

NO-A176 882

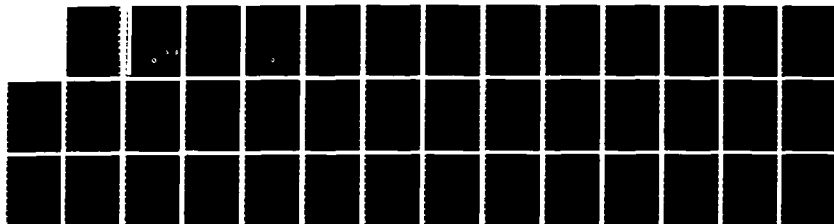
MODHLL: A PROGRAM FOR MODIFYING COMPUTER
REPRESENTATIONS OF SHIP HULLS(U) DEFENCE RESEARCH
ESTABLISHMENT ATLANTIC DARTMOUTH (NOVA SCOTIA)
D HALLY ET AL DEC 86 DREA-TM-86/224

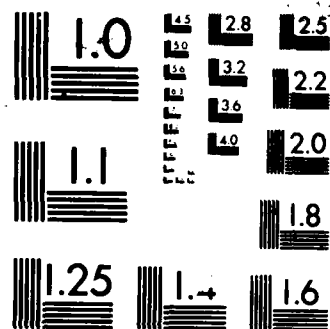
1/1

UNCLASSIFIED

F/G 13/18

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNLIMITED DISTRIBUTION



National Defence
Research and
Development Branch

Défense Nationale
Bureau de Recherche
et Développement

TECHNICAL MEMORANDUM 86/224

December 1986

AD-A176 802

MODHLL:
A PROGRAM FOR MODIFYING COMPUTER
REPRESENTATIONS OF SHIP HULLS

DAVID HALLY - NORMAND GILBERT

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited

DTIC
ELECTE
FEB 19 1987
S D

FILE COPY

Defence
Research
Establishment
Atlantic



Centre de
Recherches pour la
Défense
Atlantique

Canada

88 2 19 037

DEFENCE RESEARCH ESTABLISHMENT ATLANTIC

9 GROVE STREET

P.O. BOX 1012
DARTMOUTH, N.S.
B2Y 3Z7

TELEPHONE
(902) 426.3100

CENTRE DE RECHERCHES POUR LA DÉFENSE ATLANTIQUE

9 GROVE STREET

C.P. 1012
DARTMOUTH, N.É.
B2Y 3Z7

UNLIMITED DISTRIBUTION



National Defence
Research and
Development Branch

Défense nationale
Bureau de recherche
et développement

MODHLL:
A PROGRAM FOR MODIFYING COMPUTER
REPRESENTATIONS OF SHIP HULLS

David Hally - Normand Gilbert

December 1986

Approved by B.F. Peters A/Director/Technology Division

DISTRIBUTION APPROVED BY

A/O/TD

TECHNICAL MEMORANDUM 86/224

**Defence
Research
Establishment
Atlantic**



**Centre de
Recherches pour la
Défense
Atlantique**

Canada

Abstract

Systematic series of hulls based on a parent hull have long been used by naval architects and hydrodynamicists to determine the effects of hull form on flow and resistance. This memorandum describes an interactive computer program, MODHLL, which creates modified hulls from an existing parent hull. Although the parent hull requires a large data file for its definition, each modified hull is defined by only a few extra words of data since only the way it differs from the parent need be defined. Thus a whole series of hull forms may be defined using only a little extra storage space than the parent.

The program is based on a generalization of Lackenby's method in which the hull is modified by moving the stations between the fore and aft perpendiculars. Each modified hull is specified by choosing values for the length, breadth, draft, depth and either

1. the fore and the aft prismatic coefficients, or
2. the total prismatic coefficient and the longitudinal centre of buoyancy.

Résumé

Un ensemble systématique de coques basées sur une coque mère est depuis longtemps utilisé par les ingénieurs des constructions navales et les spécialistes en hydrodynamique pour déterminer les effets de la forme des coques sur l'écoulement et la résistance. Ce mémoire décrit un programme d'ordinateur interactif, le MODHLL, qui produit des formes de coques modifiées à partir d'une coque mère existante. Quoique un imposant fichier de données soit nécessaire pour la définition de la coque mère, chacune des coques modifiées n'est définie qu'au moyen de quelques mots supplémentaires puisqu'il n'est nécessaire de définir que la manière dont elles diffèrent de la coque mère. Ainsi il est possible de définir tout un ensemble de formes de coques en n'utilisant qu'un peu plus d'espace de stockage qu'il n'en faut pour la coque mère.

Le programme est basé sur une généralisation de la méthode de Lackenby par laquelle la coque est modifiée en déplaçant les couples de tracé entre les perpendiculaires avant et arrière. Chacune des coques modifiées est spécifiée par le choix de valeurs pour la longueur, la largeur, le tirant d'eau, la hauteur et

1. les coefficients prismatiques avant et arrière ou
2. le coefficient prismatique total et le centre de carène longitudinal.

Contents

Abstract	ii
Table of Contents	iii
List of Figures	v
List of Tables	v
Notation	vi
1 Introduction	1
2 Coordinate Systems	2
3 Coordinate Transformations	3
4 B-Splines	6
5 Calculation of the ξ_i	8
6 Monotonicity of the Transformation	10
7 Calculation of X_B , ϕ_F , and ϕ_A	11
8 Representation of the Draft Waterline	12
9 Concluding Remarks	13
Appendix	14
A MODHLL User's Guide	14
A.1 Defining the Terminal Characteristics	15
A.2 Loading a Parent Hull File	15
A.3 Saving a Modified Hull	16

Accession For	
NTIS CRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

A.4	Modifying the Hull	18
A.4.1	Changes in the Draft	18
A.4.2	Changes which Affect $x'(x)$	18
A.5	Printing Hull Characteristics	20
A.6	Plotting the Curves of Sectional Areas	21
A.7	Plotting $x'(x)$	21
A.8	Modification of Splines	22
A.8.1	Changing the Amount of Smoothing	23
A.8.2	Changing the Data Point Errors	23
A.8.3	Changing the Stiffness Weights	24
A.9	Spline Trouble-shooting	25
B	Modifications to the HLLFLO Hull Representation	26
	References	28

List of Figures

1	The coordinate systems	3
2	B-spline representation	7
A.1	Spline of Normalized Curve of Sectional Areas	16
A.2	Curves of Sectional Area	21
A.3	Transformation curve for a modified hull	22

List of Tables

1	Roots and Coefficients for 4-Point Gaussian Quadrature	12
2	Summary of Suggested Changes in p_1 and p_2	20

Notation

$a, b, c, \dots, k, l$: coefficients of the polynomial segments defining the hull transformation

$A(x)$: the sectional area at station x

B : the half-breadth of the hull

$B_i(x)$: the i^{th} B-spline basis function of order 4

$B_{i,k}(x)$: the i^{th} B-spline basis function of order k

c_i : weights used in Gaussian quadrature

D : the depth of the hull

d_i : the derivative of the function at point x_i

e_i : the error associated with the i^{th} data point of a spline function

$f(x, y, \dots)$: a function of $x, y, \dots$

$f^x(x, y, \dots)$: first partial derivative of $f(x, y, \dots)$ with respect to x : $\partial f(x, y, \dots)/\partial x$

$g(x, y, \dots)$: a function of $x, y, \dots$

$g^x(x, y, \dots)$: the first partial derivative of $g(x, y, \dots)$ with respect to x : $\partial g(x, y, \dots)/\partial x$

k : the order of a spline

L : the length of the hull between perpendiculars

N : the number of B-splines used to generate a spline curve

p_1, p_2 : the breakpoints of the transformation $x'(x)$

p'_1, p'_2 : the stations on the original hull which correspond to p_1 and p_2 respectively

s : non-cartesian hull coordinate

$\text{sgn}(i)$: sign of i

t_i : the roots of the Legendre polynomial of fourth order

$w(x')$: function defining the waterline

x : station number

- X : cartesian hull coordinate
- X_B : longitudinal centre of buoyancy
- Y : cartesian hull coordinate
- $y(x, s)$: function defining the Y coordinate of the hull in terms of the non-cartesian coordinates (x, s) .
- Z : cartesian hull coordinate
- $z(x, s)$: function defining the Z coordinate of the hull in terms of the non-cartesian coordinates (x, s) .
- Z_d : the design draft of the hull
- Z_w : the draft of the hull
- $\alpha_i, \gamma_i, \Delta_i$: parameters used in the test for monotonicity of the transformation $x'(x)$
- β_i : spline coefficients for the waterline curve or curve of sectional areas
- ϕ_A : aft prismatic coefficient
- ϕ_F : forward prismatic coefficient
- ϕ : total prismatic coefficient
- χ^2 : the chi-square of a spline
- ξ_i : spline coefficients ($i = 1, 4$)

Note: Variables with primes refer to the parent hull: e.g. X'_B is the longitudinal centre of buoyancy of the original hull, while X_B is the longitudinal centre of buoyancy for the modified hull.

1 Introduction

Systematic series of hulls based on a parent hull have long been used by naval architects and hydrodynamicists to determine the effects of hull form on flow and resistance. Since computer programs are now used for calculating detailed hydrodynamic performance of hulls, an efficient means of generating computer representations of series of hulls is both useful and convenient for determining the relations between hull form and performance. The suite of programs HLLFLO[1], developed at DREA, may be used to calculate the flow around a ship hull; however, HLLFLO's original system for representing hulls in the computer[2,3] required the generation of a separate representation of each hull. This is not only a time consuming task, but also can require large amounts of disk storage since each hull is stored in a file containing several thousand words of data.

This memorandum describes an interactive computer program, MODHLL, which creates modified hulls from an existing parent hull. Each modified hull is defined by only 35 to 70 words of data (to be compared with the several thousand for the parent) since only the way it differs from the parent need be defined. Thus a whole series of hull forms may be defined using only a little extra storage space than the parent.

MODHLL is based on a generalization of Lackenby's method[4] in which a hull is modified by moving the stations between the fore and aft perpendiculars. Unlike Lackenby's method, MODHLL also allows the draft to be changed independently of the hull depth. Each modified hull is specified by choosing values for the length, breadth, draft, depth and either

1. the fore and the aft prismatic coefficients, or
2. the total prismatic coefficient and the longitudinal centre of buoyancy.

Modification of the draft leads to some confusion of terminology which is best dealt with at the outset. On the parent hull, the "draft" and the "design draft" are, by definition, one and the same; however, on a modified hull the "draft" and the "design draft" may be different. The draft of a modified hull is a value specified by the MODHLL user and may be used to account for changes in hydrostatic condition. On the other hand, the design draft of a modified hull is defined to be the design draft of the parent scaled by the relative change in the depth of the two hulls. That is,

$$Z_d = \frac{Z'_d D}{D'} \quad (1.1)$$

where Z_d is the design draft of the modified hull, Z'_d is the design draft of the parent, D is the depth of the modified hull, and D' is the depth of the parent. The "design waterline" is the waterline which yields the design draft; the "draft waterline" is the waterline which yields the draft.

All HLLFLO programs will accept the modified hull representations generated by MODHLL in lieu of the parent file; whether a hull is modified or is a parent is irrelevant to the remainder of the HLLFLO programs.

2 Coordinate Systems

The hull is embedded in a cartesian coordinate system whose axes are aligned with the waterlines and centreplane. The three cartesian coordinates are denoted (X, Y, Z) (see Figure 1) and are defined as follows.

- X increases from bow to stern. $X = 0$ is the forward perpendicular and $X = L$ is the aft perpendicular, where L is the length between perpendiculars. The planes of constant X are hull sections.
- Y increases from the centreplane to the point of maximum breadth. $Y = 0$ is the centreplane. Owing to the symmetry of the hull, it is usually not necessary to decide whether Y increases to port or to starboard; however, to maintain a right-handed coordinate system, Y should increase to starboard. Surfaces of constant Y are parallel to the centreplane.
- Z increases from the baseline to the point of maximum height above the baseline. The surfaces of constant Z are waterplanes. $Z = 0$ is the waterplane which contains the baseline.

The computer representation of the hull requires a two dimensional coordinate system (x, s) which specifies points lying on the surface of the hull. Given a point (x, s) on the hull surface, one must be able to calculate the cartesian components (X, Y, Z) of that point. X, Y , and Z can then be regarded simply as functions of x and s . In the system currently used at DREA[2,3], x and X are simply proportional to one another:

$$X(x, s) = \frac{xL}{20} \quad (2.1)$$

where x runs from 0 at the bow to 20 at the stern and is called the station number. The coordinate s runs around the side of the ship and has values in the range $[0, 1]$. The keel always has the value $s = 1$, but the line $s = 0$ depends upon the extent of the hull represented; it may be the sheerline, the design waterline, or some other arbitrary waterline. If the hull has no knuckles, s is proportional to the arclength around each station; if it has knuckles, there is no simple interpretation of the coordinate s . The coordinates x and s are not orthogonal. Figure 1 shows the two different coordinate systems.

The functions $Y(x, s)$ and $Z(x, s)$ have the form

$$Y(x, s) = By(x, s) \quad ; \quad Z(x, s) = Dz(x, s) \quad (2.2)$$

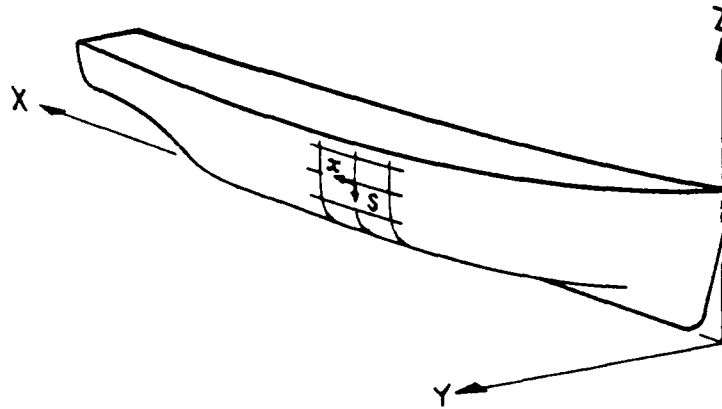


Figure 1: The coordinate systems

where B is the half-breadth of the hull and D is its depth. The functions $y(x, s)$ and $z(x, s)$ are complicated and, of course, are different for every hull (see Appendix B). They are given in terms of a large number of spline coefficients (usually several thousand) which must be determined whenever a new hull is to be represented. These coefficients are stored in a data file. Notice that the dimensions of the hull can be changed simply by changing the values of L , B , and D ; there is no need to alter the functions $y(x, s)$ and $z(x, s)$. This is the simplest form of modification allowed by MODHLL.

3 Coordinate Transformations

Lackenby[4] discusses a simple method by which hulls can be modified by moving the hull sections. The hull modification is done by performing a simple coordinate transformation on the coordinate x . This amounts to a shifting of each section forward or aft by an amount that is dependent on x . That is, if x' is the coordinate on the original hull and $x(x')$ on the modified hull, then station x' on the original hull is shifted to station $x(x')$ on the modified hull. The forward and aft perpendiculars are left invariant so that the length of the ship between perpendiculars remains constant.

Lackenby's method involves shifting only those stations forward and aft of any parallel midbody. Of course, shifting the stations in the parallel mid-body would cause no change of the ship's form. The coordinate transformation used is of the form

$$x = a'x'^2 + b'x' \text{ for } x' < p'_1 \quad (3.1)$$

$$= d'(20 - x')^2 + e'(20 - x') + 20 \text{ for } p'_2 < x' < 20 \quad (3.2)$$

where x is the coordinate describing the modified hull, x' is the coordinate describing the original hull, p'_1 is the station on the hull at which the parallel mid-body starts, and p'_2 is the station at which the parallel mid-body ends. If no parallel mid-body exists, there is an implicit assumption that the tangents to the hull at the maximum section are all parallel to the centreplane, otherwise the new hull would not be smooth at the mid-section. This method is not sufficiently general for warship hulls which, in general, will not satisfy this implicit criterion. MODHLL uses a generalization of Lackenby's method which relaxes this assumption.

The coordinate transformation used is similar to that of Lackenby, but extends over the mid-section as well as the ends of the hull. It is then no longer required that p'_1 and p'_2 be the start and end of parallel mid-body: they may be any two stations satisfying $0 < p'_1 \leq p'_2 < 20$.

If the old hull was given by $X'(x', s)$, $Y'(x', s)$, and $Z'(x', s)$, the functional relations $X(x, s)$, $Y(x, s)$, $Z(x, s)$ for the modified hull are given by

$$X(x, s) = X'(x'(x), s) \quad (3.3)$$

$$Y(x, s) = Y'(x'(x), s) \quad (3.4)$$

$$Z(x, s) = Z'(x'(x), s) \quad (3.5)$$

thus, the requirement is that x' be given in term of x , in contrast to Lackenby's transformation.

The coordinate transformation used is represented as a piecewise cubic polynomial with breakpoints¹ at p_1 and p_2 , where $0 < p_1 < p_2 < 20$; p_1 and p_2 are otherwise arbitrary. If p_1 and p_2 are chosen to be the start and the end of the parallel mid-body, then the transformation assumes the same form as that of Lackenby except for the reversal of roles of x and x' and the higher order of the polynomials used. Flow calculations which use the hull representation sometimes require that $X(x, s)$, $Y(x, s)$, and $Z(x, s)$ be twice differentiable. This in turn requires that the coordinate transformation have two continuous derivatives at p_1 and p_2 . This cannot be done unless the polynomial pieces are of at least fourth order (i.e. cubic polynomials). Since there are three polynomial segments,

$$x'(x) = a + bx + cx^2 + dx^3 \text{ for } 0 \leq x \leq p_1 \quad (3.6)$$

$$= e + fx + gx^2 + hx^3 \text{ for } p_1 \leq x \leq p_2 \quad (3.7)$$

$$= i + jx + kx^2 + lx^3 \text{ for } p_2 \leq x \leq 20 \quad (3.8)$$

there are 12 coefficients to determine. The continuity requirements at p_1 and p_2 give six relations which must be satisfied. The last requirement is that positions of the bow and the stern do not change; thus $x = 0$ must correspond to $x' = 0$ and $x = 20$ to $x' = 20$. This gives an additional

¹Breakpoints are the places at which the piecewise polynomial segments are connected. For more information on numerical calculations using piecewise polynomials, see de Boor[5]

two conditions, leaving 4 degrees of freedom inherent in the coordinate transformation: i.e. there are four parameters that may be altered when specifying how the hull is to change.

The transformation is defined by specifying four parameters altered in one of the following two ways:

1. specifying X_B , ϕ , p'_1 , and p'_2 , or
2. specifying ϕ_F , ϕ_A , p'_1 , and p'_2 .

ϕ_F , ϕ_A , ϕ , X_B , p'_1 and p'_2 are defined as follows.

1. ϕ_F is the prismatic coefficient of the forebody measured from midships².

$$\phi_F = \frac{1}{10A(10)} \int_0^{10} A(x) dx \quad (3.9)$$

where $A(x)$ is the curve of sectional areas. Note that $A(x)$ is straightforward to calculate from the curve of sectional areas on the original hull since $A(x) = A'(x'(x))$.

2. ϕ_A is the prismatic coefficient of the afterbody measured from midships.

$$\phi_A = \frac{1}{10A(10)} \int_{10}^{20} A(x) dx \quad (3.10)$$

3. $\phi \equiv (\phi_A + \phi_F)/2$, the total prismatic coefficient.

4. X_B is the longitudinal centre of buoyancy.

$$X_B = \frac{1}{20A(10)\phi} \int_0^{20} xA(x) dx \quad (3.11)$$

5. p'_1 is the station on the original hull which correspond to p_1 . This may be used to specified the start of a parallel mid-body. Thus, p_1 is set to the desired start of the parallel mid-body on the the new hull, and p'_1 is specified to be the start of the parallel mid-body on the original hull.

6. p'_2 is the station on the original hull which correspond to p_2 . This may be used to specify the end of a parallel mid-body.

It is important to note that the sectional area curve for a modified hull is always calculated using the design waterline (see Section 1), even if the draft has been changed. It was decided to calculate $A(x)$ in this way for three reasons.

²Although the fore and aft prismatic coefficients are properly measured from the section of maximum area, if they are measured in this way, changes in them produce little change in the form of hulls with near-parallel mid-body. Hence, for the purposes of this memorandum, it is better to calculate them from the mid-section.

1. The interpretation of station 0 and 20 as the forward and aft perpendiculars is maintained: i.e. they are the stations at which the hull surface, the centreplane and the design waterline intersect.
2. Changes in draft will not affect the calculated prismatic coefficients. To maintain constant prismatic coefficients when the draft is changed, it is not necessary to change the hull form, as would be the case if the sectional areas were calculated from the draft waterline.
3. MODHLL represents the draft waterline in a way that remains invariant if the transformation $x'(x)$ is changed (see Section 8). Thus the draft waterline is independent of changes in hull form. However, future modifications of MODHLL will allow calculation of the draft waterline for different states of ship's trim. At non-zero trim angles, the draft waterline will necessarily become dependent on changes in hull form as well as the trim angles themselves. If the curve of sectional areas were to be calculated using the draft waterline, to maintain constant prismatic coefficients while changing the angle of trim would require a change in hull form. Moreover, calculation of the transformation $x'(x)$ (see Section 5) would be greatly complicated by the fact that any change in hull form would require recalculation of the draft waterline.

4 B-Splines

Piecewise polynomials such as that described above are often called splines. A convenient way of representing a spline is in term of B-spline basis functions (see de Boor[5] for an explanation of B-splines). This allows the representation of the transformation in the following form:

$$x' - x = \sum_{i=1}^4 \xi_i B_i(x) \quad (4.1)$$

where $B_i(x)$ is the i^{th} B-spline corresponding to the knot sequence $\{0, 0, 0, p_1, p_2, 20, 20, 20\}$ ¹. By choosing only three identical knots at each endpoint one ensures that $B_i(0) = B_i(20) = 0$ and hence that the forward and aft perpendiculars remain invariant under the transformation. Figure 2 shows the four B-splines when $p_1 = 8$ and $p_2 = 12$. The spline coefficients ξ_i are adjusted to produce the transformation desired. If the change in the hull is small, then each ξ_i is small: i.e. $\xi_i \ll 20$.

Changes in the the ξ_i cause the following changes.

¹A knot sequence is a generalization of the sequence of breakpoints; each element of the sequence is called a knot. The B-spline basis functions depend upon the positions of the knots. See de Boor[5] for an exposition of the theory of splines.

- Increasing ξ_1 or ξ_2 increases ϕ_F and ϕ , decreases X_B (i.e. moves the longitudinal centre of buoyancy further forward), and increases p'_1 (implying a relative lengthening of the mid-body on the new hull).
- Increasing ξ_3 or ξ_4 decreases ϕ_A and ϕ , decreases X_B , and increases p'_2 (implying a relative shortening of the mid-body on the new hull).

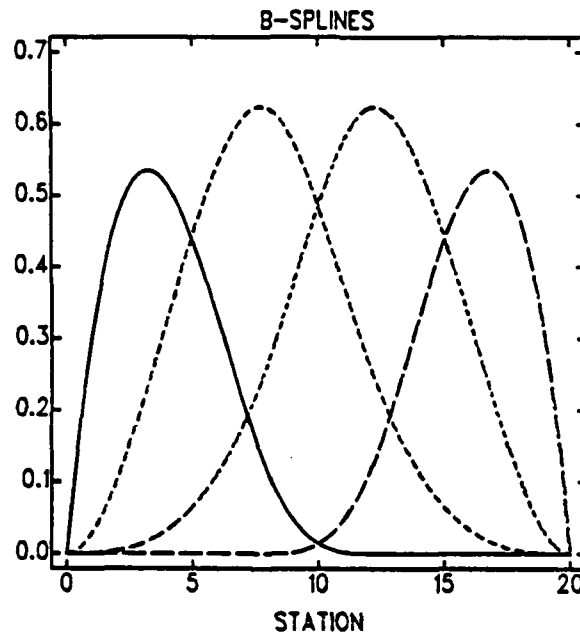


Figure 2: B-spline representation

5 Calculation of the ξ_i

If each of the specified quantities were linear in the ξ_i , the calculation of the ξ_i would amount to the solution of a linear system of equations in four unknowns, as was the case in Lackenby's method; however, due to the inversion of the roles of x and x' , neither the prismatic coefficients or the longitudinal centre of buoyancy are linear in the ξ_i since they appear in the argument of the non-linear function $A'(x')$ used to evaluate $A(x)$. The method for calculating the ξ_i is as follows.

Since p'_1 and p'_2 are always specified, ξ_1 and ξ_4 can be determined in terms of ξ_2 and ξ_3 . Note that $B_1(p_2) = 0$ and $B_4(p_1) = 0$. Therefore,

$$p'_1 - p_1 = \xi_1 B_1(p_1) + \xi_2 B_2(p_1) + \xi_3 B_3(p_1) \quad (5.1)$$

$$p'_2 - p_2 = \xi_2 B_2(p_2) + \xi_3 B_3(p_2) + \xi_4 B_4(p_2) \quad (5.2)$$

ξ_2 and ξ_3 are found by a simple secant search.

Let

$$f(x, y) = a \quad (5.3)$$

$$g(x, y) = b \quad (5.4)$$

be two simultaneous equations to be solved for x and y . Expanding to first order around x_n and y_n one gets

$$f(x_n, y_n) + f^x(x, y)(x - x_n) + f^y(x, y)(y - y_n) \approx a \quad (5.5)$$

$$g(x_n, y_n) + g^x(x, y)(x - x_n) + g^y(x, y)(y - y_n) \approx b \quad (5.6)$$

where the superscripts denote partial derivatives by x or y . If x_n and y_n are previous approximations to the roots of equations (5.3) and (5.4) can be solved to yield a better approximation:

$$x_{n+1} = x_n + \frac{g^y(x, y)(a - f(x, y)) - f^y(x, y)(b - g(x, y))}{f^x(x, y)g^y(x, y) - f^y(x, y)g^x(x, y)} \quad (5.7)$$

$$y_{n+1} = y_n - \frac{g^x(x, y)(a - f(x, y)) - f^x(x, y)(b - g(x, y))}{f^x(x, y)g^y(x, y) - f^y(x, y)g^x(x, y)} \quad (5.8)$$

The derivatives of f and g may be approximated by simple finite differences of previous approximations.

The following algorithm is used to find suitable approximations for the values of x and y .

$n = 0$

Guess x_n, y_n

Calculate $f(x_n, y_n), g(x_n, y_n)$

Guess x_{n+1}

Calculate $f(x_{n+1}, y_n), g(x_{n+1}, y_n)$

$$f^z = \frac{f(x_{n+1}, y_n) - f(x_n, y_n)}{x_{n+1} - x_n} \quad g^z = \frac{g(x_{n+1}, y_n) - g(x_n, y_n)}{x_{n+1} - x_n}$$

Guess y_{n+1}

Calculate $f(x_{n+1}, y_{n+1}), g(x_{n+1}, y_{n+1})$

Repeat

$n = n + 1$

$$f^y = \frac{f(x_n, y_n) - f(x_n, y_{n-1})}{y_n - y_{n-1}} \quad g^y = \frac{g(x_n, y_n) - g(x_n, y_{n-1})}{y_n - y_{n-1}}$$

$$x_{n+1} = x_n + \frac{g^y(a - f(x_n, y_n)) - f^y(b - g(x_n, y_n))}{f^z g^y - f^y g^z}$$

Calculate $f(x_{n+1}, y_n), g(x_{n+1}, y_n)$

$$f^z = \frac{f(x_{n+1}, y_n) - f(x_n, y_n)}{x_{n+1} - x_n} \quad g^z = \frac{g(x_{n+1}, y_n) - g(x_n, y_n)}{x_{n+1} - x_n}$$

$$y_{n+1} = y_n + \frac{f^z(b - g(x_{n+1}, y_n)) - g^z(a - f(x_{n+1}, y_n))}{f^z g^y - f^y g^z}$$

Calculate $f(x_{n+1}, y_{n+1}), g(x_{n+1}, y_{n+1})$

Until sufficient accuracy or too many iterations

When ϕ_F and ϕ_A are specified, the independent variables of both functions are ξ_2 and ξ_3 . Thus, substitute the following values into the preceding algorithm:

$$\begin{aligned} x_n &\rightarrow \xi_{2,n} \\ y_n &\rightarrow \xi_{3,n} \\ a &\rightarrow \phi_F \\ b &\rightarrow \phi_A \\ f(x_n, y_n) &\rightarrow \phi_E(\xi_{2,n}, \xi_{3,n}) \\ g(x_n, y_n) &\rightarrow \phi_A(\xi_{2,n}, \xi_{3,n}) \end{aligned}$$

Whenever ϕ_F and ϕ_A are to be evaluated (i.e. at the steps requiring the calculation of f and g), ξ_1 and ξ_4 must first be determined using equations (5.1) and (5.2). ϕ_F and ϕ_A are then calculated using equations (3.9) and (3.10).

Similarly, the longitudinal centre of buoyancy and the total prismatic coefficient also depend on ξ_2 and ξ_3 . Hence, when the total prismatic coefficient and the longitudinal centre of buoyancy are specified, then substitute the following values into the preceding algorithm:

$$\begin{aligned} x_n &\rightarrow \xi_{2,n} \\ y_n &\rightarrow \xi_{3,n} \end{aligned}$$

$$\begin{aligned}
a &\rightarrow X_B \\
b &\rightarrow \phi \\
f(x_n, y_n) &\rightarrow X_B(\xi_{2,n}, \xi_{3,n}) \\
g(x_n, y_n) &\rightarrow \phi(\xi_{2,n}, \xi_{3,n})
\end{aligned}$$

6 Monotonicity of the Transformation

The function describing the modified hull ($x'(x) = x + \sum_{i=1}^4 \xi_i B_i(x)$) must increase with the station number, x , and must never exceed 20 or be negative because the functions $X'(x', s)$, $Y'(x', s)$ and $Z'(x', s)$ are not defined in those intervals. Note that the latter two conditions are special cases of the first one because the knot sequence used for the B-splines ensures that $B_i(0) = 0$ and $B_i(20) = 0$.

Fritsch and Carlson[6] have shown necessary and sufficient conditions for a cubic interpolant to maintain monotonicity. Let $f(x)$ be the cubic interpolant, let $\{x_i, i = 1, \dots, n\}$ be its breakpoints and

$$\begin{aligned}
\Delta_i &= \frac{f_{i+1} - f_i}{x_{i+1} - x_i} \\
d_i &= \frac{df}{dx}(x_i) \\
\alpha_i &= \frac{d_i}{\Delta_i} \\
\gamma_i &= \frac{d_{i+1}}{\Delta_i} \\
\text{sgn}(a) &\equiv \text{the sign of } a.
\end{aligned}$$

Their conditions for monotonic data sets can then be summarized as follows. If $\text{sgn}(d_i) = \text{sgn}(d_{i+1}) = \text{sgn}(\Delta_i) > 0$, then $f(x)$ is monotonic on $[x_i, x_{i+1}]$ if and only if either

1. $\alpha_i + \gamma_i - 2 \leq 0$; or
2. $\alpha_i + \gamma_i - 2 > 0$; and either
 - (a) $2\alpha_i + \gamma_i - 3 \leq 0$; or
 - (b) $\alpha_i + 2\gamma_i - 3 \leq 0$; or
 - (c) $\alpha_i - \frac{(2\alpha_i + \gamma_i - 3)^2}{3(\alpha_i + \gamma_i - 2)} \geq 0$

This test is used to check the monotonicity of the transformation between the modified hull and its parent. If the test fails, MODHLL requires either that the modified hull be discarded or that it be further modified until $x'(x)$ is monotonic.

7 Calculation of X_B , ϕ_F , and ϕ_A

The values for X_B , ϕ_F , and ϕ_A are obtained from equations (3.11), (3.9), and (3.10), each of which involves the curve of sectional areas, $A(x)$. This function is itself defined in terms of a line integral around the perimeter of section x as follows.

$$A(x) = - \int Y dZ = - \int_0^1 Y(x, s) \frac{\partial Z}{\partial s}(x, s) ds \quad (7.1)$$

Moreover, the curve of sectional areas of a modified hull is given by $A(x) = A'(x'(x))$ where $A'(x')$ is the curve of sectional areas of the parent hull. Hence, only $A'(x')$ need be calculated. To avoid performing this integral many times, $A'(x')$ is calculated once for 21 values of x' and the resulting values are splined. The resulting spline approximation of $A'(x')$ is then used when calculating X_B , ϕ_F , and ϕ_A . This is much more efficient than integrating equation (7.1) many times whenever X_B or the prismatic coefficients are calculated.

The spline of the curve of sectional areas is calculated using the algorithm BSMTH described by Hally[7]. It is represented in terms of B-splines as follows.

$$A'(x') = \sum_{n=1}^N \beta_n B_n(x') \quad (7.2)$$

where:

- β_n is the n^{th} spline coefficient,
- N is the number of B-splines, and
- B_n is the n^{th} B-spline of fourth order corresponding to the knot sequence $\{0,0,0,0,2,3,4,\dots,16,17,18,20,20,20,20\}$.

Each of the above integrals is calculated numerically by sub-dividing its range into a number of smaller intervals and using four-point Gaussian quadrature on each interval. Since the simple linear transformation $t = (2x - a - b)/(b - a)$ will translate any interval $[a,b]$ into $[-1,1]$ provided $b \neq a$, one need only consider integrals over the interval $[-1,1]$.

$$\int_a^b f(x) dx = \int_{-1}^1 f\left(\frac{(b-a)t + b + a}{2}\right) \frac{(b-a)}{2} dt = \int_{-1}^1 g(t) dt \quad (7.3)$$

where

$$g(t) = f\left(\frac{(b-a)t + b + a}{2}\right) \frac{(b-a)}{2} \quad (7.4)$$

The integral of $g(t)$ is approximated by

$$\int_{-1}^1 g(t) dt \approx \sum_{i=1}^n c_i g(t_i) \quad (7.5)$$

for some known points t_i and coefficients c_i . The method of Gaussian quadrature chooses these $2n$ parameters optimally so that all polynomials of order $2n$ or less are calculated exactly. The t_i for which this occurs are the roots of the n^{th} order Legendre polynomial. The values of x_i and c_i when $n = 4$ (used by MODHLL), are given in Table 1.

Roots, t_i	Coefficients, c_i
.8611363116	.3478548451
.3399810436	.6521451549
-.3399810436	.6521451549
-.8611363116	.3478548451

Table 1: Roots and Coefficients for 4-Point Gaussian Quadrature

8 Representation of the Draft Waterline

The draft waterline of the hull is a curve $s = w(x')$ which gives the s -value of the draft waterline as a function of the station number on the parent hull x' . The function $w(x')$ is defined implicitly by

$$Z'(x', w(x')) = Z_w \quad (8.1)$$

where Z_w is the draft. The draft waterline on a modified hull is given by $s = w(x'(x))$. Although the HLLFLO suite of programs includes procedures for solving equation (8.1) for $w(x')$ (see Hally[8]), they are not very efficient. Hence, a spline representation of the curve $w(x')$ is calculated and stored in the hull data file, so that the s -value of the draft waterline may be obtained quickly. The spline is calculated using the algorithm BSMTH described by Hally[7]. It is represented in terms of B-splines as follows.

$$w(x') = \sum_{n=1}^N \beta_n B_{n,k}(x') \quad (8.2)$$

where:

- β_n is the n^{th} spline coefficient,
- N is the number of B-splines,
- k is the order of the spline (usually 4), and
- $B_{n,k}$ is the n^{th} spline corresponding to a set of knots in which there are k knots at each end-point of the draft waterline and the remaining $(N + k) - 2k$ knots are equally spaced.

If the depth of the hull, D , is changed but Z_w remains constant, the draft waterline function $w(x')$ also changes. To avoid the recalculation of $w(x')$, whenever D is changed by a factor α (i.e. $D \rightarrow \alpha D$) Z_w is also scaled by α (i.e. $Z_w \rightarrow \alpha Z_w$). In this case, $w(x')$ remains invariant so that the waterline curve need not be recalculated or resplined.

9 Concluding Remarks

In order to see how the flow and the resistance around a ship change, it is useful and convenient to generate a slightly altered hull from one whose computer representation is already defined. The MODHLL program uses a generalization of Lackenby's method on which the new hull is created by specifying the new characteristics of the hull such as the prismatic coefficients and centre of buoyancy. Furthermore, it is also possible to change the hull's dimensions and its draft without affecting the basic characteristics of the hull. The modified hull is defined from the parent hull by means of a simple spline transformation relating the station numbers x and x' on the respective hulls.

For the sake of efficiency, the program uses a spline representation to define the sectional area curve all along the hull. This curve gives the sectional area at any station on the hull and is used for calculating the prismatic coefficients and the centre of buoyancy of both modified and parent hulls.

Appendix A MODHLL User's Guide

This appendix is a user's guide for the program MODHLL. MODHLL has been designed to be integrated within the HLLFLO suite of programs and therefore many implementation decisions have been made to provide consistency with the other programs in HLLFLO. The first five chapters of the HLLFLO User's Guide[1] provide overall information about HLLFLO.

MODHLL itself is written in ANSI standard FORTRAN 77 but calls HLLFLO procedures which have minor non-ANSI features: see the HLLFLO User's Guide, Chapter 5 for details. MODHLL calls procedures from the spline libraries HLLYSP[7] and BSPLIN[5], from the GETWRD Package[9], and from the PLOT10[10] or IGL[11] graphics libraries. In addition, procedures are called from the HLLFLO libraries CMDSUB, HCRSUB, HLLSUB, MSGSUB, PLTSUB, and SMVSUB.

MODHLL is designed to be run on a Tektronix graphics terminal or equivalent. Its graphics are generated by the HLLFLO PLTSUB library which in turn call procedures from either the PLOT10 library or the IGL library. See the HLLFLO User's Guide, Chapter 4 for details.

On the DEC-20/60 at DREA, a simple method for running MODHLL is available. A new EXEC command may be defined by typing

```
@DECLARE PCL PS:<HALLY>HLLFLO
```

at the EXEC level of command. This uses the PCL extension to the TOPS-20 EXEC to define the new command HLLFLO. By giving the command

```
@HLLFLO MODHLL plot-library
```

MODHLL is automatically compiled, loaded, and run. If the file MODHLL.EXE already exists on the directory to which you are connected, it will be run. Otherwise a search is made for .REL versions of all necessary library files. If an .REL version of a file cannot be found, the source file on PS:<HALLY> is compiled. Once .REL versions of all necessary files have been created, they are linked to create the file MODHLL.EXE which is then run. New .REL and .EXE files remain on the connected directory for subsequent use. The sub-command *plot-library* is one of PLOT10 or IGL according to the plotting library you wish to use.

MODHLL uses a command language to allow you to define how hulls are to be modified: see Chapter 3 of the HLLFLO User's Guide[1] for details. The prompt

```
command=>
```

indicates that MODHLL is awaiting command. The allowed commands and their functions are described in the following sections. The command

`command=>EXIT`

is used to stop execution of the program.

A.1 Defining the Terminal Characteristics

Upon entering MODHLL the message

Entering MODHLL ...

terminal type? =>

is written on the terminal screen. This is the prompt to identify the characteristics of the terminal which you are using. See the HLLFLO User's Guide, Chapter 4 for an explanation of the possible responses. The terminal characteristics may be changed by giving the command

`command=>TERMINAL=`

followed by the characteristics of the terminal. Typically you will plot a graph in MODHLL then use this command to redefine the plotter to be the laser-printer rather than the terminal so that a hard copy of the graph can be obtained.

A.2 Loading a Parent Hull File

At all times MODHLL saves the characteristics of two hulls: the parent and a modified form of it. Before modifications of a hull can be made, a parent hull must first be loaded into the program memory. A prompt for the name of the file containing the parent hull data occurs when you enter MODHLL, immediately after the prompts for the terminal characteristics. Subsequently a new parent hull can be loaded by giving the command

`command=>LOAD`

after which you will be prompted the name of the parent hull data file. The file name may be at most 30 characters long including directory specifications and extensions. If no extension is given, the default on the DREA DEC-20/60 is .DAT. The input file must be of the same format as an output file from the HLLFLO program SMHULL (see Appendix B) or an output file from MODHLL (see Section A.3). In the latter case, the name of the file containing the parent hull data is read from the file containing the modified hull data.

When a parent hull is loaded, the modified hull is set to be the same as the parent. This causes all characteristics of the previous modified hull to be lost from the program memory. If you have created a modified hull and then give the LOAD command before saving it in a file (see Section A.3), you are asked whether you wish to save the previous modified hull before loading the new parent. This prevents loss of time if you carelessly forget to save the modified hull you have just created.

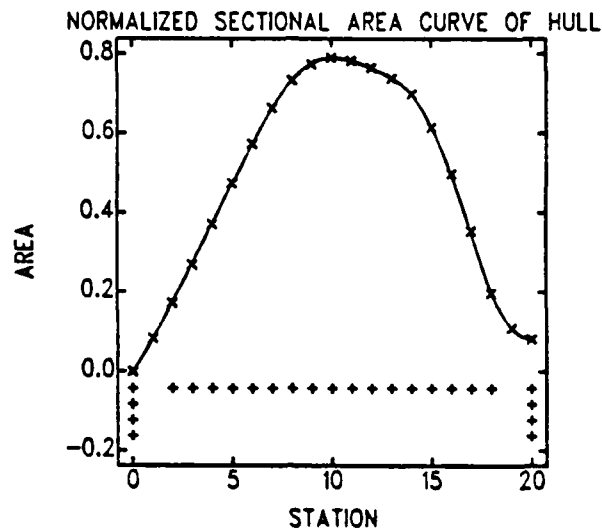


Figure A.1: Spline of Normalized Curve of Sectional Areas

Whenever a parent hull is loaded, a spline representation of its curve of sectional areas is calculated (see Section 7). A message is written on the screen when the calculation begins and ends. You are then asked whether you would like to examine a display of the spline curve. It is usually wise to do so since splines will sometimes contain unwanted wiggles that could cause problems when calculating the prismatic coefficients. If you answer YES, a display such as that in Figure A.1 is plotted. Notice that it is actually the curve $A'(x')/BZ_d$ that is splined, where B is the half-breadth and Z_d is the design waterline of the parent hull.

When the display has been plotted the prompt

sub-command=>

appears in the top left corner of the screen. If the spline curve is not to your liking it may be altered as described in Section A.8. To accept the spline curve use the sub-command CONTINUE. The screen will be cleared and the characteristics of the parent hull will be displayed. MODHLL then awaits further commands.

A.3 Saving a Modified Hull

A modified hull may be saved by giving the command

command=>SAVE

You are prompted for a title for the hull (for use on graphic displays generated by HLLFLO graphics programs), and for the name of the file in which to store the data. The former is a

string of at most 60 characters, the latter a string of at most 30 characters.

You cannot save a hull whose x -transformation has not yet been calculated (since this transformation is integral to the definition of the modified hull), nor can you save a hull whose x -transformation is not monotonic.

The file created, which contains all relevant information defining a modified hull, can be read subsequently by other programs (for example, a program calculating the flow around this modified hull).

The modified hull data file is written in the following format.

<u>Variable</u>	<u>Format</u>
'MODIFIED HULL'	(A)
HLLFIL	(A)
TITLE	(A)
SLNGTH, BDTH, DPTH	(3G15.7)
P1, P2	(2G15.7)
XI(I), I=1,...,4	(4G15.7)
If $ZWTR \neq DPTH$:	
ZWTR	(G15.7)
KWTR, NWTR	(2I5)
TWTR(J), J=1,...,KWTR+NWTR	(5G15.7)
BWTR(J), J=1,...,NWTR	(5G15.7)

where:

- 'MODIFIED HULL' is a character string flag used to indicate that the file contains data describing a modified hull rather than a parent,
- HLLFIL is the name of the file containing the parent hull data,
- SLNGTH, BDTH and DPTH are respectively the length, the half breadth and the depth of the ship,
- P1 is the station number p_1 ,
- P2 is the station number p_2 ,
- XI is an array containing the values of $\xi_i, i = 1, \dots, 4$,
- ZWTR is the draft of the ship,
- KWTR is the order of the spline used to define the position of the draft waterline,
- NWTR is the number of B-splines used to define the position of the draft waterline,
- TWTR is the knot sequence used to define the position of the draft waterline, and
- BWTR are the spline coefficients used to define the position of the draft waterline.

A.4 Modifying the Hull

As described in Section A.2, MODHLL keeps the characteristics of a parent hull and a modified hull in its memory. The modified hull starts out as an exact replica of the parent but can be modified by the commands described in this section. The parent hull remains invariant until a new parent is loaded using the LOAD command. The command to change the modified hull is

`command=>CHANGE parameter value`

where *parameter* is one of AFT-PRISMATIC-COEF, DEPTH, DRAFT, FORWARD-PRISMATIC-COEF, HALF-BREADTH, LCB, LENGTH, P1, P1', P2, P2', or TOTAL-PRISMATIC-COEF and *value* is the new value for that parameter. The default value for p_1 and p_1' is 8.0. The default value for p_2 and p_2' is 12.0. The default values for the remaining parameters are the corresponding values on the parent hull.

A.4.1 Changes in the Draft

When the draft of the hull is changed, a new draft waterline curve $w(x')$ must be calculated. As described in Section 8, $w(x')$ is approximated by a spline. Before the spline can be calculated, its order (the order of the polynomial segments of which it is comprised) and the number of B-splines used to represent it must be specified (see de Boor[5] for further information about splines). Reasonable values for these variables are 4 and 20 respectively. The values are input as sub-commands to the command CHANGE DRAFT. The full command has the form

`command=>CHANGE DRAFT value-of-draft order-of-spline no-of-B-splines`

so that a typical command might be

`command=>CHANGE DRAFT 15.5 4 20`

After the command has been given, MODHLL calculates the spline and then displays it so that you can examine it. When the display has been plotted the prompt

`sub-command=>`

appears in the top left corner of the screen. If the spline curve is not to your liking it may be altered as described in Section A.8. To accept the spline curve use the sub-command CONTINUE. The screen will be cleared and MODHLL then awaits further commands.

A.4.2 Changes which Affect $x'(x)$

A change in the value of the length, half-breadth, depth, or draft of the hull in no way changes the transformation $x'(x)$. If only these parameters are changed the values of ξ_i which define this transformation need not be recalculated; however, a change in any of the other parameters requires that the ξ_i be recalculated before the modified hull can be saved or its curve of sectional areas can be plotted. The command to calculate the transformation is

command=>CALCULATE-X-TRANSFORMATION

Note that you can make as many changes to the hull as you wish before calculating the ξ_i .

As described in Section 3, the x-transformation can be calculated using the values of either X_B and ϕ or of ϕ_F and ϕ_A . If only the values of p_1 , p'_1 , p_2 , and p'_2 are changed, the default is to use X_B and ϕ . Otherwise, MODHLL will determine which parameters have been changed and calculate the x-transformation appropriately. For example, if only ϕ_F has been changed, ϕ_F and ϕ_A will be used. In the case of an ambiguity (e.g. if both X_B and ϕ_A have been changed), you will be prompted to clarify which set of parameters you wish to be used.

Whenever an x-transformation is calculated successfully, the parameters of the modified hull described by that transformation become the new defaults. For example, suppose the current default value for X_B is 10.4 and of ϕ_F is 0.61. You change ϕ_F to 0.59 and the x-transformation is calculated successfully. MODHLL recalculates X_B using the new transformation obtaining a value of 10.4782. This becomes the new default value for X_B and the default for ϕ_F is 0.59.

When calculating the ξ_i , MODHLL prints on the terminal a record of the number of iterations performed so far. If ten iterations are performed without convergence, the calculation is ended. MODHLL stores the values of the ξ_i which gave the most accurate fit to the parameters you specified. You are given the option of retaining the hull as defined by these "best" ξ -values as the new modified hull. Remember that if you do so, the default values for all parameters will be set to the values for this hull. If you do not accept the "best" ξ -values, you must make further changes and recalculate the transformation.

Sometimes when an attempt to calculate the x-transformation fails, you will wish to restore all defaults to the values for the parent hull. This may be done by giving the command

command=>RESET

As noted in Section 4, the values of p_1 , p'_1 , p_2 , and p'_2 affect the values of the prismatic coefficients for the modified hull as compared with their values for the parent hull. The most common cause of failure in the calculation of the x-transformation is to specify values of the p 's which tend to increase a prismatic coefficient, while requiring that the prismatic coefficient actually decrease, or vice versa. For example, setting $p_1 < p'_1$ while requiring that ϕ_F decrease will likely cause trouble. Before starting the calculation of the x-transformation, MODHLL checks the values of the p 's and of the required changes in the prismatic coefficients. If it suspects there may be difficulties, you will be notified of the problem and asked whether you wish to continue with the calculation anyway. The following message is typical.

command=>CALC

P2-P1 = 4.000 and P2'-P1' = 3.000 tending to increase the total prismatic coefficient. But you wish to decrease the total prismatic coefficient. The iteration in the calculation of the x-transformation is likely to diverge.

Do you wish to proceed with the calculation (Y or N)?

Table 2 summarizes the changes suggested in the p 's to avoid difficulties when changing the hull parameters.

Parameter Modification	Suggested Change in p 's
$X_B > X'_B$	$p_2 + p_1 > p'_2 + p'_1$
$\phi > \phi'$	$p_2 - p_1 > p'_2 - p'_1$
$\phi_F > \phi'_F$	$p_1 < p'_1$
$\phi_A > \phi'_A$	$p_2 > p'_2$

Table 2: Summary of Suggested Changes in p_1 and p_2

A.5 Printing Hull Characteristics

After changing a few hull parameters, it is easy to forget the value of a specific parameter on the modified or parent hull. The values of all parameters can be printed on the terminal using the commands

command=>PRINT-MODIFIED-HULL

and

command=>PRINT-PARENT-HULL

The following is an example of output generated by PRINT-MODIFIED-HULL.

The characteristics of the current hull are:

```

P1' = 8.50   P1 = 8.00
P2' = 11.00  P2 = 12.00
Longitudinal centre of buoyancy = 10.3732
  Total prismatic coefficient = 0.5917
  Forward prismatic coefficient = unknown
  Aft prismatic coefficient = unknown
Length between perpendiculars = 112.000 M
      Half breadth = 7.260 M
      Depth = 11.500 M
      Draft = 8.125 M

```

The unknown values of ϕ_F and ϕ_A occur because either X_B and or ϕ has been changed but the x -transformation has not yet been recalculated, so that the values for ϕ_F and ϕ_A which correspond to the given values of X_B and ϕ have not yet been determined. If the x -transformation had already been determined, the values for ϕ_F and ϕ_A would be included.

A.6 Plotting the Curves of Sectional Areas

A display of the curve of sectional areas of both the modified hull and the parent hull can be generated by giving the command

```
command=>PLOT-SECTIONAL-AREAS
```

An example of the display is shown in Figure A.2.

Note that, even if the draft has been modified, the area of each section is defined to be the area below the design waterline: the waterline of the parent hull perhaps scaled by a change in depth (see Sections 1 and 8). For example, suppose the depth of the parent hull is equal to 20 ft and its design waterline is 15 ft above the baseline. If the depth of the ship is modified to a value of 25 ft, then the design waterline for the modified hull is 18.75 ft from the baseline. Even if the draft of the modified hull is subsequently changed to 17 ft, the sectional areas are still calculated using the value 18.75 ft.

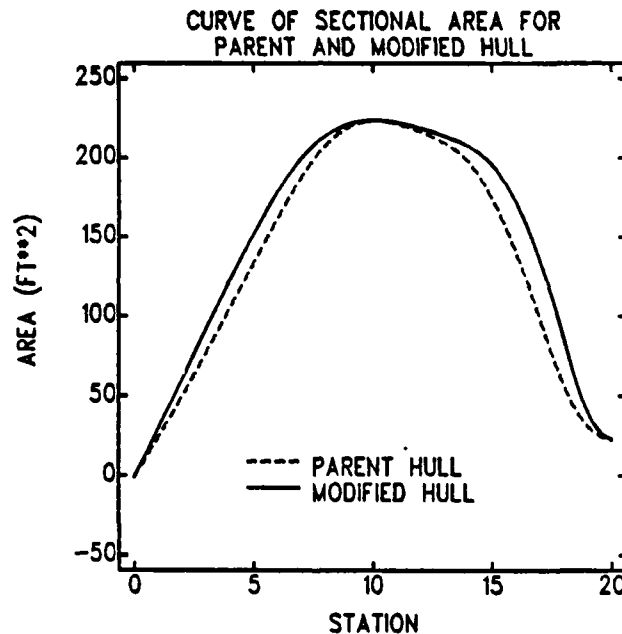


Figure A.2: Curves of Sectional Area

A.7 Plotting $x'(x)$

A display of the currently defined transformation function $x'(x)$ can be generated by giving the command

```
command=>PLOT-X-TRANSFORMATION
```

An example of the display is shown in Figure A.3. The straight line on the display indicates the transformation which defines an unmodified hull ($x'(x) = x$).

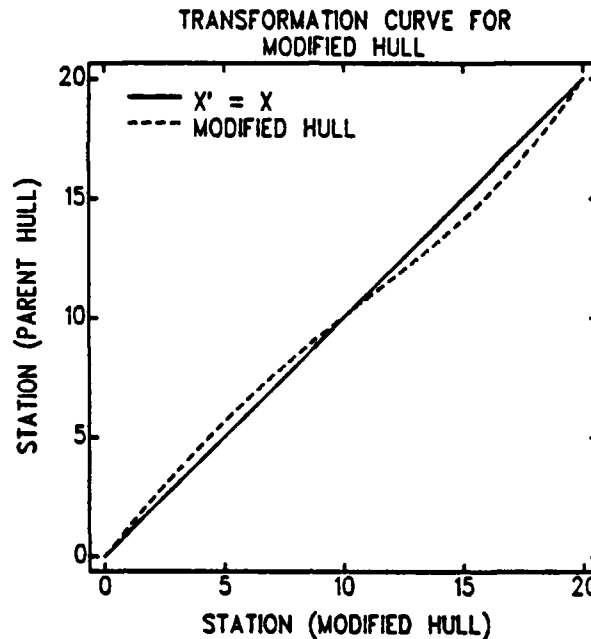


Figure A.3: Transformation curve for a modified hull

A.8 Modification of Splines

When a spline of either the sectional area curve or the design waterline of the parent hull has been calculated and plotted, MODHLL enters a sub-command mode which allows you to alter the spline if it is not to your liking. Figure A.1 is an example of the appearance of the terminal display at this point. The positions of the knots used for the spline are shown by the crosses near the bottom of the display.

The prompt

sub-command=>

which appears at the top left of the screen indicates that MODHLL is awaiting your command. If the spline is satisfactory, use the sub-command

sub-command=>CONTINUE

to continue; however, if you wish to make changes to the spline, use the command

sub-command=>CHANGE *parameter*

one or more times to alter the spline parameters. The sub-command

sub-command=>RESPLINE

will then recalculate and display the modified spline.

The following sections describe the spline parameters which may be changed in order to modify the spline curve.

A.8.1 Changing the Amount of Smoothing

The splines calculated by MODHLL are smoothing splines and hence, the desired amount of smoothing must be specified. (The spline is calculated by the HLLYSP library subroutine BSMTH whose algorithm is discussed by Hally[7].) This is done by establishing an appropriate χ^2 for the spline defined by

$$\chi^2 = \sum_{n=1}^N \frac{(f(x_n) - y_n)^2}{e_n^2} \quad (\text{A.1})$$

where $(x_n, y_n), n = 1, \dots, N$ are the data points, $f(x)$ is the function which defines the spline, and e_n is an error associated with the n^{th} data point. The accuracy of the spline is defined to be $\sqrt{\chi^2/N}$. If all the e_n are 1.0, then the accuracy of the spline is the same as the mean distance of the data points from the spline curve. Conversely, if the accuracy is 1.0, the distance of the spline from each data point will be *on average* equal to its error. The accuracy of the spline may be changed using the command

command=>CHANGE AMOUNT-OF-SMOOTHING *method*

where *method* is one of FOR-ALL-SPLINES= or CALCULATED-FROM-DATA.

The command FOR-ALL-SPLINES= requires an argument: a number equal to the accuracy for each of the splines.

The command CALCULATED-FROM-DATA indicates that the accuracy of each spline is to be calculated internally by the function PRERR in the HLLYSP library[7]. This is the default. Once changed the default for the amount of smoothing remains at its new value until changed again. If the accuracy is calculated from the data by PRERR, all subsequent splines will also call PRERR until the sub-command CHANGE AMOUNT-OF-SMOOTHING is invoked again.

A.8.2 Changing the Data Point Errors

The data point errors, e_n , may also be changed. The smaller the error, the closer the spline will try to come to its corresponding data point. The appropriate command is

sub-command=>CHANGE ERRORS *method*

where *method* is one of ALL-EQUAL or INTERACTIVE-SPECIFICATION. The command ALL-EQUAL causes the value of all errors to be set to 1.0.

The command INTERACTIVE-SPECIFICATION allows the changing of the errors of individual data points via the cross-hair sub-command mode (see Section 3.1 of the HLLFLO User's Guide[1]). Upon giving the command, the data points of the current spline are replotted and

the cross-hair appears as a prompt for a single letter command. The available commands are the following.

1. E or e: Exits from the cross-hair sub-command mode to the sub-command mode.
2. N or n: A new value for the error of the data point closest to the cross-hair may be entered. A line is drawn from the data point identified to the value typed in. The line allows you to remember which error value is associated with which data point.
3. O or o: The old value of the error of the data point closest to the cross-hair is printed on the screen somewhat above or below the cross-hair position.
4. P or p: Plots the data points again. After several errors have been changed, the screen can get cluttered with old and new error values. This command allows the screen to be cleared of all these numbers.
5. ?: Causes a list of possible cross-hair sub-commands to be written above the plot.

The errors once changed retain their altered values until changed again.

A.8.3 Changing the Stiffness Weights

The stiffness weights are parameters which can be used to change the "stiffness" of the spline in the interval between any two successive knots. By increasing the stiffness weight between two knots, the spline can be made straighter there. By decreasing the stiffness weight in a given knot interval, the spline can be allowed to bend more easily in that interval. Thus, it is desirable to have large stiffness weights where the spline should vary very little, and small stiffness weights where the spline is varying rapidly. To change the stiffness weights give the command

`command=>CHANGE STIFFNESS-WEIGHTS method`

where *method* is one of ALL-EQUAL, INTERACTIVE-SPECIFICATION or CALCULATED-FROM-DATA.

The command ALL-EQUAL sets all the stiffness weights to 1. Since only the relative values of the stiffness weights are important, there is no need to allow the setting of all stiffness weights to a value other than 1.

The command INTERACTIVE-SPECIFICATION allows the changing of individual stiffness weights via the cross-hair sub-command mode (see Section 3.1 of the HLLFLO User's Guide[1]). Upon giving the command, the data points of the current spline are replotted and the cross-hair appears as a prompt for a single letter command. The knots for the spline are indicated by the row of crosses at the bottom of the plot. There is a stiffness weight associated with the interval between each pair of adjacent knots. The available commands are

- E or e: Exits from the cross-hair sub-command mode to the sub-command mode.

- N or n: A new value for the stiffness weight of the knot interval identified by the vertical cross-hair may be entered. A line is drawn from the centre of the knot interval to the position of the cross-hair. The line allows you to remember which value is associated with which knot interval.
- O or o: The old value of the stiffness weight of the knot interval identified by the vertical cross-hair is printed at the position of the cross-hair.
- P or p: Plots the data points again. After several stiffness weights have been changed, the screen can get cluttered with old and new stiffness weight values. This command allows the screen to be cleared of all these numbers.
- ?: Causes a list of possible cross-hair sub-commands to be written above the plot.

The command `CALCULATED-FROM-DATA` indicates that the stiffness weights for each spline are to be calculated by the subroutine `WTIBEG` in the `HLLYSP` library[7]. This is the default.

New values for the stiffness weights remain as defaults until changed again. If the stiffness weights are calculated from the data by `WTIBEG`, all subsequent splines will also call `WTIBEG` until the command `CHANGE STIFFNESS-WEIGHTS` is invoked again.

A.9 Spline Trouble-shooting

`MODHLL` calculates the spline coefficients for each spline by evaluating and inverting a matrix. Occasionally, the matrix will be nearly singular and the resulting spline will be wild. `MODHLL` is almost always able to detect the problem. If the spline is not plotted, the message

```
??Error while calculating spline.
Probably numerical difficulties.
```

is written on the terminal screen. If the spline is plotted, no message is written, but it will be evident from the plot that there is a problem. Often the values of the spline will extend to huge numbers: e.g. 10^{10} or 10^{20} . It is usually possible to fix the spline by changing the amount of smoothing. In the typical case, the required accuracy of the spline is too low: try increasing it by a factor of 10 or 100 and see if that works. To determine the least amount of smoothing possible, you can later decrease it again progressively until the numerical problems reappear. Numerical difficulties can also be caused if the values of the errors or of the stiffness weights are set too high or too low. The most stable situation is if all errors and all stiffness weights are set equal to 1.0.

Appendix B Modifications to the HLLFLO Hull Representation

In order to facilitate the modification of hull dimensions by MODHLL, the HLLFLO hull B-spline representation has been modified slightly. In the previous representation (see Hally[2,3]), the cartesian coordinates (X, Y, Z) were given as function of the non-cartesian coordinates (x, s) by

$$X(x, s) = \frac{xL}{20} \quad (B.1)$$

$$Y(x, s) = \sum_{n=1}^{N_x} \sum_{j=1}^{N_s} \alpha_{nj} B_{n,k_x}^{(x)}(x) B_{j,k_s}^{(s)}(s) \quad (B.2)$$

$$Z(x, s) = \sum_{n=1}^{N_x} \sum_{j=1}^{N_s} \beta_{nj} B_{n,k_x}^{(z)}(x) B_{j,k_s}^{(s)}(s) \quad (B.3)$$

where $B_{n,k_x}^{(x)}(x), n = 1, \dots, N_x$, are B-splines of order k_x corresponding to a knot sequence $t_n^{(x)}, n = 1, \dots, N_x + k_x$, and $B_{j,k_s}^{(s)}(s), j = 1, \dots, N_s$ are B-splines of order k_s corresponding to the knot sequence $t_n^{(s)}, n = 1, \dots, N_s + k_s$. In the modified scheme, the spline coefficients are non-dimensionalized using the hull half-breadth, B , and depth, D .

$$X(x, s) = \frac{xL}{20} \quad (B.4)$$

$$Y(x, s) = \sum_{n=1}^{N_x} \sum_{j=1}^{N_s} B \alpha_{nj} B_{n,k_x}^{(x)}(x) B_{j,k_s}^{(s)}(s) \quad (B.5)$$

$$Z(x, s) = \sum_{n=1}^{N_x} \sum_{j=1}^{N_s} D \beta_{nj} B_{n,k_x}^{(z)}(x) B_{j,k_s}^{(s)}(s) \quad (B.6)$$

The virtue of the new scheme is that none of the spline coefficients need be changed when B and D are changed. In the old scheme, if B were doubled, then each of the coefficients α_{nj} would need to be doubled as well.

In addition, the new representation scheme includes a spline representation of the design waterline. In the old scheme, it was assumed that the design waterline was the line $s = 0$; however, this is not practicable if one wishes to allow modifications in the draft waterline of modified hulls. Instead, as described in Section 8, the design (or draft) waterline is defined by $s = w(x')$, where $w(x')$ is a function approximated by a spline. The spline data are included

in the data file immediately following the spline coefficients describing the hull. If a hull data file read by a HLLFLO program does not include design waterline data, the HLLFLO program will assume that the design waterline is $s = 0$ and that $Z_w = D$. The format for the design waterline data is

<u>Variable</u>	<u>Format</u>
ZWTR	(G15.7)
KWTR, NWTR	(2I5)
TWTR(J), J=1,...,KWTR+NWTR	(5G15.7)
BWTR(J), J=1,...,NWTR	(5G15.7)

where:

- ZWTR is the design draft of the ship,
- KWTR is the order of the spline used to define the position of the design waterline,
- NWTR is the number of B-splines used to define the position of the design waterline,
- TWTR is the knot sequence used to define the position of the design waterline, and
- BWTR are the spline coefficients used to define the position of the design waterline.

The program SMHULL has been modified to allow the user to spline the design waterline after the splines for the hull geometry have been determined. The user prompts and sub-commands allowed for spline modification are exactly as described in Section A.8.

All the HLLFLO programs have been modified so that they use the new representation scheme. These changes should be transparent to users but for the fact that the spline coefficients in the hull data files (the SMHULL output file) will have changed by a factor of B or D (the format of the data files remains the same). Hull data files created using HLLFLO version 1.0 will need modification to be used with the updated programs. The program CNVHLL has been provided as a means for modifying the files. CNVHLL simply asks for the name of the old data file and modifies the spline coefficients within it so that they obey the new hull representation scheme. The new data is written into a file specified by the user.

Except where noted in Chapter 5 of the HLLFLO User's Guide[1], CNVHLL is written in ANSI standard FORTRAN 77. It calls subroutines from the spline library BSPLIN[5], from the GETWRD Package[9], and from the HLLFLO libraries CMDSUB, HCRSUB, and MSGSUB. Provided that the PCL command HLLFLO has been declared (see Appendix A), CNVHLL may be run on the DREA DEC-20/60 using the command

@HLLFLO CNVHLL

References

- [1] D. Hally, "HLLFLO User's Guide," Technical Communication 86/306, DREA, 1986.
- [2] D. Hally, "Numerical Approximation of Ship Hulls," Technical Memorandum 85/207, DREA, 1985.
- [3] D. Hally, "A Spline Representation of a Hull with Knuckles," Technical Memorandum 86/209, DREA, 1986.
- [4] A. Lackenby, "On the systematic variation of ship forms," *Trans. RINA* **92**, 289 (1950).
- [5] C. de Boor, *A Practical Guide to Splines* (Springer Verlag, New York, 1978).
- [6] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM J. Numer. Anal.* **17**, 238 (1980).
- [7] D. Hally, "A New Library for Calculating Smoothing Splines," Technical Memorandum 85/205, DREA, 1985.
- [8] D. Hally, "User's Manual for DSPHLL, OFFSET, and STMHLC: Programs for Graphic Display of Ship Hulls and Potential Flows Around Them," Technical Communication 85/306, DREA, 1985.
- [9] D. Hally and C. Ann Dent, "GETWRD Package," Technical Memorandum 84/D, DREA, 1984.
- [10] *PLOT10: Terminal Control System*, Tektronix Inc. Document Nos. 062-1474-00, 062-1530-00.
- [11] *PLOT10: Interactive Graphics Library*, Tektronix Inc., 1983. Document No. 4010C01.

UNLIMITED DISTRIBUTION

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D		
(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)		
1. ORIGINATING ACTIVITY Defence Research Establishment Atlantic		2a. DOCUMENT SECURITY CLASSIFICATION Unclassified
		2b. GROUP
3. DOCUMENT TITLE MODHLL: A Program for Modifying Computer Representations of Ship Hulls		
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Technical Memorandum		
5. AUTHOR(S) (Last name, first name, middle initial) Hally, David and Gilbert, Normand		
6. DOCUMENT DATE December 1986	7a. TOTAL NO. OF PAGES 37	7b. NO. OF REFS 11
8a. PROJECT OR GRANT NO.	9a. ORIGINATOR'S DOCUMENT NUMBER(S) DREA TECHNICAL MEMORANDUM 86/224	
8b. CONTRACT NO.	9b. OTHER DOCUMENT NO.(S) (Any other numbers that may be assigned this document)	
10. DISTRIBUTION STATEMENT UNLIMITED DISTRIBUTION		
11. SUPPLEMENTARY NOTES		12. SPONSORING ACTIVITY
13. ABSTRACT Systematic series of hulls based on a parent hull have long been used by naval architects and hydrodynamicists to determine the effects of hull form on flow and resistance. This memorandum describes an interactive computer program, MODHLL, which creates modified hulls from an existing parent hull. Although the parent hull requires a large data file for its definition, each modified hull is defined by only a few extra words of data since only the way it differs from the parent needs be defined. Thus a whole series of hull forms may be defined using only a little extra storage space than the parent. The program is based on a generalization of Lackenby's method in which the hull is modified by moving the stations between the fore and aft perpendiculars. Each modified hull is specified by choosing values for the length, breadth, draft, depth and either 1. the fore and the aft prismatic coefficients, or 2. the total prismatic coefficient and the longitudinal centre of buoyancy.		

11/5/86
10-978

KEY WORDS

Ship hulls
Splines
Computer programs

INSTRUCTIONS

1. **ORIGINATING ACTIVITY** Enter the name and address of the organization issuing the document.
2. **DOCUMENT SECURITY CLASSIFICATION** Enter the overall security classification of the document including special warning terms whenever applicable.
- 2b. **GROUP** Enter security reclassification group number. The three groups are defined in Appendix M of the ORB Security Regulations.
3. **DOCUMENT TITLE** Enter the complete document title in all capital letters. Titles in all cases should be unclassified. If a sufficiently descriptive title cannot be selected without classification, show title classification with the usual one-capital-letter abbreviation in parentheses immediately following the title.
4. **DESCRIPTIVE NOTES** Enter the category of document, e.g. technical report, technical note or technical letter. If appropriate, enter the type of document, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.
5. **AUTHOR(S)** Enter the name(s) of author(s) as shown on or in the document. Enter last name, first name, middle initial. If military, show rank. The name of the principal author is an absolute minimum requirement.
6. **DOCUMENT DATE** Enter the date (month, year) of Establishment approval for publication of the document.
- 7a. **TOTAL NUMBER OF PAGES** The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.
- 7b. **NUMBER OF REFERENCES** Enter the total number of references cited in the document.
- 8a. **PROJECT OR GRANT NUMBER** If appropriate, enter the applicable research and development project or grant number under which the document was written.
- 8b. **CONTRACT NUMBER** If appropriate, enter the applicable number under which the document was written.
- 9a. **ORIGINATOR'S DOCUMENT NUMBER(S)** Enter the official document number by which the document will be identified and controlled by the originating activity. This number must be unique to this document.
- 9b. **OTHER DOCUMENT NUMBER(S)** If the document has been assigned any other document numbers (either by the originator or by the sponsor), also enter this number(s).
10. **DISTRIBUTION STATEMENT** Enter any limitations on further dissemination of the document, other than those imposed by security classification, using standard statements such as:
 - (1) "Qualified requesters may obtain copies of this document from their defence documentation center."
 - (2) "Announcement and dissemination of this document is not authorized without prior approval from originating activity."
11. **SUPPLEMENTARY NOTES** Use for additional explanatory notes.
12. **SPONSORING ACTIVITY** Enter the name of the departmental project office or laboratory sponsoring the research and development. Include address.
13. **ABSTRACT** Enter an abstract giving a brief and factual summary of the document, even though it may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall end with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (TS), (S), (C), (R), or (U).

The length of the abstract should be limited to 20 single-spaced standard typewritten lines; 7 1/4 inches long.
14. **KEY WORDS** Key words are technically meaningful terms or short phrases that characterize a document and could be helpful in cataloging the document. Key words should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context.

END

3-87

DTIC