IPP-Report

Max-Planck-Institut
für Plasmaphysik

E. Strumberger and M. Hölzl

USER MANUAL: ITERATIVE COMPUTATION OF 3D IDEAL MHD EQUILIBRIA AND MAGNETIC FIELDS

IPP 5/113
May 2005

TABLE OF CONTENTS

ABSTRACT

This document is a user manual for the iterative computation of three-dimensional ideal MHD equilibria and their magnetic field structures. The computations a carried out with a system of numerical codes consisting of VACFIELD, GOURDON, DESCUR, VMEC/NEMEC, CO-TRANS and MFBE codes. Their use is described in detail. Example calculations are documented for a possible configuration of the W7-X stellarator, and some useful advice is given.

1. INTRODUCTION

The optimized, helical advanced Wendelstein 7-X (W7-X) stellarator [1, 2] is expected to reach volume-averaged β-values up to 5%. Part of the experimental flexibility will be achieved by modifying the rotational transform in the range of $5/6 \leq \iota \leq 5/4$. For the operation of the experiment and the interpretation of the experimental data the preparation of a database is needed in which possible equilibria and their magnetic field structures are stored. A system of numerical codes computes these MHD equilibria and magnetic fields for given geometry and currents of external coils, as well as defined pressure and toroidal plasma current profiles.

An overview of the code system consisting of VACFIELD, GOURDON, DESCUR, VMEC/-NEMEC, COTRANS and MFBE codes is shown in Fig. 1.

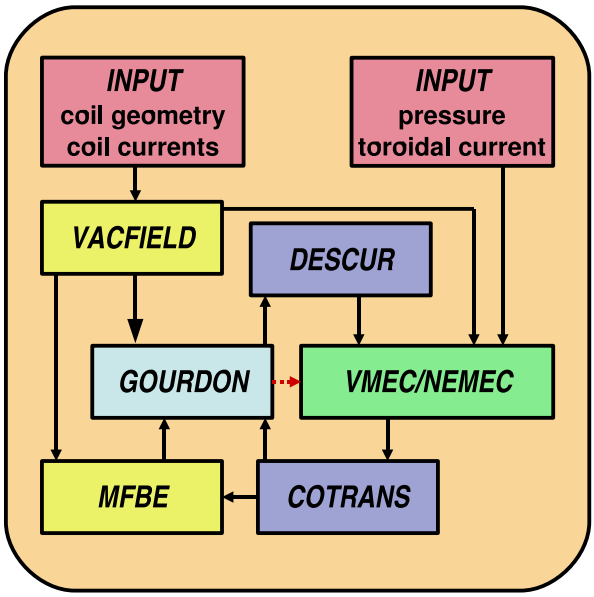


Fig. 1: Overview of the code system

- The VACFIELD code [3] is used to compute the magnetic field of external coils by means of Biot-Savart's law. Geometry and currents of the coils are the fundamental input to this code.
- The GOURDON code [4] traces field lines, and computes the rotational transform, ι.
- The DESCUR code [5] approximates the last closed magnetic surface (lcms) by a set of optimized Fourier coefficients which is used as an initial guess of the plasma boundary in the VMEC/NEMEC code.
- The VMEC/NEMEC code [6, 7] computes three-dimensional, free-boundary, ideal MHD equilibria.

- The COTRANS code [8] transforms the output data of the VMEC/NEMEC code into suitable input formats used by the GOURDON and MFBE codes.
- The MFBE code [9, 10] computes the magnetic field from VMEC/NEMEC equilibrium solutions.

A recipe for the iterative computations and a description of the results are given in chapter 2. Detailed lists of input parameters and output data are specified in the appendices.

This document is thought to be a user manual for these codes. However, it does not describe the underlying physics and numerical methods implemented in the codes. For more detailed information, we refer to the list of publications given at the end of the documentation.

Furthermore, we want to point out that the application of the code system is not restricted to stellarators, but can be used for any two- or three-dimensional toroidal device [10, 8].

2. RECIPE FOR THE COMPUTATIONS

2.0 OVERVIEW

In the free-boundary VMEC/NEMEC code the total toroidal flux, Φ_{total}, serves as a free parameter to adjust the aspect ratio of an ideal MHD equilibrium for a given external field. This parameter is determined iteratively [9, 10, 11] with the code system shown in Fig. 1. Equilibria are computed with the VMEC/NEMEC code for various values of the total toroidal flux. The corresponding magnetic fields inside and outside the plasma boundary are calculated with the MFBE code, and their last closed magnetic surfaces are determined by field line tracing using the GOURDON code. If the plasma boundary of the equilibrium lies completely inside the lcms, the toroidal flux is too small. The flux then has to be increased until the equilibrium plasma boundary agrees with the lcms of the corresponding magnetic field. Otherwise the toroidal flux has to be decreased.

For stellarators without net toroidal current, such as W7-X, the vacuum magnetic field produced by the external conductor system already exhibits closed magnetic flux surfaces. The comparison of the vacuum magnetic field structure (determined with the VACFIELD and GOURDON codes) with the field structure of the corresponding equilibrium (VMEC/NEMEC + MFBE + GOURDON solution for vanishing plasma beta) provides an excellent accuracy test of the numerical calculations.

In the following, a recipe for the iterative computations of equilibria and their magnetic field structures is given. Starting with defined coil currents and a pressure profile (for the computations described in this report a vanishing net toroidal current is assumed) the following steps have to be carried out:

1) VACFIELD code: Compute the vacuum magnetic field.

2) GOURDON code: Create a Poincaré plot of the vacuum magnetic field by tracing field lines. Determine the ι-profile. Produce an input for the DESCUR code containing the co-ordinates of the lcms and the magnetic axis.

3) DESCUR code: Determine the Fourier representation of the lcms as an initial guess for the VMEC/NEMEC code.

4) VMEC/NEMEC code: Assume a value for Φ_{total} and compute the equilibrium for $< \beta >= 0$ ($< \beta >$ is the volume-averaged plasma beta).

5) COTRANS code: Compare the plasma boundary of the equilibrium with the lcms of the vacuum magnetic field. If plasma boundary and lcms do not coincide, change Φ_{total} and compute the equilibrium again. Repeat steps 4 and 5 until both surfaces agree.

6) GOURDON code: Trace field lines of the vacuum magnetic field with starting points on equilibrium flux surfaces. The resulting magnetic surfaces should agree with the equilibrium flux surfaces.

7) COTRANS code: 'Translate' Fourier coefficients of the VMEC/NEMEC output, which describe flux surfaces and magnetic field [10], into the format used by the MFBE code.

8) MFBE code: Compute the magnetic field of the VMEC/NEMEC equilibrium on a grid suitable for the GOURDON code.

9) GOURDON code: Trace field lines by using the same starting points as in step 2 and check if the Poincaré plots agree with the plots of the vacuum magnetic field structure produced in step 2.

10) VMEC/NEMEC code: Compute a finite-β equilibrium for a value of $\Phi_{\text{total}} \leq \Phi_{\text{total}}^{\text{vac}}$ ($\Phi_{\text{total}}^{\text{vac}}$ is the total toroidal flux of the $< \beta >= 0$ equilibrium).

11) COTRANS code: Transform Fourier coefficients as in step 7.

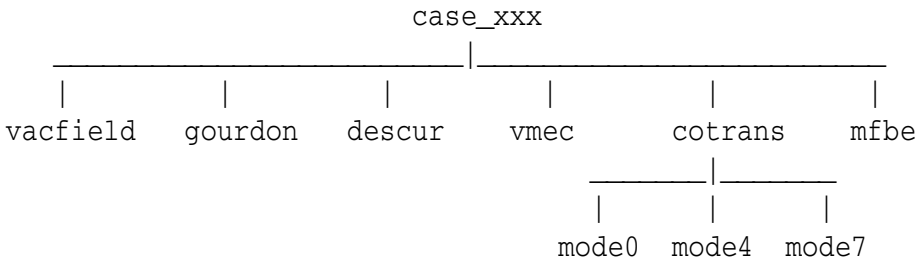
12) MFBE code: Compute the magnetic field of the finite-β equilibrium.

13) COTRANS and GOURDON codes: Determine the lcms of the equilibrium field. Depending on the relative position of the lcms with respect to the equilibrium boundary, increase or decrease Φ_{total}.

14) Repeat steps 10-13 until the correct value of the total toroidal flux is determined.

Details of the computations are described for each step in the following sections.

For the example calculations we implemented the following directory structure on our computer:



All computations belonging to the considered example are stored in the directory *case_xxx*. This directory is cut into several subdirectories, one for every code used. The directory used for COTRANS computations is further divided into three subdirectories with respect to the modes of the code.

All codes work in the same way. They take one or more input files provided by the user and/or previously used codes. The codes produce output files that can be either viewed with the XMGR plot software or used as input data for a succeeding code.

The example calculations presented in this report are computed for one pressure profile and one set of coil currents, namely, a high-iota case named *case_f*54. This configuration is characterized by a vanishing net toroidal current and $\iota = 5/4$ in the edge region.

2.1 VACUUM MAGNETIC FIELD STRUCTURE

2.1.1 COMPUTATION OF THE VACUUM MAGNETIC FIELD (STEP 1)

The VACFIELD code computes the magnetic field produced by external coils. The magnetic field used by the GOURDON and MFBE codes has to be given on a high resolution grid, while for VMEC/NEMEC calculations a different output format and less toroidal grid points are needed. Therefore, the compuations have to be done twice. Examples of standard input files, named **in_vac_xxx**, are given in Appendix A1.

2.1.2 FIELD LINE TRACING (STEP 2)

The GOURDON code needs three input files:

- the standard input file called **line_input_xxx**,
- the file called **line_points_xxx** where the starting points of the field lines are specified, and
- the file containing the magnetic field (output of the VACFIELD code (**vacfield_mfbe_xxx**) or the MFBE code (**field_mfbe_xxx**)).

Field lines traced with the GOURDON code are used in three different ways:

- Production of Poincaré plots which show the magnetic field structure.
- Computation of the ι-profile.
- Production of an input to the DESCUR code containing the co-ordinates of the magnetic axis and the lcms.

For an illustration of the magnetic field structure the following field lines are of special interest (see Figs 2a-d):

- magnetic axis (black dots in Figs 2a-d)
- extended islands inside the lcms (magenta dots in Figs 2a-d)
- lcms (red dots in Figs 2a-d)
- boundary of the extended islands outside the lcms (green dots in Figs 2a-d)
- axis of the extended islands (green dots in Figs 2a-d)
- first closed magnetic surface beyond the extended islands (not present in this example)

- last closed magnetic surface beyond the extended islands (not present in this example)
- ergodic layers (blue dots in Figs 2a-d)

Between those surfaces an appropriate number of additional (equidistant) surfaces should be plotted as well. For viewing the plots it is helpful to use XMGR parameter files to avoid adjusting all settings over and over again. In Figs 2a-d the vacuum magnetic field structure is shown at four different cross-sections of one period.

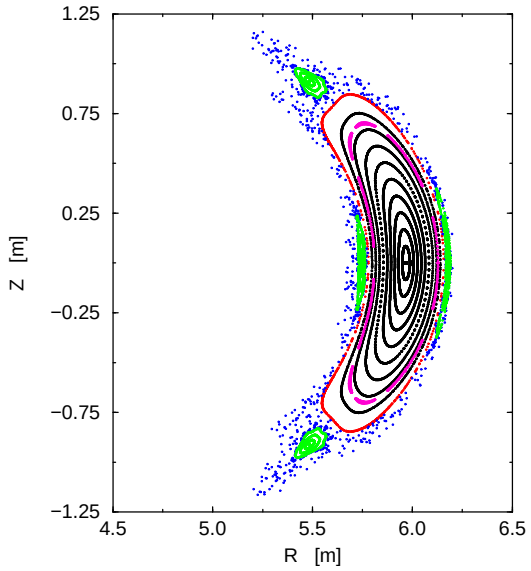


Fig. 2a: Poincaré plot for $\varphi = 0^o$.

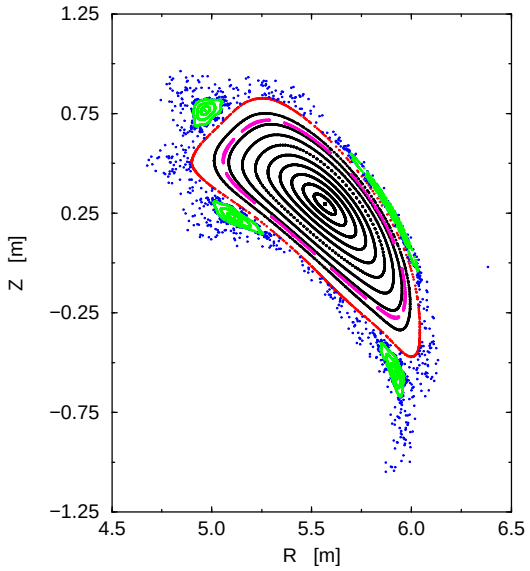


Fig. 2b: Poincaré plot for $\varphi = 18^o$.

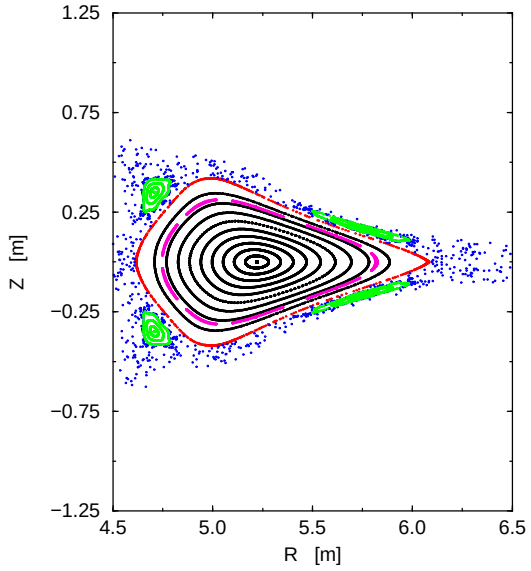


Fig. 2c: Poincaré plot for $\varphi = 36^o$.

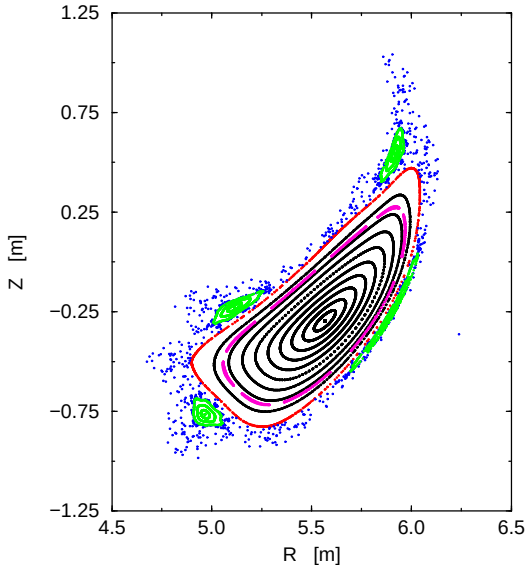


Fig. 2d: Poincaré plot for $\varphi = 54^o$.

There is no easy way to find those special surfaces. One has to search for them iteratively checking the results with XMGR plots in each step. For getting a first overview it is useful to trace many field lines covering the whole area of interest. For this purpose, the GOURDON code offers the possibility to specify only the first and the last starting point as shown in one of the example files given in Appendix A2.

Note, if there are extended islands inside the lcms, the VMEC/NEMEC code may have convergence problems. Make sure that all relevant islands of the vacuum magnetic field have been found.

The GOURDON code is able to compute the rotational transform, ι, for closed field lines. Figure 3 shows the rotational transform profile as function of the major radius, R (Z=0, φ=0), from the magnetic axis to the 5/4 island remnants (but without ergodic field lines).

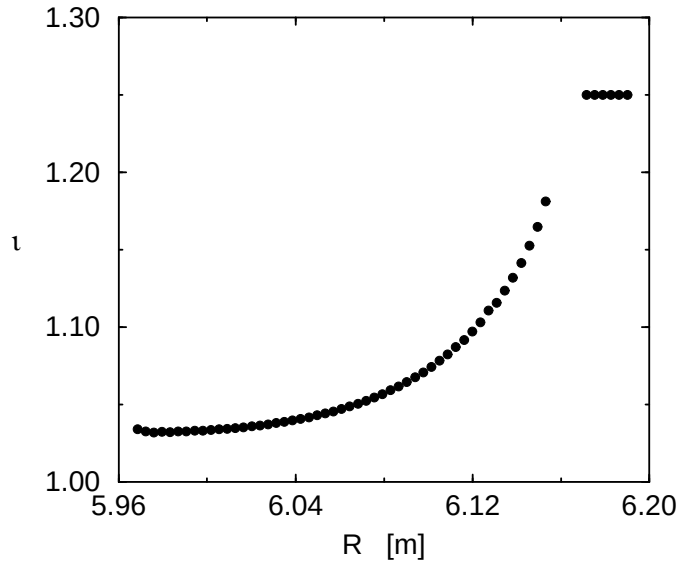


Fig. 3: Iota profile

For the production of the DESCUR input only two field lines have to be traced. One started on the magnetic axis and the other one started close to but inside the lcms.

All input parameters and example input files needed for the computations discussed above are described in detail in Appendix A2.

2.1.3 FOURIER APPROXIMATION OF THE LCMS (STEP 3)

The DESCUR code determines the Fourier representation of a specified closed magnetic surface. For this purpose, it needs the co-ordinates of the surface and of the magnetic axis. These input data are provided by the GOURDON code (output file: **data_desc_xxx**). In addition, the DESCUR code needs a standard input file, named **input_descur**, which provides the maximum poloidal (*mtor*) and toroidal (*ntor*) mode numbers of the Fourier spectrum. For most cases

$mpol = 9$, $ntor = 11$ works fine. But for the case discussed here, the number of poloidal modes, $mpol$, has to be at least 11 to obtain good results.

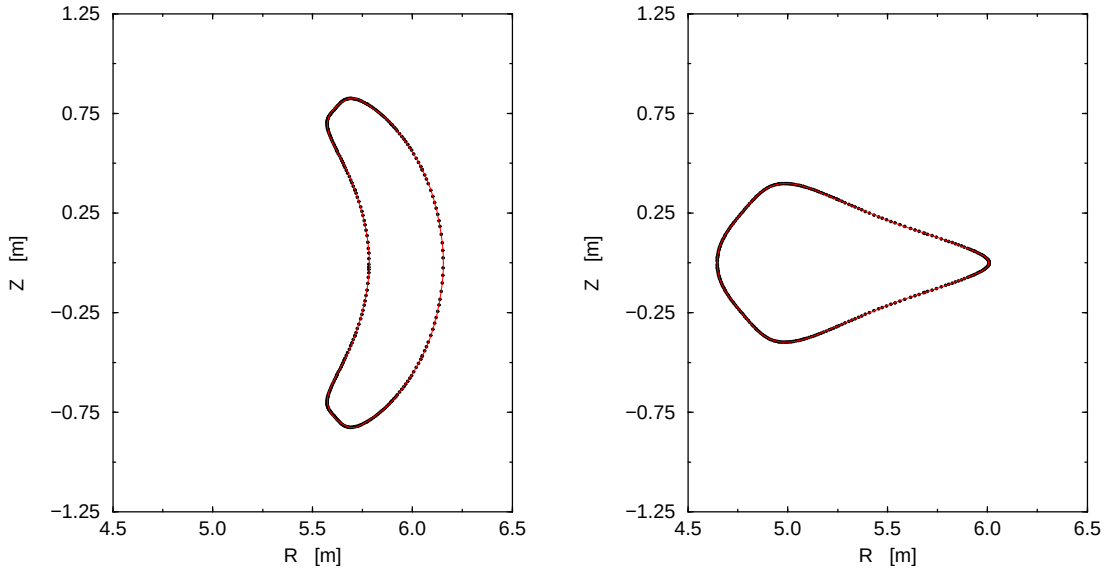


Fig. 4: Specified surface (black dots) and its approximation by the DESCUR code (red line) plotted at the cross-sections $\varphi = 0^o$ (left figure) and $\varphi = 36^o$ (right figure).

Figure 4 shows a comparison of the given surface and its approximation by the DESCUR code at two toroidal cross-sections. The DESCUR code provides these data for all toroidal cross-sections at which the surface is specified. The output files are named **plotout_000_xxx** ... **plotout_nnn_xxx** with *nnn* indicating the number of the toroidal cross-section. As a rule, some of them should be plotted with the XMGR plot software to make sure that the accuracy of the approximation is sufficient. We also recommend to use the enlargement possibilities of this software.

If the DESCUR code stops with an error message, the surface chosen in step 2 is not adequate (e.g. the co-ordinate points are not uniformly distributed in poloidal direction). Then, this calculation of step 2 has to be repeated with another starting point of the surface.

Note, the error tolerance, $ftol$, also plays a role here (see Appendix A3). If it is chosen too small, the DESCUR code might find no converged solution.

Summarized: Try to find a surface close to the lcms for which the DESCUR run converges with $mpol$ as big as necessary and $ftol$ as small as possible.

For a detailed description of the input and output files of the DESCUR code see Appendix A3.

2.2 ITERATIVE COMPUTATION OF AN EQUILIBRIUM WITH $<\beta>=0$

2.2.1 EQUILIBRIUM COMPUTATION (STEPS 4-6)

The total toroidal flux, Φ_{total}, of the equilibrium with vanishing plasma beta (no plasma pressure) has to be determined. The computations are performed in the following three steps.

STEP 4

At the beginning, a value of Φ_{total} is assumed, and the free-boundary equilibrium is computed with the VMEC/NEMEC code. For W7-X equilibria $\Phi_{\text{total}} \approx 2$ Wb is a reasonable initial guess.

The VMEC/NEMEC code needs several input files. The standard input file, named **input.xxx**, is a namelist file which contains paths and names of further input files. Those input files provide the vacuum magnetic field (output file **vacfield_vmec_xxx** of the VACFIELD code) and initial guesses of the Fourier coefficients of the magnetic axis and the plasma boundary. Doing these computations for the first time, a simple guess for the Fourier coefficients of the magnetic axis has to be made, while the Fourier coefficients of the plasma boundary are provided by the DESCUR code (output file **out_vmec_xxx**).

STEP 5

The COTRANS code provides an interface between various codes. Due to its various tasks the code is written in modular form. Each mode of the code has its special input and output files. The features relevant for step 5 are the following:

- It generates diverse XMGR plot data files, named **xmgr_..._xxx**, from the VMEC/NEMEC solution (output file **wout.xxx**).
- It produces an input to the GOURDON code, called **out_gourdon_xxx**, which contains starting points on flux surfaces of the VMEC/NEMEC equilibrium.
- It extracts the Fourier coefficients of the magnetic axis and the plasma boundary (or any other flux surface) and stores them in the output files **out_vmec_axis_xxx** and **out_vmec_lcms_xxx**. These coefficients can be used as initial guesses for the next run of the VMEC/NEMEC code.

These computations are performed by choosing mode 0 (VMEC/NEMEC $\Longrightarrow$ GOURDON) of the COTRANS code. The appropriate standard input file **input_mode0** is described in Appendix A5.

STEP 6

In order to determine Φ_{total}, the position of the equilibrium plasma boundary has to be compared with the position of the lcms of the vacuum field (identified in step 2). If the plasma boundary is located outside the lcms, Φ_{total} is too large and has to be reduced, or vice versa. Then, steps 4 and 5 have to be repeated until the correct value of the total toroidal flux is determined. As examples see Figs 5a-c, which show equilibrium flux surfaces (black solid lines) up to the plasma boundary for three different values of Φ_{total}, and the lcms of the vacuum magnetic field (red dots).

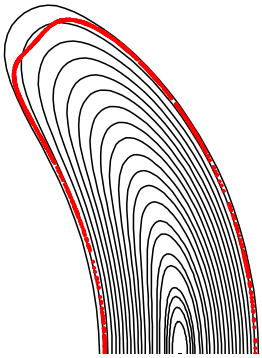


Fig. 5a: Φ_{total} too big.

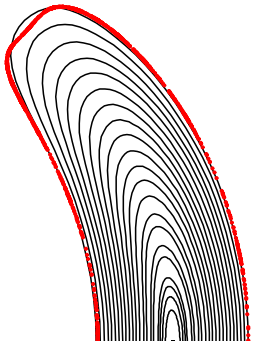


Fig. 5b: Φ_{total} correct.

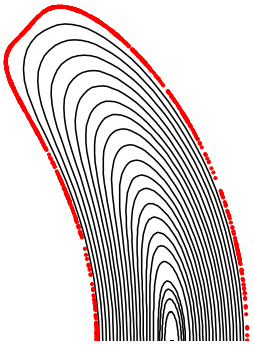


Fig. 5c: Φ_{total} too small.

Fig. 5a shows an example where the toroidal flux is too big (Φ_{total}=1.65 Wb), while in Fig. 5c the toroidal flux is too small (Φ_{total}=1.35 Wb). In Fig. 5b the plasma boundary and lcms approximately agree, that is, the toroidal flux (Φ_{total}=1.50 Wb) has the right value.

Knowing the right value of the toroidal flux, a further consistency test of the equilibrium solution is possible. Field lines with starting points on equilibrium flux surfaces (these points are given in the output file **out_gourdon_xxx** of the COTRANS code) are traced with the GOURDON code using the vacuum magnetic field provided by the VACFIELD code. The resulting magnetic surfaces have to coincide with the equilibrium flux surfaces stored in the plot files **xmgr_fluxsur_xxx** of the COTRANS code. If they do not match, the VMEC/NEMEC input parameters may have been chosen in an inadequate way. It is also possible that the considered vacuum magnetic field encloses extended islands inside the lcms. **Note**, the VMEC/NEMEC code does not take into account such islands (it assumes nested flux surfaces), and, therefore, may yield poor results. Also the shape of the lcms may play a role. If it is too complex, the VMEC/NEMEC code is not able to reproduce it in every detail. A further reason may be a bad fit of the lcms by the DESCUR code.

The results obtained for the considered configuration, presented in Fig 6, show a satisfactory, but not perfect agreement of the surfaces. The small 10/9 islands in the vacuum magnetic field are one reason for this result.

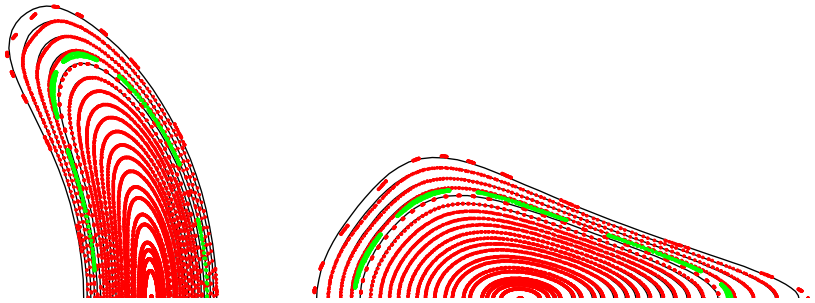


Fig. 6: Equilibrium flux surfaces (black solid lines) and vacuum magnetic flux surfaces (red dots) for the upper halves of two toroidal cross-sections, $\varphi = 0^o$ (left figure) and $\varphi = 36^o$ (right figure). The green dots represent small 10/9 islands in the vacuum magnetic field.

2.2.2 TRANSFORMATION OF THE EQUILIBRIUM OUTPUT (STEP 7)

In step 7 the Fourier coefficients of the VMEC/NEMEC output are transformed into a readable format for the MFBE code by using mode 7 of the COTRANS code. The COTRANS code needs two input files, namely the standard input file, called **input_mode7** (for details see Appendix A5), and the output file **wout.xxx** of the VMEC/NEMEC code. In this mode, the COTRANS code yields two output files. Those are the standard output file **output_xxx**, and the file containing the input to the MFBE code (named **out_four_xxx**). The parallelized version of the code (16 processors are a good choice) is recommended for these computations.

2.2.3 COMPUTATION OF THE MAGNETIC FIELD (STEP 8)

The MFBE code computes the magnetic field of a given VMEC/NEMEC equilibrium inside and outside the plasma boundary. It yields the magnetic field in cylindrical co-ordinates on a grid. Analogue to the VACFIELD output, this output, stored in file **field_mfbe_xxx**, serves as input to the GOURDON code.

The MFBE code needs three input files, namely, the standard input file **input_mfbe_xxx** (for a detailed description see Appendix A6), the Fourier representation of the equilibrium (output file **out_four_xxx** of the COTRANS code (mode 7)), and the vacuum magnetic field (output file **vacfield_mfbe_xxx** of the VACFIELD code).

The MFBE computations are time consuming. For instants, the calculations described in this report take typically ≈ 4 hours per run using 16 processors.

2.2.4 COMPARISON OF THE EQUILIBRIUM SOLUTION WITH THE VACUUM MAGNETIC FIELD STRUCTURE (STEP 9)

In this step, the vacuum magnetic field is reproduced out of the vacuum equilibrium. It should agree with the original vacuum field computed in step 1 with good accuracy. Testing the quality

of these results, field lines are traced for the same starting points using once the vacuum magnetic field and once the magnetic field of the equilibrium solution (output of the MFBE code). The results are overlayed in one Poincaré plot. The magnetic field structures are shown in Figs 7a-b for the vacuum field (black dots) and the MFBE field (red dots) at two cross-sections. Both plots show a satisfactory agreement.

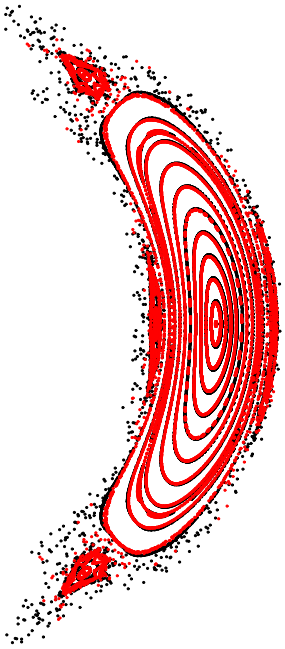


Fig. 7a: Cross-section at $\varphi = 0^o$.

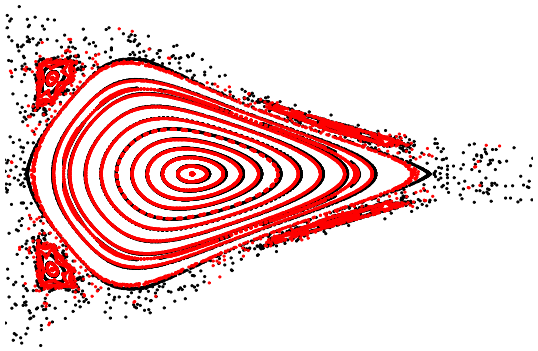


Fig. 7b: Cross-section at $\varphi = 36^o$.

2.3 ITERATIVE COMPUTAION OF FINITE-β EQUILIBRIA (STEPS 10-14)

In the last four steps, finite-β equilibria are computed. The used pressure profile is shown in Fig. 8.

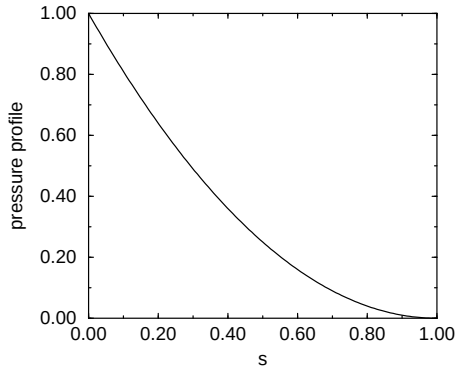


Fig. 8: Pressure profile as function of the normalized toroidal flux, *s*.

We start out with a low β-value (e.g. 1%) which is increased step by step. For each β-value the toroidal flux has to be determined iteratively.

STEP 10

First, the β-value has to be adjusted. The shape of the pressure profile is defined in the VMEC/-NEMEC input file **input.xxx**, while its absolute values are scaled with the parameter *betascale* (also defined in the input file). The resulting β-value is calculated during the VMEC/NEMEC run. The pressure has to be chosen in a way the desired β-value is obtained. For adjusting the β-value (by varying the parameter *betascale*), it is sufficient to run the VMEC/NEMEC code with low accuracy and a small number of radial surfaces (e.g. $ftol = 1 \cdot 10^{-6}$, $ns = 21$).

Now that the correct β-value is determined, the finite-β equilibrium is computed with high accuracy. The initial guess of the total toroidal flux should be equal or smaller than the value obtained for an equilibrium with smaller plasma beta.

STEP 11

Analogue to step 7, the VMEC/NEMEC equilibrium output is transformed into a MFBE compatible format using the COTRANS code (mode 7).

STEP 12

The magnetic field of the VMEC/NEMEC equilibrium is computed with the MFBE code.

STEP 13

The lcms is determined by field line tracing. Depending on the relative position of the lcms with respect to the plasma boundary of the finite-β equilibrium, the total toroidal flux has to be increased or decreased.

STEP 14

Steps 10-13 have to be repeated until lcms and plasma boundary agree. If no agreement can be found, an equilibrium with the specified β-value most probably does not exist.

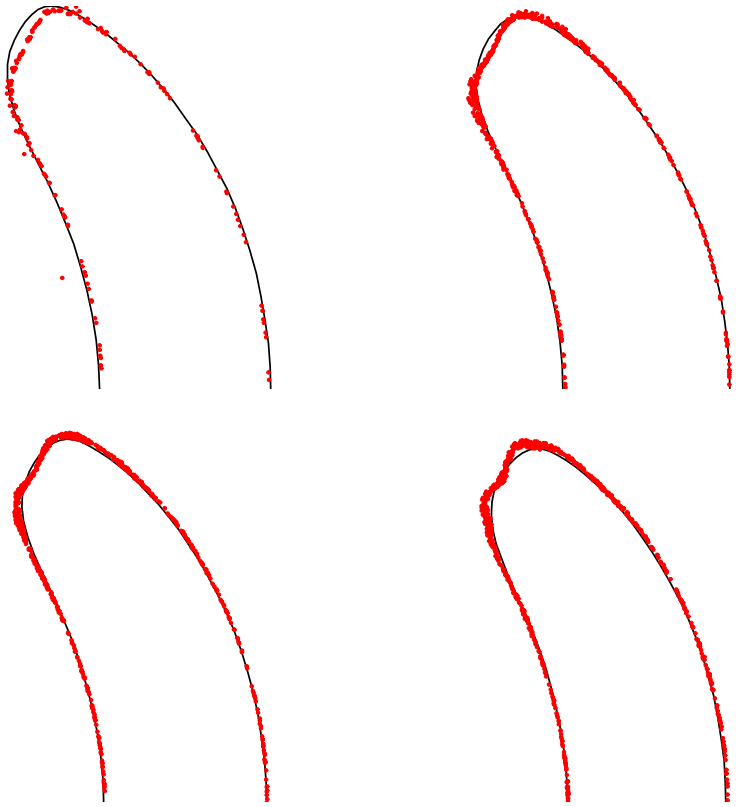


Fig. 9: Upper halves of the bean-shaped cross-section ($\varphi = 0^o$) showing the equilibrium plasma boundary (black solid line) and the lcms (red dots) of the corresponding magnetic field for four values of the total toroidal flux: $\Phi_{\text{total}} = 1.50$ Wb (upper left plot), $\Phi_{\text{total}} = 1.425$ Wb (upper right plot), $\Phi_{\text{total}} = 1.35$ Wb (lower left plot), and $\Phi_{\text{total}} = 1.275$ Wb (lower right plot).

In Fig. 9 the results for $< \beta >= 1\%$ and four toroidal flux values are shown. While in Fig. 9a the total toroidal flux is too big ($\Phi_{\text{total}} = 1.50$ Wb), it fits in Fig. 9b ($\Phi_{\text{total}} = 1.425$ Wb) and Fig. 9c ($\Phi_{\text{total}} = 1.35$ Wb), and is too small in Fig. 9d ($\Phi_{\text{total}} = 1.275$ Wb). In Fig. 9b and 9c lcms and plasma boundary agree with satisfying accuracy. Studying also other cross-sections and enlargements of the plots, we decided to choose $\Phi_{\text{total}} = 1.425$ Wb as correct value. If a more accurate fit is desired one would have to compute further equilibria with values in between (1.425 Wb $< \Phi_{\text{total}} < 1.35$ Wb).

Having determined the total toroidal flux for one β-value, steps 10-13 are repeated for another one.

Figure 10 shows the results obtained for higher β-values. For $< \beta >= 4\%$ no agreement could be found by further reducing Φ_{total}. There, the region around the 10/9 islands is already ergodized, which leads to a strong reduction of the plasma volume. Most likely, for this configuration no equilibrium exists for $< \beta > \geq 4\%$.

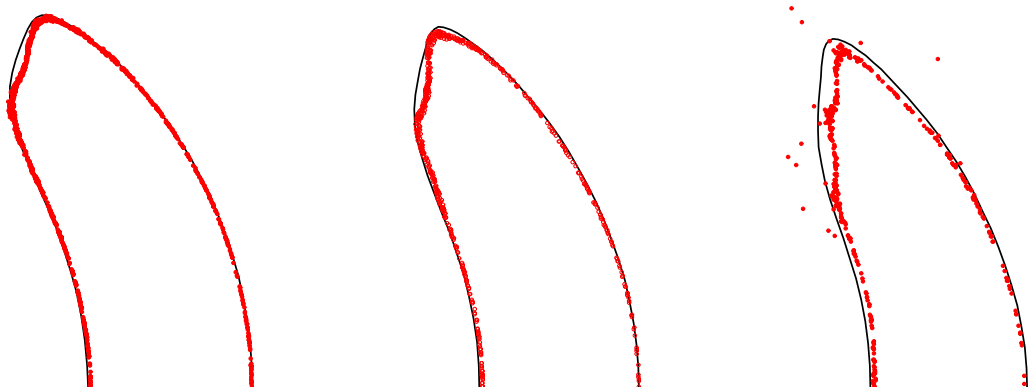


Fig. 10: Upper halves of the bean-shaped cross-sections ($\varphi = 0^{o}$) showing the equilibrium plasma boundary (black solid line) and the lcms (red dots) of the magnetic field for three cases: $< \beta >= 2\%$, $\Phi_{\text{total}} = 1.35$ Wb (left), $< \beta >= 3\%$, $\Phi_{\text{total}} = 1.22$ Wb (middle), and $< \beta >= 4\%$, $\Phi_{\text{total}} = 1.10$ Wb (right).

2.4 SUMMARY OF THE RESULTS

Finally, the results can be summarized as follows.

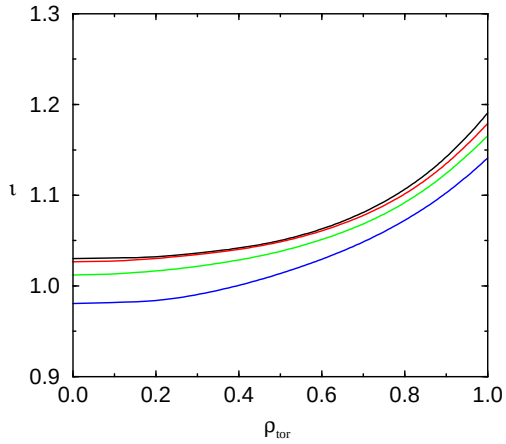


Fig. 11a: rotational transform

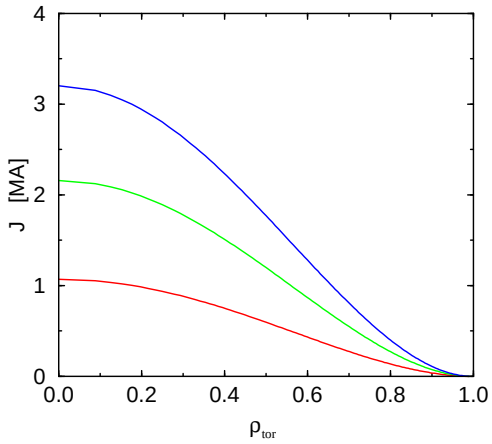


Fig. 11b: poloidal plasma current

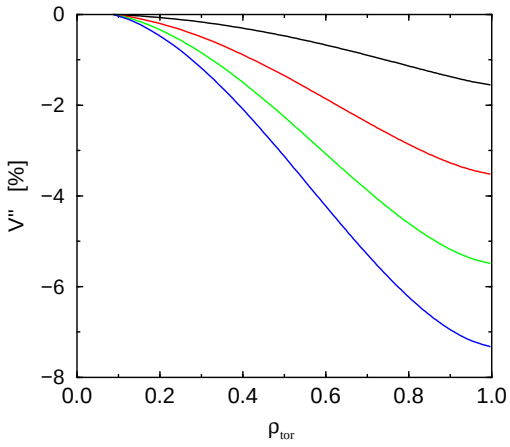


Fig. 11c: magnetic well

Figures 11a-c show the rotational transform, ι, the poloidal plasma current, J, and the magnetic well, $V'' = (V'_{\rm lcms} - V'_0)/V'_0$ ($V'_{\rm lcms}$ = specific volume on the lcms, V'_0 = specific volume on the magnetic axis) as function of $\rho_{\rm tor}$. The colours identify the β-values: $<\beta>= 0\%$ (black), $<\beta>= 1\%$ (red), $<\beta>= 2\%$ (green), and $<\beta>= 3\%$ (blue). As expected, the rotational transform decreases with increasing plasma beta, while poloidal plasma current and magnetic well increase. For $<\beta>= 3\%$, the rational rotational transform value, $\iota = 5/5$, appears inside the plasma, which might cause island formation.

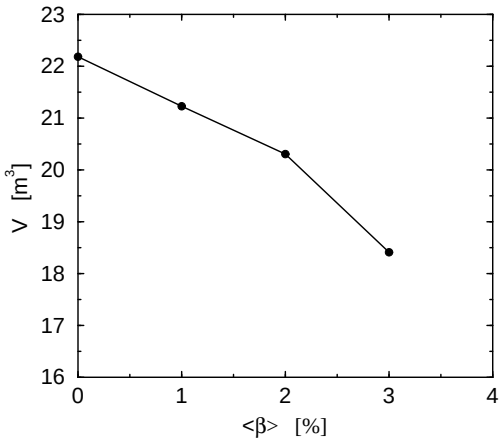


Fig. 12a: plasma volume

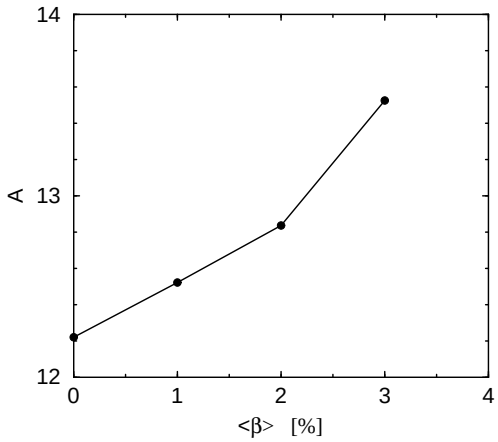


Fig. 12b: aspect ratio

Plasma volume, V, and aspect ratio, $A = R_0/a$ (R_0 = major radius, a = minor radius), as function of the volume-averaged plasma beta, $< \beta >$, are plotted in Figs 12a-b.

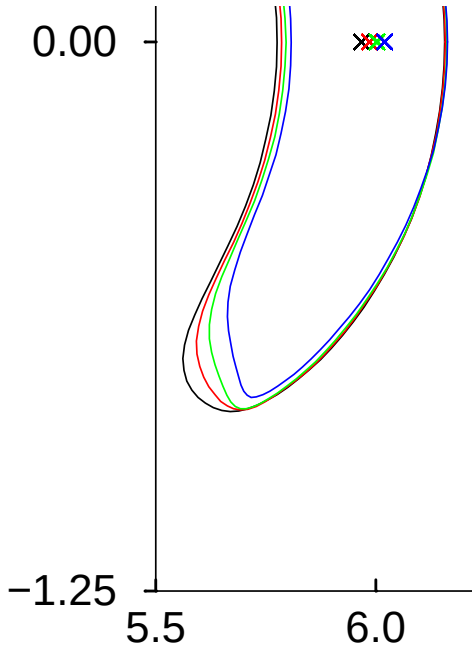


Fig. 13: Enlargement of the lower half of the bean-shaped cross-section ($\varphi = 0^o$). Plasma boundaries (solid lines) and magnetic axes (crosses) are plotted for various β-values: $< \beta >= 0$ (black), $< \beta >= 1\%$ (red), $< \beta >= 2\%$ (green), and $< \beta >= 3\%$ (blue).

In Fig. 13 the contraction of the plasma boundary and the Grad-Shafranov shift of the magnetic axis are illustrated for increasing β-values.

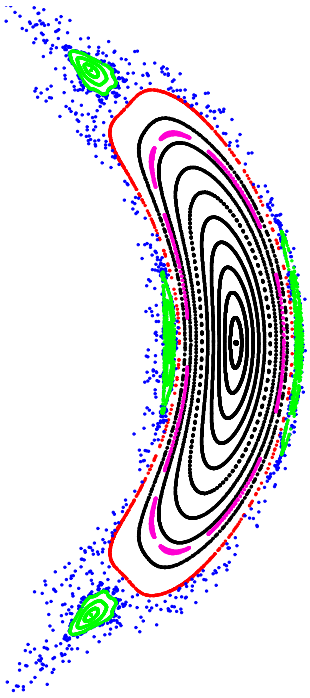


Fig. 14a: $< \beta >= 0$

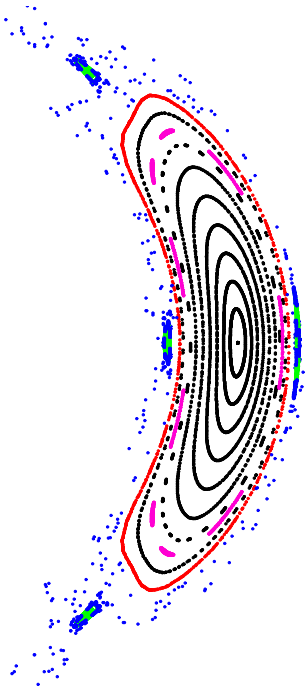


Fig. 14b: $< \beta >= 1\%$

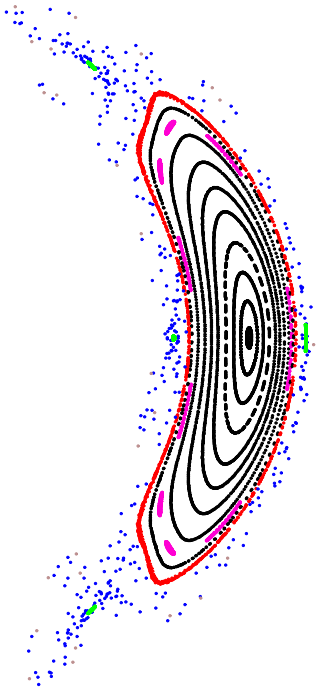


Fig. 14c: $< \beta >= 2\%$

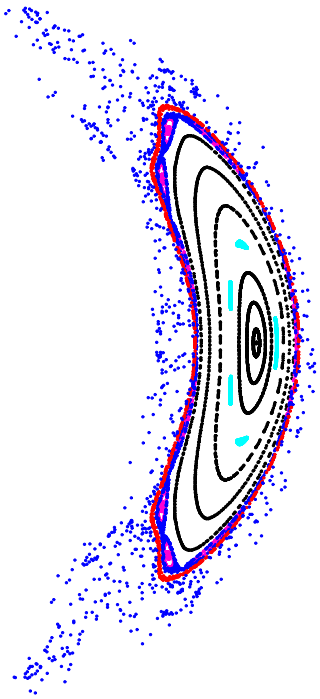


Fig. 14d: $< \beta >= 3\%$

Figures 14a-d show the magnetic field structures for $< \beta > = 0$, 1, 2 and 3% at the bean-shaped cross-section. The rotational transform already drops below $\iota = 5/5$ (cyan dots) for $< \beta >= 3\%$. In addition, the 10/9 islands (magenta dots) located close to the plasma boundary (red dots) extend with increasing plasma beta. For $< \beta >= 4\%$ those islands lead to a further ergodization and a rigorous reduction of the plasma volume. No consistent equilibrium solution could be found for $< \beta >= 4\%$. Furthermore, the 5/4 island remnants (green dots) shrink with increasing β-value. For $< \beta >= 3\%$ these islands are fully ergodized.

Finally, Mercier and resistive interchange stability are investigated with the COTRANS code (mode 4, for details see Appendix A5). Figures 15a-b show the results for various β-values ($< \beta >= 1\%$ (red), $< \beta >= 2\%$ (green), $< \beta >= 3\%$ (blue)). The equilibrium with $< \beta >= 1\%$ is stable with respect to these criteria. For higher β-values formal instability prevails around the 5/5 ($< \beta >= 3\%$) and 10/9 ($< \beta > \geq 2\%$) resonances.

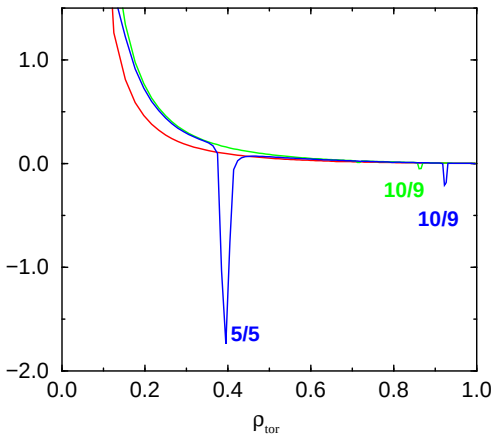


Fig. 15a: Mercier stability

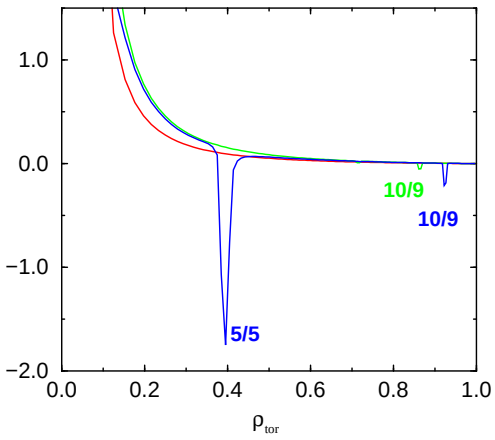


Fig. 15b: resisitive interchange stability

A complex magnetic field structure and island formation deteriorate the numerical accuracy of the considered W7-X configuration. There exist other configurations (e.g. see Ref. [11, 10]) where much better agreement between vacuum magnetic field structure and equilibrium solution ($< \beta >= 0$) can be achieved. Equilibria up to $< \beta >= 5\%$ can be found by varying coil currents and pressure profile.

References

[1] Grieger, G. et al., in Plasma Physics and Controlled Nuclear Fusion Research 1990 (Proc. 13th Int. Conf. Washington, DC, 1990) Vol. **3**, IAEA, Vienna (1991) 525.

[2] Lotz, W., Nührenberg, J., Schwab, C., in Plasma Physics and Controlled Nuclear Fusion Research (Proc. 13th Int. Conf. Washington 1990) Vol. **2**, IAEA, Vienna (1991) 603.

[3] Strumberger, E. and Schwarz E., Laboratory Report 5/112, Garching 2005. http://www.ipp.mpg.de/netreports/ipp-report_5_112.ps.

[4] Gourdon, C., "Programme Optimise de Calculs Numeriques Dans le Configurations Magnetique Toroidales", CEN Fontenay aux Roses, 1970.

[5] Hirshman, S.P. and Meier, H.K., Phys. Fluids **28** (1985) 1387.

[6] Hirshman, S.P. and Lee, D.K., Comput. Phys. Commun. **39** (1986) 161.

[7] Hirshman, S.P., van Rij, W.I., Merkel, P., Comput. Phys. Commun. **43** (1986) 143.

[8] Strumberger, E., et al., Nuclear Fusion **44** (2004) 464.

[9] Strumberger, E., Nuclear Fusion **37** (1997) 19.

[10] Strumberger, E., Merkel P. Schwarz E. and Tichmann C., Laboratory Report IPP 5/100, Garching 2002. http://www.ipp.mpg.de/netreports/ipp-report_5_100.ps.

[11] Strumberger, E., Laboratory Report IPP 2/339 Garching 1997.

[12] Nührenberg, J., Zille, R., in Theory of Fusion Plasmas (Proc. Joint Varenna-Lausanne Int. Workshop Varenna 1987), Editrice Compositori, Bologna (1988) 3.

APPENDICES

A1: VACFIELD CODE

The VACFIELD code uses a standard input file. Therein paths and names of further input files are specified which provide geometrical data of the coils.

standard input file

```
output:      'out_vac_mfbe'
out_mfbe:    'vacfield_mfbe'
type_mfbe:   'binary'
out_vmec:    'none'
type_vmec:   'binary'
symmetry:    'stellarator symmetry'
derivative:  'no'
grid:         nf    nr    nz    np
              254   128   128    5
              centre (r0,z0)  half width (dr,dz)
               r0     z0     dr     dz [m]
              5.500  0.000  1.500   1.500
main_coils:  'modular coils'
main_input:  'coils/mod_hs5v10u'
              ncoil_main
                10
current:      j_main    [MA]
               1.60
fm(i):        (i=1,...,ncoil_main)
               1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00 1.00
corr_coils:  'none'
corr_input:  'none'
              ncoil_corr
                 2
              type        current   [MA]
              SAD1         0.0
              SAD2         0.0
add_coils:   'auxilliary coils'
add_input:   'coils/aux_hs5v10u'
              ncoil_pert
                4
fa(i):        (i=1,...,ncoil_pert)   [MA]
              -0.416 -0.416 -0.416 -0.416
```

The current *j_main* is the nominal coil current for each of the modular coils, while $fm(i)$ is the relative variation of the current of coil number *i*. For example, *j_main*=1.60 MA and $fm(1)$=1.00 result in a total current of $1.60\ \text{MA} \cdot 1.00 = 1.60$ MA flowing through modular coil 1.

Note, the currents for the correction and additional coils are not given with respect to *j_main*, but in absolute values (e.g. -0.416 MA).

Being used as input to the MFBE and GOURDON codes, the magnetic field has to be calculated on a high resolution grid. Here we use 254 toroidal grid points per period, and 128 points in *R*- and *Z*-direction.

For VMEC/NEMEC calculations a different output format and less toroidal points are required. Only the first few lines of the input file have to be changed:

```
output:       'out_vac_vmec'
out_mfbe:     'none'
type_mfbe:    'binary'
out_vmec:     'vacfield_vmec_44'
type_vmec:    'binary'
symmetry:     'stellarator symmetry'
derivative:   'yes'
grid:          nf    nr    nz    np
               44   128   128    5
```

Note, nf corresponds to *nzeta* in the VMEC/NEMEC input file. It has to be four times of the toroidal mode number, *ntor* (VMEC/NEMEC input). For most cases nf=44 should be a good choice.

The computational time of the VACFIELD code strongly depends on the total number of grid points ($nf \cdot nr \cdot nz$). However, using 16 processors the computation is performed interactively within a computational time less than 15 minutes.

A detailed description of the VACFIELD code, its use, and its input and output quantities are documented in Ref. [3].

A2: GOURDON CODE

The GOURDON code is able to trace field lines and guiding centre orbits. Here input and output files are only described for field line tracing. The GOURDON uses SI units, that is, all input and output quantities are given in these units. It needs a standard input file and further input files specified therein. An example of a standard input file is shown below.

standard input file

```
modus:            'field line'
out_num:          'struc'
standart_output:  'out_line'
field_input:      '../vacfield/vacfield_mfbe'
index_input:      'none'
equipment_input:  'none'
line_input:       'line_points_struc'
diffusion:        'none'
line_plasma:      'none'
line_wall:        'none'
line_plate:       'none'
line_xmgr:        'xmgr_line'
line_iota:        'none'
line_tracing:      hpasz    npper
                   0.0100     4
line_descur:      'none'
                   ntheta
                    360
computing_time:    timema   trest
                    300.     20.
```

List of input quantities:

modus	character∗80	mode of the code
	field line	field line tracing
out_num	character∗8	suffix added to all output files
standard_output	character∗80	name of the standard output file
field_input	character∗80	name of the input file containing the magnetic field
line_input	character∗80	name of the input file containing the initial starting points of the field lines
diffusion	character∗80	not implemented up to now
line_plasma	character∗80	not implemented up to now
line_wall	character∗80	not implemented up to now
line_plate	character∗80	not implemented up to now
line_xmgr	character∗80	XMGR plot file containing the co-ordinates of the traced field lines
	none	no output
	xxx	path and name of the output file
line_iota	character∗80	computation of the rotational transform
	none	no computation

	xxx	path and name of the output file
hpasz	real	integration step size
npper	integer	number of toroidal cross-sections per period for which field line data are stored
line_descur	character∗80	production of DESCUR input data
	none	no output
	xxx	path and name of the output file
ntheta	integer	number of poloidal co-ordinate points
timema	real	maximum computation time
trest	real	program stops when the remaining computational time is smaller than *trest*

The starting points of the field lines are provided by the input file **line_points_xxx**. One may either specify individual starting points, or define a range of co-ordinates in between field lines are started. Below, an example is given for tracing individual field lines.

input **line_points_xxx** *(individual starting points)*

```
      nparam
        19
nr         R              Z          phi  ntour
1    0.596850E+01   0.000000E+00    0.0   20    magnetic axis
2    0.594953E+01   0.000000E+00    0.0  200
3    0.593056E+01   0.000000E+00    0.0  200
4    0.591159E+01   0.000000E+00    0.0  200
5    0.589262E+01   0.000000E+00    0.0  200
6    0.587365E+01   0.000000E+00    0.0  200
7    0.585468E+01   0.000000E+00    0.0  200
8    0.583571E+01   0.000000E+00    0.0  200
9    0.581674E+01   0.000000E+00    0.0  200
10   0.579777E+01   0.000000E+00    0.0  200
11   0.577880E+01   0.000000E+00    0.0  200   lcms
12   0.576310E+01   0.000000E+00    0.0  200   ergodic layer
13   0.576300E+01   0.000000E+00    0.0  200   surface of 5/4 islands
14   0.575762E+01   0.000000E+00    0.0  200
15   0.575223E+01   0.000000E+00    0.0  200
16   0.574685E+01   0.000000E+00    0.0   20   axis of 5/4 islands
17   0.572000E+01   0.000000E+00    0.0  200   outer ergodic area
```

```
18   0.575403E+01   0.700829E+00    0.0  200   axis of 10/9 islands
19   0.575400E+01   0.696000E+00    0.0  200   surface of 10/9 islands
```

List of input quantities

nparam	integer	number of field lines to be traced
nr	integer	field line number
R	real	*R*-co-ordinate
Z	real	*Z*-co-ordinate
phi	real	φ-co-ordinate
ntour	integer	number of toroidal turns

Note, a smaller value for *ntour* (e.g. 20) may be chosen if the field line forms an *O*-point (e.g. magnetic axis, axis of an island). One may take the advantage to write a comment behind important starting points as done in the example above.

Below we show an example for tracing field lines in a specified equidistant area range.

input **line_points_xxx** *(equidistant starting points)*

```
nparam
 100
 nr              R           Z           phi    ntour
  1           5.96850     0.0000        0.00     80
100           6.20000     0.0000        0.00     80
```

Here the GOURDON code would trace 100 field lines starting on equidistant points between $R = 5.97$ and $R = 6.20$.

Note, you also have the possibility to start field lines on equilibrium flux surfaces. For this purpose, the output file **out_gourdon_xxx** provided by the COTRANS code (mode 0) is used as input file (see also Appendix A5).

Rename the input files from above into **line_input_iota** and **line_points_iota** to compute the ι-profile. Only few of the input parameters have to be changed (only the changed lines of the standard input file **line_input_iota** are shown):

```
out_num:          'vac_01_iota'

line_input:       'line_points_iota'

line_xmgr:        'none'
line_iota:        'xmgr_iota'
line_tracing:      hpasz    npper
                   0.0100    64
```

In the input file **line_points_iota** the starting points of ergodic field lines have to be deleted, because the rotational transform is only defined for closed field lines. Additional starting points may be useful to illustrate details. The parameter *npper* has to be chosen big enough ($npper = 64$ works fine).

For the production of the DESCUR input the original input files should be copied to ones with the extension **desc**.

```
out_num:          'desc'

line_input:       'line_points_desc'

line_xmgr:        'none'
line_iota:        'none'
line_tracing:      hpasz    npper
                   0.0100    64
line_descur:      'data'
                   ntheta
                    300
```

In the file **line_points_desc** only the starting points of the magnetic axis and the lcms are left. Running the DESCUR code, it may be necessary to repeat this computation with a surface which is actually not the lcms but close to and inside it. In that case, it is ususally sufficient to vary the starting point by approximately one millimeter. Again, the parameter *npper* has to be chosen big enough ($npper = 64$ works fine).

The GOURDON code generates a standard output file and several XMGR plot files. The standard output file, named **out_line_xxx**, contains input quantities, as well as useful information about the traced field lines. The output file **data_desc_xxx** provides the input to the DESCUR code (see also Appendix A3). An example is given below:

output **data_desc_xxx** *(input to the DESCUR code)*

```
ntheta  nphi   nfp  isymm  isort
 300     64     5    2       33
magnetic axis
  raxis       zaxis
5.9685000   0.0000000
5.9664697   0.0331695
5.9604604   0.0658674
 :           :
5.9664727  -0.0331937
plasma boundary
  rin          zin
5.6448468   0.8084756
5.7589188   0.2078079
5.5878046  -0.7453986
 :           :
6.1322529   0.2248175
```

List of input quantities

ntheta	integer	number of co-ordinate points in poloidal direction
nphi	integer	number of co-ordinate points in toroidal direction (per period)
nfp	integer	number of periods
isymm	integer	symmetry of the configuration
	0	no symmetry
	1	axisymmetry
	2	stellarator symmetry
isort	integer	kind of poloidal ordering
	0	no ordering, input poinst are already poloidally ordered
	> 0	number of the toroidal cross-section where the co-ordinate points are ordered (new ordering method of the DESCUR code is used)
	< 0	the original ordering method of the DESCUR code is used

co-ordinates of the magnetic axis (for one period)

raxis(j)	real	R-co-ordinate of the magnetic axis (j=1,...,*nphi*)
zaxis(j)	real	Z-co-ordinate of the magnetic axis (j=1,...,*nphi*)

co-ordinates of the magnetic surface stored along the field line

rin(j)	real	R-co-ordinate of the surface point
zin(j)	real	Z-co-ordinate of the surface point
		(j=1,...,$ntheta \cdot nphi$)

Note, the GOURDON code yields $isort = nphi/2 + 1$. If another value of *isort* is needed as input to the DESCUR code, this parameter has to be modified manually.

Furthermore, the GOURDON code produces XMGR plot files. These files are named:

xmgr_line_xxxr_nnn	R, Z-co-ordinates of the field line at cross-section *nnn*
xmgr_iota_xxx	rotaional transform, ι, as function of the radial co-ordinate, R, for specified Z- and φ-co-ordinates

A3: DESCUR CODE

The DESCUR code uses SI units and needs two input files. An example of the standard input file, named **input_descur**, is given below. Therein a further input file is specified.

standard input file

```
suffix:              'vac'
in_coordinates:      '../gourdon/data_desc'
mode numbers:         mpol   ntor
                      11      11
num. parameters:      niter  nstep  iwrite   ivmec
                      6000    200   'short' 'datalist'
                      ftol       pexp    qexp
                      5.e-7      2.0     2.0
```

List of input quantities:

suffix	character*30	suffix added to all output files
in_coordinates	character*100	input file containing the co-ordinate points of magnetic axis and flux surface
mpol	integer	number of poloidal modes
ntor	integer	number of toroidal modes

niter	integer	number of iterations
nstep	integer	print step size
iwrite	character$*$10	length of the output
	long	
	short	
ivmec		output file format (file used as VMEC/NEMEC input)
	datalist	data list
	namelist	namelist file
ftol	real	accuracy
pexp	real	optimization parameter, p
qexp	real	optimization parameter, q

Note, the parameters p and q should be as large as needed, but not unreasonable large. It is advisable to fix $p+q=Q$ at some reasonable value. $Q \leq 7$ has been found to be numerically feasible. For more details see Ref. [5].

The DESCUR code needs as further input the co-ordinates of the flux surface which will be approximated by Fourier coefficients. For three-dimensional configurations also the co-ordinates of the magnetic axis are required. Here these data are provided by the GOURDON code (output file **data_desc_xxx**, for details see Appendix A2).

The DESCUR code produces several output files. An example of the standard output file, named **output_xxx** is given below. It contains the input parameters of the standard input file, and information about the convergence behaviour of the optimization of the Fourier coefficients.

standard output file

```
suffix:              vac

in_coordinates:      ../gourdon/data_desc

mode numbers:        mpol   ntor
                      11     11
num. parameters:     niter  nstep  iwrite   ivmec
                     6000     200  short   datalist
                     ftol       pexp    qexp
                   0.50E-06     2.00    2.00
```

```
 ITERATIONS    RMS ERROR    FORCE GRADIENT    <M>       m
                  Fitting toroidal plane:          1
     1       0.107E-01       0.179E-01      4.06      10
   200       0.261E-02       0.413E-03      2.02      10
   400       0.261E-02       0.168E-05      2.03      10
   459       0.261E-02       0.449E-06      2.03      10
                  Fitting toroidal plane:          2
     1       0.116E-01       0.185E-01      3.64      10
     :          :               :            :     :   :
     :          :               :            :     :   :
   256       0.126E-02       0.448E-06      1.82      10
                  Fitting toroidal plane:         33
     1       0.790E-02       0.216E-01      2.90      10
   200       0.135E-02       0.134E-04      1.84      10
   246       0.134E-02       0.460E-06      1.84      10

 ANGLE CONSTRAINTS WERE APPLIED
 BASED ON RM**2 + ZM**2 SPECTRUM WITH P =     2.00 AND Q =     2.00
```

The above output file shows the convergence behaviour of every toroidal cross-section (see Ref. [5] for the description of the quantities *RMS ERROR*, *FORCE*, *GRADIENT*, etc.).

Depending on the format parameter *ivmec*, the Fourier coefficients are either written as namelist file or as data file. These files are named **out_vmec_xxx**. The namelist file has the structure used by the VMEC/NEMEC input file (for details see Appendix A4). The data file has the following form:

data file **out_vmec_xxx** *(input to the VMEC/NEMEC code)*

```
  mbound nbound
    10     11
 m     n          rbc            zbs
  0     0     5.5516E+00     0.0000E+00
  0     1     2.7256E-01    -2.1088E-01
  0     2     6.4267E-04    -7.8118E-03
  0     3     2.8596E-04    -2.3622E-03
  0     4    -1.9583E-03     2.2338E-03
  0     5     3.1088E-04     2.1354E-04
  0     6    -4.5107E-04     2.9402E-04
  0     7     6.3141E-05     1.8628E-04
  0     8    -3.7526E-04    -2.0637E-04
  0     9     2.6914E-04     3.7526E-04
```

```
  0    10     8.3236E-05     3.3175E-04
  0    11     1.6422E-04     1.5755E-04
  1   -11     7.2957E-05    -7.5354E-06
  1   -10     2.6509E-05    -3.7527E-05
  :     :          :              :
  :     :          :              :
 10    11    -5.5224E-06     5.4567E-06
```

List of output quantities:

mbound	integer	maximum poloidal mode number
nbound	integer	maximum toroidal mode number
m	integer	poloidal mode number (m=0,...,*mbound*)
n	integer	toroidal mode number (-*nbound* $\leq$ n $\leq$ *nbound*)
rbc(m,n)	real	Fourier coefficients of the radial co-ordinate
zbs(m,n)	real	Fourier coefficients of the *Z*-co-ordinate

Note, in case of a non-stellarator symmetric equilibrium also the asymmetric Fourier coefficients *rbs* and *zbc* are listed.

Furthermore, the DESCUR code produces XMGR plot files. These files are named **plotout_nnn_xxx** (for details see Sect. 2.1.3) and **p_sort_nnn_xxx**. The latter are only generated if the input parameter *iwrite* is set to *iwrite* =’long’. Those files contain the poloidally ordered input data (co-ordinates of the magnetic surface).

A4: VMEC/NEMEC CODE

The VMEC/NEMEC code uses SI units. Several input files serve as input to the code. The standard input file named **input.xxx** is a namelist file. It contains paths and names of further input files. Below an example of a standard input file is given.

standard input

```
mgrid_file = '/afs/ipp/u/ers/run/W7-X/case_f54/vacfield/vacfield_vmec_44',
user_pressure = 'none',
user_itor     = 'none',
user_iota     = 'none',
user_fouraxis = '/afs/ipp/u/ers/run/W7-X/case_f54/vmec/out_vmec_axis_01'
user_fourlcms = '/afs/ipp/u/ers/run/W7-X/case_f54/descur/out_vmec_vac',
lfreeb        = .true.,
```

```
pres_profile  = 'polynome',
itor_profile  = 'none',
iota_profile  = 'none',
woutf         = 'binary'
ns_array      =  11,    33,    65,    129,
ftol_array    =  1.e-6, 5.e-8, 5.E-9, 5.0E-11,
delt          =  0.20,
niter         =  3000,
nstep         =   200,
nvacskip      =     8,
mpol          =    12,
ntor          =    11,
ntheta        =    48,
nzeta         =    44,
iasym         =     0,
nfp           =     5,
ncurr         =     1,
curtor        =   0.0,
phiedge       =  1.50,
gamma         =   0.0,
betascale     =   1.0,
am            =  3.342951500E-02, -6.6859032E-02, 3.342951500E-02,
ai            =  0.0,              0.0,             0.0,
ac            =  0.0,              0.0,             0.0,
raxis_co(0)   =  0.0, zaxis_si(0) = 0.0,
raxis_co(1)   =  0.0, zaxis_si(1) = 0.0,
  :           :   :        :      :  :
  :           :   :        :      :  :
raxis_co(11)  =  0.0, zaxis_si(12)= 0.0,
rbc(0,0)      =  0.0, zbs(0,0)  = 0.0, rbs(0,0)  = 0.0, zbc(0,0)  = 0.0,
rbc(1,0)      =  0.0, zbs(1,0)  = 0.0, rbs(1,0)  = 0.0, zbc(1,0)  = 0.0,
  :           :   :        :    :  :        :    :  :        :    :  :
  :           :   :        :    :  :        :    :  :        :    :  :
rbc(11,12)    =  0.0, zbs(11,12 = 0.0, rbs(11,12)= 0.0, zbc(11,12)= 0.0,
```

List of input quantities:

input file names and profile types

mgrid_file	character∗80	path and name of the input file containing the vacuum magnetic field
	none	no vacuum magnetic field, a fixed-boundary equilibrium is computed

	xxx	path and file name
user_pressure	character∗80	path and name of the input file containing the mass profile
user_itor	character∗80	path and name of the input file containing the toroidal current profile
user_iota	character∗80	path and name of the input file containing the ι-profile
user_fouraxis	character∗80	path and name of the input file containing the Fourier coefficients of the magnetic axis
	none	the Fourier coefficients, *raxis_co*, *zaxis_si*, listed in the namelist file are used
	xxx	path and file name
user_fourlcms	character∗80	path and name of the input file containing the Fourier coefficients of the plasma boundary
	none	the Fourier coefficients, *rbc*, *zbs*, *rbs*, *zbc*, listed in the namelist file are used
	xxx	path and file name

working parameters

lfreeb	logical	fixed- or free-boundary equilibrium
	true	a free-boundary equilibrium is computed
	false	a fixed-boundary equilibrium is computed
pres_profile	character∗40	kind of the mass profile
	user	profile provided by the input file **user_pressure**
	polynome	profile defined by a polynome of order 10; its coefficients, $am(i)$ (i=0,...,10), are defined in the namelist file
itor_profile	character∗40	kind of the toroidal current profile
	user	profile provided by the input file **user_itor**
	polynome	profile defined by a polynome of order 10; its coefficients, $ac(i)$ (i=0,...,10), are defined in the namelist file
iota_profile	character∗40	kind of the ι-profile
	user	profile provided by the input file **user_iota**
	polynome	profile defined by a polynome of order 13, its coefficients, $ai(i)$ (i=0,...,13), are defined in the namelist file
woutf	character∗10	format of the output file **wout.xxx**
	binary	
	ascii	
ns_array(i)	integer	numbers of radial surfaces which are used for the iterations (i=1,...,100)

ftol_array(i)	real	the iterations with $ns\_array(i)$ radial surfaces are performed with accuracy $ftol\_array(i)$ (i=1,...,100)
delt	real	step size of the iteration
niter	integer	number of iterations made with $ns\_array(i)$ radial surfaces and accuracy $ftol\_array(i)$
nstep	integer	the result of every $nstep^{\text{th}}$ iteration step is written in the output file **threed1.xxx**
nvacskip	integer	in case of a free-boundary calculation the boundary conditions are taken into account in every $nvacskip^{\text{th}}$ iteration step
mpol	integer	total number of poloidal modes (m=0,...,$mpol-1$)
ntor	integer	maximum toroidal mode number ($-ntor \leq$ n $\leq ntor$)
ntheta	integer	number of poloidal grid points (should be three or four times of $mpol$)
nzeta	integer	number of toroidal grid points (should be three or four times of $ntor$)

physical parameters

iasym	integer	symmetry of the equilibrium
	0	stellarator symmetry
	1	no symmetry
nfp	integer	number of periods
ncurr	integer	choice between toroidal current and ι-profile
	0	ι-profile is used
	1	toroidal current profile is used
curtor	real	total toroidal current (only used for $ncurr = 1$)
phiedge	real	total toroidal flux
gamma	real	compressibility (see Ref. [6])
	0.	mass profile = pressure profile
	$> 0.$	mass profile $\neq$ pressure profile
betascale	real	scale factor of the mass profile
am(i)	real	polynomial coefficients of the mass profile
ai(i)	real	polynomial coeffcients of the ι-profile
ac(i)	real	polynomial coeffcients of the toroidal current profile
raxis_co(n)	real	symmetric Fourier coefficients of the magnetic axis (radial co-ordinate, n=0,..,$ntor$)
raxis_si(n)	real	asymmetric Fourier coefficients of the magnetic axis (radial co-ordinate, n=0,...,$ntor$)
zaxis_si(n)	real	symmetric Fourier coefficients of the magnetic axis (Z-co-ordinate, n=0,...,$ntor$)

zaxis_co(n)	real	asymmetric Fourier coefficients of the magnetic axis (Z-co-ordinate, n=0,...,*ntor*)
rbc(n,m)	real	symmetric Fourier coefficients of the plasma boundary (radial co-ordinate, n=-ntor,...,*ntor*, m=0,...,*mpol*)
rbs(n,m)	real	asymmetric Fourier coefficients of the plasma boundary (radial co-ordinate, n=-*ntor*,...,*ntor*, m=0,...,*mpol*)
zbs(n,m)	real	symmetric Fourier coefficients of the plasma boundary (Z-co-ordinate, n=-*ntor*,...,*ntor*, m=0,...,*mpol*)
zbc(n,m)	real	asymmetric Fourier coefficients of the plasma boundary (Z-co-ordinate, n=-*ntor*,...,*ntor*, m=0,...,*mpol*)

Here the VMEC/NEMEC code computes free-boundary equilibria, that is, the vacuum magnetic field has to be provided in an input file specified by the parameter $mgrid\_file$. Furthermore, the initial guesses of the Fourier coefficients of the magnetic axis and the lcms are given in the input files defined by the parameters $user\_fouraxis$ and $user\_fourlcms$.

If the VMEC/NEMEC code is executed for the first time, the initial guess for the lcms is taken from the **out_vmec_xxx** file of the DESCUR code. The initial guess for the magnetic axis has to be taken from some similar case.

For further executions of the VMEC/NEMEC code, the results of previous calculations can be used as initial guesses. These data are extracted form the VMEC/NEMEC output **wout.xxx** and written into the files **out_vmec_axis_xxx** and **out_vmec_lcms_xxx** by the COTRANS code.

The parameters ns_array and $ftol\_array$ define the numbers of surfaces and the accuracies which are used for the computations. Usually, the VMEC/NEMEC code starts with a small number of surfaces (e.g. $ns\_array = 11$) and a low accuracy (e.g. $ftol\_array = 1.e-7$). As soon as this accuracy or the maximum number of iterations, *niter*, is reached, the next entries are aimed at (e.g. $ns\_array = 33$ and $ftol\_array = 1.e-8$). For the following entries this scheme is repeated. The parameter $nvacskip$ tells the VMEC/NEMEC code how often to take into account the boundary conditions. For example, $nvacskip = 4$ means that in every fourth step the boundary conditions are checked and the necessary changes are made. Values between 4 and 10 seem to be reasonable. This parameter affects considerably the performance of the VMEC/NEMEC code.

A good choice for the accuracies seems to be something such as 1.e-7, 1.e-8, 1.e-9, 5.e-10. Changes may reduce or increase the computational time and improve or deteriorate the results. Some tests with different values might be useful.

The parameter *delt* determines how much the Fourier coefficients are changed during one iteration step. This parameter influences the convergence significantly.

The computational time of the VMEC/NEMEC code also strongly depends on the poloidal and toroidal mode numbers. A typical computational time is ≈ 4 hours for the calculations reported in this document. When executing the VMEC/NEMEC code for the first time, it usually is enough to use *11* and *33* surfaces to get an idea of the toroidal flux (for details see steps 4-6).

Then, the computational time is considerably reduced.

The VMEC/NEMEC code generates the output files **threed1.xxx** and **wout.xxx**. The first file stores standard output quantities, such as input parameters, information about the convergence of the iteration, and important equilibrium quantites (plasma beta, $< \beta >$, plasma volume, V, aspect ratio, A, etc.). The equilibrium solution (Fourier representations of flux surfaces and magnetic field) is written into the file **wout.xxx**.

A5: COTRANS CODE

The COTRANS code uses SI units. It needs a standard input file and further input files defined therein. The code generates various output files and XMGR plot files. Input and output files are described below for the modes 0, 4, and 7 of the code.

Mode 0: interface VMEC/NEMEC $\Longrightarrow$ GOURDON code

Mode 0 of the COTRANS code transforms the representation of the VMEC/NEMEC equilibrium from flux co-ordinates into cylindrical co-ordinates. As input it needs the standard input file **input_mode0**, and the VMEC/NEMEC output file **wout.xxx**.

standard input file

```
out_num:         'vac_01'
input_code:      'VMEC/NEMEC'
output_code:     'GOURDON'
in_coordinate:   '../vmec/wout.vac_01'
format_type:     'binary'
transformation from flux into cylindrical co-ordinates
working_param.:  ifn   nu  nv   ustart   vstart   ntour
                  20  100   4    0.25      0.0      80
                 iplot  jfl
                   1    120
```

List of input quantities:

mode of the code

out_num	character∗25	suffix added to all output file names
input_code	character∗80	code providing the input data
	VMEC/NEMEC	VMEC/NEMEC code
output_code	character∗80	code obtaining the output data
	GOURDON	GOURDON code

non-standard input file name

in_coordinate	character∗80	VMEC/NEMEC output file name (**wout.xxx**)
format_type	character∗80	type of the file format
	binary	
	ascii	

working parameters

flux surface plot co-ordinates (XMGR plot file **xmgr_fluxsur_xxx***)*

ifn	integer	number of flux surfaces
nu	integer	number of poloidal grid points per flux surface
nv	integer	number of toroidal cross-sections

preset of initial data used for field tracing in the GOURDON code (XMGR plot file **out_gourdon_xxx***)*

ustart	real	poloidal flux co-ordinate
vstart	real	toroidal flux co-ordinate
ntour	integer	number of toroidal turns

writing of XMGR plot files

iplot	integer	generation of XMGR plot files
	0	no plot files
	1	reduced number of plot files
	2	all plot files

writing of the Fourier coefficients of an outer surface which will serve as initial guess for a new VMEC/NEMEC calculation (output file **out_four_lcms_xxx***)*

jfl	integer	number of the outer flux surface

Note, if ifn and/or jfl exceed the number of surfaces computed with the VMEC/NEMEC code, they are reduced to this number. The parameter ifn also specifies the number of data sets written into output file **out_gourdon_xxx**.

Mode 0 generates various output files and XMGR plot files. Below all output files and some of the XMGR plot files are listed:

output_xxx	standard output file containing all input quantities and important quantities of the VMEC/NEMEC equilibrium
out_gourdon_xxx	input file to the GOURDON code containing individual starting points of field lines on flux surfaces (equivalent to file **line_points_xxx** in Appendix A2)
out_vmec_axis_xxx	input file to the VMEC/NEMEC code containing the initial guess of the magnetic axis
out_vmec_lcms_xxx	input file to the VMEC/NEMEC code containing the initial guess of the lcms
xmgr_iota_rtor_xxx	rotational transform, ι, as function of ρ_{tor}
xmgr_jpol_rtor_xxx	total poloidal current, J, as function of ρ_{tor}
xmgr_fluxsur_nnn_xxx	equilibrium flux surfaces at toroidal cross-section *nnn*

Mode 4: Mercier and resistive interchange stability

Mode 4 of the COTRANS code investigates the VMEC/NEMEC equilibrium with respect to Mercier and resistive interchange stabilities. It needs two input files, namely the standard input file **input_mode4**, and the VMEC/NEMEC output file **wout.xxx**.

standard input file

```
out_num:         'beta_03b'
input_code:      'VMEC/NEMEC'
output_code:     'MERCIER STABILITY'
in_coordinate:   '../../vmec/wout.beta_03b'
format_type:     'binary'
computation of Boozer co-ordinates
working_param:    inu  inv  mpol  ntor ilambda iwrite iplot ilinear npktf
                   4    4    20    20     1      0     0      1      1
```

Here only the working parameters are described. For the description of the other quantities see mode 0.

List of input quantities:

working parameters

inu	integer	$nu = inu \cdot (mpol - 1)$ number of poloidal grid points
inv	integer	$nv = inv \cdot ntor$ number of toroidal grid points

mpol	integer	total number of poloidal modes ($0 \leq$ m $\leq$ *mpol*-1)
ntor	integer	maximum toroidal mode number (*-ntor* $\leq$ n $\leq$ *ntor*)
ilambda	integer	computation of the single-valued function λ (see Ref. [12])
	0	λ of the VMEC/NEMEC output is used
	1	λ is recalculated in the COTRANS code
iwrite	integer	length of the output
	0	short output
	1	detailed output
iplot	integer	output of XMGR plot data
	0	only plot files of the results
	1	reduced number of plot files
	2	all plot files
ilinear	integer	interpolation method applied to the Fourier coefficients in radial direction
	0	cubic spline interpolation
	1	linear interpolation
npktf	integer	number of inner radial points neglected and recalculated with cubic spline interpolation

Note, the working parameters of mode 4 are chosen in an adequate way for W7-X configurations, and, normally, don't have to be changed.

Mode 4 of the COTRANS code produces a standard output file called **output_xxx**, and several XMGR plot files, e.g.:

xmgr_well_rtor_xxx	magnetic well, V'', as function of ρ_{tor}
xmgr_mercier_rtor_xxx	Mercier stability as function of ρ_{tor}
xmgr_resistive_rtor_xxx	resistive interchange stability as function of ρ_{tor}

The standard output file lists all quantities of the standard input file, important quantities of the VMEC/NEMEC equilibrium, and quantities concerning the stability criteria.

Mode 7: interface VMEC/NEMEC $\Longrightarrow$ MFBE code

Mode 7 of the COTRANS code transforms the VMEC/NEMEC output into an appropriate MFBE input. It requires the standard input file **input_mode7**, and the VMEC/NEMEC output file **wout.xxx**.

standard input file

```
out_num:         'vac_01'
input_code:      'VMEC/NEMEC'
output_code:     'MFBE'
in_coordinate:   '../../vmec/wout.vac_01'
format_type:     'binary'
out_coordinate: 'out_four'
format_type:     'binary'
preparation of the MFBE input
working_param:    inu  inv  mpol  ntor ilambda iwrite iplot ilinear npktf
                    4    4    20    20      1      0     0       1       4
```

List of input quantities:

non-standard output file name

out_coordinate	character∗80	MFBE input file name
format_type	character∗80	type of the file format
	binary	
	ascii	

For the meaning of the other parameters see descriptions of mode 0 and mode 4.

Mode 7 generates two output files, namely the standard output file called **output_xxx**, and the MFBE input file named **out_four_xxx**. The standard output file lists all quantities of the standard input file, important quantities of the VMEC/NEMEC equilibrium, and some MFBE input quantities.

A6: MFBE CODE

Here an example of the standard input file (**input_mfbe_xxx**) is given with parameters adjusted to the W7-X stellarator. Normally, only the input file names *in_equilibrium* and *in_field* have to be changed, if calculations are performed for this fusion device.

standard input file

```
in_equilibrium:'/afs/ipp/u/ers/run/W7-X/case_f54/cotrans/mode7/out_four..'
format_type:   'binary'
in_code:       'COTRANS'
```

```
in_field:      '/afs/ipp/u/ers/run/W7-X/case_f54/vacfield/vacfield_mfbe'
format_type:   'binary'
out_field:     '/afs/ipp/u/ers/run/W7-X/case_f54/mfbe/field_mfbe_vac_01'
format_type:   'binary'
configuration: 'W7-X'
                r0m    a0m
                5.5    0.55
stop:          'out_boundary'
test:          'no'
                nrt1  nrt2  nzt1  nzt2  npht1  npht2
                 60   120    70    70     1      1
complex integration
                jubmin   icompd   epscom
                 2000       7      1.e-3
determination of the curvilinear co-ordinates
                xtol_c05pbe  ifail_c05pbe  seps   dus   nus  icout
                   1.e-5          1       1.e-5   0.05  10    20
determination of the minimum distance
                e1_e04bbe e2_e04bbe db_e04bbe maxcall_e04bbe ifail_e04bbe
                 1.e-8      1.e-8     0.30         20             1
equidistant grid for surface integration
                jud  jvd
                320  600
non-equidistant grid for surface integration
                nc1u   nc1v   r2x    r3x   r4x
                45     45   1.e-3   1.5  0.15
maximum difference between Bc and B
                dbphimax  dbrmax  dbzmax
                  0.50     0.50    0.50
outer boundary
                msr  nsr  mnsr
                 7    4    59
                 m    n     crc          crs          czs          czc
                 0    0  5.5178E+00  0.0000E+00   0.0000E+00  0.0000E+00
                 :    :     :           :            :           :
                 :    :     :           :            :           :
                 6    4 -4.3190E-04  0.0000E+00  -1.8510E-04  0.0000E+00
```

List of input quantities:

in_code	character*80 VMEC/NEMEC	name of the code providing the equilibrium input the VMEC/NEMEC output file **wout.xxx** serves directly as input

COTRANS	the VMEC/NEMEC output is modified by the COTRANS code; its output file **out_four_xxx** serves as input

A description of the other parameters will be found in Ref. [10].