

**Lectures**

# **Weight Optimization of a Cruise Liner under Stress Restrictions**

Jörg Riedel

# Weight Optimization of a Cruise Liner under Stress Restrictions

Jörg Riedel (Bauhaus-University Weimar)

## Introduction

The search for an optimal design of a load structure is normally determined by several competing assessment criteria. In weight optimization light designs often conflict with stress limitations and stability criteria.

This paper is introducing the optimization of a cruise liner (figure 1). Steel should be economized by the variation of plate thicknesses, whereas the stresses as results of two loading cases are limited by acceptable stress values. Stability was not considered. The static calculations for the ship are complicated, because the ship is 300 meter long and has a fine structured load system. Beyond that, the optimization of 30000 single wall thicknesses is a difficult or even insolvable task for common optimization strategies. That is why the choice of an appropriate strategy is crucial for the success of the optimization. Therefore, self-regulating evolution strategies and a method for automatic grouping of the variables have been developed. The optimization and the parallel solver calls were implemented in the programm `SIANG` and the static calculations were executed in ANSYS. To guarantee an automatic procedure, a bidirectional interface between the programs had to be set up.

## Model

Within the ANSYS finite element model of the ship, the form, material, load structure, both load cases and the start design are determined. Inside the ship, plenty profiles and different steel plates with irregular distances exist. Normally, the distance between vertical structure elements is less than 8 meters. Compared to the total size of the ship, the ship has a fine structured, cellular load system. Both load cases are determined by regulations regarding the class and the length of the ship. It is assumed, that a swell with a wave length slightly shorter than the ship causes the worst bending stresses. The first load case describes the situation where the ship is carried by a wave ahead and rear, so that the middle part of the ship is sagging (figure 2). The second load case corresponds to the situation where the highest uplift takes place at the middle part of the ship and front and back of the ship are sagging, the so called hogging (figure 3).

For numerical reasons, the model has to be statically supported by some imaginary points. To avoid artificial reaction forces at these points, the acceleration field, which affects the weight loads of the masses, is manipulated in a way that uplift forces and mass forces compensate each other and the ship floats (see (ANSYS, 1998) 15.2 Inertial Relief). The manipulation of angle and path acceleration amounts approximately 1%. Therefore, the resulting inaccuracy can be neglected without loss of significance.

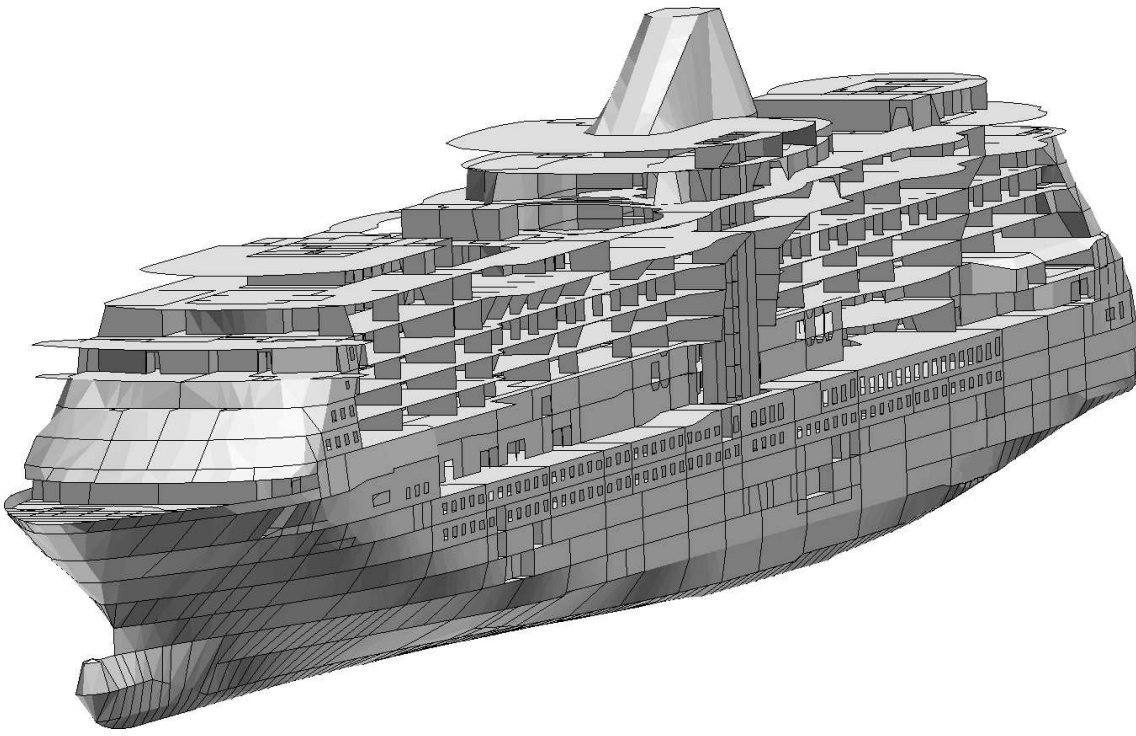


Figure 1. cruise liner

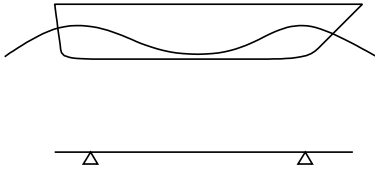


Figure 2. Loadcase Sagging

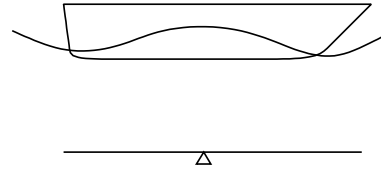


Figure 3. Loadcase Hogging

### Objective function of the optimization

The design vector  $\mathbf{t}$  of the optimization task consists of 30000 independent steel plate thicknesses  $t_i$ , each with an area  $A_i$ . The aim is to minimize the steel volume regarding to  $\mathbf{t}$ .

$$\min(Z(\mathbf{t})) = \min\left(\sum_i t_i A_i\right) \quad i = 1, 2, \dots, 30000 \quad t_i \in [t_{min}, t_{max}] \quad A_i \in R \quad (1)$$

The objective function  $Z$  is a linear combination of the variable  $\mathbf{t}$  and therefore describes a 30000-dimensional hyper plane. The absolute extrema for  $t_i = t_{min}$  or  $t_i = t_{max}$  are achieved trivially.

$$\begin{aligned}
\min(Z(\mathbf{t})) &= \sum_i t_i A_i = t_{\min} \sum_i A_i \\
\max(Z(\mathbf{t})) &= \sum_i t_i A_i = t_{\max} \sum_i A_i
\end{aligned} \tag{2}$$

The introduction of stress limits makes the optimization even more complicated, because the stresses depending on  $\mathbf{t}$  and resulting from different load cases have to be determined by numeric calculations. For each element  $i$ , following constraints have to be maintained:

$$\begin{aligned}
g_{1,i}(\mathbf{t}) &= |\sigma_{x,i}| < \sigma_{\text{zul}} \\
g_{2,i}(\mathbf{t}) &= |\sigma_{y,i}| < \sigma_{\text{zul}} \\
g_{3,i}(\mathbf{t}) &= |\tau_{xy,i}| < \tau_{\text{zul}}
\end{aligned} \tag{3}$$

where  $\sigma_x$ ,  $\sigma_y$  and  $\tau_{xy}$  are the membrane stresses at the centre of element  $i$ . Regarding the variables  $\mathbf{t}$ , the constraints are nonlinear. Thus, a change in the wall thickness of element  $i$  effects a rearrangement of stress and can cause violations of constraints in other elements. The hyperplane objective function is surrounded by these nonlinear restrictions.

### Automatic grouping of variables

In consideration of the size of the model, the required calculation time of nearly 2 hours per design and given project deadlines, the amount of calculations in advanced is limited to approx. 1000. In contrast, there are 30000 variable wall thicknesses to be optimized. All considered optimization strategies will need to gain information on the change of the objective value when certain variables vary. That is why it is obligatory to group the variables and to apply the variation on all variables of a group simultaneously.

The sensitivity of  $t_i$  can be interpreted as potential which violates the constraints of all elements, if the wall thickness  $t_i$  is differentially modified. With the help of this assumption, variables can be found which are robust regarding the constraints and which accordingly modify the design vector. Thus, for each variable  $t_i$  a complete statical calculation (30000 calculations) would be necessary, which is not realistic due to the time limitation of the project. Therefore, the definition of sensitivity was reduced in such a way that no additional numerical calculations were required.

Now, the sensitivity of the variable  $t_i$  describes the potential of a differential change of the wall thickness  $t_i$  that leads to the violation of the constraints of element  $i$ . The state of forces of element  $i$  is assumed to be constant, because the flow of the forces is rarely influenced by a change of the wall strength  $t_i$ . Thus, the sensitivity  $\zeta$  is defined as follows:

$$\zeta(i) = \left| \frac{d}{dt} t \max \left\{ \frac{\max(|\sigma_i|)}{\sigma_{\text{zul}}}, \frac{\max(|\tau_i|)}{\tau_{\text{zul}}} \right\} \right| \tag{4a}$$

The membrane stress is indirect proportional to  $t$  and therefore equation (4a) can be simplified:

$$\zeta(i) = \frac{\max \left\{ \begin{array}{l} \frac{\max(|\sigma_i|)}{\sigma_{zul}} \\ \frac{\max(|\tau_i|)}{\tau_{zul}} \end{array} \right.}{t} \quad (4b)$$

The grouping into 40 groups happens with decreasing sensitivity. The size  $G$  of a group  $j$  is determined nearly quite arbitrarily by the formula:

$$G(j) = \text{int} \left( \left( \frac{75 + 3j}{39} \right)^{4,93574} + 20 \right) \quad j = 1, 2, \dots, 40 \quad (5a)$$

The operator  $\text{int}(x)$  eliminates the post decimal positions of  $x$ , so that

$$\sum_{j=1}^{40} G(j) = 30000 \quad (5b)$$

### Self-regulating evolution strategy

Regarding other optimization strategies, evolution strategies (ES) have important advantages for this special task. The strategy is controlled by 3 mechanisms: selection, mutation and reproduction.

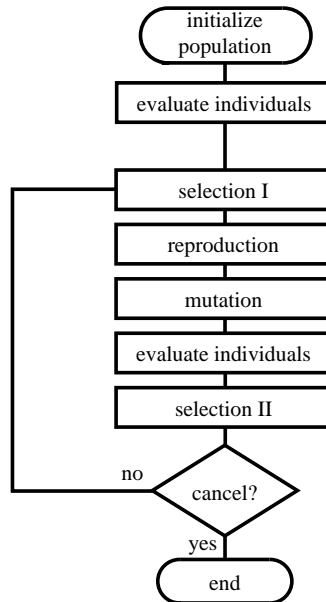


Figure 4. flowchart evolution strategy

The content of each module in figure 4 is not strictly determined and has to be adjusted with experience to the relevant task. More information about evolution strategies can be found in (J. Riedel, 1998), ((Michalewicz, 1994) and (E. Herz, 2000). In the following passage, the implementation of single modules for the optimization of the ship is explained.

### *Initialize population*

A population administrates the design vectors  $\mathbf{t}$  of several realizations of the ship. Amongst others, the size of the population, the amount of design variables, their borders and their start vectors are determined during the initialization. In this special adaption, two populations are administrated in parallel. The population  $P1$  consists of 4 realizations, 30000 design variables with  $t_i \in [t_{min}, t_{max}]$ . The four start designs of the first generation ( $n = 0$ ) are extracted out of the given numeric model and fulfil the constraints. The population  $P2$  is a reduced representation of  $P1$  which holds the first variable of each group of each designvector, which means 4 realizations with 40 representative design variables. For the self-regulation mechanism the vector  $\mathbf{r}$  is initialized with 0.

### *Evaluate individuals*

For each design vector  $\mathbf{t}$  from the population  $P1$ , the objective value  $Z(\mathbf{t})$  is calculated and the stresses of both load cases are achieved by a numeric calculation in ANSYS. On the basis of these stress results, the constraints are checked (eq. 3) and the sensitivities are calculated. The expensive numeric analyses are independent from each other, so that they can be done in parallel.

### *Selektion I*

All 4 design vectors of population  $P1$  are selected for reproduction.

### *Reproduction*

For selected vectors, off springs are created by using the uniform crossover method (L.J. Eshelman, 1989) and (E. Herz, 2000)). Thus, for every 2 design vectors, variables are exchanged with a probability of 50%.

### *Mutation*

Before the mutation, variables of the off springs are grouped according the best realizations regarding sensitivity, like described in the chapter “automatic grouping of the variables”. In each case, the first variable of a group represents the other variables. The generation  $n$  of population  $P2$  is generated by the  $4 \times 40$  representative variables of  $P1$ . Variables from the population  $P2$  are involved in the mutation mechanism with a probability of 7%.

The mutation mechanism for variable  $j$  is determined by a normal gaussian distribution with the mean value  $t_j$  and a standard deviation  $\sigma(j, f_n)$  according to equation 6.

$$\sigma(j, f_n) = f_n \frac{0,04(40 - j) + 0,20(j - 1)}{39} \quad (6a)$$

Factor  $f_n$  serves the self regulation and should prevent the creation of too many violating design vectors. As there are no experiences with self regulating ES, the equation,

$$f_n = f_{n-1} \cdot \begin{cases} 0,9 & \text{wenn } \sum_{k=n-5}^{n-1} r_k \geq 1 \\ 1,1\bar{1} & \text{sonst} \end{cases} \quad \text{mit } f_0 = 1,0 \quad (6b)$$

was used without further reason. New random values for the variables are generated by the normal gaussian distribution based on the standard deviation  $\sigma(j, f_n)$ . For the backward transformation, the differences between the representative variables  $t_j$  from  $P2$  and their original value in  $P1$  are ascertained. This differences are applied to all variables from group  $j$  in  $P1$ . Variables which exceed the domain  $[t_{min}, t_{max}]$  are reduced to the corresponding boundary.

### *Evaluate individuals*

Like stated above.

### *Selektion II*

Realizations that violate one or more constraints are eliminated from the population and their amount is registered in  $r_n$ . The best 4 out of the remaining realizations and those of the last generation are constituting the new population  $P1$  of generation  $n + 1$ .

### *Interruption*

In this case, the time for the optimization is restricted by the given dead line.

## **Results**

Beginning with the start design of the numeric model, figure 4 shows the process of the optimization. The beginning was satisfying, the optimization seemed to converge after 200 generations. But a detailed examination pointed out that the factor for the self-regulation mechanism  $f_n$  has become very small and the optimization was not able to escape a steady state. After a few experiments with the generations after 330, factor  $f_{390}$  was set back to 1.0. The further performance indicates that the self-regulation mechanism is strongly interfering with the optimization. The preliminary results after 748 generations and almost 3000 calculations shows, that a significant material savings potential still remains.

## **Acknowledgement**

The presented work could not have been done without *CAD-FEM*. Sincere thanks also to *Jörn Nelz* for the cooperation regarding the interfaces between the used programs.

## **Literature**

ANSYS (1998), *ANSYS Theory Reference*, SAS IP, Inc., online documentation.

E. Herz, O. Huth, e. a. (2000), *SLang* the Structural Language, Institut für Strukturmechanik, Bauhaus-Universität Weimar, Weimar.

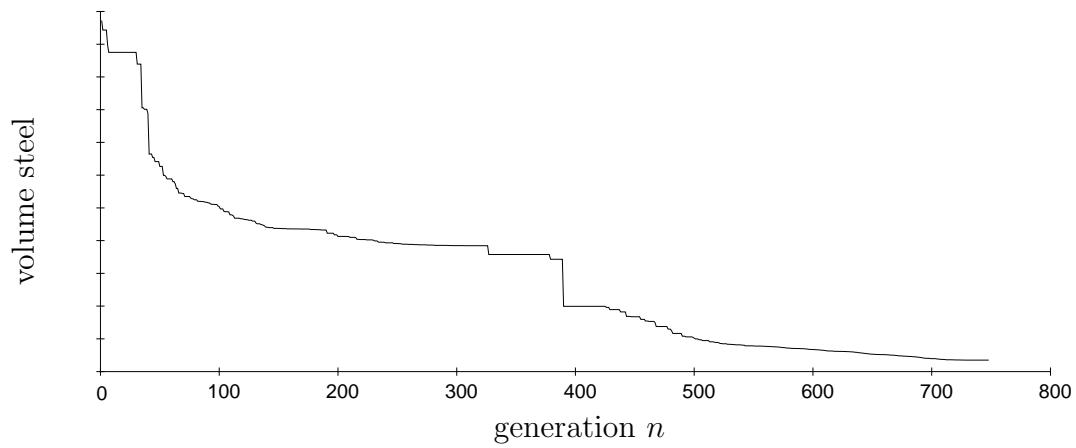


Figure 5. Verlauf der Optimierung

J. Riedel, D. Roos, e. a. (1998), *4. Institutskolloquim*, Institut für Strukturmechanik, Bauhaus-Universität Weimar, Weimar.

L.J. Eshelman, R.A. Caruana, J. S. (1989), "Biases in the Crossover Landscape," *Proceedings of the Third International Conference on Genetic Algorithms, ICGA'89*, 10–19.

Michalewicz, Z. (1994), *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer-Verlag, Berlin USA.