=4, 5 and 7 at and on either side of the inclination flip are presented in Fig. 1.1. The switching of the solution between components of the leading unstable left eigenvector is apparent. However, the line

```
ORIENTABLE (    0.6491192560D-03)
```

at N=35 would seem to contradict the detection of the inclination flip at this point. Nonetheless, the important fact is the zero of the test function; and note that the value of the variable indicating the orientation is small compared to its value at the other regular points.

Finally, we remark that the Newton step in the dummy parameter PAR(20) performed above is crucial to obtain convergence. Indeed, if we try to continue the homoclinic orbit and the solution of the adjoint equation directly by setting

```
IRS = 2
```

in auttest.f and running

```
@autoh test
@svaut test2
```

we obtain the output

```
  BR   PT TY LAB    PAR(4)      ...       PAR(8)        PAR(10)
   1    2 MX   8  0.000000E+00 ...   -3.620305E-11  0.000000E+00
```

indicating a no-convergence error.

1.2 Non-orientable resonant eigenvalues

Inspecting the output of the computations performed in the previous section we observe the existence of a non-orientable homoclinic orbit at label 6 corresponding to PT=40. We restart at this label, with the first continuation parameter being once again a =PAR(1), by changing constants in auttest.f according to

```
IRS = 6   DS = -0.05   NMX = 20   ICP(1) = 1
```

Running

```
@autoh test
```

the output is given by

```
 BR   PT TY LAB    PAR(1)      ...       PAR(8)        PAR(10)
  1    8 UZ   8  2.341636E-09  ...   -5.160054E-12   -7.697015E-10
  1   20 EP   9 -4.916798E-01  ...   -2.077344E-12   -7.551938E-10
```

which we choose to append to the previous data

```
@apaut test
```

(however, for plotting purposes it may sometimes be advantageous to @svaut to save the data to a new file). The file d.test contains the lines

```
 * DETECTION OF SINGULAR POINT : ...
 BRANCH  1 N=   8 IT= 0 ...
 BRANCH  1 N=   8 IT= 1 ...
 EIGENVALUES
```

```
 ...
 NON-ORIENTABLE (   -0.1314592670D+05)
 PSI(  1)
 USZR        FUNCTION =   0.468E-08
```

indicating that a non-orientable resonant bifurcation occurred at that point.

1.3 Orbit flip

In this subsection we compute an orbit flip. To this end we restart from the original explicit solution, but with $\alpha = 1$, $\beta = 0$ and without computing the orientation. We begin by separately performing continuation in $(a, \tilde{\mu})$ and $(b, \tilde{\mu})$ in order to reach the parameter values $(a, b) = (0.5, 0.25)$. First, we change the following constants in `auttest.f` to read

```
                ITWIST = 0   IRS = 0   DS = 0.05
             PAR(4)= 1.0   PAR(5)= 0.0   PAR(11) = 10.0
            NPUSZR = 1   NPSI = 0   PUSZR = PAR(1) - 0.5
```

and run the program

```
                                @autoh test
```

to get the following output

```
  BR   PT TY LAB      PAR(1)      ...       PAR(8)
   1    1 EP   1   0.000000E+00  ...   0.000000E+00
   1    8 UZ   2   4.999999E-01  ...  -7.122160E-05
   1   20 EP   3   1.668793E+00  ...  -2.219671E-02
```

Saving this data via

```
                                @svaut test
```

will over-write the previous output (which you should therefore have copied elsewhere if you had wished to keep). On changing `auttest.f`

```
          IRS = 2   NMX = 30   ICP(1) = 2   PUSZR = PAR(2) - 3.0
```

and rerunning

```
                                @autoh test
                                @apaut test
```

we obtain

```
  BR   PT TY LAB       PAR(2)      ...       PAR(8)
   1   20      4    2.613317E+00  ...  -3.727036E-11
   1   24 UZ   5    3.000000E+00  ...  -1.743765E-10
   1   30 EP   6    3.597855E+00  ...  -3.090265E-10
```

Next we perform continuation with respect to μ =PAR(7)

```
          IRS = 5   NMX = 20   ICP(1) = 7   PUSZR = PAR(7) - 0.25
```

and again run

```
                                @autoh test
                                @apaut test
```

to produce

```
  BR   PT TY LAB    PAR(7)       ...       PAR(8)
   1    5 UZ   7  2.500000E-01  ...   6.732622E-02
   1   20 EP   8  1.737059E+00  ...   4.631532E-01
```

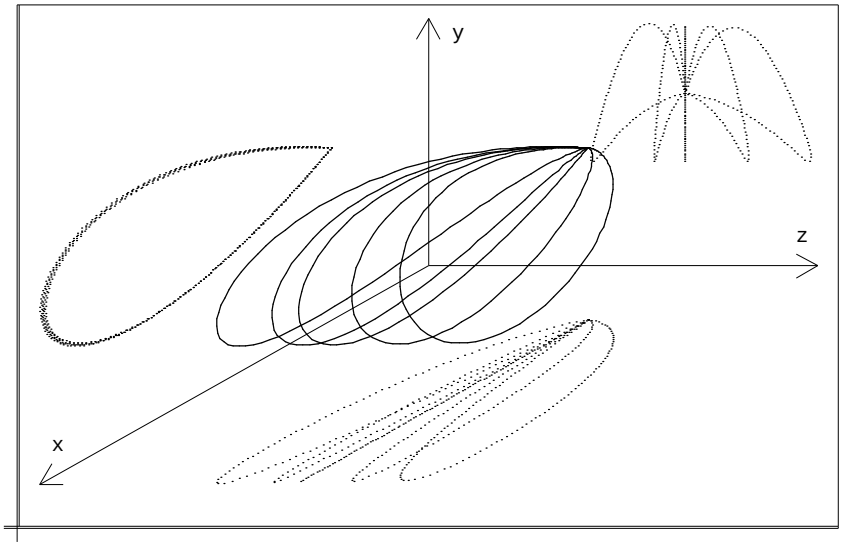

Figure 1.2: Orbits either side of the orbit flip bifurcation. The critical orbit is contained in the (x, y)-plane

The final step consists of computing in the other direction in μ towards $\mu = 0$ with an appropriate test function for an orbit flip activated. We make the following alterations

```
IRS = 7   DS = -0.05   NPUSZR = 0   NPSI = 1   IPSI(1) = 11
```

and rerun

```
@autoh test
@apaut test
```

The output yields an orbit flip bifurcation

```
 BR   PT TY LAB     PAR(7)       ...      PAR(8)
  1    5 UZ   9  -5.008722E-08   ...   -1.370673E-08
  1   20 EP  10  -1.486437E+00   ...   -4.081331E-01
```

at approximately $(\mu, \tilde{\mu}) = 0$ which is the value predicted by theory. Note that the critical orbit is contained in the (x, y)-plane (see Fig. 1.2).

2. Predator-prey model by M. Scheffer

Consider the following system of two equations (Scheffer 1995)

$$\begin{cases} \dot{X} &= RX\left(1 - \dfrac{X}{K}\right) - \dfrac{A_1 XY}{B_1 + X} + D_0 K \\ \dot{Y} &= E_1 \dfrac{A_1 XY}{B_1 + X} - D_1 Y - \dfrac{A_2 ZY^2}{B_2^2 + Y^2}. \end{cases} \tag{2.1}$$

The values of all parameters except (K, Z) are set as follows:

$$R = 0.5,\ A_1 = 0.4,\ B_1 = 0.6,\ D_0 = 0.01,\ E_1 = 0.6,\ A_2 = 1.0,\ B_2 = 0.5,\ D_1 = 0.15.$$

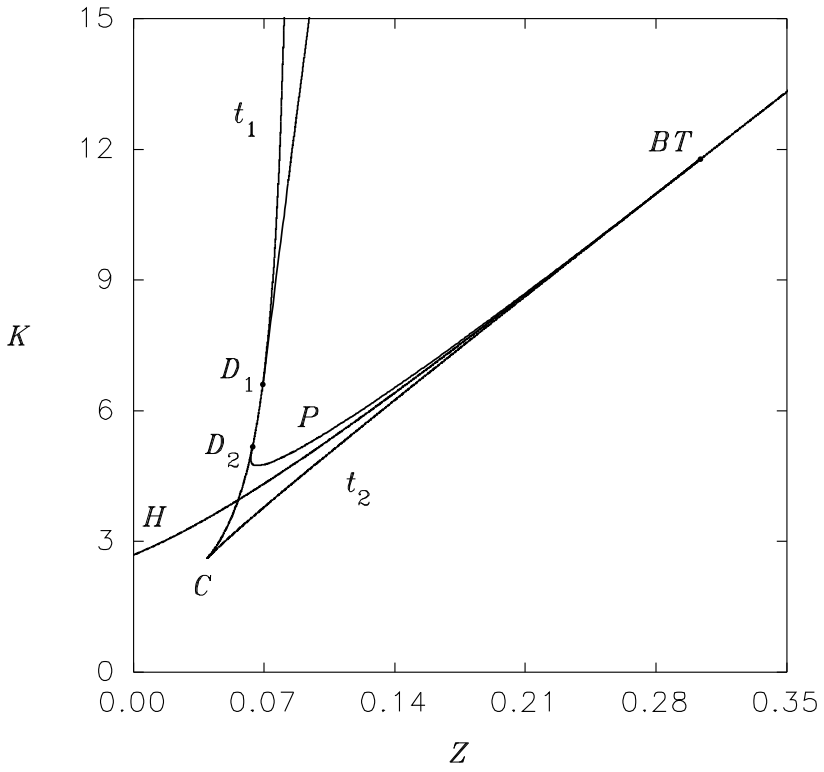

Figure 2.1: Parametric portrait of the predator-prey system

The parametric portrait of the system (2.1) on the (Z, K)-plane is presented in Figure 2.1. It contains fold ($t_{1,2}$) and Hopf (H) bifurcation curves, as well as a homoclinic bifurcation curve P. The fold curves meet at a cusp singular point C, while the Hopf and the homoclinic curves originate at a Bogdanov-Takens point BT. Only the homoclinic curve P will be considered here, the other bifurcation curves can be computed using AUTO86 or LOCBIF (Khibnik, Kuznetsov, Levitin & Nikolaev 1993).

2.1 Continuation of central saddle-node homoclinics

Local bifurcation analysis shows that at $K = 6.0$, $Z = 0.06729762\ldots$, the system has a saddle-node equilibrium

$$(X^0, Y^0) = (5.738626\ldots, 0.5108401\ldots),$$

with one zero and one negative eigenvalue. Direct simulations reveal a homoclinic orbit to this saddle-node, departing and returning along its central direction (i.e., tangent to the null-eigenvector).

Starting from this solution, stored in the file `pstpnt.dat`, we continue the saddle-node central homoclinic orbit with respect to the parameters K and Z by running

```
@autoh marten
```

in the directory `marten`. The file `autmarten.f` contains approximate parameter values corresponding to the homoclinic orbit,

$$K = \texttt{PAR(1)} = 6.0,\ Z = \texttt{PAR(2)} = 0.06729762,$$

as well as the coordinates of the saddle-node

$$X^0 = \texttt{PAR(12)} = 5.738626,\ Y^0 = \texttt{PAR(13)} = 0.5108401,$$

and the length of the truncated time-interval

$$T_0 = \texttt{PAR(11)} = 1046.178$$

Since a homoclinic orbit to a saddle-node is being followed, we also set

```
IEQUIB = 2   NUNSTAB = 0   NSTAB = 1
```

and monitor two test-functions to detect non-central saddle-node homoclinic orbits:

```
NPSI = 2   IPSI(1) = 15   IPSI(2) = 16
```

Among the output there is a line

```
 BR   PT TY LAB    PAR(1)        L2-NORM      ...    PAR(2)      ...
  1   23 UZ   4  6.610455E+00  6.254906E+00   ...  6.932481E-02 ...
```

indicating that a zero of the test function `IPSI(1)=15` (see the output in `fort.9`) has been accurately located. This means that at

$$D_1 = (K^1, Z^1) = (6.610458\ldots, 0.06932482\ldots)$$

the homoclinic orbit to the saddle-node becomes *non-central*, namely, it returns to the equilibrium along the stable eigenvector, forming a non-smooth loop. Save the output in the usual way

```
@svaut marten
```

Repeating computations in the opposite direction along the curve by editing the constants

```
IRS = 1   DS = -0.01
```

in `autmarten.f` and running it

```
@autoh marten
```

one obtains

```
 BR   PT TY LAB    PAR(1)        L2-NORM      ...    PAR(2)      ...
  1   29 UZ   8  5.180308E+00  4.819672E+00   ...  6.385499E-02 ...
```

which means another non-central saddle-node homoclinic bifurcation occurs at

$$D_2 = (K^2, Z^2) = (5.180308\ldots, 0.06385499\ldots).$$

Save the output by typing

```
@apaut marten
```

2.2 Switching between saddle-node and saddle homoclinic orbits

Now we can switch to continuation of saddle homoclinic orbits at the located codim 2 points D_1 and D_2. For this, make the following changes in `autmarten.f`:

```
NUNSTAB = 1   IEQUIB = 1   IRS = 4   DS = 0.01   DSMAX = 0.5   NMX = 40
```

and set

```
IPSI(1) = 9   IPSI(2) = 10
```

to monitor for nonhyperbolic equilibria along the homoclinic locus. On running

```
@autoh marten
```

we get the following output

```
 BR   PT TY LAB    PAR(1)        L2-NORM      ...    PAR(2)       ...
  1   10     10  6.968309E+00  6.605262E+00   ...  7.038941E-02   ...
  1   20     11  8.698321E+00  8.288411E+00   ...  7.539922E-02   ...
  1   30     12  1.214421E+01  1.162850E+01   ...  8.554196E-02   ...
  1   40 EP  13  1.573264E+01  1.510738E+01   ...  9.639685E-02   ...
```

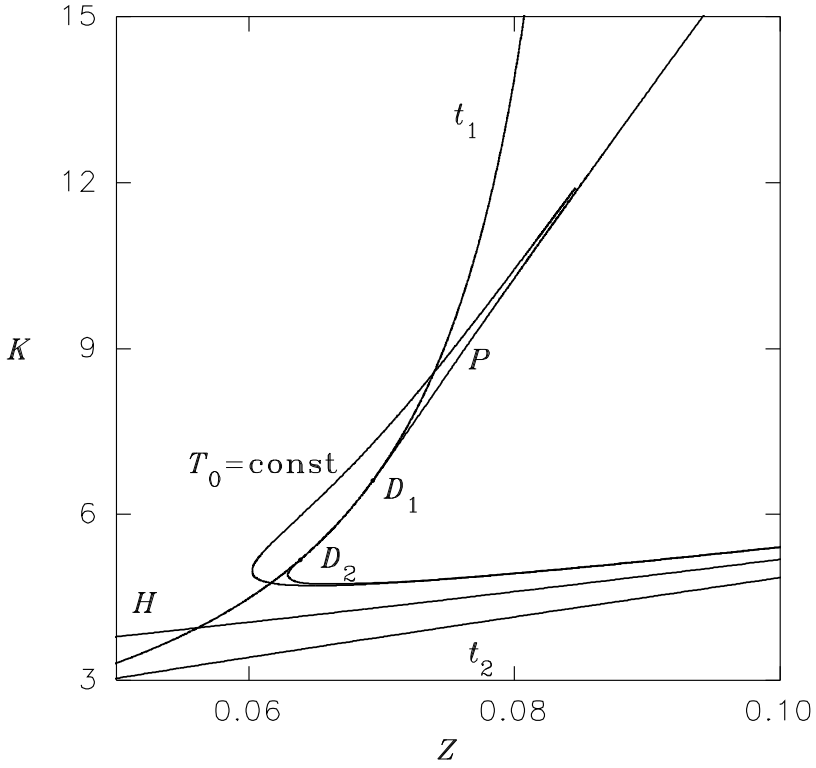

Figure 2.2: Approximation by a large-period cycle

This is the upper branch of P in Figure 2.1. Append this data to the stored results

```
@apaut marten
```

Notice that restarting in the opposite direction with `IRS=10, DS=-0.01` will detect the same codim 2 point D_1 but now as a zero of the test-function `IPSI(1)=9`

BR	PT	TY	LAB	PAR(1)	L2-NORM	...	PAR(2)	...
1	10	UZ	14	6.610459E+00	6.254910E+00	...	6.932482E-02	...

Note that the value of `PAR(1)` differs from that at label `4` only in the final decimal place. Actually, the program runs further and eventually computes the point D_2 and the whole lower branch of P emanating from it, however, the solutions between D_1 and D_2 should be considered as spurious[2], therefore we do not save this data. The reliable way to compute the lower branch of P is to restart `@autoh marten` from the point `LAB=8` by setting

```
IRS = 8     NMX = 50
```

in `autmarten.f`. This gives the lower branch of P approaching the Bogdanov-Takens point BT (see Figure 2.1)

BR	PT	TY	LAB	PAR(1)	L2-NORM	...	PAR(2)	...
1	10		14	4.990545E+00	4.610448E+00	...	6.305166E-02	...
1	20		15	4.944760E+00	4.196061E+00	...	8.054530E-02	...
1	30		16	6.992942E+00	5.204561E+00	...	1.554437E-01	...
1	40		17	1.121011E+01	7.433009E+00	...	2.867595E-01	...
1	50	EP	18	1.168976E+01	7.693878E+00	...	3.011494E-01	...

[2] The program actually computes the saddle-saddle heteroclinic orbit bifurcating from the non-central saddle-node homoclinic at the point D_1, see Champneys et al. (1995, Fig. 2), and continues it to the one emanating from D_2.

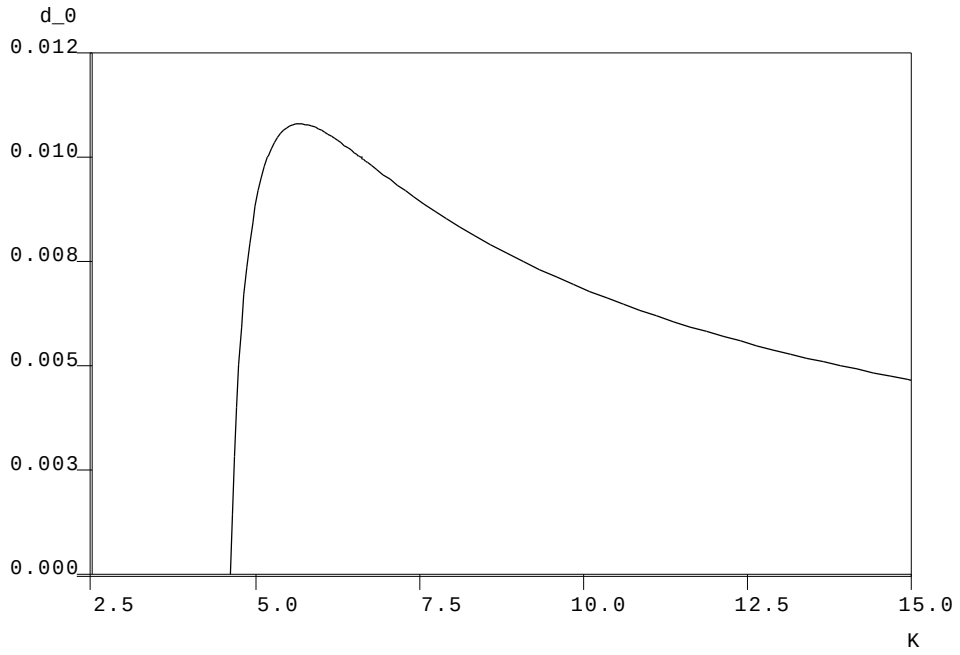

Figure 2.3: Projection onto the (K, D_0)-plane of the three-parameter curve of non-central saddle-node homoclinic orbit

Upon appending this data to the stored results

```
@apaut marten
```

one could now plot the entire data to reproduce both branches of the curve P shown in Figure 2.1.

It is worthwhile to compare the homoclinic curves computed above with a curve $T_0 = const$ along which the system has a limit cycle of constant large period $T_0 = 1046.178$, which can easily be computed using AUTO86 or LOCBIF. Such a curve is plotted in Figure 2.2. It obviously approximates well the saddle homoclinic loci of P, but demonstrates much bigger deviation from the saddle-node homoclinic segment D_1D_2. This happens because the period of the limit cycle grows to infinity while approaching both types of homoclinic orbit, but with *different asymptotics*: as $\|\alpha - \alpha^*\|^\gamma$, where $\gamma = \mu_1/\lambda_1$ in the saddle homoclinic case, and as $-\ln\|\alpha - \alpha^*\|$ in the saddle-node case.

2.3 Three-parameter continuation

Finally, we can follow the curve of non-central saddle-node homoclinic orbits in three parameters. The extra continuation parameter is D_0=`PAR(3)`. To achieve this we restart at label `4`, corresponding to the codim 2 point D_1. We return to continuation of saddle-node homoclinics, but append the defining equation $\psi_{15} = 0$ to the continuation problem. This is achieved by making the following changes to `autmarten.f`.

```
NUNSTAB = 0   IEQUIB = 2   IRS = 4   DS = 0.001   DSMAX = 0.1   NFIXED = 1
```

To specify the free parameters, one can type

```
NFREE = 3   ICP(1) = 3   ICP(2) = 1   ICP(3) = 2
```

Notice that we consider D_0 as the first continuation parameter because AUTO86 detects limit points with respect to this parameter. To this end we also set

```
ILP = 1   NPUSZR = 1   NPSI = 0
```

the second of which is used to activate the first user-defined output function in PUSZR which detects the intersection with the plane $D_0 = 0.01$, while the last discharges monitoring of test functions. Upon running using

```
@autoh marten
```

we get among other output

```
 BR    PT TY LAB    PAR(3)       ...    PAR(1)          PAR(2)       ...
  1    20 LP  20  1.081234E-02   ...  5.673620E+00   6.608181E-02   ...
  1    26 UZ  21  1.000000E-02   ...  5.180308E+00   6.385499E-02   ...
```

the first line of which represents the D_0 value at which the homoclinic curve P has a tangency with the branch t_2 of fold bifurcations. Beyond this value of D_0, P consists entirely of saddle homoclinic orbits. The data at label 21 reproduces the coordinates of the point D_2. The results of this computation and a similar one starting from D_1 in the opposite direction (DS=-0.001) are displayed in Figure 2.3.

3. Koper's extended Van der Pol model

The example file autkoper.f in the directory koper contains the equations

$$\begin{cases} \dot{x} &= \varepsilon_1^{-1}\,(k\,y - x^3 + 3\,x - \lambda) \\ \dot{y} &= x - 2\,y + z \\ \dot{z} &= \varepsilon_2(y - z), \end{cases} \tag{3.1}$$

with $\varepsilon_1 = 0.1$ and $\varepsilon_2 = 1$ (Koper (1994, 1994)).

3.1 The primary branch of homoclinics

First, we solve for a homoclinic orbit using the homotopy method (ISTART=3). To do this, we take the AUTO86 constants as initially specified in autkoper.f, which already contains approximate parameter values for a homoclinic orbit, namely λ =PAR(1)=-1.851185, k =PAR(2)=-0.15. We begin with continuation in $2T$=PAR(11);

```
@autoh koper
```

Among the output there is the line

```
 BR    PT  TY LAB    PAR(11)       L2-NORM      ...    PAR(17)     ...
  1    26  UZ   3  1.908778E+01  1.693730E+00   ... -5.845273E-11 ...
```

which indicates that a zero has been located of the artificial parameter ω_1 = PAR(17), which measures the distance of the solution at the right-hand endpoint from the linearized stable manifold. The continuation ends in a no convergence error

```
  1    46  MX   6  2.197655E+01  1.687785E+00   ...  1.149913E+00 ...
```

as the right-hand endpoint leaves the saddle close to its unstable manifold. We can save this output in the usual way

```
@svaut koper
```

However, upon plotting the data at label 3 (see Figure 3.1) it can be noted that although the right-hand projection boundary condition is satisfied, the solution is still quite away from the equilibrium. The right-hand endpoint can be made to approach the equilibrium by performing a further continuation in T with the right-hand projection condition satisfied (PAR(17) fixed) but with λ allowed to vary. That is, edit autkoper.f, so that the following two constants are altered to read

```
IRS = 3     ICP(2) = 1
```

Note that values are assigned to the variables ICP(..) at two places in autkoper.f depending on whether ISTART=3 or ISTART=1,2. Here we change ICP(2) at the *second* place, that is, in line 222 of autkoper.f. Running AUTO86 again using

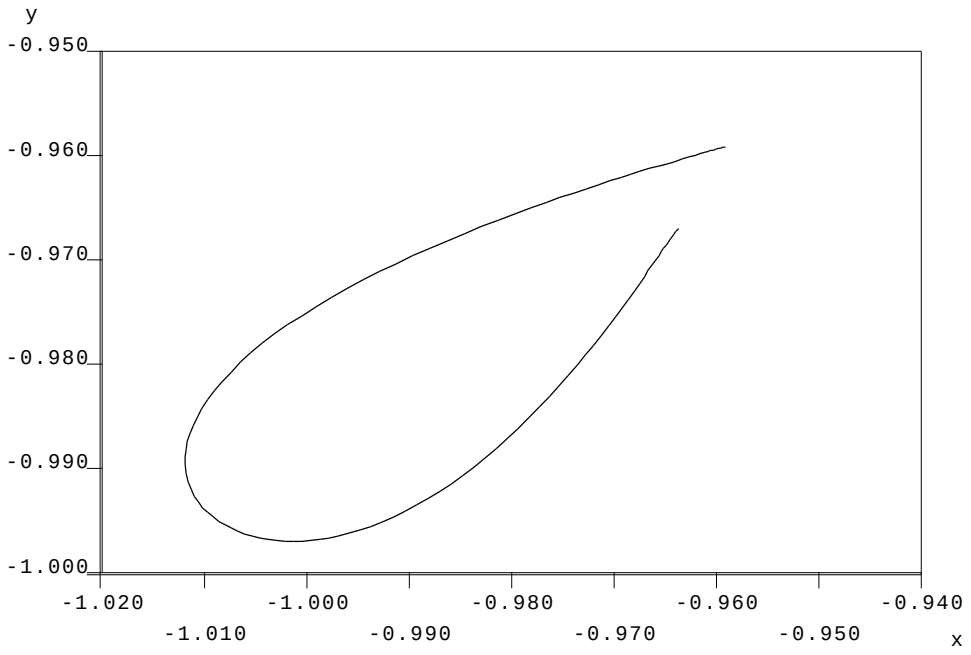

Figure 3.1: Projection on the (x, y)-plane of solutions of the boundary value problem with `PAR(11) = 19.08778`

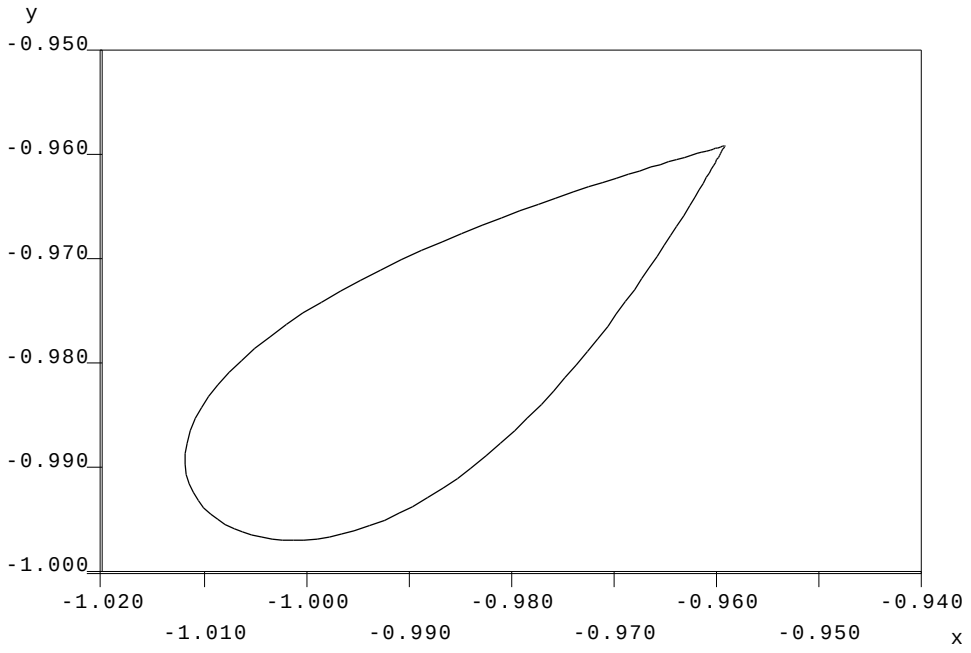

Figure 3.2: Projection on the (x, y)-plane of solutions of the boundary value problem with `PAR(11) = 60`

```
@autoh koper
```

the output at label 10

```
  BR   PT TY LAB    PAR(11)       L2-NORM      ...     PAR(1)      ...
   1   34 UZ  10  6.000000E+01  1.672806E+00  ...  -1.851185E+00 ...
```

provides a good approximation to a homoclinic solution (see Figure 3.2). We add this data to that already obtained in the usual way

```
@apaut koper
```

The second stage to obtain a starting solution is to add a solution to the modified adjoint variational equation. This is done by a simple two-step process. First we add some trivial data to that defining the homoclinic orbit at label 10.

```
@adjaut 10 koper koper2
```

The new data is now stored in `q.koper2`. We start with a Newton step in a trivial parameter in order to solve the linear adjoint equation. To do this, make the following changes to the constants in `autkoper.f`

```
ITWIST = 1   ISTART = 1   IRS = 10   NMX = 2   NPR = 2
```

and save the new file to `autkoper2.f`. Continuation is now performed with respect to the dummy parameter `PAR(20)`. Upon running AUTO86 and saving the output

```
@autoh koper2
@apaut koper2
```

the output at the second point contains the converged homoclinic solution (variables `(U(1), U(2), U(3))` and the adjoint `(U(4), U(5), U(6))`). We now have a starting solution and are ready to perform two-parameter continuation.

We make the following changes in `autkoper2.f`

```
IRS = 11   DS = 0.02   DSMAX = 0.2   NMX = 60   NPR = 2
  ICP(1) = 1   (the first occurrence, line 176 in autkoper2.f)
```

Note that this small value of `NPR=2` is kept in order to produce detailed output near the inclination-flip points computed below; if the user has a limited filespace available, then we recommend taking a larger value, say `NPR=10`. Then we run again

```
@autoh koper2
```

Among the output we find two zeroes of the test function `IPSI(2)=13` (see the output in `fort.9`), which gives the accurate location of two inclination-flip bifurcations,

```
 BR   PT TY LAB    PAR(1)        ...      PAR(2)         PAR(10)    ...
  1   14 UZ  18 -1.801663E+00    ...  -2.002655E-01  -2.317688E-08  ...
  1   20 UZ  21 -1.568756E+00    ...  -4.395466E-01   2.578296E-09  ...
```

and a point at which the equilibrium undergoes a saddle-node bifurcation (a zero of the test function `IPSI(1)=9`), namely a non-central saddle-node homoclinic orbit

```
  1   54 UZ  38  1.765060E-01    ...  -2.405332E+00   8.769192E-09  ...
```

Any output beyond the point `LAB=38` is spurious[3]. Note from this output, that at each computed point (not just the codim 2 points) the artificial parameter ϵ =`PAR(10)` is zero to within the allowed tolerance, as it should be theoretically. This output is saved rather than appended to the previous data,

[3] See footnote 2 or Champneys et al. (1995, Fig. 2).

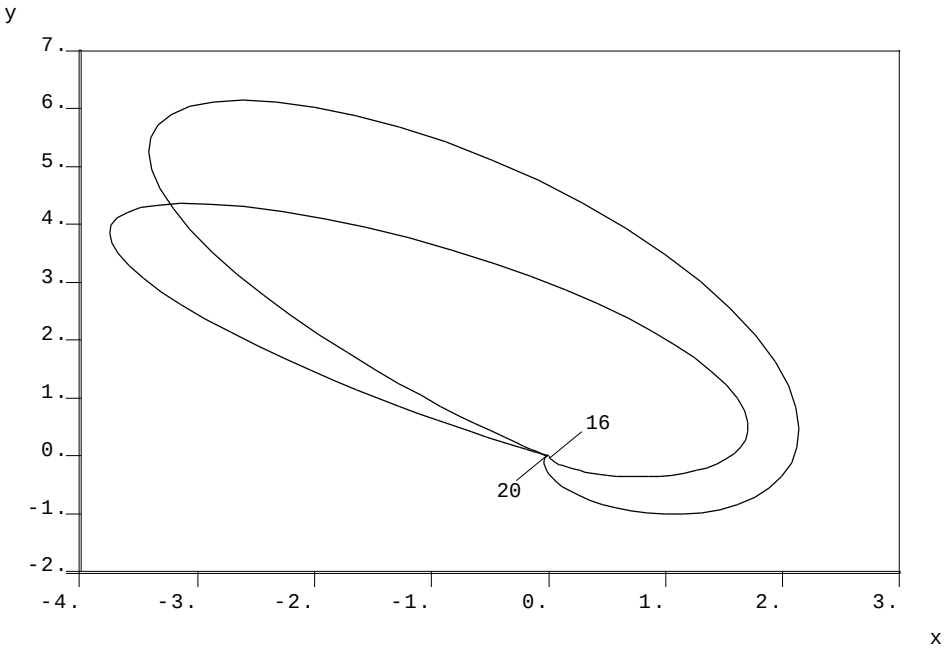

Figure 3.3: Projection on the (x, y)-plane of solutions $\varphi(t)$ at `LAB=16` (`PAR(1)=-1.825470, PAR(2)=-0.1760749`) and `LAB=20` (`PAR(1)=-1.686154, PAR(2)=-0.3183548`)

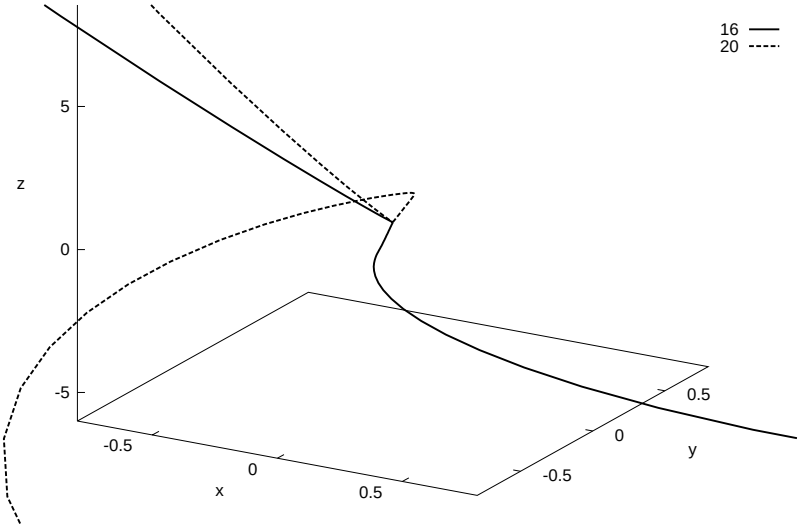

Figure 3.4: Three-dimensional blow-up of the solution curves $\varphi(t)$ at `LAB=16` (solid line) and `LAB=20` (dotted)

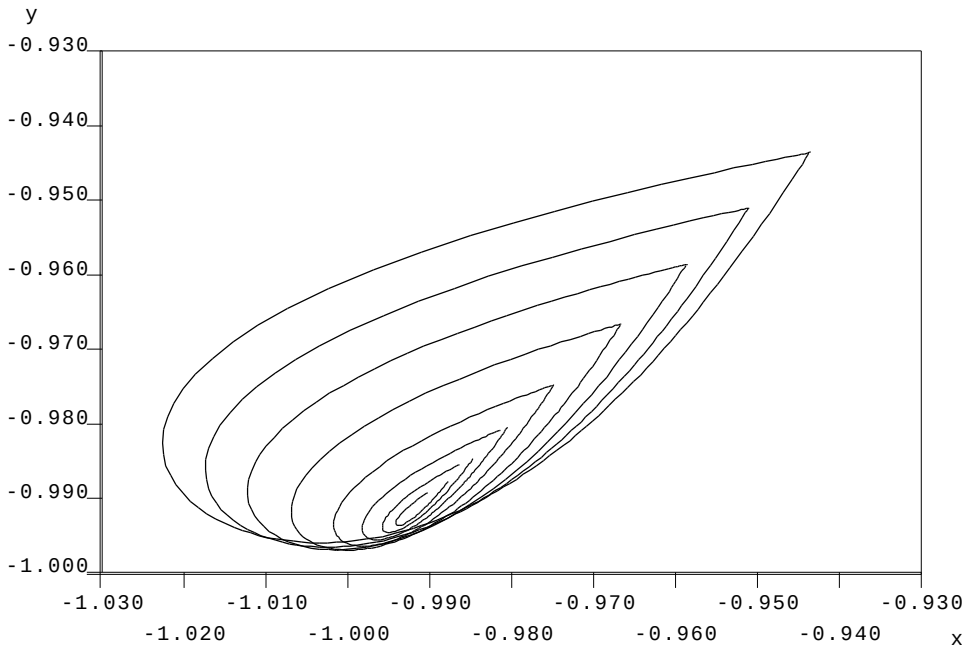

Figure 3.5: Computed homoclinic orbits approaching the BT point

```
@svaut koper2
```

since that contained the artificial Newton step which no longer need. Figure 3.3 presents solutions $\varphi(t)$ of the modified adjoint variational equation (for details see Sandstede (1995*c*)) at `LAB=16` and `20` which are points on the homoclinic branch before and after the first detected inclination flip. A blow-up of the region close to the origin of this figure is shown in Figure 3.4 illustrating the flip of the solutions of the adjoint equation while moving through the bifurcation point. Note that the data in Figure 3.4 was plotted after first performing an additional continuation of the solutions at both points with respect to the truncation interval `PAR(11)`.

Continuing in the other direction by changing `autkoper2.f`

```
IRS = 12   DS = -0.02   DSMAX = 0.05   NPR = 10
```

and running

```
@autoh koper2
```

we approach a Bogdanov-Takens point.

```
 BR   PT TY LAB    PAR(1)        ...      PAR(2)         PAR(10)     ...
  1   60 EP  47 -1.949015E+00   ...  -5.120101E-02  -8.332154E+01  ...
```

Note that the numerical approximation has ceased to become reliable, since `PAR(10)` has now become large. To follow the homoclinic orbit to the BT point with more precision, we would need to first perform continuation in $2T$ (`PAR(11)`) to obtain a more accurate homoclinic solution (see Section 3.3.2 below). We could now plot the locus of homoclinic bifurcations so far obtained, after first appending this data to that already obtained for the branch of homoclinics

```
@apaut koper2
```

Phase portraits of homoclinic orbits between the BT point and the first inclination flip (in fact, between labels `12` and `18` and from labels `42` to `46`) are depicted in Figure 3.5, note how the computed homoclinic orbits approaching the BT point have their endpoints well away from the equilibrium, again showing that we need to take a larger truncation interval.

3.2 More accuracy and saddle-node homoclinic orbits

To analyze the branch of homoclinic orbits further, we first perform continuation in T from one point on the previously computed branch in order to obtain an approximation of the homoclinic orbit over a longer interval. This is necessary for parameter values near a non-hyperbolic equilibrium (either a saddle-node or BT) where the convergence to the equilibrium is slower. First, we pick a point well away from the non-hyperbolic equilibrium (label 12) and remove the data for the computation of the adjoint variational equation, because inclination flips will not be involved in what follows.

```
@delhaut 12 koper2 koper3
```

The following changes are then made to `autkoper2.f`

```
ITWIST = 0   DS = 2.0   DSMAX = 10.0   NMX = 100   NPR = 20
             NPSI = 2   RL1 = 2000.0   ICP(1) = 11
```

with the resulting file renamed as `autkoper3.f`. We then run this new file

```
@autoh koper3
```

The last point in this computation is

```
 BR   PT TY LAB    PAR(11)        L2-NORM      ...      PAR(2)      ...
  1  100 EP  17  9.855682E+02  1.661016E+00    ...  -1.516927E-01 ...
```

We can now repeat the computation of the branch of saddle homoclinic orbits from this point, by saving this data

```
@svaut koper3
```

setting

```
IRS = 17   DS = 0.02   DSMAX = 0.1   ICP(1) = 1
```

in `autkoper3.f` and running in the usual way

```
@autoh koper3
```

The saddle-node point is now detected at

```
 BR   PT TY LAB    PAR(1)       L2-NORM       ...      PAR(2)       ...
  1   46 UZ  20  1.764948E-01  7.472675E-01   ...  -2.405358E+00   ...
```

Note that the parameter values differ from that at the previously-computed saddle-node homoclinic point only in the fifth decimal place. We save this output as a new file

```
@svaut koper4
```

Replacing

```
DS = -0.01   DSMAX = 0.02   NMX = 20   NPR = 10
```

in `autkoper3.f` and rerunning

```
@autoh koper3
```

results in a more accurate approximation to the curve of homoclinics approaching the BT point

```
 BR   PT TY LAB    PAR(1)        L2-NORM       ...      PAR(2)       ...
  1   10     18 -1.945502E+00  1.714829E+00    ...  -5.474625E-02   ...
  1   20 EP  19 -1.950577E+00  1.717653E+00    ...  -4.962832E-02   ...
```

Note that we do not save this output.

To switch to continuation of the central saddle-node homoclinic curve in two parameters from the non-central saddle-node homoclinic orbit at `LAB=20`, we make the following changes to `autkoper3.f`

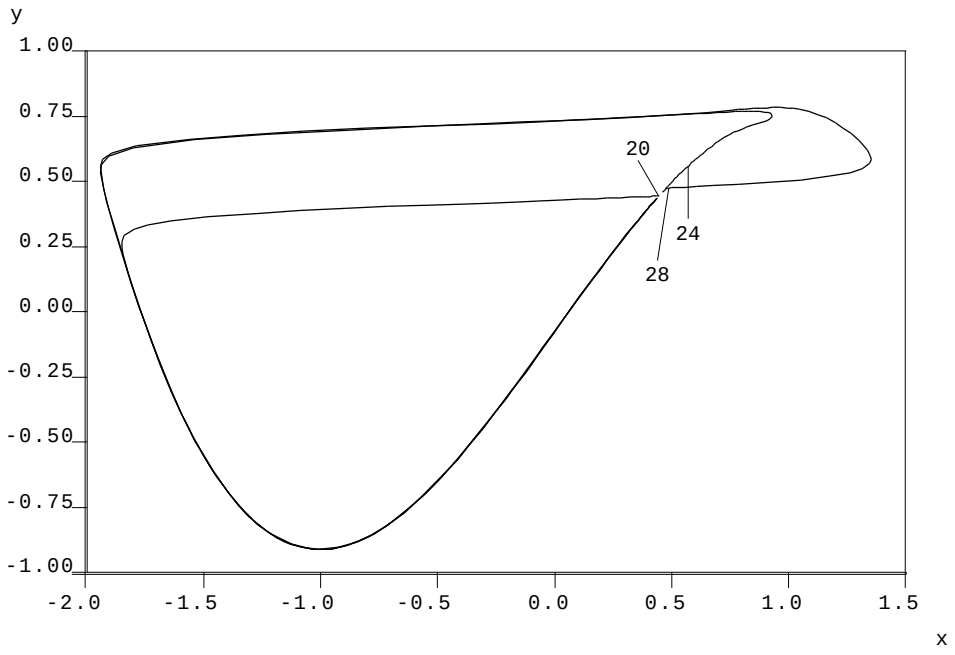

Figure 3.6: Two non-central saddle-node homoclinic orbits (LAB=20 and 28), and a central saddle-node homoclinic orbit between these two points (LAB=24).

```
NSTAB = 1   IEQUIB = 2   IRS = 20   DS = 0.01
NMX = 30   NPR = 5   IPSI(1) = 15   IPSI(2) = 16
```

which we rename to `autkoper4.f`. Upon running, among the output, we get

```
@autoh koper4
@apaut koper4
```

```
 BR   PT TY  LAB    PAR(1)       L2-NORM     ...     PAR(2)      ...
  1   28 UZ  28  1.764896E-01  7.491793E-01 ...  -2.405369E+00 ...
```

At LAB=28, the branch of homoclinic orbits once again leaves the locus of saddle-nodes in a second non-central saddle-node homoclinic bifurcation (a zero of ψ_{16}). Using the 2d function in `@plaut` at this stage to plot a phase space diagram (see Figure 3.6) shows clearly that, between the two codimension-two points (labels 20 and 28), the homoclinic orbit rotates between the two components of the 1D stable manifold, i.e. between the two boundaries of the center-stable manifold of the saddle node. The overall effect of this process is the transformation of a nearby "small" saddle homoclinic orbit to a "big" saddle homoclinic orbit (i.e. with two extra tuning points in phase space).

Finally, we can switch to continuation of the big saddle homoclinic orbit from the new codim 2 point. To this end we change constants according to

```
NSTAB = 2   IEQUIB = 1   IRS = 28   DSMAX = 0.2
       NMX = 400   NPR = 40   NPSI = 0
```

in `autkoper4.f`, rerun and append the data

```
@autoh koper4
@apaut koper4
```

Note that AUTO86 takes a large number of steps near the line PAR(1)=0, while PAR(2) approaches $-2.189\ldots$ (which is why we chose such a large value of NMX). This particular computation ends at

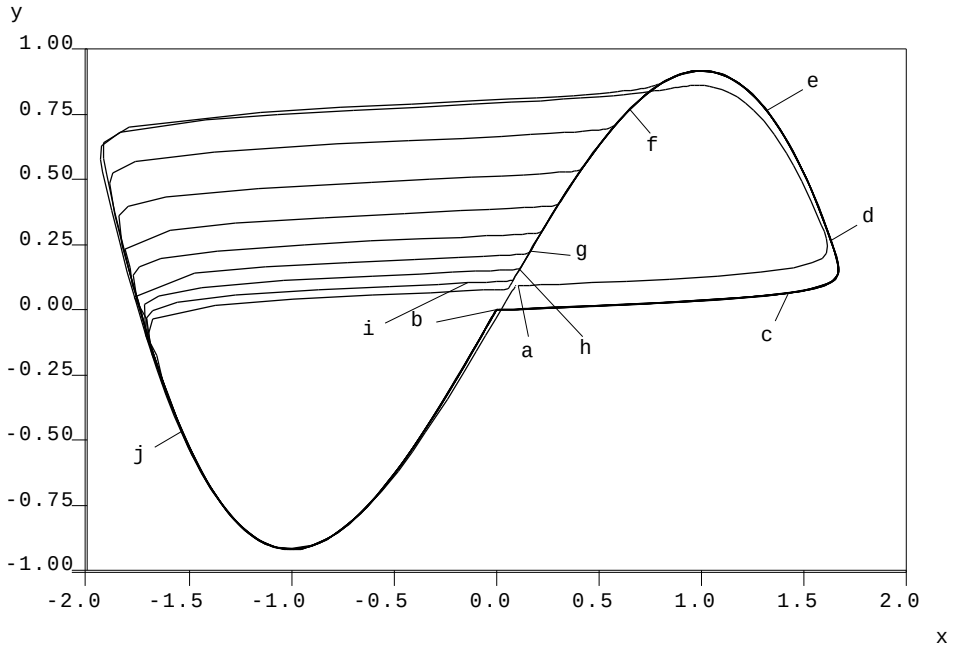

Figure 3.7: The big homoclinic orbit approaching a figure-of-eight

```
 BR   PT TY LAB    PAR(1)       L2-NORM       ...     PAR(2)      ...
  1  400 EP  39 -2.026640E-04  2.194528E-01  ...  -2.189919E+00 ...
```

By plotting phase portraits of the last ten orbits approaching this end point (see Figure 3.7) we see a "canard-like" like transformation of the big homoclinic orbit to a pair of homoclinic orbits in a figure-of-eight configuration. That we get a figure-of-eight is not a surprise because `PAR(1)=0` corresponds to a symmetry in the differential equations (Koper 1994); note also that the equilibrium, stored as (`PAR(12), PAR(13), PAR(14)`), approaches the origin as we approach the figure-of-eight homoclinic.

3.3 Three parameter continuation

We now consider curves in three parameters of each of the codimension-two points encountered in Section 3.3.2, by freeing the parameter ε = `PAR(3)`. In order to continue in three parameters the inclination flips detected at label `18`, we make the following changes to `autkoper2.f`

```
      IRS = 18   DS = -0.2   DSMAX = 0.5   NMX = 35
  NPSI = 1   NFIXED = 1   IFIXED(1) = 13   NFREE = 3
        ICP(1) = 3   ICP(2) = 1   ICP(3) = 2
```

and run

```
@autoh koper2
```

We save the output to a new file

```
@svaut koper5
```

Among the output there is a codimension three point (zero of ψ_9) where the neutrally twisted homoclinic orbit collides with the saddle-node curve

```
 BR   PT TY LAB    PAR(3)       ...    PAR(1)        PAR(2)      ...
  1   32 UZ  51  5.744773E-01  ...  1.282702E-01 -2.519325E+00 ...
```

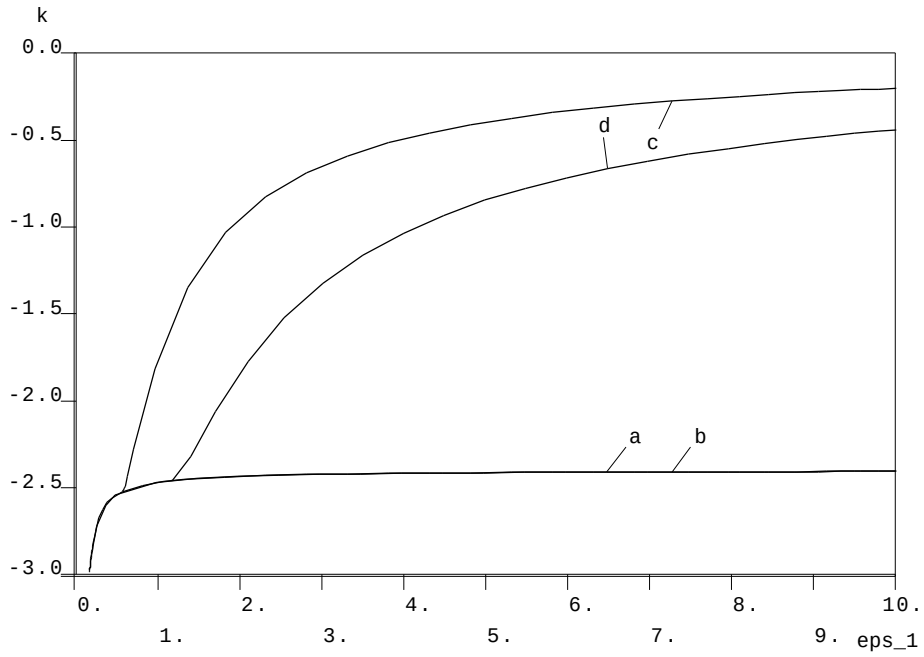

Figure 3.8: Projection onto the (PAR(3),PAR(2))-plane of the non-central saddle-node homoclinic orbit curves (labeled a and b) and the inclination-flip curves (labeled c and d)

We continue the other detected inclination flip (at LAB=21), by changing

```
IRS = 21   NMX = 25
```

in autkoper2.f, run the driver and append the data

```
@autoh koper2
@apaut koper5
```

Again we find a point at which the inclination-flip curve collides with that of the saddle-node homoclinic orbits

```
 BR   PT TY LAB     PAR(3)      ...      PAR(1)          PAR(2)     ...
   1   23 UZ  50  1.171705E+00 ...  1.535420E-01 -2.458100E+00  ...
```

To continue the non-central saddle-node homoclinic orbits it is necessary to work on the data without the solution $\varphi(t)$. We therefore restart from the data at LAB=20 and LAB=28 saved in koper4. We could continue these codim 2 points in two ways, either by appending the defining condition $\psi_{16} = 0$ to the continuation of saddle-node homoclinic orbits (IEQUIB=2, etc.), or by appending $\psi_9 = 0$ to the continuation of a saddle homoclinic orbit. The first approach was used in the example in Section 3.2, for contrast we shall adopt the second approach here. We achieve this by changing autkoper4.f according to

```
IRS = 20   DS = -0.2   DSMAX = 0.5   NMX = 50   NPR = 10
  NFREE = 3   ICP(1) = 3   NFIXED = 1   IFIXED(1) = 9
```

running in the usual way and saving the data to a new file

```
@autoh koper4
@svaut koper6
```

Similarly, we restart from label 28 by setting IRS=28 in autkoper4.f and rerun

```
@autoh koper4
@apaut koper6
```

The projection onto the (ε, k)-plane of all four of these codimension-two curves is given in Figure 3.8. The intersection of the inclination-flip lines with one of the non-central saddle-node homoclinic lines is apparent. Note that the two non-central saddle-node homoclinic orbit curves are almost overlaid.

4. Electronic circuit of Freire *et al*

Consider the following model of a three-variable autonomous electronic circuit (Freire, Rodríguez-Luis, Gamero & Ponce 1993)

$$\left\{\begin{array}{rcl} r\dot{x} &=& -\nu x + \beta(y-x) - A_3 x^3 + B_3(y-x)^3, \\ \dot{y} &=& -\beta(y-x) - z - B_3(y-x)^3, \\ \dot{z} &=& y. \end{array}\right. \tag{4.1}$$

The equations are included in the HomCont example file `autcircuit.f` in the directory `circuit`. We begin by reading in data from `pstpnt.dat` for a saddle-focus homoclinic orbit at $\beta = 0.6$, $\nu = -0.721309$, $r = 0.6$, $A_3 = 0.328578$ and $B_3 = 0.933578$, which was obtained by shooting over the time interval $2T =$`PAR(11)`$= 36.13$. We wish to follow the branch in the (β, ν)-plane, but first we perform continuation in (T, ν) to obtain a better approximation to a homoclinic orbit. Running AUTO86

```
@autoh circuit
```

yields the output

```
 BR   PT TY LAB    PAR(11)       L2-NORM       ...       PAR(2)
 1     1 EP   1  3.613000E+01  2.140388E-01   ...  -7.213090E-01
 1    21 UZ   2  1.000000E+02  1.286637E-01   ...  -7.213093E-01
 1    42 UZ   3  2.000000E+02  9.097897E-02   ...  -7.213093E-01
 1    50 EP   4  2.400000E+02  8.305206E-02   ...  -7.213093E-01
```

which we save in dummy files `p.circuit1`, `q.circuit1` and `d.circuit1`

```
@svaut circuit1
```

Note that $\nu =$`PAR(2)` remains constant during the continuation as the parameter values do not change, only the solution. We now restart at `LAB=3`, corresponding to a time interval $T = 200$, and change the principal continuation parameter to be β. To this end, the following changes are made to the file `autcircuit.f`

```
      IRS = 3    DS = -0.01    DSMAX = 0.05
       ICP(1) = 1    NMX = 30    NPR = 30
1 PUSZR=PAR(1)-0.1      2 PUSZR=PAR(1)+0.1
```

which is then saved as a new file `autcircuit1.f`. Rerunning AUTO86 via

```
@autoh circuit1
```

we get the output

```
 BR   PT TY LAB    PAR(1)        L2-NORM       ...       PAR(2)
  1    9 UZ   5  4.535585E-01  1.246500E-01   ...  -7.256936E-01
  1   17 UZ   6  1.000000E-01  2.228733E-01   ...  -9.196704E-01
  1   20 UZ   7 -6.218301E-09  2.754461E-01   ...  -1.026452E+00
  1   24 UZ   8 -1.000000E-01  3.711805E-01   ...  -1.154211E+00
  1   30 EP   9 -3.247670E-01  5.681331E-01   ...  -1.508869E+00
```

which is saved to `p.circuit` etc.

```
@svaut circuit
```

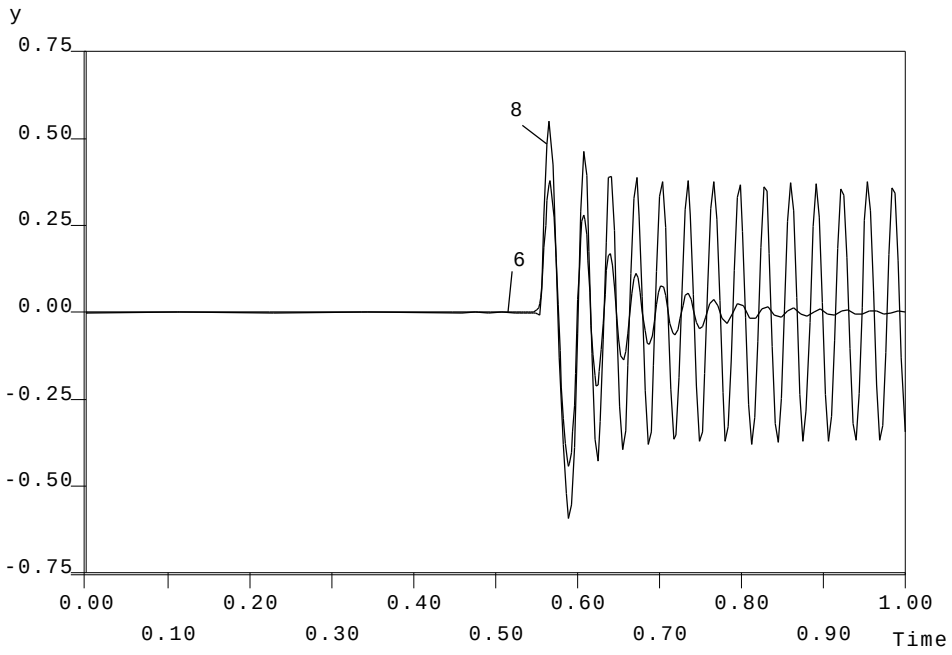

Figure 4.1: Solutions of the boundary value problem at labels 6 and 8 either side of the Shilnikov-Hopf bifurcation

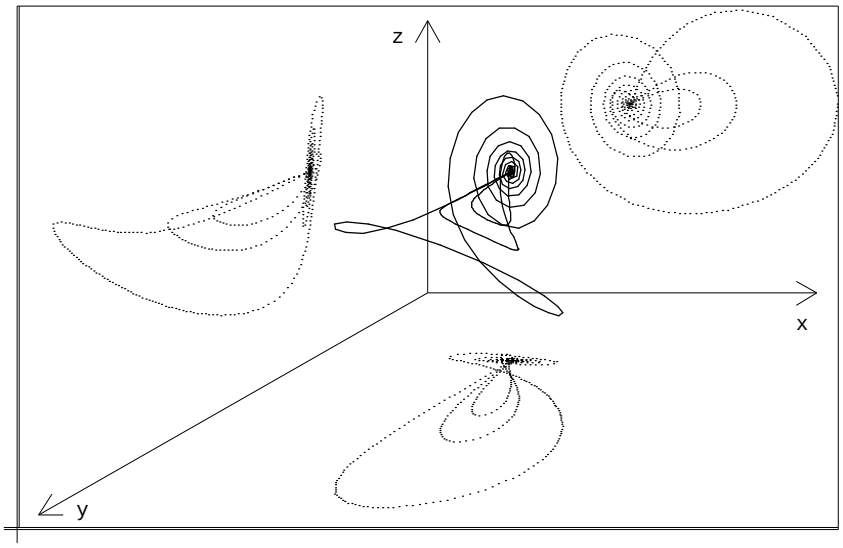

Figure 4.2: Phase portraits of three homoclinic orbits on the branch, showing the saddle-focus to saddle transition

Inspecting the output in `d.circuit`, we see that label `5` corresponds to neutrally-divergent saddle-focus ($\psi_5 = 0$), while label `7` corresponds to a local bifurcation ($\psi_9 = 0$) which we note from the eigenvalues there corresponds to a *Shilnikov-Hopf* bifurcation. Labels `6` and `8` are the user-defined output points, the solutions at which are plotted in Fig. 4.1. Note that solutions beyond label `7` (e.g. the plotted solution at label `8`) do not correspond to homoclinic orbits, but to *point-to-periodic* heteroclinic orbits (c.f. Section 2.2.1 of Champneys et al. (1995)).

To continue the locus in the other direction, we make the following changes to `autcircuit1.f`

```
IRS = 5    DS = 0.01
```

save the file to `autcircuit.f`, rerun and append the data

```
@autoh circuit
@apaut circuit
```

The output

```
 BR   PT TY LAB    PAR(1)        L2-NORM      ...     PAR(2)
  1    9 UZ  10  5.912146E-01  9.305334E-02  ...  -7.203984E-01
  1   13 UZ  11  7.428710E-01  4.869731E-02  ...  -7.590573E-01
  1   21 UZ  12  7.746145E-01  1.020295E-02  ...  -7.746679E-01
  1   23 MX  13  7.746301E-01  9.871745E-03  ...  -7.746785E-01
```

contains a neutral saddle-focus (a *Belyakov* transition) at `LAB=10` ($\psi_4 = 0$), a double real leading eigenvalue (saddle-focus to saddle transition) at `LAB=11` ($\psi_2 = 0$) and a neutral saddle at `LAB=12` ($\psi_4 = 0$). Data at several points on the complete branch are plotted in Fig. 4.2. The computation ends at a no convergence error `TY=MX` owing to the homoclinic branch approaching a Bogdanov-Takens singularity at small amplitude. To compute further towards the BT point one would first need to continue to a higher value of `PAR(11)`.

5. A HETEROCLINIC EXAMPLE

The following system of five equations by Rucklidge & Mathews (1995)

$$\begin{cases} \dot{x} &= \mu z + x y - z u, \\ \dot{y} &= -y - x^2, \\ \dot{z} &= -\nu z + x u, \\ \dot{u} &= -\dfrac{\sigma}{4} u - \dfrac{\sigma Q}{4\pi^2} v + \dfrac{3(1+\sigma)}{4\sigma} x z, \\ \dot{v} &= \dfrac{\zeta}{4} u - \dfrac{\zeta}{4} v \end{cases} \tag{5.1}$$

have been used to describe shearing instabilities in fluid convection. The equations possess a rich structure of local and global bifurcations. Here we shall reproduce a single curve in the (σ, μ)-plane of codimension-one heteroclinic orbits connecting a non-trivial equilibrium to the origin for $Q = 0$ and $\zeta = 4$. The defining problem is contained in the HOMCONT example file `autshear.f` in the directory `shear`, and starting data for the orbit at $(\sigma, \mu) = (0.5, 0.163875)$ is stored in `pstpnt.dat`, with a truncation interval of `PAR(11)=85.07`.

We begin by computing towards $\mu = 0$

```
@autoh shear
@svaut shear
```

which yields the output

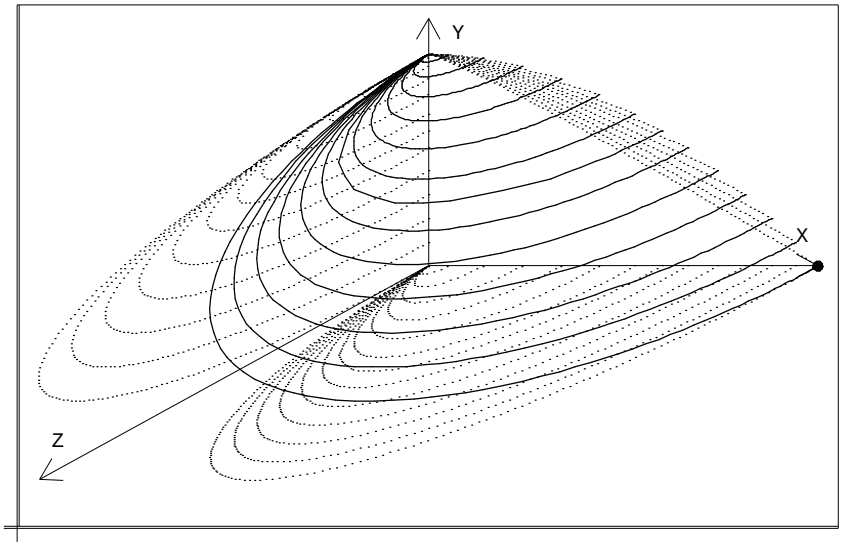

Figure 5.1: Projections into (x, y, z)-space of the family of heteroclinic orbits.

```
 BR   PT TY LAB    PAR(3)        L2-NORM      ...     PAR(1)
  1    1 EP   1  5.000000E-01  4.059140E-01   ...  1.638750E-01
  1    5      2  4.377582E-01  3.618591E-01   ...  1.280061E-01
  1   10      3  3.611938E-01  3.060609E-01   ...  8.742184E-02
  1   15      4  2.854734E-01  2.495465E-01   ...  5.285155E-02
  1   20      5  2.104957E-01  1.921718E-01   ...  2.611675E-02
  1   25      6  1.359888E-01  1.337901E-01   ...  8.835810E-03
  1   30 EP   7  6.104020E-02  7.642498E-02   ...  1.167419E-03
```

We restart in the other direction by making the following changes in `autshear.f`

```
IRS = 1     DS = 0.02
```

rerunning, and saving

```
@autoh shear
@apaut shear
```

to get the output

```
 BR   PT TY LAB    PAR(3)        L2-NORM      ...     PAR(1)
  1    5      8  5.624535E-01  4.496259E-01   ...  2.015640E-01
  1   10      9  6.417016E-01  5.032011E-01   ...  2.508191E-01
  1   15     10  7.221099E-01  5.556790E-01   ...  3.014721E-01
  1   20     11  8.037082E-01  6.069428E-01   ...  3.527945E-01
  1   25     12  8.865009E-01  6.568957E-01   ...  4.042239E-01
  1   30 EP  13  9.704713E-01  7.054625E-01   ...  4.553292E-01
```

The results of both computations are presented in Fig. 5.1, from which we see that the orbit shrinks to zero as `PAR(1)`$=\mu \to 0$.

Appendix I
Source code auttest.f

```
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C               AUTTEST.F
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C
C   An Authomcont example program for the general test example
C
C       VERSION  2.0         Last revision 7/95
C-----------------------------------------------------------------------
C
c     ------------------
      subroutine problem
c     ------------------
c
c This sets up various various constants which are problem specific
c and puts them in a common block /PROB/
c It also sets up several run specific constants to be used in INIT
c
c This is the only routine that should need changing from
c one run to another on the same problem even if the
c dimensions of the stable or unstable manifold or the number of
c frozen parameters etc. vary
c
c The constants set are the following
c
c NPROB ...........  problem size (i.e. dimension of phase-space)
c NFREE, ICP(20) ..  number and labels of free parameters
c NFIXED, IFIXED(20) number and labels of fixed conditions
c NPSI, IPSI(20) ..  number and labels of test functions for degenerate
c                      homoclinic orbits (see the function PSI in AUTLIBH.F)
c NSTAB, NUNSTAB ..  dimensions of the stable and unstable manifolds
c NPUSZR ..........  number of user defined functions (defined in PUSZR)
c                      for output
c
c IEQUIB   =0 if equilibrium  is specified explicitly in PEQUIB
c          =1 if equilibrium is to be solved for during continuation
c          =2 if homoclinic orbits to saddle-node are followed and
c               equilibrium is to be solved for during continuation
c               in this case one has to supply initial data of PAR(11+K)
c               (K=1,NPROB) for equilibrium solution
c ITWIST   =0 orientation not computed
c          =1 orientation computed via adjoint variational equation
```

```
c ISTART   =1 if starting data is read from pstpnt.dat when IRS=0; this data
c               must be t,U at each point with t in [0,1]; multi column format
c          =2 if an explicit solution is specified in the subroutine PSTPNT
c          =3 if the "artificial parameter" approach is used for starting
c
c  The other constants are as described in the AUT086 manual
c
c  The bifurcation parameter is ICP(1)
c  and this may be equal to the truncation interval T=PAR(11)
c  Note that for a non-degenerate homoclinic orbits NFREE=NFIXED+2.
c
      IMPLICIT DOUBLE PRECISION(A-H,O-Z)
      COMMON /BLBCN/ NDIM,IPS,IRS,ILP,ICP(20),PAR(20)
      COMMON /BLCDE/ NTST,NCOL,IAD,ISP,ISW,IPLT,NBC,NINT
      COMMON /BLDLS/ DS,DSMIN,DSMAX,IADS
      COMMON /BLLIM/ NMX,NUZR,RL0,RL1,A0,A1
      COMMON /BLMAX/ NPR,MXBF,IID,ITMX,ITNW,NWTN,JAC
      COMMON /PROB/ IPSI(20),IFIXED(20),ITWIST,ISTART,IEQUIB,NFIXED,
     +     NPSI,NFREE,NPROB,NUNSTAB,NSTAB,NPUSZR
      COMMON /ACC/COMPZERO
      COMMON /BLEPS/ EPSL(20),EPSU,EPSS
c
c problem dependent
c
      NPROB = 3
      NUNSTAB = 1
      NSTAB = 2
      IEQUIB = 0
      ITWIST = 0
      ISTART = 2
c
c restart
c
      IRS = 0
c
c step length
c
      DS    = 0.05D0
      DSMIN = 1.0D-04
      DSMAX = 0.1D0
c
c stop
c
      NMX = 10
      NPR = 20
c
c parameter/phase space region
c
      RL0 = -100.0D0
      RL1 = 100.0D0
      A0  = -100.0D0
      A1  = 100.0D0
c
c limit point and bifurcation detection
```

```
c
      ILP = 0
      ISW = 1
      ISP = 0
c
c mesh
c
      NTST = 35
      NCOL = 4
      IAD  = 1
      IADS = 1
c
c
c machine precision
c
      COMPZERO = 1.0e-13
c
c relative tolerances
c
      EPSU = 1.0E-7
      EPSS = 1.0E-7
      DO I=1,20
         EPSL(I) = 1.0E-7
      END DO
c
c maximal iterations
c
      ITNW = 8
      ITMX = 8
c
c step length weigths
c
c      THETAU=1.0d0
c      THETAL(1)=1.0d0
c      THETAL(2)=0.0d0
c
c output
c
c      IPLT = 0
c      IID  = 2
c
c the initial values of all (non-artificial) parameters if IRS=0
c
      IF (IRS.EQ.0) THEN
c
C Parameter   (only PAR(1) up to PAR(9) available for the user)
C
C
C   d/dt x = ax + by - ax^2 + (timu - alpha z) x(2-3x)
C   d/dt y = bx + ay - 1.5 bx^2 - 1.5 axy - (timu - alpha z) 2y
C   d/dt z = cz + mu x + gamma xz + alpha beta (x^2(1-x)-y^2)
C
C   a
         PAR(1)= 0.0D0
```

```
C
C   b
         PAR(2)= 1.0D0
C
C   c
         PAR(3)= -2.0D0
C
C   alpha
         PAR(4)= 0.0D0
C
C   beta
         PAR(5)= 1.0D0
C
C   gamma
         PAR(6)= 0.0D0
C
C   mu
         PAR(7)= 0.0D0
C
C   timu (tilde mu_2)
         PAR(8)= 0.0D0
c
c
c -PAR(11) is reserved for the truncation interval
c
         PAR(11) = 20.0D0
c
c -if IEQUIB=1, PAR(12)--PAR(12+NPROB) are reserved for the equilibrium
c
         IF (IEQUIB.NE.0) THEN
            PAR(12) = 0.0D0
            PAR(13) = 0.0D0
            PAR(14) = 0.0D0
         ENDIF
      ENDIF
c
c ********************
c regular continuation
c ********************
c
      IF(ISTART.NE.3) THEN
C
c free parameters (user defined)
c
         NFREE  = 2
         ICP(1) = 1
         ICP(2) = 8
c
c fixed conditions and test functions
c
         NPUSZR  = 1
c
         NFIXED    = 0
         IFIXED(1) = 0
```

```
c
         NPSI    = 0
 IPSI(1) = 1
 IPSI(2) = 13
c
c ********************************
c starting solutions using homotopy  (only if ITWIST=0)
c ********************************
c
      ELSE
c
         CONTINUE
c
      ENDIF
c
      RETURN
      END
c
c     ------------------------------
      SUBROUTINE PEQUIB(J,XEQUIB,PAR)
c     ------------------------------
c
c  co-ordinates of the equilibrium if IEQUIB=0
c  j=1 => t=-infty,  j=2 => t=+infty
c
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION PAR(20),XEQUIB(20)
c
c the origin
c
        DO I=1,20
           XEQUIB(I) = 0.0D0
END DO
c
      RETURN
      END
c
C     ----------------------
      SUBROUTINE PF(F,U,PAR)
c     ----------------------
c
c the right-hand sides of the O.D.E. du(i)/dt = pf(i)
c
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION U(*),PAR(20),F(*)
c
c test example
c
      F(1)= PAR(1)*U(1) + PAR(2)*U(2) - PAR(1)*U(1)*U(1)
     1    + (PAR(8) - PAR(4)*U(3)) * U(1)*(2.0D0 - 3.0D0*U(1))
      F(2)= PAR(2)*U(1) + PAR(1)*U(2)
     1    - 1.5D0*PAR(2)*U(1)*U(1) - 1.5D0*PAR(1)*U(1)*U(2)
     2    - (PAR(8) - PAR(4)*U(3)) * 2.0D0*U(2)
      F(3)= PAR(3)*U(3) + PAR(7)*U(1) + PAR(6)*U(1)*U(3)
```

```
     1   + PAR(4)*PAR(5)*(U(1)*U(1)*(1.0D0-U(1))-U(2)*U(2))
c
      RETURN
      END
c
c     -------------------------
      SUBROUTINE PDFDU(PA,U,PAR)
c     -------------------------
c
c returns as PA the jacobian of PF at U
c
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION PAR(20),U(20),PA(20,20)
c
      PA(1,1)= PAR(1) - 2.0D0*PAR(1)*U(1)
     1    + (PAR(8)-PAR(4)*U(3)) * (2.0D0-6.0D0*U(1))
      PA(1,2)= PAR(2)
      PA(1,3)= - PAR(4) * U(1)*(2.0D0-3.0D0*U(1))
c
      PA(2,1)= PAR(2) - 3.0D0*PAR(2)*U(1) - 1.5D0*PAR(1)*U(2)
      PA(2,2)= PAR(1) - 1.5D0*PAR(1)*U(1)
     1    - (PAR(8)-PAR(4)*U(3)) * 2.0D0
      PA(2,3)= 2.0D0*PAR(4)*U(2)
c
      PA(3,1)= PAR(7) + PAR(6)*U(3)
     1    + PAR(4)*PAR(5) * U(1)*(2.0D0-3.0D0*U(1))
      PA(3,2)= -2.0D0*PAR(4)*PAR(5) * U(2)
      PA(3,3)= PAR(3) + PAR(6)*U(1)
c
      RETURN
      END
C
c     ---------------------
      FUNCTION PUSZR(I,PAR)
c     ---------------------
c
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION PAR(20)
      COMMON /PROB/ IPSI(20),IFIXED(20),ITWIST,ISTART,IEQUIB,NFIXED,
     +     NPSI,NFREE,NPROB,NUNSTAB,NSTAB,NPUSZR
c
c User-defined functions of parameters (artificial or otherwise)
c for which accurate location and the ability to plot and restart
c are required. Identical to the ususal AUTO86 routine USZR.
c
c ********************
c regular continuation
c ********************
c
c
      IF (ISTART.NE.3) THEN
c
          PUSZR = PAR(1) - 0.25D0
          RETURN
```

```
c
c ********************************
c starting solutions using homotopy
c ********************************
c
      ELSE
c
         CONTINUE
c
      ENDIF
      END
c
c
c     ----------------------
      SUBROUTINE PSTPNT(U,T)
c     ----------------------
c
c Substitute the explicit solution U=U(T)
c The length of the time interval is PAR(11), the solution U(T)
c will be computed symmetrically with respect to T=0.
c
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
      DIMENSION U(*)
c
c test example (a=0,b=1)
c
      U(1) = 1.0D0 - ( (1.0D0-DEXP(T))/(1.0D0+DEXP(T)) )**2
      U(2) = 4.0D0 * DEXP(T) * (1.0D0-DEXP(T)) / (1.0D0+DEXP(T))**3
      U(3) = 0.0D0
c
      RETURN
      END
c
```

References

Bai, F. & Champneys, A. (1994), Numerical detection and continuation of saddle-node homoclinic bifurcations of codimension one and two, Technical report, University of Bath. Mathematics Preprint 94-04.

Champneys, A. & Kuznetsov, Y. (1994), 'Numerical detection and continuation of codimension-two homoclinic bifurcations', *Int. J. Bifurcation & Chaos* **4**, 795–822.

Champneys, A., Kuznetsov, Y. & Sandstede, B. (1995), 'A numerical toolbox for homoclinic bifurcation analysis'. To appear in *Int. J. Bifurcation & Chaos*.

Doedel, E. & Kernévez, J. (1986), AUTO: Software for continuation problems in ordinary differential equations with applications, Technical report, California Institute of Technology. Applied Mathematics.

Doedel, E., Friedman, M. & Monteiro, A. (1993), On locating homoclinic and heteroclinic orbits, Technical report, Cornell Theory Center; Center for Applied Mathematics, Cornell University.

Doedel, E., Keller, H. & Kernévez, J. (1991*a*), 'Numerical analysis and control of bifurcation problems: (I) Bifurcation in finite dimensions', *Int. J. Bifurcation and Chaos* **1**, 493–520.

Doedel, E., Keller, H. & Kernévez, J. (1991*b*), 'Numerical analysis and control of bifurcation problems: (II) Bifurcation in infinite dimensions', *Int. J. Bifurcation and Chaos* **1**, 745–772.

Freire, E., Rodríguez-Luis, A., Gamero, E. & Ponce, E. (1993), 'A case study for homoclinic chaos in

an autonomous electronic circuit: A trip from Takens–Bogdanov to Hopf–Shilnikov', *Physica D* **62**, 230–253.

Khibnik, A., Kuznetsov, Y., Levitin, V. & Nikolaev, E. (1993), 'Continuation techniques and interactive software for bifurcation analysis of ODEs and iterated maps', *Physica D* **62**, 360–371.

Koper, M. (1994), Far-from-equilibrium phenomena in electrochemical systems, PhD thesis, Univeriteit Utrecht, Holland.

Koper, M. (1995), 'Bifurcations of mixed-mode oscillations in a three-variable autonomous Van der Pol-Duffing model with a cross-shaped phase diagram', *Physica D* **80**, 72–94.

Rucklidge, A. & Mathews, P. (1995), 'Analysis of the shearing instability in nonlinear convection and magnetoconvection'. Submitted to *Nonlinearity*.

Sandstede, B. (1995*a*), Constructing dynamical systems possessing homoclinic bifurcation points of codimension two, WIAS Preprint No. 149, Berlin.

Sandstede, B. (1995*b*), Convergence estimates for the numerical approximation of homoclinic solutions, In Preparation.

Sandstede, B. (1995*c*), Numerical computation of homoclinic flip-bifurcations, In Preparation.

Scheffer, M. (1995), 'Personal communication'.