Use of a Genetic Algorithm for Compact Stellarator Coil Design

William H. Miner, Jr. and Prashant M. Valanju

Fusion Research Center

The University of Texas at Austin

Steven P. Hirshman

Oak Ridge National Laboratory

Art Brooks and Neil Pomphrey

Princeton Plasma Physics Laboratory

December 14, 2000

A new global optimization technique for designing stellarator coils has been developed and applied to the design of coils for the National Compact Stellarator Experiment. Using this technique we have found coil sets with fewer coils and lower current densities than those obtained with traditional

methods. In this paper, we describe the new coil design procedure which uses a Genetic Algorithm as the core optimization method and present the resulting advanced coil designs.

1 Introduction

The National Compact Stellarator Experiment (NCSX) has been proposed to investigate the physics of high-beta, low-aspect-ratio, quasi-axisymmetric (QA) stellarator plasmas. An NCSX plasma can be regarded as a specific helical perturbation of an axisymmetric tokamak plasma. The required perturbation is such that on each flux surface the magnetic field strength, when expressed as a function of magnetic (Boozer) coordinates, is independent of toroidal angle. QA stellarators offer the combined advantages of tokamak neoclassical confinement levels and stellarator stability.

The first stage of NCSX design requires identification of a baseline plasma configuration with attractive physics properties. A three-field-period, aspect ratio 3.5 QA equilibrium that is stable to kink and ballooning modes at $\beta = 4\%$ has been obtained [1] using a configuration optimization procedure developed by Nührenberg and Zille [2]. Fourier coefficients of the plasma boundary shape were varied to minimize a target function which expresses the desired physics properties of the NCSX plasma. The target function includes values of the rotational transform at the magnetic axis and the plasma edge (global shear), a measure of the degree of quasisymmetry (goodness of

transport), measures of kink and ballooning stability, and boundary limits describing geometric constraints such as aspect ratio, maximum or minimum radius, and maximum plasma height. The VMEC [3] MHD equilibrium code computes a new equilibrium for each boundary modification, from which the optimizer target function can be evaluated. When one or more attractive equilibria have been identified the process of designing coils to support the equilibria can begin.

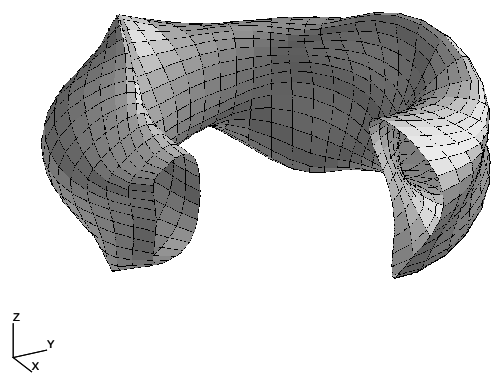

Figure 1: A three dimensional rendering of the C82 plasma boundary.

In this paper we will focus on a coil design for a baseline equilibrium named C82. The last closed flux surface of C82 is shown in Fig. 1. On this surface the coil set must impose the condition that the normal component of the total magnetic field must vanish,

$$\mathbf{B} \cdot \hat{\mathbf{n}} = 0 \ , \tag{1}$$

where

$$\mathbf{B} = \mathbf{B}_{plasma} + \mathbf{B}_{ext}. \tag{2}$$

Here $\hat{\mathbf{n}} = \nabla s/|\nabla s|$ is the unit normal to the plasma bounding surface, $s(x, y, z)$, $\mathbf{B}_{plasma}$ is the magnetic field due to currents which flow in the plasma, and $\mathbf{B}_{ext}$ is the magnetic field due to all external field coils. There is a possibility for building the NCSX device on the existing Princeton Beta Experiment (PBX) platform and of making use of the PBX toroidal and poloidal magnetic field coils. It is convenient, therefore, to write $\mathbf{B}_{ext}$ as the sum

$$\mathbf{B}_{ext} = \mathbf{B}_{tok} + \mathbf{B}_{stell}, \tag{3}$$

where $\mathbf{B}_{tok}$ is the (known) magnetic field from any given tokamak coils, and $\mathbf{B}_{stell}$ is the magnetic field from the stellarator coils whose design is being sought.

Following the NESCOIL [4] procedure developed by Peter Merkel, $\mathbf{B}_{stell}$ can be produced by a surface current, $\vec{j}$, on a current winding surface (CWS) that encloses the plasma. The surface current is divergence-free and can be written in terms of the surface gradient ∇_s of a current potential $\Phi(u, v)$:

$$\vec{j} = \nabla s' \times \nabla_s \Phi(u, v). \tag{4}$$

Here $s'(x, y, z)$ defines the shape of the CWS and u, v are poloidal, toroidal angles on that surface. Once the potential is determined [4] a set of coils can be defined by selecting N_c appropriate contours of Φ and interpreting each contour as a filamentary coil carrying an amount of current that is

proportional to the change in potential midway between each chosen contour and its two chosen neighbors. In the limit as $N_c \to \infty$ the discrete coil system reproduces exactly the magnetic field of the current sheet. For a practical coil system, however, we would like to choose an efficient coil set with the following minimum set of properties: (1) the number of coils should be small, (2) the reconstruction errors (measured by how well Eqn. 1 is satisfied) should be small, and (3) the maximum coil current should be small to minimize resistive dissipation. An optimization algorithm for choosing the "best" set of N_c contours is the primary topic of this paper.

Fig. 2 shows a contour plot of the NESCOIL current potential for C82 on a CWS (see Fig. 3) that is conformal with the plasma boundary, with each point of the CWS a normal distance of 18 cm from the plasma. In NCSX this normal separation is determined by engineering constraints: it is the smallest coil-plasma separation that allows room for plasma facing components, vacuum vessel, etc., in a R =1.5 m device. As the separation distance is increased, both the normal magnetic field error at the plasma boundary and the current sheet density on the CWS increase. The simplest algorithm for cutting coils from the current sheet is to choose N_c contours equally spaced in Φ, with equal current in each of the coils. This, in fact, was the algorithm used in coil designs for Helias and ATF[4]. However, the NCSX plasma has a smaller aspect ratio and is more highly shaped (due to the requirements of quasisymmetry and high-beta stability) and it becomes extremely beneficial to optimize the coil selection procedure: the number

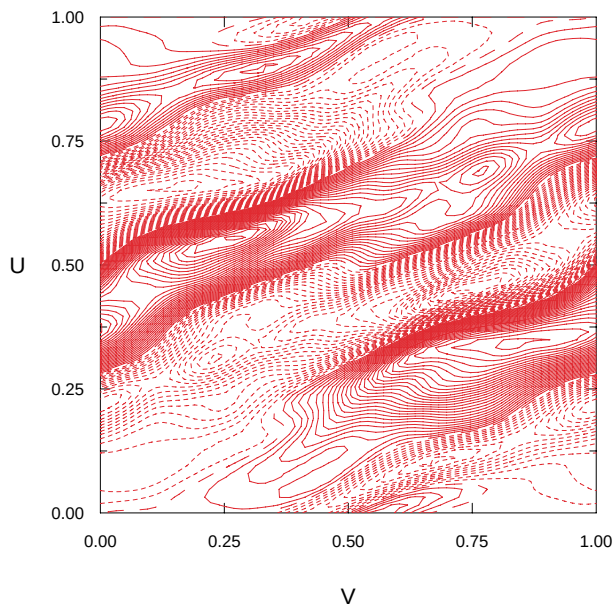

Figure 2: A contour plot of the surface current potential for C82. Solid lines represent contours of positive potential and dashed lines represent contours of negative potential.

of stellarator coils and the maximum current required by the coils can be significantly decreased.

Initially, an optimization code was written which selected coil contours based on minimizing a weighted sum of the squares of the net normal component of the magnetic field at the plasma boundary (which, ideally, should vanish) and the maximum current in the selected coils. A Levenberg-Marquardt algorithm [5] was used as the minimization procedure. Modest improvements in coil design were obtained when compared with the simple equipotential

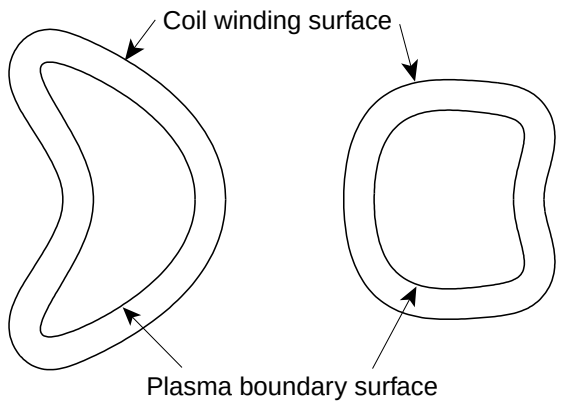

Figure 3: Cross sections of the plasma boundary and the coil winding surface it two toroidal planes.

spacing algorithm (smaller B-normal fitting errors and 10-20% smaller maximum coil currents for a fixed number of coils). However it became clear that the final Levenberg-Marquardt solutions were trapped in **local** minima of the minimizing target function and that considerable gains might be achieved if a **global** minimization algorithm were applied.

In the coil selection problem we are confronted with the daunting combinatorial task of selecting a relatively small number of potential contours (coils) from a large pool of contours (the full current sheet) and requiring that they reconstruct the plasma boundary with sufficient accuracy to reproduce the desired physics of the original equilibrium. A specific example will demonstrate the enormity of the problem. In Fig. 2 there are 60 equally spaced contours of the C82 current sheet. These produce 98 individual coils

(some levels are multivalued, as can be seen from the saddle region of the figure). If we seek a coil design with only 10 coils per field period, then all combinations of these 98 coils taken 10 at a time - which leads to

$$\frac{98!}{10!(98-10)!} = 1.4 \times 10^{13} \text{ combinations!}$$

- would have to be examined to guarantee finding a global minimum of the target function (minimum normal $\mathbf{B}$, together with any current density minimization).

Thus, the combinatorial optimization problem confronting us consists of a *state space*, a target *cost function* to be minimized, and a discrete but very large set of possible solutions. From previous calculations, we also know that the solution space exhibits many local minima. One approach to this problem is to apply a Genetic Algorithm (GA) [6] which is a non-gradient based optimization method that has been successfully applied to several problems of this type [7]. The GA works with a population of *individuals* each of which represents a possible solution to the optimization problem. Each individual is assigned a 'fitness' according to the value of the cost function. The fittest individuals, those with the lowest value fo the cost function, are allowed to reproduce by cross-breeding, thereby producing a new generation of individuals (population of new possible solutions) that contains a high proportion of the best characteristics of the previous generation. In this way, over generations, the good characteristics (low cost) are spread throughout the population and the most promising areas of the search space can be

explored in a computationally efficient manner.

2 The Genetic Algorithm

2.1 Overview

An overview of the steps in the GA is shown in Fig. 4. For further information and different applications the interested reader is referred to [8-12].

To apply the GA to the problem of cutting discrete coils for a stellarator, we proceed as follows, First the current potential $\Phi(u, v)$ defining a current sheet on a specified current winding surface is discretized into a large number of equally–spaced contours (eg., see Fig. 2). These contours represent the pool of *individuals* from which the final, much smaller subset of coils will be chosen. The GA will be used to decide which coils are to be retained, based on their evaluated *fitness* (matching B_{err}^{avg} and current density targets). Let N_Φ denote the total number of coil contours. Also, let N_c denote the number of coils desired in the final coil design. The GA will select the "best" N_c coils out of the available pool of N_Φ. Let a binary string with N_Φ bits describe the energized states of coils in the pool. The j'th element of the string describes the state of the j'th coil: if its value is "1" it is energized (retained with a finite current); if its value is "0" it is not. Any N_Φ bit pattern comprising N_c elements with value "1" and $N_\Phi - N_c$ elements with value "0" describes a coil system whose fitness to represent the final coil design can be tested. Each such binary string (bit pattern) is termed a *chromosome*.

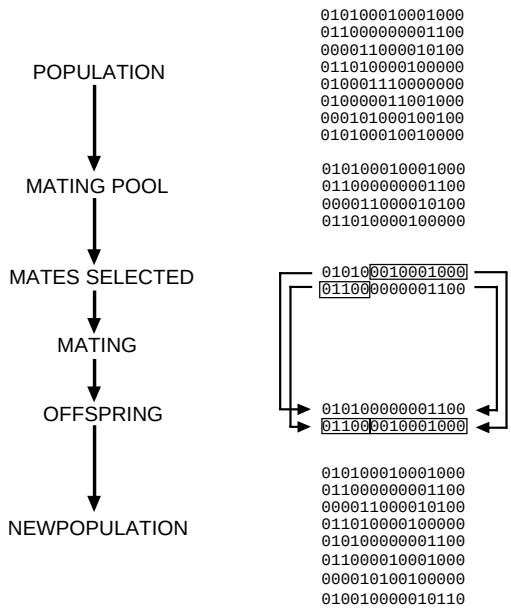

Figure 4: An overview of the steps in the GA. Mutation is not explicitly shown. If it occurs, it takes place before the new offspring are included in the new population.

An initial *population* of N_{pop} chromosomes is selected. Each chromosome has a random bit pattern. The *fitness* of each chromosome in the population is determined by optimizing the values of the currents in each energized coil so that a weighted sum of two quantities is minimized. The first quantity is B_{err}^{avg}, the r.m.s. normal component of the magnetic field error at the plasma boundary normalized by the local total magnetic field. This is a measure of how well the coil system defined by the chromosome can reconstruct the

designing baseline equilibrium. In practice, it has been found that $B_{err}^{avg} \lesssim 1\%$ is a requirement for adequate reconstruction. The second quantity is the maximum current, I_c^{max} in an energized coil. Engineering constraints require that $I_c^{max} \lesssim 15\text{kA/cm}^2$. Further details of the calculations of B_{err}^{avg} and I_c^{max} are given in Sections 4 and 6.

After the fitness of each chromosome has been ranked, the half of the population that is most fit is placed in a *mating pool* and the other half is discarded. Mating *parents* are chosen by tournament selection [9]. In this procedure, a subset of possible parents is randomly chosen from the mating pool. From this subset, the one with the best fitness becomes a parent. This procedure is repeated until the required number of parents is obtained. We have chosen a simple reproductive evolution process in which two parents create two offspring using a technique called *single point crossover*. Since each chromosome consists of N_Φ bits, each parent can be divided into two pieces at $N_\Phi - 1$ places. For each parent couple an integer random number is generated such that $1 \leq n_{ran} \leq N_\Phi - 1$. Each parent chromosome is then divided at that place in two pieces. The left part of the first parent chromosome is spliced with the right part of the second parent chromosome, and vice-versa, to form two offspring. Fig. 5 illustrates this form of reproduction for two chromosomes with $N_\Phi = 15$ bits, corresponding to a contour pool of fifteen coils of which four are energized.

Before the offspring are added to the population, the possibility of mutation during reproduction is considered. Let $0 \leq P_{mut} \leq 1$ be a chosen

	binary string
parent	0100100 10001000
parent	0101000 01000100
offspring	010010001000100
offspring	010100010001000

Figure 5: The mating process using uniform crossover.

probability that a mutation occurs. A random number $0 \leq P_{ran} \leq 1$ is generated; if $P_{ran} \leq P_{mut}$ a mutation occurs. If not the offspring are added to the population as is. Two types of mutation have been used in our implementation of the GA - *jump mutations* and *creep mutations* [13]. Random numbers $0 \leq P_{jump} \leq 1$ and $0 \leq P_{creep} \leq 1$ are generated and a jump/creep mutation occurs if $P_{ran} \leq P_{jump/creep}$. In a jump mutation two randomly located bits of an offspring chromosome are interchanged. In a creep mutation two neighboring bits are interchanged. Examples of the mutation processes are shown in Fig. 6. It is the diversity of reproduction in the GA inhibits trapping of the population in local minima.

The completion of the steps outlined in Fig. 4 constitutes a *generation*. The procedure is repeated, beginning with a new population, until a satisfactory solution is obtained or a stated maximum number of generations has been examined.

	binary string
jump mutation before	$0100\hat{1}0010001 0\hat{0}0$
jump mutation after	$0100\hat{0}0010001 0\hat{1}0$
creep mutation before	$01010000\widehat{1}000100$
creep mutation after	$01010001\widehat{0}000100$

Figure 6: Examples of the two mutation processes.

2.2 Fixing the Number of Coils in a Set

In the usual GA, the range of the variable represented by the binary string is set such that the mimimum value is given by the string consisting of all zeros and the maximum value is given by the string consisting of all ones. Therefore, in the usual GA any number of bits may be "on" in any chromosome. In the case of the coil selection process, a fixed number coils, N_c, is sought. Therefore, an additional step has to be taken in order to preserve this number.

After the GA has produced a new individual, it is checked to determine its length, i. e., the number of bits that are "on". If there are more than N_c bits "on" the chromosome should either be discarded or it must be modified. Since discarding the individual would then require an additional mating, and we would not be guaranteed that the new chromosome would be too long, we chose to modify the chromosome. The chromosome is modified using the following procedure. If a bit is "off", the next bits are checked until an "on"

bit is found. A random number $0 \leq P_{change} \leq 1$ is generated. If $P_{change} \leq .5$ the bit is turned "off" and if $P_{change} > .5$ it remains unchanged. Then, starting at the next bit the procedure is repeated until the number of bits that are "on" equals N_c. Note that if a chromosome has less than N_c bits "on" it is allowed in the population because coils sets with fewer coils are desirable.

3 Computational Details

3.1 The Inductance Matrix

Previous methods for finding stellarator coils have used a Fourier representation for the shape of filamentary coils on the CWS and a Fourier representation for the shape of the CWS itself [14]. The Fourier coefficients of both the filaments and the CWS are allowed to vary in an optimization loop which minimizes a combination of engineering constraints. Such techniques are very flexible, but the calculation of the magnetic field at the plasma boundary is computationally expensive. Each time the coils are moved the magnetic field on the plasma boundary must be recomputed using the Biot-Savart law.

For the GA, the coil pool from which the evolving population is constructed is fixed. An inductance matrix, $\mathbf{G}$, of dimension $N_{uv} \times N_\Phi$ relates currents in each of the N_Φ coils in the pool to the normal magnetic field at

N_{uv} points on the plasma boundary,

$$\mathbf{G} \cdot \mathbf{I} = \hat{\mathbf{n}} \cdot \mathbf{B}_{stell}. \tag{5}$$

The full $\mathbf{G}$ matrix can be calculated once outside of the optimization loop, then re-used within the loop when calculating the magnetic field on the plasma boundary due to each member of the population in each generation. This simply requires compacting the full inductance matrix into an $N_{uv} \times N_c$ reduced matrix, $\mathbf{G}_c$, obtained by eliminating those columns of $\mathbf{G}$ which correspond to coils that are not energized. The magnetic field is then evaluated by the simple matrix multiplication

$$\mathbf{G}_c \cdot \mathbf{I}_c = \hat{\mathbf{n}} \cdot \mathbf{B}_{stell}. \tag{6}$$

3.2 Calculation of coil currents

After the GA has chosen to energize a particular coil subset, the current in each of these N_c coils must be evaluated. The choice is that vector $\mathbf{I}_c$ which minimizes

$$\chi^2 = |\mathbf{G}_c \cdot \mathbf{I}_c - \hat{\mathbf{n}} \cdot (\mathbf{B}_{tok} + \mathbf{B}_{plasma})|^2. \tag{7}$$

The solution to this subsidiary minimization problem is obtained by singular value decomposition(SVD) [5], giving

$$\mathbf{I}_c = \mathbf{V} \cdot [\mathrm{diag}(1/w_j)] \cdot (\mathbf{U}^T \cdot \hat{\mathbf{n}} \cdot (\mathbf{B}_{tok} + \mathbf{B}_{plasma})), \tag{8}$$

where

$$\mathbf{G}_c = \mathbf{U} \cdot \mathbf{w} \cdot \mathbf{V}^T \tag{9}$$

is the SVD of the compressed inductance matrix. $\mathbf{U}$ and $\mathbf{V}$ are $N_{uv} \times N_c$ and $N_c \times N_c$ unitary matrices (eigenfunctions, respectively, of $\mathbf{G}\mathbf{G}^T$ and $\mathbf{G}^T\mathbf{G}$), and $\mathbf{w}$ is a $N_c \times N_c$ diagonal matrix of "singular values" w_j (the square roots of eigenvalues of $\mathbf{G}\mathbf{G}^T$ and $\mathbf{G}^T\mathbf{G}$).

4 Results

4.1 Coil studies

In general, the current potential on a prescribed CWS enclosing a chosen plasma can be written as the sum of a secular and a periodic contribution[4]:

$$\begin{aligned} \Phi(u,v) &= c_t u + c_p v \\ &+ \sum_{m=0,M,n=-N,N}^{M,N} \Phi_{mn} \sin 2\pi(mu + nv). \end{aligned} \tag{10}$$

Non-zero secular coefficients c_t and c_p correspond to coil topologies with net poloidal and toroidal currents (e.g., stellarator modular coils or wavy PF coils). A saddle coil design, with $c_t = c_p = 0$ is being considered for NCSX. For this design the toroidal flux requirements of the plasma are provided by external tokamak field coils. The saddle coils provide the required stellarator magnetic field for producing the rotational transform.

We seek a saddle coil design to support an equilibrium configuration named C82 (see Fig. 1). The major plasma parameters for C82 are $<R>= 1.45m, <a>= 0.42m, \beta = 4\%, I_p = 200kA, B_T = 1.2 - 2T$. A current sheet solution was determined on a CWS conformal with the plasma boundary and

separated from it by a normal distance of 18 cm. $N_\Phi = 60$ contours of the current potential are shown in Fig. 2. These contours form the coil pool for the GA.

4.2 Benchmark for Comparison of GA Results

In order to evaluate the effectiveness of the GA, it is useful to establish a benchmark with which to compare the GA results. We begin by noting two criteria that are needed for an efficient coil design. The first is an engineering constraint: in order to operate the coils reliably and cool them effectively, the coil current density must be kept below $15kA/cm^2$. This puts a constraint on the minimum allowable separation between coils. The second criterion is a physics constraint: the field produced by the coils must provide a satisfactory match to the field from the target equilibrium in order for that target to be *reconstructed*. The average error is

$$B_{err}^{avg} \equiv \frac{1}{N} \sum_{i=1}^{N} |\mathbf{B}_i \cdot \hat{\mathbf{n}}| / |\mathbf{B}_i| \tag{11}$$

where N is the total number of points at which the field is calculated on the plasma boundary. In practice, it has been found that the average fitting error, B_{err}^{avg} must be $\lesssim 1\%$ for an NCSX plasma.

The benchmark was obtained by taking a fixed number of current potential contours, equally spaced in Φ, with equal currents in all coils, and finding the minimum numbers of coils required to get $B_{err}^{avg} < 1\%$. In order to satisfy this limit, 26 coils per field period were required, carrying a max-

imum current of $14.7kA/cm^2$. This result is tabulated in Table 1, where, for comparison, we also show the fitting error obtained from the continuous NESCOIL current–sheet solution (a lower bound for any discrete coil calculation) and the current–sheet current density. Table 1.

Source	Solution	$B_{err}^{avg}(\%)$	$B_{err}^{max}(\%)$	$J^{max}(kA/cm^2)$
NESCOIL	Current Sheet	0.22	2.0	
Equi-$\Delta\Phi$	13 coils per half period	0.95	7.0	14.7

Table 1: NESCOIL current sheet solution and results using 13 coils per half period, chosen equally spaced in Φ.

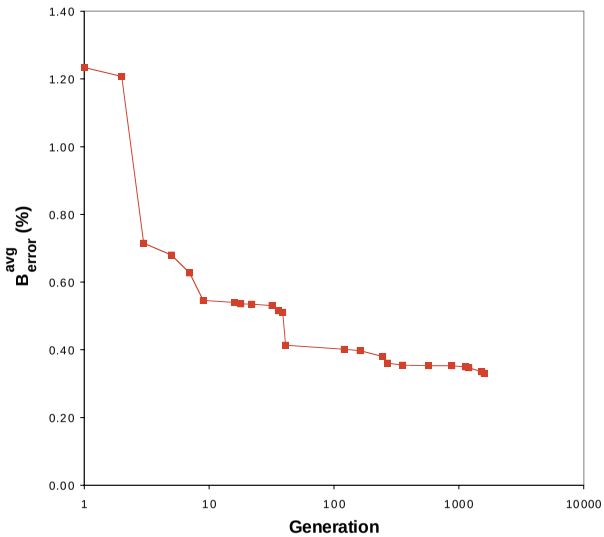

Figure 7: Average error obtained by the GA as a function of the number of generations for 12 coils per field period.

Our initial GA calculation selected 24 coils per period and ran for 3000 generations. The results are shown in Fig. 7. Although 3000 generations were computed, no better solution was found after generation 1600, the last point on the curve. The GA found a solution with average $B_{err}^{avg} = 0.33\%$ and maximum $B_{err}^{max} = 2.55\%$ an improvement over the similar equally spaced coil set of nearly a factor of three. There are two important points to be noted in this figure. First is the rapid rate of convergence, the number of generations is shown on a logarithmic scale. Second is the accuracy attained. As previously mentioned, good reconstructions have been obtained with errors $\lesssim 1\%$, and the GA has found solutions well below that limit.

The goal of this work is to find the minimum number of coils with current densities below the engineering limit that can still reproduce the magnetic field at the plasma boundary accurately enough that the target plasma can be reconstructed. The results shown in Fig. 7 suggest that we have sufficient margin in B_{err}^{avg} to decrease the number of coils and still meet the 1% error criterion.

4.3 Decreasing the number of coils

Depending on whether the original coil pool is obtained by contouring the current potential with an odd or an even number of contours, a helical coil associated with the contour value $\Phi = 0$ may or may not be present in the coil pool. The $\Phi = 0$ contour is shown in Fig. 8 for the three periods of the CWS. Each helical coil wraps around the plasma twice in the toroidal

direction (V) and three times in the poloidal direction (U) before joining back on itself. The helical coil is clearly a special topology among the saddle coils. Since its effect on reconstructing the target plasma is not a-priori known, it was deemed important to include – at least initially – the zero potential coil in the available pool.

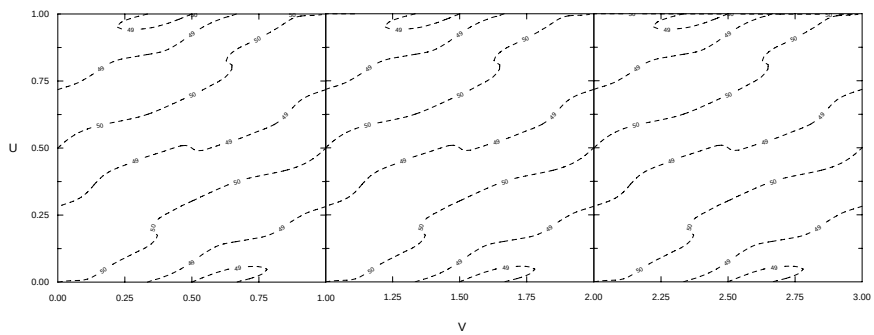

Figure 8: The two helical coils obtained from the zero value of the current potential.

Thus, beginning with the NESCOIL current potential we choose two sets of contours to provide the candidate coils, namely 60 and 61 contour levels. In each case, because of coil degeneracies previously mentioned in the Introduction, a total of 98 individual coils are in each pool. In the 60 contour case this leads to 49 independent coils; the other 49 are their stellarator–symmetric images. In the case of 61 contours, there are 50 independent coils; with 48 saddle coils and their stellarator symmetric images and the 2 helical coils described earlier.

We begin by targeting only B_{err}^{avg} in the cost function and ignore the

engineering constraint on the current density. The GA was asked to select 14 coils per period from the set of possible candidates that minimize B_{err}^{avg}. The GA ran for 3000 generations and found a set of coils that reduced B_{err}^{avg} to $\approx 0.5\%$, a little more than twice the theoretical minimum NESCOIL current–sheet solution error of 0.22%. The number of coils was then reduced by one and the process repeated until an the maximum acceptable error of 1% was exceeded. The results are shown in Fig. 9. For between 8 and 14 coils per period the error decreases almost linearly. Only in the case of 6 coils per period does the error exceed 1%. None of the final coil solutions contained a helical coil.

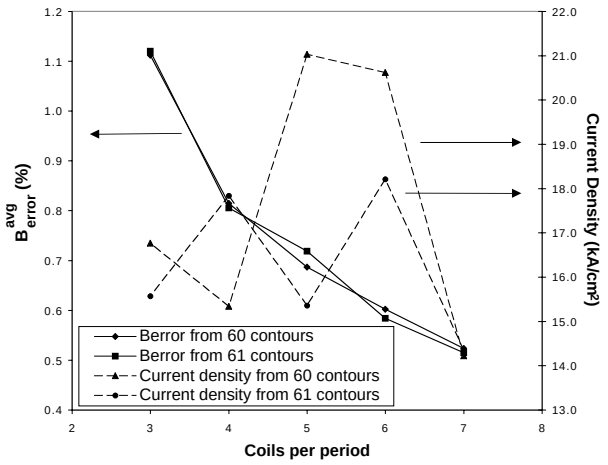

Figure 9: Average error and current density as a function of the number of coils per half period.

4.4 Targeting a combination of B_{err}^{avg} and I_c^{max}

Almost all of the coil solutions shown in Fig. 9 have maximum coil current densities, I_c^{max}, that exceed $15kA/cm^2$. To obtain coil solutions that satisfy both the physics reconstruction and engineering coil current density criteria it is necessary to target both B_{err}^{avg} and I_c^{max} in the GA cost function.

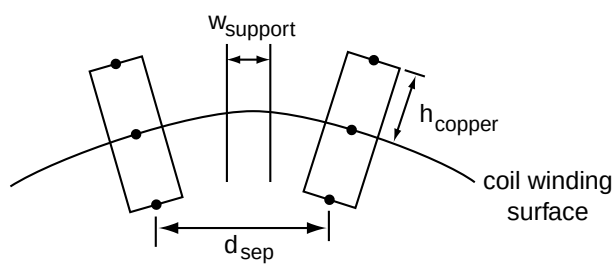

Figure 10: The minimum coil separation varies with the curvature of the current carrying surface.

In this paper, we use a crude estimate for the coil current density. It approximates way the finite extent ("build") of the coils, as well as the coil supports ("ligaments"). A schematic is shown in Fig. 10. Let i and j denote neighboring coils, carrying coil currents I_i and I_j. The coil centroids lie on the CWS and are separated by a distance d_{sep}. Neighboring coils are assumed to have the same width w, and same height h. Finally, let w_s denote the width of the coil support located between the coils labelled i and j. On the CWS there are as many neighboring coil pairs as there are individual coil members. An averaged coil current density is calculated for the coil pair ij

by averaging the currents over the adjacent coil blocks:

$$J_{ij} = \frac{0.5(I_i + I_j)}{h_{copper}(d_{sep} - w_{support})}. \tag{12}$$

The maximum coil current density is then

$$I_c^{max} = \max_{[ij]} J_{ij}. \tag{13}$$

Assumed coil build and support parameters are $h = 7.0cm$ and $w_s = 1.0cm$. The coil currents and separation distances are calculated by the GA for each candidate coil solution (population member).

The composite target cost function can we written:

$$\mathcal{C} = (1.0 - \alpha)B_{err}^{avg} + \alpha I_c^{max}/I_{norm}, \tag{14}$$

where I_{norm} is a scale factor used to normalize the magnitude of the current density so that it will be of the same order as the error. In the previous section we used $\alpha = 0$ and therefore the current density played no part in the determination of the error. Now, since we are looking for solutions with $I_c^{max} \leq 15kA/cm^2$, we scan α between 0 and 1. The lowest rms B-errors with current densities below this upper bound are shown in Fig. 11.

5 Conclusions

The GA has is an extremely useful tool for finding coils that meet the engineering constraints for a compact stellarator. The GA found coil sets with

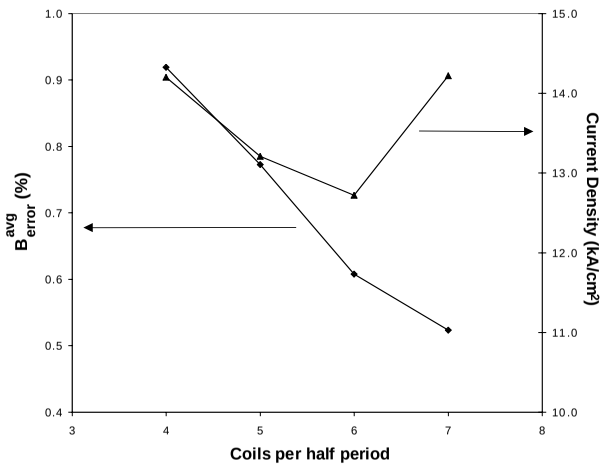

Figure 11: The minimum error obtained where $J_{max} \leq 15kA/cm^2$

both fewer coils per period and lower current densities than had been obtained using previous algorithms. The final results are compared to the constant $\Delta\Phi$, constant current case, in Table 2. The obtained from the the

Source	CPHP	$avg(B_{error})$	$max(B_{error})$	J_{max}
equal	13	.95	7.0	14.7
GA	7	.52	2.8	14.2
GA	6	.61	3.8	12.7
GA	5	.77	5.7	13.2
GA	4	.92	5.0	14.2

Table 2: Comparison of the constant $\Delta\Phi$, constant current case with the GA results.

GA also provide better access to the plasma by reducing the number of coils while still maintaining a constant value of B_{err}^{avg}. Figure 12 shows the coil set found by using the equally spaced contours (26 coils per period) and a coil set found by the GA (8 coils per period). The red surface represents the plasma boundary. In particular,note that the GA was able to eliminate the coil that previously obstructed access at the outboard mid-plane.

Acknowledgments

One author (WHM) would like to thank J. C. Wiley for suggesting the GA as a technique for solving this problem and to thank D. L. Carroll for a copy of his GA code which he has made available at http://www.staff.uiuc.edu/~carroll/ga.html. This work was supported by the USDOE under grant DE-FG03-95ER54296 and contract nos. DE-AC020-76-CHO3073 and DE-AC05-00OR22725.

References

[1] A. Reiman, et al., Plasma Phys. and Controlled Fusion, **41**, B273 (1999).

[2] J. Nührenberg and R. Zille, Phys. Lett. A **129**, 113 (1988).

[3] S. P. Hirshman, W. I. van Rij, and P. Merkel, Comp. Phys. Comm. **43**, 143 (1986).

[4] P. Merkel, Nucl. Fusion **27**, 867 (1987).

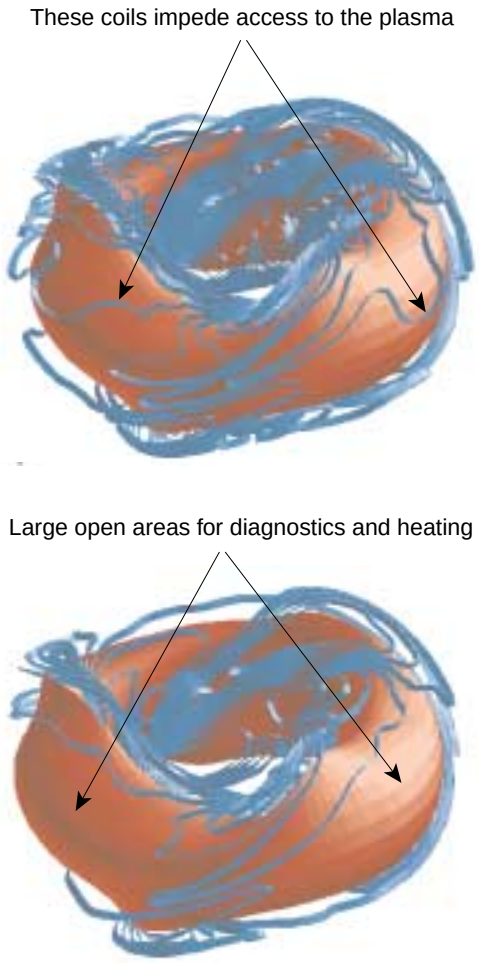

Figure 12: A comparison between the coil set found using equally spaced contours (the upper figure) and the coil set found by the GA (the lower figure), both having almost equal error and current density.

[5] W. H. Press, et al., *NUMERICAL RECIPIES The Art of Scientific Computing* (Cambridge University Press, Cambridge, 1990) p. 523.

[6] J. H. Holland, *Adaptation in natural and artificial systems : an introductory analysis with applications to biology, control, and artificial intelligence* (Ann Arbor, 1975).

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, (Addison-Wesley, 1989).

[8] M. Mitchell, *An Introduction to Genetic Algorithms* (The MIT Press, Cambridge, MA, 1996).

[9] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms* (John Wiley & Sons, Inc., New York, 1998).

[10] M. D. Vose, *The Simple Genetic Algorithm* (The MIT Press, Cambridge, MA, 1999).

[11] Z. Michalemicz, *Genetic Algorithms + Data Structures = Evolution Programs* (Springer, New York, 1996).

[12] E. Falkenauer, *Genetic Algorithms and Grouping Problems* (John Wiley & Sons, Ltd., Chichester, 1998).

[13] D. L. Carroll, AIAA J. **34**, 338 (1996).

[14] M. Drevlak, Fusion Tech. **33**, 106 (1998).