

A HEURISTIC SOLUTION FOR THE STATIC STABILITY PROBLEM IN CONVENTIONAL SUBMARINES

M. SINCLAIR

**Department of Industrial Engineering
University of Toronto
Toronto, Ontario M5S 1A4
CANADA**

and

M.M. WINSTANLEY

**Department of Mechanical Engineering
University of Stellenbosch
7600 Stellenbosch
SOUTH AFRICA**

ABSTRACT

Seafaring vessels must meet a few very obvious conditions in order to be of any use. The one condition is that they should stay afloat when loaded. The other condition is that they should float upright and with a level keel. The technical terms for these conditions are gravitational stability and trim. In practice some very precise numerical values are calculated for certain parameters in order to describe these conditions. After each refit of a submarine, the numerical values for the parameters referred to above must be recalculated. In the conventional submarines used by the South African Navy these new values must then be translated into the physical redistribution of weights housed in bays in the keel of the submarine. This paper discusses the problem of how this redistribution must be done in order to minimize the amount of work necessary. A heuristic algorithm is proposed by means of which a satisfactory redistribution may economically be computed.

1. INTRODUCTION

In order for any seafaring vessel to be effective, it must of course provide adequate buoyancy to give support for the ship and its contents. Furthermore, it must float in the proper attitude and remain upright when loaded with passengers and cargo.

This involves the problems of gravitational stability and trim [1], which will be discussed in this paper for a conventional class of submarine. *Trim*, as used in this paper, defines the longitudinal inclination of the vessel. The discussion will be limited to static conditions in calm water.

Before a submarine is submerged, considerable effort is expended to ensure that the following conditions are met:

- (i) The total weight of the submarine, with its loads and ballast, will be very nearly equal to the weight of the water it will displace when submerged. This condition is called *neutral buoyancy*.
- (ii) The center of gravity of the submarine will be very nearly in the same longitudinal position as the centre of buoyancy of the submerged submarine. This condition is called *zero trim*.
- (iii) The center of gravity of the submarine will be lower than the center of buoyancy of the submerged submarine. This condition is called *positive stability*.

A submarine on the surface, with loads adjusted and positioned so that the first two conditions will be met upon filling the main ballast tanks, is said to be in *diving trim*. It is not practicable to achieve exact balance of weight and buoyancy, or to bring the center of gravity precisely to the same longitudinal position as the center of buoyancy. It is also not necessary, since minor deviations can be corrected with the use of regulating tanks (see Figure 1). For an explanation of the principles involved in stability of vessels, see [5].

Some modern submarines have so-called *active trim-tank systems*, with the aid of which they can accept or reject sea water to obtain a level trim and a neutral submerged buoyancy for standard conditions. These submarines can thus be kept stable with their trim-tanks. However, the Daphne class submarines operated by the South African Navy have a closed circuit fresh water trim tank system. These submarines have regulating tanks, which are to allow for small deviations from the standard condition, e.g. extra crew members. The trim tanks cannot be used to

ensure stability after a major refit. Therefore, a further calculation must be done in order to determine the positioning of fixed weights in the keel such that the required trim and buoyancy after a major refit is achieved. These weights are located in 32 bays and their number and distribution in these bays are used to obtain the necessary stability (see Figure 1).

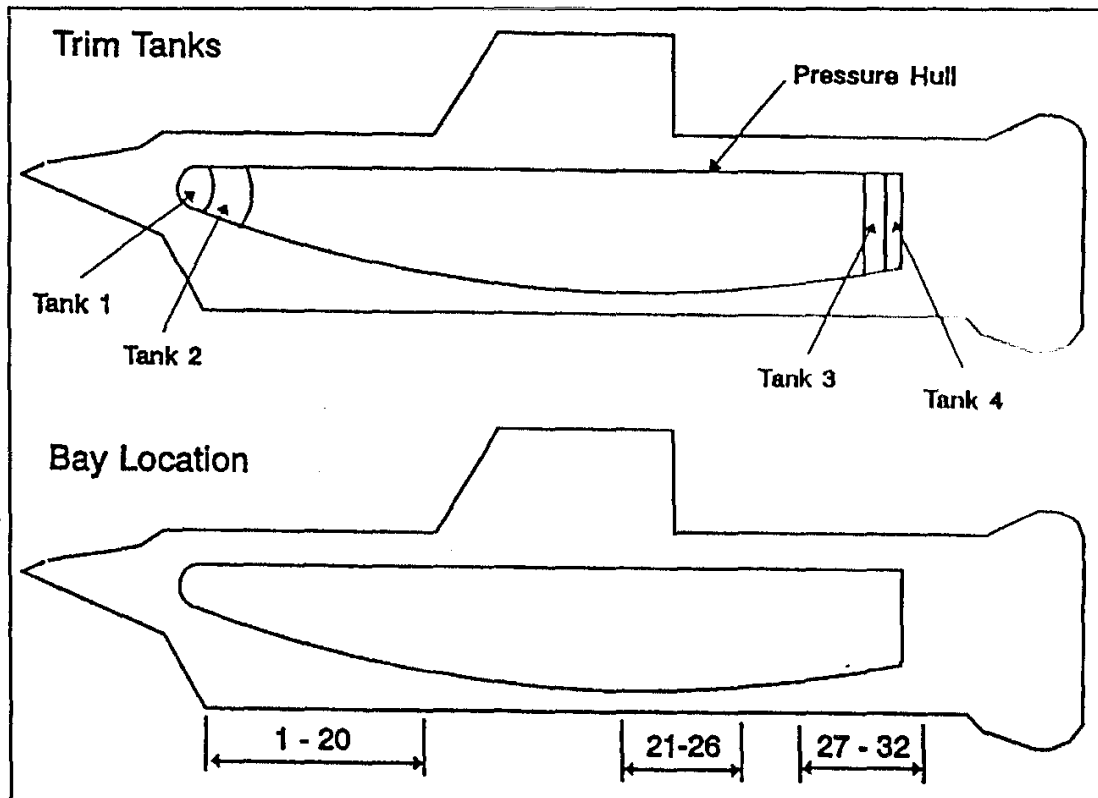


Figure 1: The trim tanks and bay locations.

After construction or every major refit of a submarine, a so-called "trim and inclining experiment" must be conducted to provide the basis for the stability calculations. This experiment is done to ensure that the stability characteristics of the submarine have remained the same as those prescribed by the original design conditions. From these experiments the necessary data are obtained in the form of corrections to mass and longitudinal moment, as well as the present configuration of weights in the 32 bays. The problem then becomes a combinatorial one, namely

How should the weights be redistributed in the bays in order to obtain the prescribed trim and stability, while minimizing the number of bays to be opened?

The cost of removal or insertion of the weights are also important. It was decided, however, that not much can be done about the number of weights that had to be shifted around. Therefore, the number of bays that had to be opened was the main objective of the study.

Until recently, the decision on which bay to open and how many weights to move from one bay to the other, was taken by a process of trial and error. No structured approach was implemented. This paper will describe the mathematical formulation of the problem, a heuristic solution method and the final results for one instance of the problem.

2. THE WEIGHT DISTRIBUTION PROBLEM

Suppose a trim and inclining experiment had been done on a conventional submarine of the type described above. Then it is known what correction is needed for the mass of the submarine, as well as in the moment about one of the endpoints of the submarine, usually the aft (back) end, in order to bring the submarine to stable equilibrium. The question then is: How should the weights presently in the weight bays be shifted in order to ensure that the resulting change in mass and moment will ensure that there is an equilibrium moment about the endpoint chosen? This redistribution of weights must be done in such a way that the lowest number of weight bays has to be opened.

The traditional operations research approach to the solution of combinatorial optimization problems such as the one defined above is to construct a linear integer programming model. Although this approach turned out not to be the best one for the problem under discussion, such a model does provide useful insight into relationships in the system under investigation. Since we are not going to solve the model in the end, we shall present a scaled down version of the original model in order to facilitate the discussion on the best approach to the solution our

problem. The following definitions are needed for this model formulation:

Parameters:

- M: required extra moment about the chosen endpoint of the submarine, determined through the trim and inclining experiment.
- N: number of weights in the bays. This ensures the desired mass correction.
- r: percentage by which final extra moment may differ from required moment M.
- u_i : maximum number of weights allowed in bay i.
- m_i : moment of one weight in bay i about the chosen endpoint of the submarine.
- p_i : current number of weights packed into bay i.
- n: number of bays.

Variables:

- x_i : number of weights packed into bay i.
- d_i : 0/1 variable indicating whether bay i must be opened (= 1) or not (= 0).

MODEL P:

$$\text{Minimize } \sum_{i=1}^n d_i$$

Subject to:

$$M(1-r) \leq \sum_{i=1}^n m_i x_i \leq M(1+r) \dots \dots \dots (1)$$

$$\sum_{i=1}^n x_i = N \dots \dots \dots (2)$$

$$p_i - x_i \leq u_i d_i, \text{ for all } i$$

$$x_i - p_i \leq u_i d_i, \text{ for all } i$$

$$0 \leq x_i \leq u_i, \text{ integer, for all } i$$

$$d_i = 0/1, \text{ for all } i.$$

For the practical problem under consideration, model (P) would have been large and therefore difficult to solve routinely on a personal computer. After consultation with the potential users, it was decided not to follow this route, for the following reasons:

1. Sophisticated hardware and software would have been necessary for this approach. Neither the funds, nor the expertise to make full use of such an approach was available on site.
2. The aim was to incorporate the solution to this problem into a decision support system, which was written in Pascal. It is difficult to incorporate a sophisticated commercial software package into such a decision support system. Furthermore, it would not have been cost effective to buy an expensive commercial package and then use it only in a rather unsophisticated manner. Personal experience with some of the less sophisticated, and therefore less expensive software packages, made us decide not to stake our reputation on them!

In view of these reasons, it was decided to develop an inexpensive, tailor made heuristic solution method, to be discussed in the next section.

3. HEURISTIC SOLUTION METHOD

Many heuristic solution methods exist for a variety of combinatorial optimization problems such as the one formulated in the previous section. One of the authors has recently completed a study of such methods [2] and could thus make an informed decision on which approach to choose for the problem under consideration. The criteria used in this decision were: Objective function value, CPU time used and ease of implementation. Taking all of these into account, it was decided to use the Great Deluge Algorithm (GDA, [3]).

The GDA was originally designed for the maximization of unconstrained objective functions. The GDA for constrained minimization problems such as the one under consideration can be summarized as in Figure 2.

It is obvious that the GDA is easy to implement on a computer. The details of the computer implementation will be explained with reference to Figure 2.

```
1. Choose:
    initial feasible solution;
    F := function value(initial solution);
    UP > 0;
    LEVEL > 0;

2. REPEAT
    Choose new solution := small stochastic perturbation of old solution;
    IF new solution feasible THEN
        E := function value(new solution);
        IF E < LEVEL THEN
            old solution := new solution;
            LEVEL := LEVEL - UP*(LEVEL - E);
            F := E;
        END IF
        Keep track of best solution found so far;
    END IF;
UNTIL (for long time no improvement in function OR too many iterations).
```

Figure 2: GDA for minimization.

The choice of a heuristic for the solution of the weight allocation problem led to a substantial simplification in the formal formulation of the problem. Most of the variables and constraints in the formulation in the previous section were necessary only to enforce various logical constraints. When a heuristic is used, such explicit formal representation of logical constraints becomes unnecessary.

In the mathematical model in the previous section, the variables d_i are used to count the number of bay doors to be opened. These variables, and all the

constraints necessary only to ensure their correct values, can be eliminated by changing the objective function slightly. This can be achieved if, instead of minimizing the number of doors to be opened, we minimize the total deviation of the current weight distribution (represented by the variables x_1, x_2, \dots, x_n) from the original weight distribution (represented by the known numbers p_1, p_2, \dots, p_n). The new objective function used by the heuristic was:

$$\sum_{i=1}^n (x_i - p_i)^2$$

It is obvious that as the current and new packing arrangements approach each other, the objective function value is reduced. In addition to the reduction in the number of variables in the model, the choice of this quadratic objective function was also preferred because previous experience with the GDA indicated that the heuristic worked very well for such objectives.

In order to ensure that the solution being considered is feasible (see the test in the second line of step 2 in Figure 2), constraints (1) and (2) must be tested, as well as the simple constraints $x_i \leq u_i$ for all i . However, it is not so straightforward to generate the initial feasible solution. The following procedure was used to do this:

1. Fill all bays from aft with the required number of weights (N), filling each bay to capacity before placing weights in the next bay. If the moment of these weights about the aft end exceeds the upper bound in constraint (1) moment, then no solution exists for the problem. STOP. Else, go to step 2.
2. Repeat until the moment of the weights satisfy constraint (1):
 - Shift one weight at a time further forward according to the following scheme:
 - First try shifting a weight between two bays with the smallest lever difference (difference in m_i values). If this is not possible, try the two bays with the second smallest lever difference, etc.
3. If step 2 fails to determine a solution satisfying constraint (1), then no

feasible solution exists. STOP.

The computations in this procedure were simplified by arranging the lever differences from smallest to largest before the time.

A stochastic perturbation of a current feasible solution (see Figure 1) was effected as follows: Choose two bays at random and call them bays number 1 and 2. If both bays are empty, abandon the effort and choose another two bays at random. If only one of the bays (say bay 1) contains weights, effect a perturbation by shifting a weight from bay 1 to bay 2. If both bays contain weights, shift a weight from one, chosen at random, to the other. An additional constraint was placed on the random bay selection, namely that once the number of weights in a bay equalled p_i , the number of weights in the bay could not be altered again. With this extra constraint the algorithm converged much more quickly to a local minimum, while the objective function values obtained were very good.

Initial values for the remaining parameters (see step 1 in Figure 2) were chosen as shown in Table 1:

Parameter name	Parameter value	Description
UP	0,01	Rate of decrease of water level.
LEVEL	100000	Initial water level .
Maxex	35000	Maximum number of iterations.

Table 1: Initial values of parameters.

The parameter values in Table 1 were chosen after some experimentation with the algorithm. Previous experience with the solution of problems with quadratic objective functions also played a role in this choice. However, no full scale parameter study was done.

The algorithm was coded in Turbo Pascal 6 and run on an ordinary IBM compatible personal computer (80286 processor). On a set of real data, the following comparison can be made between the previous trial and error approach and the results obtained with the heuristic:

Trial and error solution: 6 doors opened out of 32. Time required: 3 hours.

Heuristic solution: 7 doors opened out of 32: Time required: 5 minutes.

4. SUMMARY AND CONCLUSIONS

We studied the problem of arranging the weights in the keel of a conventional submarine after a refit. After formulating the problem as a linear integer problem, we decided on practical grounds to develop a heuristic solution method. The heuristic forms part of a decision support system which includes the trim and inclining experiment [4]. The decision support system was validated by using real data obtained from the South African Navy. The users of the decision support system expressed their satisfaction with the results obtained by the heuristic.

We expected resistance from the users on the fact that the heuristic is non-deterministic in the sense that it will not always supply the same result for the same set of data. To our surprise, they preferred to rerun the program a few times in order to generate a few alternative solutions, which left them with the option of making the final decision. The obvious reason for this is that there are some aspects of the problem that cannot be modelled and thus do not influence the solution generated by the heuristic.

The final lesson learnt from this project is that customers can gain much from a simple custom made method which fits in with their existing systems as opposed to a sophisticated software package which is difficult to integrate with their current systems. Our experience is that some consultants need to consider this option carefully when advising clients.

REFERENCES

- [1] L.L. GOLDBERG, Intact stability, in *Principles of Naval Architecture, Volume I: Stability and Strength* (E.V. Lewis, Ed.), The Society of Naval Architects and Marine Engineers, Jersey City, NJ, 63 - 142 (1988).
- [2] M. SINCLAIR, Comparison of the performance of modern heuristics for combinatorial optimization on real data. *Computers and Operations Research*, **20**(7), 687 - 695 (1993).
- [3] G. DUECK, New optimization heuristics: the Great Deluge algorithm and the Record-to-Record Travel, *Journal of Computational Physics*, **104**, 86 - 92 (1993).
- [4] M.M. WINSTANLEY, *Trim and stability computer program for conventional submarines*, Unpublished Final Year Thesis, Department of Mechanical Engineering, University of Stellenbosch (1992).
- [5] K.J. RAWSON and E.C. TUPPER, *Basic Ship Theory*, Longmans, London (1968).