Optimization of neoclassical transport in stellarators

B. Seiwald[1], V. V. Nemov[2], V. N. Kalyuzhnyj[2], S. V. Kasilov[2] and W. Kernbichler[1]
[1]Institut für Theoretische Physik - Computational Physics, Petersgasse 16, A-8010 Graz, Austria
[2]Institute of Plasma Physics, National Science Center "Kharkov Institute of Physics and Technology",
Ul. Akademicheskaya 1, 61108 Kharkov, Ukraine
E-mail: Bernhard.Seiwald@itp.tugraz.at

***Abstract*: For some of the existing fusion devices with toroidal magnetic confinement optimizations are planned. For stellarators where the finite plasma pressure causes a weak influence on the equilibrium the optimization can be done in real-space coordinates. In the present work a new code for optimizing stellarators with fixed coil design is developed.**

***Keywords*: Optimization, Neoclassical Transport, Magnetic Field Line Integration, Fusion Plasmas**

I. Introduction

In the present project a new and fast code for optimizing existing stellarators, based on the existing code for calculating neoclassical transport, is presented. The code is able to deal with realistic vacuum fields and the optimization can be done without any restrictions to the complexity of the field.

The simulation could show the amount of possible improvement of existing stellarators when modifying the currents or changing the position and angle of the coils. The proposed tool can find optimal configurations with respect to neoclassical transport in the long-mean-free-path regime.

Various codes exist for optimizing fusion devices where equilibrium codes are necessary (e.g., [1] and [2]). For some of the stellarators, namely the Uragan-2M (U-2M) [3], [4] and the TJ-II [5] (on which this work is focused), the plasma causes only a week influence on the MHD equilibrium. Therefore, there is no need for running MHD equilibrium codes like VMEC [6]. This allows to optimize the total stored energy in a plasma fully in real space (RS) coordinates.

Improvement of the above mentioned devices can be done through changes of the coil currents (and/or coil positions). The total stored energy depends on neoclassical transport properties (see Sec. II). An important quantity for the evaluation of the neoclassical transport is the so-called effective ripple $\epsilon_{\rm eff}$. The effective ripple is computed by the field line integration technique with the help of a Biot-Savart code. The obtained results for total stored energy reflect only $1/\nu$-transport (which is a very important transport regime in stellarators), but, nevertheless, this quantity is a main objective for neoclassical optimization studies.

For a wide class of optimization algorithms the computation of gradients of the target quantity is needed. Here, the computation of the gradients cannot be done practically with an appropriate accuracy. As a consequence of this, the well known Simulated Annealing algorithm has been chosen (see Sec. III).

II. Physics

Neoclassical transport plays an important role in stellarator physics due to its impact on particle confinement properties. The neoclassical transport theory shows that, compared to the classical transport theory for a magnetic field with straight field lines, the complication of the field configuration and hence more complex trajectories of particles in stellarators lead to a considerable increase of transport coefficients. The effective ripple $\epsilon_{\rm eff}$ is a measure for neoclassical transport across the magnetic flux surfaces. For good confinement $\epsilon_{\rm eff}$ should be small. The effective ripple, which is a function of the effective radius of the flux surface [7], [8], can be used to compute the total stored energy in plasma by solving the heat conductivity equation using a given particle density profile (see A). It is assumed that the heat source is located at the magnetic axis. The radial dependence of $\epsilon_{\rm eff}$ can be computed by a field line tracing code [7] where the magnetic field is directly computed from field coils without transformation to magnetic coordinates. Additionally, a proper detection of islands and ergodic zones has to be ensured.

A. Total stored energy in plasma

As fitness parameter for the optimization the total stored energy in the plasma volume in case of an energy source which is localized at the magnetic axis is used. Cold ions are assumed and the ambipolarity condition is used in order to eliminate the explicit dependency of the radial electric field in the system.

It is assumed that the temperature profile is defined by the heat conductivity equation

$$\frac{1}{r}\frac{\partial}{\partial r} r \kappa_\perp \frac{\partial T}{\partial r} + Q(r) = 0\,, \tag{1}$$

and the boundary conditions

$$T(a) = 0 \quad \text{and} \quad r\frac{\partial T}{\partial r}\Big|_{r=0} = 0,$$

where r is an effective radius and a is the effective radius of the boundary of the plasma. The energy source $Q(r)$ is located at the magnetic axis and is written as $Q(r) = (Q_0/r)\delta(r)$.
The particle and energy flux across the flux surfaces is proportional to $\epsilon_{eff}^{3/2}$ (for the $1/\nu$ regime). The heat conductivity coefficient is

$$\kappa_\perp = A \epsilon_{\rm eff}^{3/2} T^{7/2},$$

where A is a constant which does not change during the optimization. The quantity $\kappa_\perp$ is proportional to the temperature T and $\epsilon_{\rm eff}^{3/2}$ and, therefore, computation of $\epsilon_{\rm eff}^{3/2}$ for sets of computed magnetic surfaces is an essential part of the optimization procedure.

Integrating the temperature profile one obtains the normalized total stored energy as

$$\hat{W} = \int_0^a dr\, r\, \hat{n}(r) \left(\int_r^a \frac{dr'}{r'\, \epsilon_{\rm eff}^{3/2}(r')} \right)^{2/9}, \qquad (2)$$

where the normalization factor stays constant during optimization. For computing the total stored energy, the last closed flux surface has to be detected.

B. *Effective radius*

The effective radius of a magnetic flux tube is defined in [7] in differential form, $Sdr = dV$, where S is the area of the magnetic surface and V is the volume enclosed by the magnetic surface. The computation of this quantity requires the calculation of many magnetic surfaces. The accuracy is low, if just a few surfaces are used. One can introduce a different definition of an effective radius $r_{\rm eff} = 2V/S$, which can be calculated during a single field line integration. Dividing the volume V, limited by the magnetic surface, by the surface area S, one obtains

$$\begin{aligned} r_{\rm eff} &= \frac{2V}{S} = \frac{2}{3} \frac{\int dS \mathbf{r} \cdot \frac{\nabla\psi}{|\nabla\psi|}}{\int dS} = \frac{2}{3} \frac{< \mathbf{r} \cdot \nabla\psi >}{< |\nabla\psi| >} \\ &= \frac{2}{3} \lim_{L\to\infty} \frac{\int_0^L \frac{dl}{B} \mathbf{r} \cdot \nabla\psi}{\int_0^L \frac{dl}{B} |\nabla\psi|}, \end{aligned} \qquad (3)$$

where $\nabla\psi$ is the vector normal to the flux surface, $\mathbf{r}$ is a radius vector, B the module of the magnetic field and dl a distance measured along the magnetic field line. The definition of the effective radius (3) is used in Eq. 2 for the optimization.

The normal vector $\nabla\psi$ is computed during the field line integration. This can be done if its direction is known at the beginning of the integration (see [9]). The direction of $\nabla\psi$ is always given at the $z = 0$ in the $\varphi = 0$ plane due to symmetry properties of the magnetic flux surface of this point.

For configurations like the TJ-II "standard" configuration the effective radius (3) is close to the same quantity defined in [7], since the shape of the cross-sections of magnetic surfaces changes only little with radius (see Fig. 1).

In Eq. 2 the integration variable is the effective radius of the flux surfaces. Due to this it is necessary to detect islands and "bad" flux surfaces (see Sec. C).

C. *Detection of islands and "bad" field lines*

A magnetic flux tube which closes itself after several toroidal turns exhibits on a cross section an island structure (see Fig. 2) and is, therefore, called magnetic island, while a "good" magnetic flux tube (enclosed by the magnetic surface) is closed after one turn. Magnetic islands correspond to field lines where the rotational transform ι is a rational number. The rotational transform ι is defined as the average slope of the magnetic field line forming a flux surface. The effective radius of nested magnetic surfaces increases from the magnetic axis towards the plasma boundary. The radius of a magnetic island is very small compared to the radius of neighboring magnetic surfaces, see Fig. 3. In

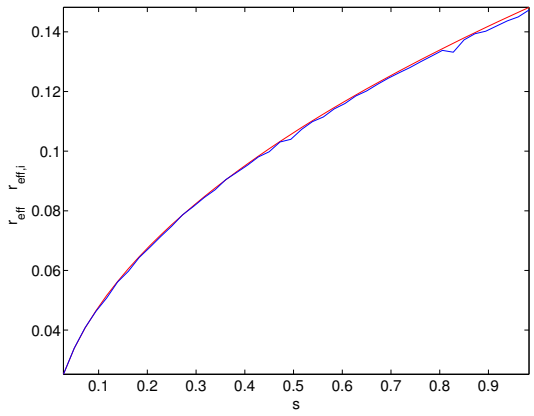

Figure 1. The effective radius calculated as proposed in [7] (red) compared to the effective radius as proposed in Eq. 3 (blue) with the field line integration for the TJ-II "standard" configuration.

Fig. 3 the island appears at $R_{\rm beg} = 1.647$m. The computed effective radius of the island is $r_{\rm eff} = 0.0019$m, which is considerably smaller than the effective radius of the neighboring good surfaces with an effective radius of about 0.11m and 0.13m. This island appears close to the last closed surface in Fig. 5.

The detection of magnetic islands is done in the $\varphi = 0$ plane because, there, $R_{\rm beg}$ is known (see Sec. B). For this purpose the angle α between the vectors $\mathbf{P}$ and $\nabla\Psi$ is used (see Fig. 2). The vector $\mathbf{P}$ points from the magnetic axis MA to a point P on the magnetic surface. The magnetic axis is the innermost (degenerated) magnetic surface with $r = 0$. For a "good" magnetic surface $\cos\alpha$ is negative for all points on the surface. If the magnetic axis is not located inside the flux tube, like for magnetic islands, the numbers of positive and negative values for $\cos\alpha$ are close to each other.

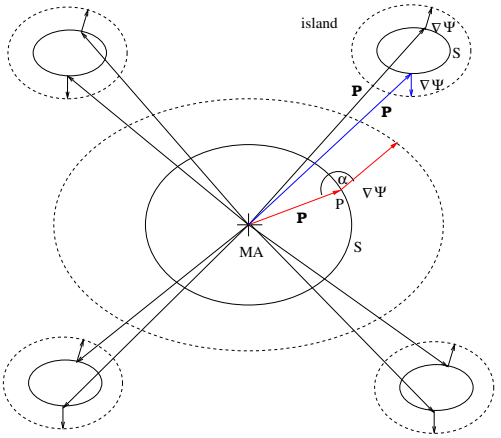

Figure 2. Island detection: Here a "good" flux surface and an island corresponding to $\iota = 1/4$ is shown. MA ... magnetic axis, S ... magnetic surface.

The magnetic axis is found by fitting the effective radius $r_{\rm eff}$ vs. the starting value $R_{\rm beg}$ for the field line integration (see Fig. 3). For the magnetic axis $r_{\rm eff}$ is zero. For fitting $r_{\rm eff}(R_{\rm beg})$ only the nested flux surfaces can be used, therefore, islands have to be excluded.

The normal vector $\nabla\Psi$ is sensitive to islands close to a magnetic surface. For this flux surface $\nabla\Psi$ shows a characteristic convexity. This is shown in Fig. 4.

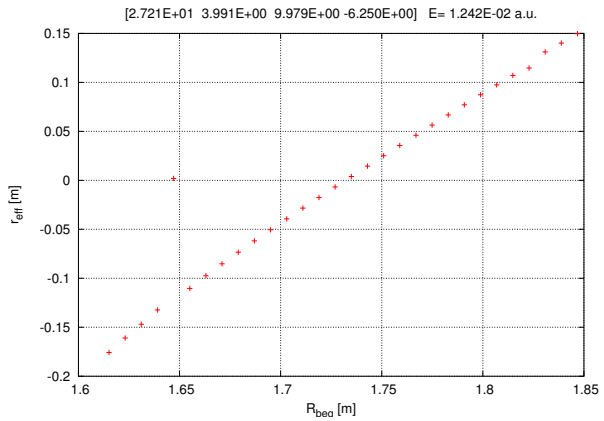

Figure 3. Finding the magnetic axis for the "standard" configuration of the TJ-II in the $\varphi = 0$ plane. The point at $R_{\mathrm{beg}} = 1.647$m with $r_{\mathrm{eff}} = 0.0019$m belongs to an island and is not used for finding the magnetic axis. Negative values of r_{eff} result from a inward directed $\nabla\Psi$ at the beginning of the field line integration.

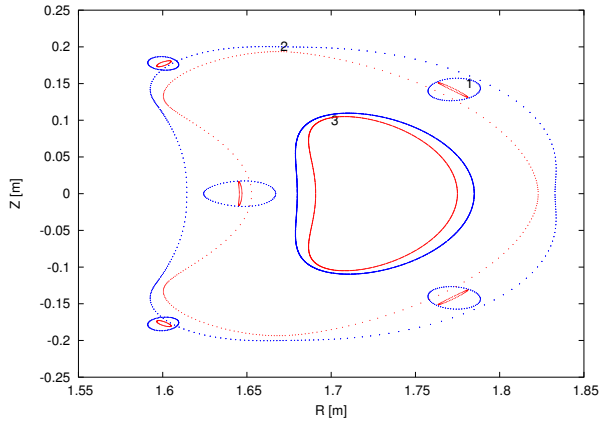

Figure 4. Flux surfaces (red) and the corresponding $\nabla\Psi$ (blue) for an island chain (1) and two flux surfaces (2,3).

Field lines which belong to a stochastic zone do not form a magnetic surface. For such field lines the module of $\nabla\Psi$ increases continuously. This behavior can be explained by the fact that, in this case, the function Ψ (as well as $\nabla\Psi$) is not a single valued function of the position. Therefore, these field lines can be easily marked as "bad" field lines.

III. Optimization

The proper computation of the physical target quantity is a key feature of the optimization code designed for improving existing helical devices through (minor) changes in coil currents (and/or coil positions) [8],[10]. The optimizer uses the Simulated Annealing algorithm, which is a stochastic method for optimization. Simulated Annealing (SA), first presented by Kirkpatrik et al. [11], simulates in principle the annealing process of solid state matter. It is used for solving combinatorial optimization problems and has been proven to be a good technique for a lot of applications [12], [13]. This method uses stochastic processes to "scan" the parameter space and selects "test configurations". From each test configuration a fitness parameter (here $E = -\hat{W}$) is calculated. Comparing the fitness parameters according to the simulated annealing process gives the desired solution.

Like the physical process, the algorithm behaves "non-greedy" during the cooling process and gets "greedy" when the artificial temperature T is zero. If the algorithm tends to get stuck in a local minimum it has a good chance to get out of it by itself. This features will be clarified in the following explanation of SA.

A. Theory and algorithm of Simulated Annealing

The major elements of the algorithm are (like for the physical process):

1) Choice of a proper starting temperature T_a and starting configuration C_a.
2) For a constant temperature T a thermodynamical equilibrium has to be reached.
3) Cooling: decrement of the temperature.
4) Stopping if the temperature T_e is low enough.

This scheme has a few parameters, which have to be adapted for each problem separately. Mostly these parameters cannot be calculated analytically and have to be found during test runs. For optimizing stellarators this procedure is not practicable because the computational time for computing the energy of one configuration is rather high (up to two hours on a PC cluster for parallelized computation; for parallelization see Sec. B) compared to the computational time of classical problems like the traveling salesman problem (up to seconds). Due to this fact it is desirable to "transform" sensitive parameters to less sensitive parameters and let the algorithm adjust some parameters itself [14]. In the following the theory and the algorithm are explained shortly.

We start at point 2 - reaching a thermodynamical equilibrium at a constant temperature. This means that the path through the configuration space has to be long enough. For making the path, a rule is needed, which proposes from a configuration C a neighboring configuration C'. For program stellopt-RS-SA one of the parameters on which the configurations depend is chosen randomly. For this parameter a new value is, again, chosen randomly. Then the energy $E(C')$ is computed and compared to $E(C)$. Accepting and rejecting configurations as it is done in the well known Metropolis algorithm gives the desired equilibrium distribution in the configuration space. This leads to the algorithm:

1) Choice of a starting configuration C_a and computation of the energy $E(C_a)$.
2) Creating a configuration C' out of C using a proper rule and computation of the energy $E(C')$; let $\eta = (E(C') - E(C))/T$, with the artificial and proper scaled temperature T (to absorb the Boltzmann-constant k_B).
3) If $\eta \leq 0$ accept C' as new configuration: $C' \rightarrow C$ and go on at step 2 or stop if the path is long enough for the temperature T.
4) If $\eta > 0$ choose a random number r, distributed equally in $0 \leq r < 1$. If $r < \exp(-\eta)$, accept C', $C' \rightarrow C$, otherwise reject C' and go on with step 2.

From point 3 it can be seen, that a configuration C' which is better than a configuration C is always accepted. In point 4 of the algorithm, one can see that a configuration C' can be accepted with a certain probability despite the fact that

C' is worse compared to C. This mechanism allows the algorithm to move away from a local minimum.

For a fixed temperature T the averages of E and E^2 are defined as

$$< E >= \frac{1}{L}\sum_k E(k) \quad \text{and} < E^2 >= \frac{1}{L}\sum_k E^2(k).$$

Furthermore, the heat capacity

$$C(T) := \frac{d < E >}{dT} = \frac{< (\delta E)^2) >}{T^2} \tag{4}$$

is needed. For the length of the path the number of accessible neighbors is used, which is widely proposed in literature. The next question is about a proper starting temperature T_a. Integration of the heat capacity (Eq. 4) leads to

$$< E > (T) - < E > (\infty) \approx -\frac{< (\delta E)^2) > (\infty)}{T}.$$

Here $< E > (\infty)$ is the energy and $< (\delta E)^2) > (\infty)$ is the variance for $T = \infty$. In other words, $T = \infty$ means that each proposed configuration is accepted. T_a is chosen in a way, that $E(T_a)$ is in the range of the variance for $T = \infty$. This leads to

$$< E > (T_a) =< E > (\infty) - \sqrt{< (\delta E)^2 > (\infty)}$$

and as an estimation for the starting temperature

$$T_a = \sqrt{< (\delta E)^2 > (\infty)}. \tag{5}$$

Now a cooling strategy is needed. A widely used empirical rule is $T_k = T_a q^k$, with $0 < q < 1$ e.g. $q = 0.95$. This scheme does not take care of phase transitions, where the temperature should be lowered carefully.

For a proper cooling strategy, the occupation probability $w(C, T)$ for two temperatures T_k and T_{k+1} should be close to each other. This can be formulated as

$$\frac{1}{1+\delta} < \frac{w(C, T_k)}{w(C, T_{k+1})} < 1 + \delta.$$

For $w(C, T)$ the Boltzmann distribution for equilibrium is used:

$$w(C, T) \sim \exp\left(-\frac{E(C) - E(0)}{T}\right),$$

where $E(0)$ is the energy of the lowest/best value for E. This leads to the scheme for cooling

$$T_{k+1} = \frac{T_k}{1 + \frac{T_k}{3\sqrt{<(\delta E)^2>}} \ln(1+\delta)}, \tag{6}$$

where δ is a control parameter, which is fixed at the beginning. For small δ values small temperature steps are done and for big δ the steps are big. This scheme automatically reduces the temperature slowly near phase transitions, because $< (\delta E)^2 >$ is big there. For the stopping criterion is requested, that $< E > (T_e) - E_0$ is small. This is formulated as

$$\frac{< E > (T_e) - E(0)}{< E > (T_a) - < E > (T_e)} < \varepsilon,$$

which leads to the criterion

$$\frac{< (\delta E)^2 > (T_e)}{T_e\,(< E > (T_a) - < E > (T_e))} < \varepsilon, \tag{7}$$

where all quantities at the left hand side can be computed during the run of SA.

B. Parallelization

The computational time for integrating a sufficient number of field lines for a magnetic field configuration is very high. Due to the weak influence of the finite plasma pressure on the equilibrium, there is no need for running MHD equilibrium codes and magnetic field lines can be computed independently of each other. Therefore, it is possible to parallelize the computation of magnetic field lines. For parallelizing the message passing interface (MPI), a very flexible system, has been chosen. For stellopt-RS-SA a master/slave model has been implemented. The advantage of this kind of speeding up computation is, that very high speed for computing the total stored energy for one configuration is reached on multiprocessor systems or clusters of computers. This is of advantage if scans of parts of the phase space are needed.

The scheme for the parallelized computation is as follows:

```
MPI_INIT()
MPI_COMM_RANK(myid)
IF (myid .EQ. 0) THEN        ! M A S T E R
 C1 := SetInitialSolution()
 T := WarmingUp()
 do
        do
                C2 := Neighbor(C1)
                (E1 = Energy(C1))
                MPI_Bcast()
                MPI_Send()
                MPI_Irecv()
                E2 = Energy(C2)
                Δ Energy := E2 - E1
                if Δ Energy < 0 or Accept(Δ Energy, T)
                        C1 := C2
        until Equilibrium()
        T :=          DecrementT()
       until Frozen()
ELSE      ! S L A V E
       MPI_Bcast()
       MPI_Irecv()
       Field Line Integration(), Stability()
       MPI_Send()
END IF

Message Passing Interface MPI
algorithm structure      (solver)
problem specific functionality      (user implemented)
temperature scheduling      (scheduler)
```

Due to the stochasticity of SA, a proposed configuration C' can be very close to another configuration C. This might not change the physical properties and the total stored energy significantly. For further reduction of the computational time, only discrete values of the coil currents (and coil positions) are used. This means that a grid is spanned over the parameter space defined by all variable quantities. The grid size is chosen in a way, that the total stored energy in plasma does not change excessively from one grid point to a neighboring grid point. So the function

will not artificially become too spiky. A proper grid size is determined by preliminary scans.

If an interesting minimum is found, the optimization is started in this minimum or close to it with a substantially refined grid and a reduced allowed parameter space to improve this minimum. Furthermore, all computed configurations are stored in a proper data structure in a database for restarts of the optimizer. Therefore, it is not necessary to compute one and the same configuration twice.

IV. Results

For the first run of the optimization, four currents (the toroidal coil current, I_{tor}, the helical coil current, I_{hel}, the horizontal coil current corresponding to the central coil, I_{hor}, and the horizontal coil current corresponding to the vertical field coils, I_{vert}) have been varied. For calculation of the total stored energy two models for the particle density n have been applied. For one of these model n has been assumed constant, for the other one a parabolic profile $n = 1 - \alpha * (r/a)^2$, with $\alpha = 0.8$, has been assumed.

In Fig. 5 the cross section of the $\varphi = 0$ plane is plotted for the TJ-II "standard" configuration. Various configurations with an enhanced energy content have been found. For the "best" configuration the energy content is about 15% enhanced compared to the "standard" configuration. In Fig. 6 the cross section for the "best" configuration is shown. Comparing both cross sections an island chain close to the last closed surface, which is fully inside the vacuum vessel, can be seen at the plot for the "standard" configuration (see Fig. 5). For the configuration shown in Fig. 6 the island chain appears outside the last closed surface. In principle, these islands could be used for an island divertor.

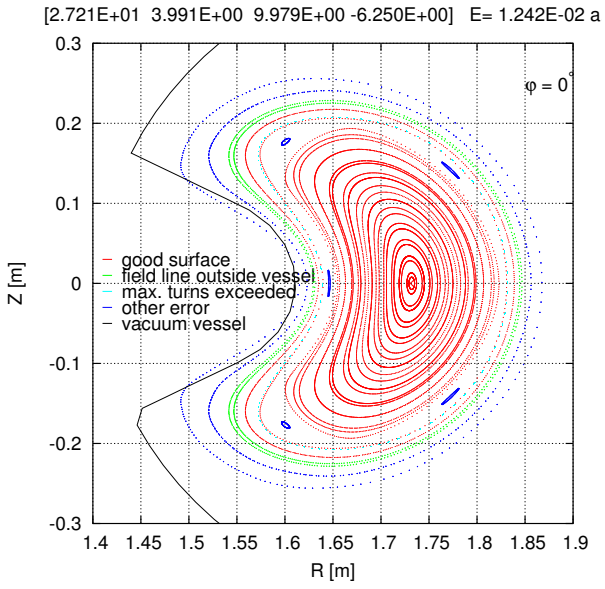

Figure 5. Cross section of the TJ-II "standard" configuration.

In Fig. 7 the values for the effective ripple, $\epsilon_{\rm eff}$, for both configurations are presented. It can be seen that for the "standard" configuration (green) $\epsilon_{\rm eff}$ is bigger than for the configuration with enhanced total stored energy thus indicating an enhanced transport across the flux surfaces for the "standard" configuration.

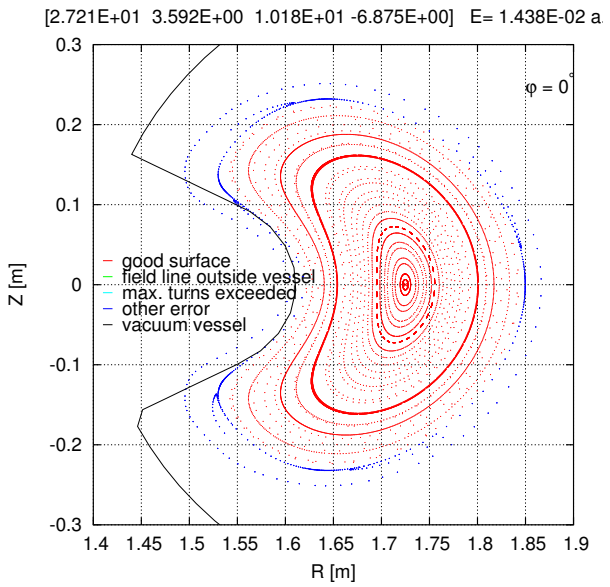

Figure 6. Cross section of a TJ-II configuration with enhanced total stored energy.

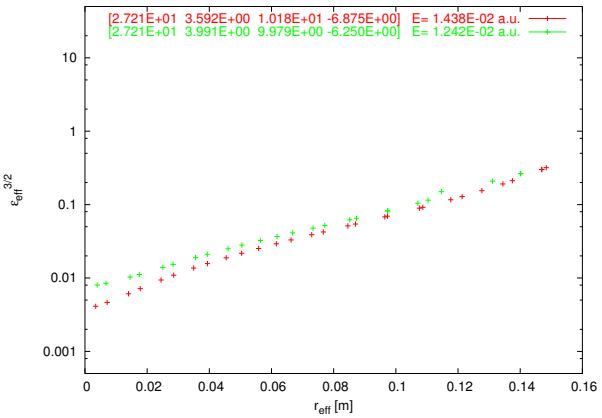

Figure 7. Effective ripple $\epsilon_{\rm eff}^{3/2}$ vs. effective radius $r_{\rm eff}$ for the TJ-II. red . . . configuration with enhanced total stored energy (see Fig. 6); green . . . "standard" configuration (see Fig. 5)

In figures 8 - 11 cuts for the total stored energy for both, the constant particle density profile (red) and the parabolic particle density profile (green), along the axes of the parameter space are shown. The "standard" configuration is marked with a blue cross. The positions of the blue crosses in figures 8 - 11 indicate that the TJ-II "standard" configuration is a good operation point since small changes of the coil currents do not strongly affect the energy confinement.

V. Summary

A new tool for optimizing existing stellarators, based on the technique for evaluating the effective ripple $\epsilon_{\rm eff}$ is presented and has been applied to TJ-II. The magnetic field computed directly from the coil currents is used for the computation of the effective ripple. Configurations with enhanced total stored energy in plasma have been found. Comparing the "best" found configuration with the "standard" configuration it can be seen that neoclassical transport across the flux surfaces is reduced.

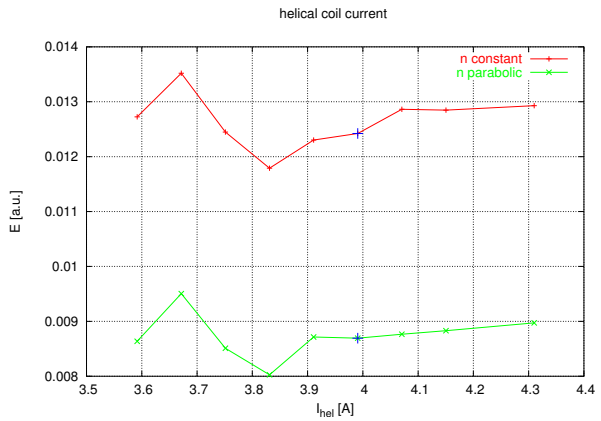

Figure 8. Normalized stored energy $\hat{W}$ for two particle density profiles n. Cut along the I_{hel}-axis of the parameter space. The "standard" configuration is marked with a blue cross.

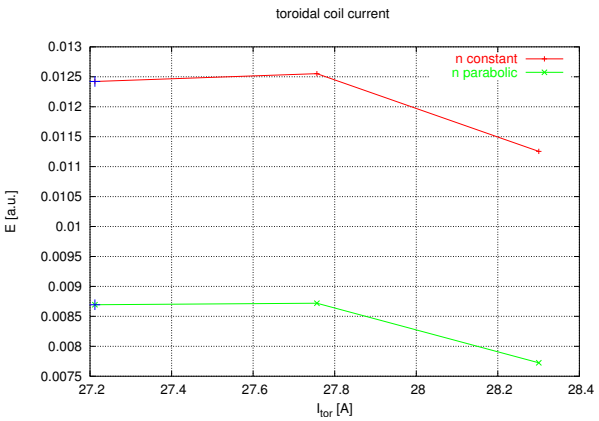

Figure 10. Normalized stored energy $\hat{W}$ for two particle density profiles n. Cut along the I_{tor}-axis of the parameter space. The "standard" configuration is marked with a blue cross.

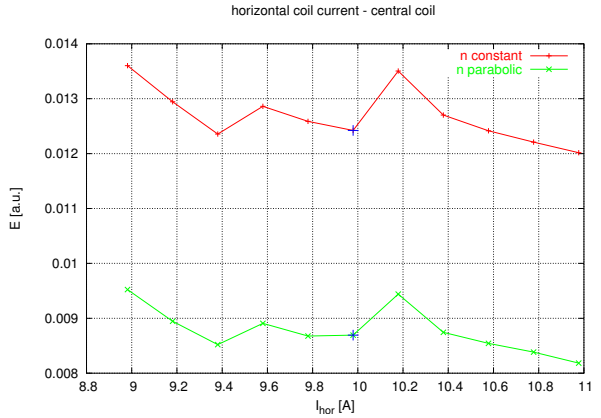

Figure 9. Normalized stored energy $\hat{W}$ for two particle density profiles n. Cut along the I_{hor}-axis of the parameter space. The "standard" configuration is marked with a blue cross.

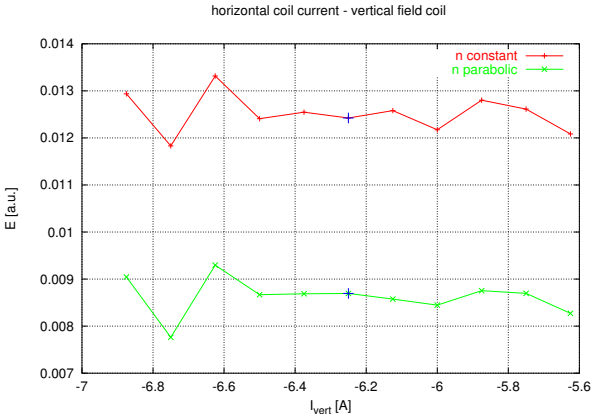

Figure 11. Normalized stored energy $\hat{W}$ for two particle density profiles n. Cut along the I_{vert}-axis of the parameter space. The "standard" configuration is marked with a blue cross.

VI. Acknowledgments

The authors wish to thank V. Tribaldos for providing necessary data for TJ-II and for discussions.

This work has been carried out within the Association EURATOM-ÖAW and with funding from the Austrian Science Fund (FWF) under contract P16797-N08. The content of the publication is the sole responsibility of its authors and it does not necessarily represent the views of the Commission or its service.

References

[1] H. E. Mynick, N. Pomphrey, "Control-matrix approach to stellarator design and control", Phys. Plasmas, 7, pp. 4960-4971, 2000.

[2] H. E. Mynick, N. Pomphrey, S. Ethier, "Exploration of stellarator configuration space with global search methods", Phys. Plasmas, 9, pp. 869-876, 2002.

[3] O. S.Pavlichenko for the U-2M group, "First results from the "URAGAN-2M" torsatron", Plasma Phys. Control. Fusion, 35, pp. B223-B230, 1993.

[4] C. D. Beidler, N. T. Besedin, V. E. Bykov, et al., "Physics Studies for the Uragan-2M Torsatron", Proc. of 13 IAEA Conf. on Nuclear Fusion, Washington, Vol.2, Vienna: IAEA, pp.663, 1991.

[5] C. Alejaldre, J. Alonso, L. Almoguera, et al., "First plasmas in the TJ-II flexible Heliac", Plasma Phys. Control. Fusion, 41, pp. A539-A548, 1999.

[6] S. P. Hirshman, W. I. van Rij, P. Merkel, "Three-dimensional free boundary calculations using a spectral Green's function method", Comp. Phys. Commun., 43, pp. 143-155, 1986.

[7] V. V. Nemov, S. V. Kasilov, W. Kernbichler and M. F. Heyn, "Evaluation of $1/\nu$ Neoclassical Transport in Stellarators", Phys. Plasmas, 6, pp. 4622, 1999.

[8] B. Seiwald, V. V. Nemov, S. V. Kasilov, W. Kernbichler, "Optimization of Stellarators with Respect to Neoclassical Transport in Real Space", Proc. of 29th EPS Conference on Plasma Phys. and Contr. Fusion, Montreux, 17-21 June 2002, ECA, 26B, P-4.099, 2002.

[9] V. V. Nemov, "Calculations of the Magnetic Surface Function Gradient and Associated Quantities in a Torsatron", Nucl. Fusion, 28, pp. 1727–1736, 1988.

[10] B. Seiwald, V. V. Nemov, V. N. Kalyuzhnyj, S. V. Kasilov, W. Kernbichler, "Optimization of neoclassical transport in URAGAN-2M", 31st EPS Conference on Plasma Phys. London, 28 June - 2 July 2004, ECA, 28B, P-5.118, 2004.

[11] S. Kirkpatrick, C. D. Gelatt Jr., M. P. Vecchi, "Optimization by Simulated Annealing", Science, Vol. 220, No. 4598, pp. 671-680, 1983.

[12] E. Aarts, J. Korst, "Simulated Annealing and Boltzmann Machines", Wiley, 1989.

[13] R. V. V. Vidal (Ed.), "Applied Simulated Annealing", Springer-Verlag, 1993.

[14] J. Schnakenberg, "Algorithmen in der Quantentheorie und Statistischen Physik", Zimmermann-Neufang, 1995.