

SHIP PRODUCTION COMMITTEE
FACILITIES AND ENVIRONMENTAL EFFECTS
SURFACE PREPARATION AND COATINGS
DESIGN/PRODUCTION INTEGRATION
HUMAN RESOURCE INNOVATION
MARINE INDUSTRY STANDARDS
WELDING
INDUSTRIAL ENGINEERING
EDUCATION AND TRAINING

September 1979
NSRP 0006

THE NATIONAL SHIPBUILDING RESEARCH PROGRAM

Proceedings of the REAPS Technical Symposium

Paper No. 9: SHIPDS-SHIPLO: A Two-Phase Programming System for the Design of Surfaces in Shipbuilding

U.S. DEPARTMENT OF THE NAVY
CARDEROCK DIVISION,
NAVAL SURFACE WARFARE CENTER

Report Documentation Page

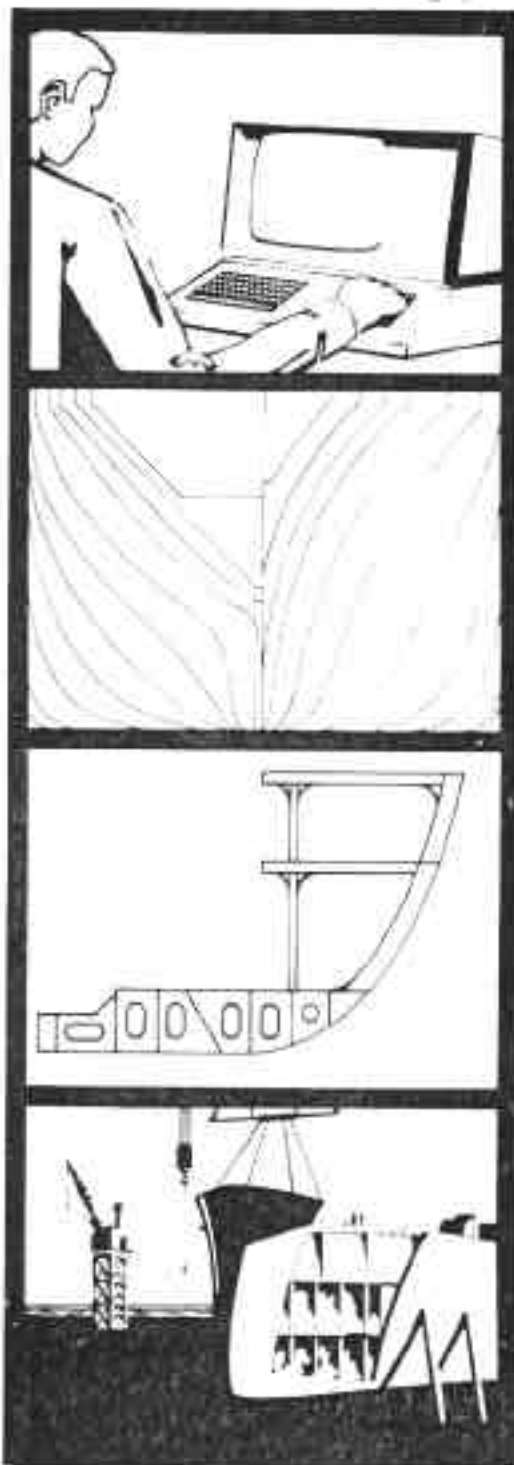
Form Approved
OMB No. 0704-0188

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE SEP 1979	2. REPORT TYPE N/A	3. DATES COVERED -	
4. TITLE AND SUBTITLE The National Shipbuilding Research Program Proceedings of the REAPS Technical Symposium Paper No. 9: SHIPDS-SHIPLO: A Two-Phase Programming System for the Design of Surfaces in Shipbuilding		5a. CONTRACT NUMBER	
		5b. GRANT NUMBER	
		5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)		5d. PROJECT NUMBER	
		5e. TASK NUMBER	
		5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Surface Warfare Center CD Code 2230 - Design Integration Tools Building 192 Room 128 9500 MacArthur Blvd Bethesda, MD 20817-5700		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)	
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release, distribution unlimited			
13. SUPPLEMENTARY NOTES			
14. ABSTRACT			
15. SUBJECT TERMS			
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified	SAR
			18. NUMBER OF PAGES 26
			19a. NAME OF RESPONSIBLE PERSON

DISCLAIMER

These reports were prepared as an account of government-sponsored work. Neither the United States, nor the United States Navy, nor any person acting on behalf of the United States Navy (A) makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness or usefulness of the information contained in this report/manual, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or (B) assumes any liabilities with respect to the use of or for damages resulting from the use of any information, apparatus, method, or process disclosed in the report. As used in the above, "Persons acting on behalf of the United States Navy" includes any employee, contractor, or subcontractor to the contractor of the United States Navy to the extent that such employee, contractor, or subcontractor to the contractor prepares, handles, or distributes, or provides access to any information pursuant to his employment or contract or subcontract to the contractor with the United States Navy. ANY POSSIBLE IMPLIED WARRANTIES OF MERCHANTABILITY AND/OR FITNESS FOR PURPOSE ARE SPECIFICALLY DISCLAIMED.



R ESEARCH
E AND
A NGINEERING
S FOR
P AUTOMATION
S AND
RODUCTIVITY
IN
SHIPBUILDING

Proceedings of the
REAPS Technical Symposium
September 11-13, 1979
San Diego, California

**SHIPDS-SHIPLO: A TWO-PHASE PROGRAMMING SYSTEM
FOR THE DESIGN OF SURFACES IN SHIPBUILDING**

**Andreas Weichbrodt
Computer Science Department
University of Utah
Salt Lake City, Utah**

Mr. Weichbrodt is currently working on his masters thesis, "A System for Computer Aided, Interactive Design of General Arrangements and Simulation of Assembly Sequences (GENASS)", at the University of Utah's Computer Science Department. He is a graduate of the Technical University of Berlin's Department of Naval Architecture and Marine Engineering, where he conducted research on "The Development of a General Purpose Software System for Optimization of Ship Design". Mr. Weichbrodt has also had significant practical training for shipbuilding and marine engineering in Japan and South Africa, and as a member of the German Navy.

S H I P D S - S H I P L O

A TWO PHASE PROGRAMMING SYSTEM FOR
SURFACE REPRESENTATION IN
SHIPBUILDING AND ENGINEERING

Andreas Weichbrodt

Computer Science Department, University of Utah
Salt Lake City, Utah 84 112

ABSTRACT

Several types of surface interpolation techniques exist which are defined over triangular or rectangular surface patches. They all have proven their usefulness in Computer Aided Design.

This paper describes a first approach to a programming system that combines triangular and rectangular interpolation methods and applies them to a typical design problem in shipbuilding, the representation of a ship hull.

The system is designed such that it requires only those data which are available from a common shiplines graph and produces the output in the most general form, i.e. as a set of points $(X, Y, Z(X, Y))$ for the single patches. This output will then be processed in a post-processor fashion by some particular graphical or production device.

INTRODUCTION

Computer Aided Geometric Design (CAGD) is a specialized field in Computer Aided Design (CAD) and focuses on the mathematical representation of arbitrarily shaped univariate curves and bivariate surfaces. It provides several interpolation methods which already have been incorporated in the design process in many companies.

Interpolation over rectangular and triangular surface patches are two important schemes used for surface representation. They have been used mainly to provide the surface data of arbitrarily shaped objects for the purpose of stress analyses and aerodynamic computations. However, the two schemes have not been applied in conjunction with one another yet to solve design problems. Therefore optimal design features for general purpose applications have not been achieved yet, because previous design systems were strongly affected by the advantages and disadvantages of the single methods.

One of the objectives of the project to be described in this paper was to combine the advantages of these two methods and to eliminate or at least reduce the disadvantages to a minimum. Thus the efficiency of the rectangular interpolation schemes is exploited in parts of the surface (e.g. mid-section of a ship's hull) where rectangular patches can be used without any problem. For the more complex shaped parts of the surface (e.g. bow and stern of a ship) the more elaborate but therefore more adaptable triangular schemes are employed (see Fig. 1).

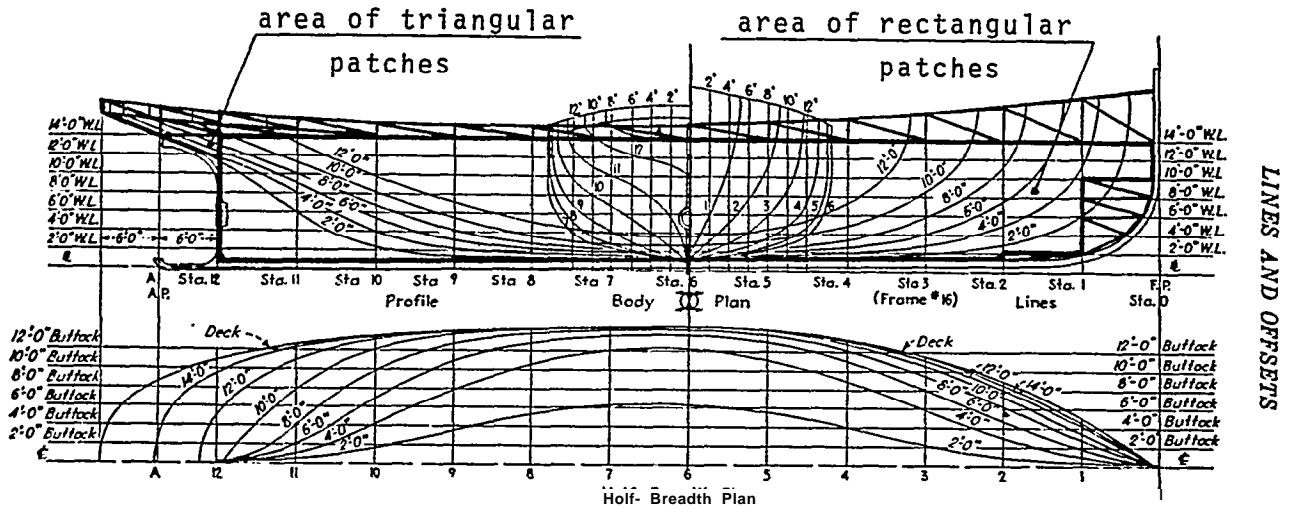


Fig. 1: Areas of rectangular and triangular interpolation on a ship hull surface

The programming system for the surface representation is not intended to be used for interactive ship hull design or fairing. It is meant for the more production oriented phase of the entire design process, at which stage a fairly accurate lines graph (e.g. ship lines graph) is expected to be available.

1. PROGRAM DESCRIPTION

1.1 Input Data for the Program SHIPDS

C^0 - and C^1 -data which can be obtained from the ship lines graph are the only input to the programming system. These are the positional $(X_i, Y_i, Z(X_i, Y_i))$ and the slope $(DZ(X_i, Y_i)/DX, DZ(X_i, Y_i)/DY)$ data at each of the vertices of the single patch i (see Fig. 2).

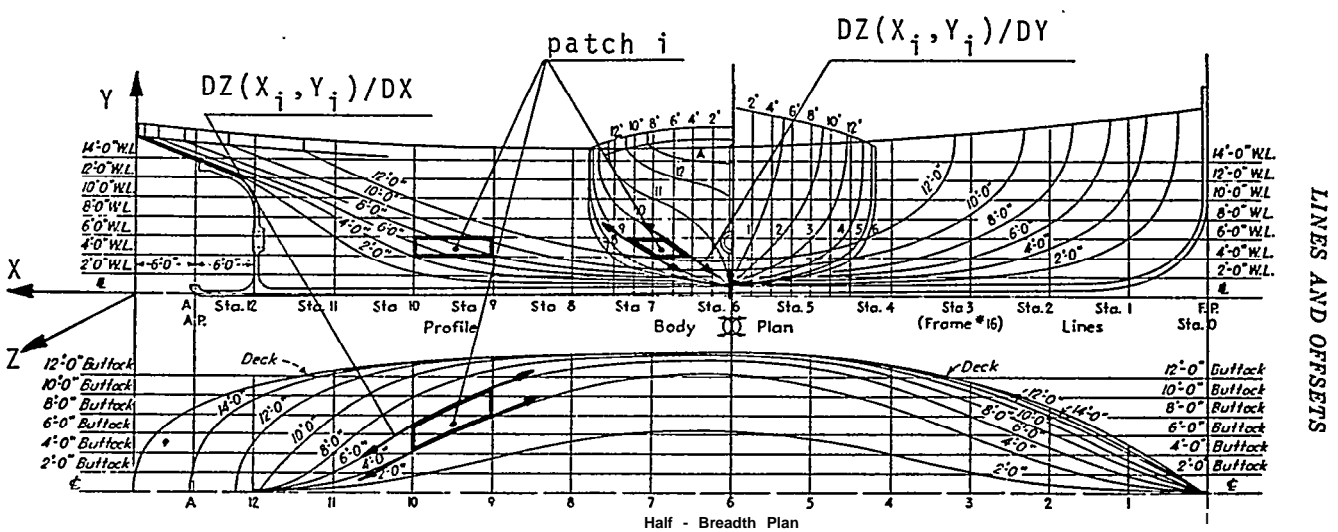
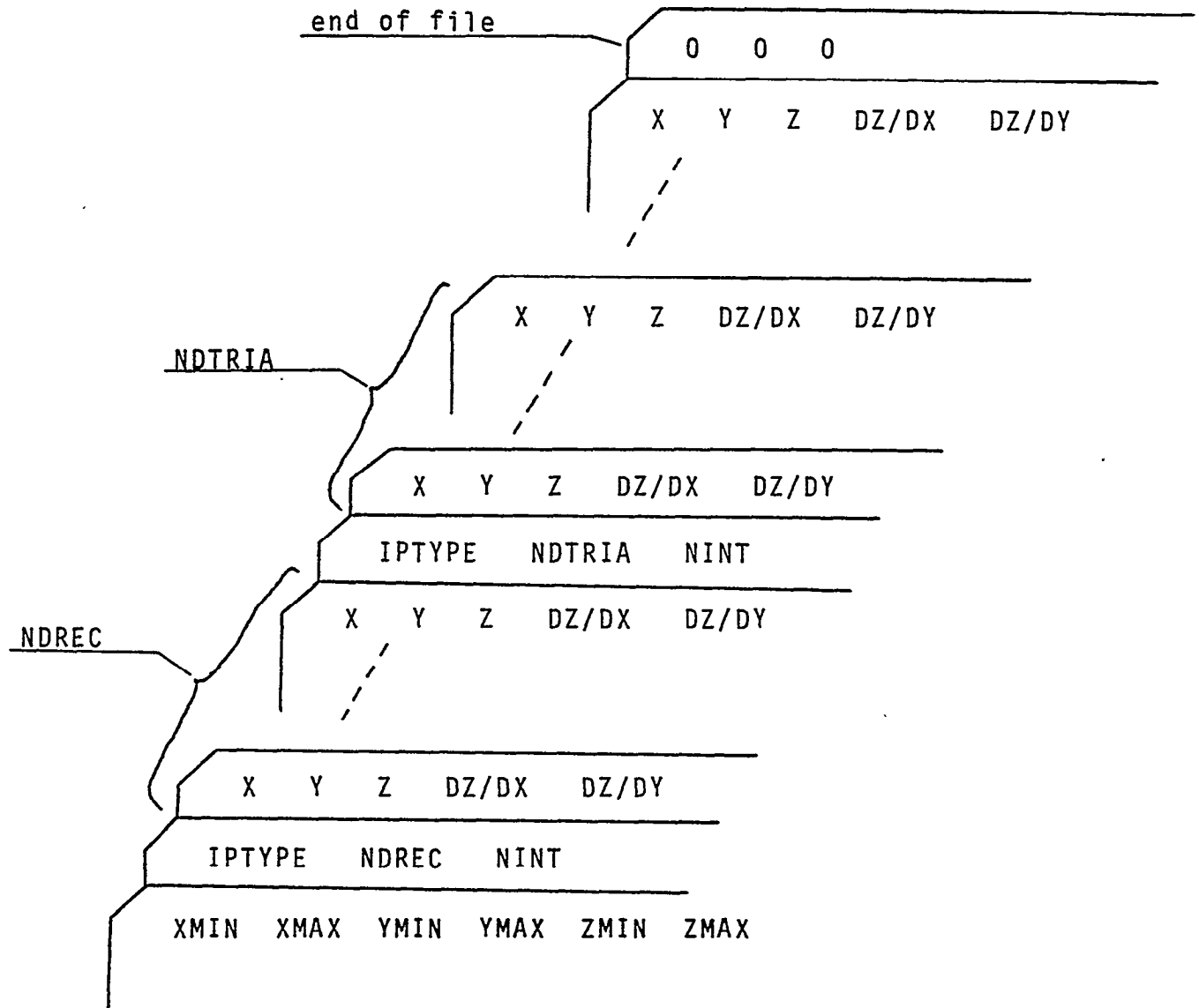


Fig. 2: C^0 - and C^1 - data obtained from the ship lines graph

The patches can be defined easily on the grid that is used to draw the ship lines. But it does not need to be the same grid if bigger patches are sufficient for the required interpolation accuracy.

The principal input sequence in form of data cards is shown in Fig. 3. Fig. 4 illustrates this sequence in the form of records in a data file for the particular case of harbor tug Huntington 7/.



NDREC = number of rectangular patches in a set
NDTRIA = number of triangular patches in a set
IPTYPE = patch type, 3: triangular patch, 4: rectangular patch
NINT = number of intervals on the interpolated patch
X, Y, Z = position at each vertex of the patch
DZ, DZ = gradient at each vertex of the patch
DX DY
XMN, YMN, ZMN = lower limits for the X, Y, Z data
XMAX, YMAX, ZMAX = upper limits for the X, Y, Z data

Fig. 3: Principle sequence of input data for the program SHIPDS

0.0	108.0	-54.0	54.0	54.0	54.0	
4	36	5				
0.0	10.0	0.0	0.44	0.00	}	1. rectangular patch
8.0	10.0	3.5	0.44	0.34		
0.0	12.0	0.0	0.53	0.00		
8.0	12.0	4.2	0.50	0.25		
0.0	12.0	0.0	0.53	0.00	}	2. rectangular patch
8.0	12.0	4.2	0.50	0.25		
0.0	14.0	0.0	0.60	0.00		
8.0	14.0	4.8	0.53	0.15		
8.0	2.0	0.0	0.04	0.54		
16.0	2.0	0.8	0.09	1.09		
8.0	4.0	1.0	0.14	0.50		
16.0	4.0	3.0	0.28	1.00		
:						
:						
3	10	5				
8.0	10.0	3.5	0.44	0.34	}	1. triangular patch
0.73	8.0	0.0	0.28	0.15		
0.0	10.0	0.0	0.44	0.00		
8.0	10.0	3.5	0.44	0.34	}	2. triangular patch
0.73	8.0	0.0	0.28	0.15		
8.0	8.0	2.8	0.39	0.38		
:						
:						
:						
3	2	5				
96.0	12.0	5.0	-1.38	4.00	}	last triangular patch
97.8	14.0	8.6	-0.94	1.75		
96.0	14.0	9.5	-0.54	1.31		
102.0	14.0	0.0	-4.50	9.42		
97.8	12.0	0.0	-3.00	6.28		
97.8	14.0	8.6	-0.94	1.75		
0	0	0				

Fig. 4: Example of an input sequence illustrated on the harbor tug Huntington

There is no restriction imposed on the order of patch definition nor on the number of patches in a set. The variable NINT controls the grid size on which the interpolation values are computed or, in other words, controls the density of interpolation points on a surface patch. The values zero for the patch type IPTYPE, number of patches NPATCH and number of intervals on a patch NINT, signal the end of the input stream and terminate the execution of the program SHIPDS.

1.2 Output Data from the Program SHIPDS and Input Data for the Program SHIPLO

All patches are processed independently and sequentially as they are defined in the input stream (see Fig. 3 and Fig. 4). Therefore the sequence of output data from the program SHIPDS is of similar form. Each interpolated patch is described accordingly by a set of triples, the coordinates of the interpolated points X,Y,Z(X,Y). Fig. 5 illustrates the principal arrangement of the output data in form of data cards. Fig. 6, however, demonstrates this arrangement for the particular case of harbor tug Huntington in the form of records in the output file. Here also the values zero for the patch type IPTYPE and the number of interpolated points on the patch NDRES indicate the end of the output file from the program SHIPDS.

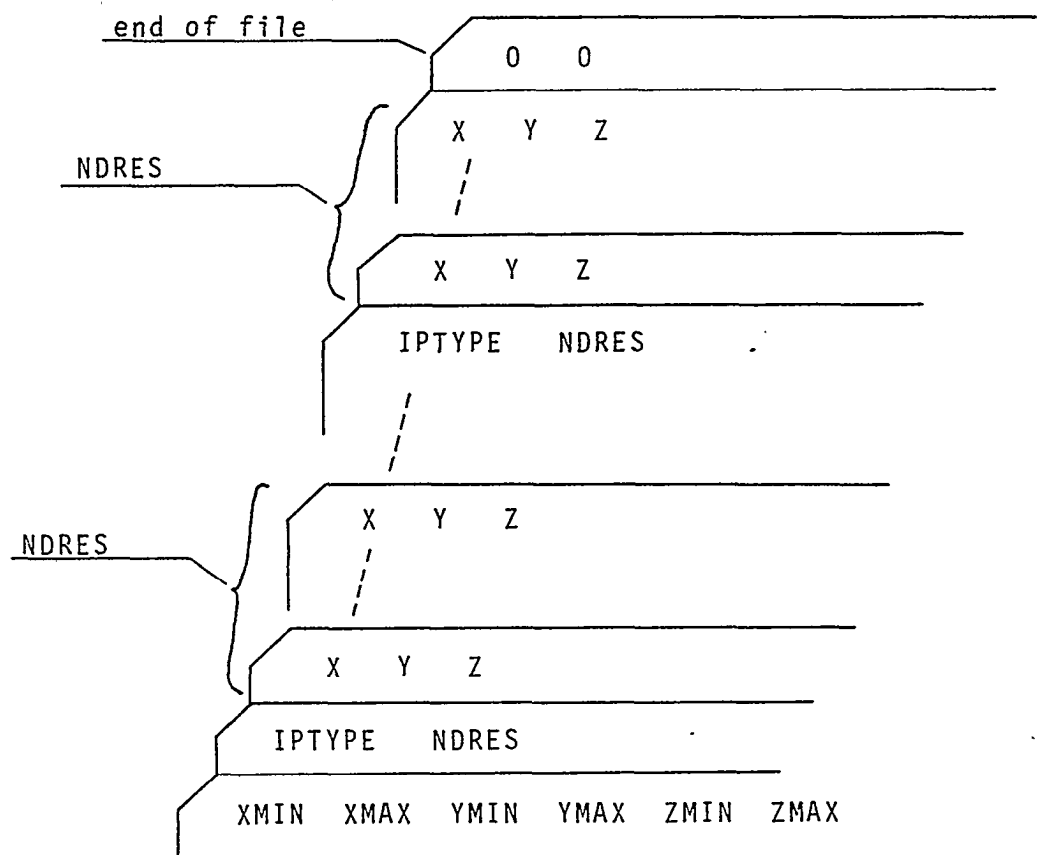
The output data from the program SHIPDS are also treated as input data to the program SHIPLO, which represents any kind of graphical or production device.

It should be mentioned that the interpolation points are computed on a grid of lines with constant values of the free variables. This property normally can be exploited by the interface program for the plotting device. In case of rectangular patches the grid of the interpolation points coincides with the $X = \text{const.}$ and $Y = \text{const.}$ lines (see Fig. 7). In the case of triangular patches, however, the points are computed on lines with constant barycentric coordinates. Those lines are parallel to the edges of a triangular patch (see Fig. 7).

1.3 Structure of the Programming System SHIPDS - SHIPLO

The system is called a "two phase" programming system, since it is separated into two distinct, and to some extent independent, parts.

SHIPDS represents the "first phase" of the system and performs the interpolation computations over the rectangular or triangular patches. The main objectives for the design of this part of the system were portability and flexibility. This could be accomplished by using standard FORTRAN IV as the most widely used programming



IPTYPE = patch type; 3: triangular patch; 4: rectangular patch
NDRES = number of data for single patch
X, Y, Z = coordinates of the interpolated points on the patch
XMIN, YMIN, ZMIN = lower limits for the X, Y, Z data (from input-file)
XMAX, YMAX, ZMAX = upper limits for the X, Y, Z data

Fig. 5: Principle sequence of output data from the program SHIPDS and sequence of input data for the program SHIPLO

.000000E+00	108.000	-54.0000	54.0000	-54.0000	54.0000
4	36				
.000000E+00	10.0000	.000000E+00	}	first rectangular patch	
.000000E+00	10.4000	.000000E+00			
.000000E+00	10.8000	.000000E+00			
.000000E+00	11.2000	.000000E+00			
.000000E+00	11.6000	.000000E+00			
.000000E+00	12.0000	.000000E+00			
1.60000	10.0000	.701920			
1.60000	10.4000	.732037			
1.60000	10.8000	.763048			
1.60000	11.2000	.794062			
1.60000	11.6000	.823938			
1.60000	12.0000	.851520			
3.20000	10.0000	1.40096			
.					
.					
4.80000	12.0000	2.55264			
6.40000	10.0000	2.79808			
6.40000	10.4000	2.91963			
6.40000	10.8000	3.04862			
6.40000	11.2000	3.17573			
6.40000	11.6000	3.29159			
6.40000	12.0000	3.38688			
8.00000	10.0000	3.50000			
8.00000	10.4000	3.64384			
8.00000	10.8000	3.79632			
8.00000	11.2000	3.94688			
8.00000	11.6000	4.08496			
8.00000	12.0000	4.20000			
.					
.					
.					
3	21		}	last triangular patch	
97.8000	14.0000	8.60000			
97.8000	13.6000	7.65952			
97.8000	13.2000	6.27456			
97.8000	12.8000	4.49984			
97.8000	12.4000	2.39008			
97.8000	12.0000	.000000E+00			
98.6400	14.0000	7.80506			
98.6400	13.6000	6.62360			
98.6400	13.2000	5.01034			
98.6400	12.8000	2.79933			
98.6400	12.4000	-.319996E-02			
99.4800	14.0000	6.81869			
99.4800	13.6000	5.19569			
99.4800	13.2000	3.08825			
99.4800	12.8000	.145519E-08			
100.320	14.0000	5.36979			
100.320	13.6000	3.13624			
100.320	13.2000	.479994E-02			
101.160	14.0000	3.18726			
101.160	13.6000	.639985E-02			
102.000	14.0000	.000000E+00			
0	0				

Fig. 6: Example of an output sequence (harbor tug Huntington)

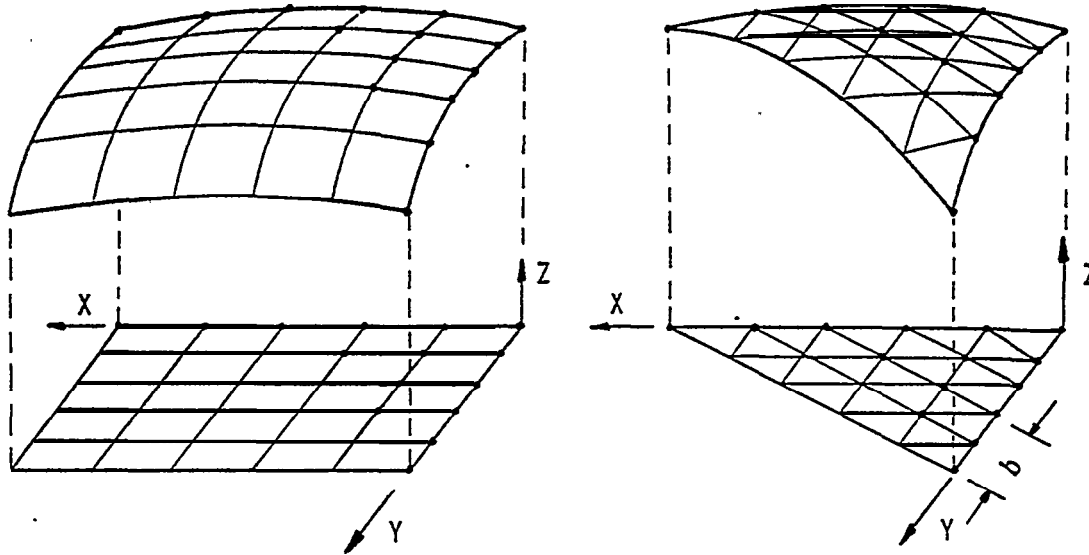


Fig. 7: Interpolation grid of a rectangular and triangular surface patch

language in engineering and by following strictly the rules of the standard.

Exploiting the FUNCTION- and SUBROUTINE-features the program SHIPDS could be designed in a very modular structure (see Fig. 8), such that the rectangular and triangular interpolation schemes as a whole could be replaced by other techniques easily. This also permits the use of those interpolation programs in other than just the SHIPDS-environment.

The programs RPINT for rectangular patch and TPINT for triangular patch interpolation themselves are of very modular structure. Thus different techniques for interpolating the edge functions of a patch or approximating tangential and cross boundary derivatives along the edges could be substituted by better and/or more appropriate ones.

The freedom of choice for the different interpolation methods, however, is restricted by the type of data they need. Since most techniques do not require more than the C^0 and C^1 data as described in chapter 1.1 at the vertices of a patch and since those data all can be obtained from the ship lines graph (see Fig. 2), the system deliberately was confined exclusively to those data.

The output data from SHIPDS and also input data for SHIPLO had to be provided in a most general form, because the type of graphical or production device is not known in advance and will vary from application to application. The essential information each kind of graphical or production device will need is the coordinates of the interpolation points on a patch together with the number of points thereon and the code for the type of the patch (see Fig. 5 and Fig. 6).

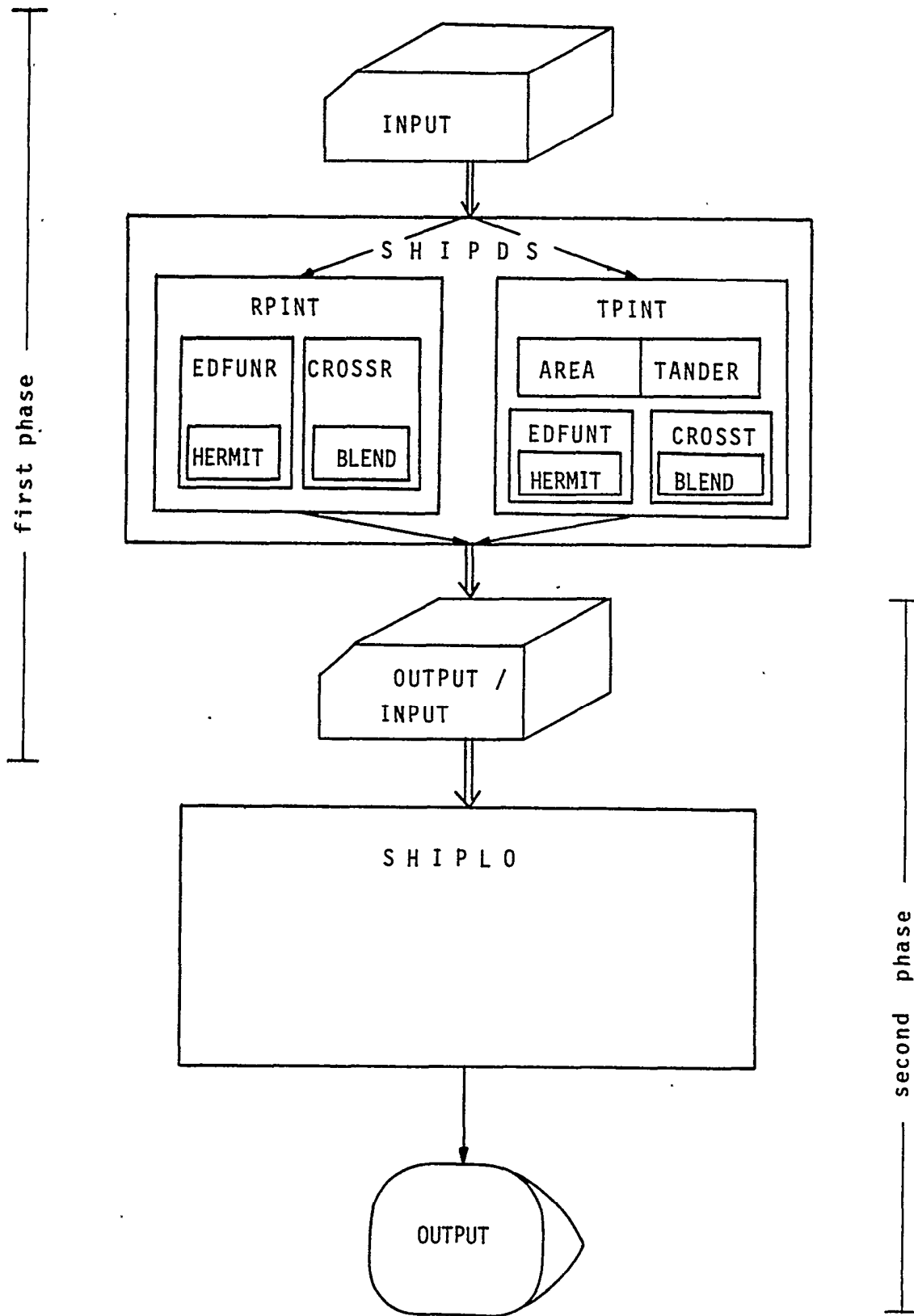


Fig. 8: Structure of the programming system SHIPDS - SHIPLO

The "second phase" of the programming system, represented by the program SHIPLO, is highly device dependent. It is an interface program that needs to be modified or completely rewritten for a new application of the system. It, therefore, was written as a single program and very closely adjusted to the output device.

2. THEORY

2.1 Interpolation Schemes over Rectangular Patches

The parametric interpolation method over rectangles was introduced by S. A. Coons [1]. Therefore they are also called "Coons" patches.

This method uses the boolean sum $(P1 \oplus P2)F$ that interpolates to the values of the corners and edges of a rectangular patch. It is defined as follows:

$$(P1 \oplus P2)F = P1F + P2F - P1P2F \quad (1)$$

with the projectors

$$P1F = \begin{bmatrix} h_0(u) & h_1(u) & \bar{h}_0(u) & \bar{h}_1(u) \end{bmatrix} \begin{bmatrix} F(0,v) \\ F(1,v) \\ F_{10}(0,v) \\ F_{10}(1,v) \end{bmatrix} \quad (2)$$

$$P2F = \begin{bmatrix} F(u,0) & F(u,1) & F_{01}(u,0) & F_{01}(u,1) \end{bmatrix} \begin{bmatrix} h_0(v) \\ h_1(v) \\ \bar{h}_0(v) \\ \bar{h}_1(v) \end{bmatrix} \quad (3)$$

$$P1P2F = \begin{bmatrix} h_0(u) & h_1(u) & h_0(u) & h_1(u) \end{bmatrix} \begin{bmatrix} B \end{bmatrix} \begin{bmatrix} h_0(v) \\ h_1(v) \\ \bar{h}_0(v) \\ \bar{h}_1(v) \end{bmatrix} \quad (4)$$

and with

$$B = \begin{bmatrix} F(0,0) & F(0,1) & F_{01}(0,0) & F_{01}(0,1) \\ F(1,0) & F(1,1) & F_{01}(1,0) & F_{01}(1,1) \\ F_{10}(0,0) & F_{10}(0,1) & F_{11}(0,0) & F_{11}(0,1) \\ F_{10}(1,0) & F_{10}(1,1) & F_{11}(1,0) & F_{11}(1,1) \end{bmatrix} \quad (5)$$

and the cubic Hermite functions

$$\begin{aligned} h_0(t) &= (1-t)^2 (2t+1) & \bar{h}_0(t) &= (1-t)^2 t \\ h_1(t) &= t^2 (-2t+3) & \bar{h}_1(t) &= t^2 (t-1), \quad t \in [0,1]. \end{aligned} \quad (6)$$

$$F(u,0), F(u,1), F(0,v), F(1,v) \quad (7)$$

denote the edge functions of the patch (see Fig. 9) and

$$\left. \begin{matrix} F_{01}(u,0) \\ F_{01}(u,1) \end{matrix} \right\} = \left\{ \begin{matrix} DF(u,0)/Du \\ DF(u,1)/Dv \end{matrix} \right\} \text{ and } \left\{ \begin{matrix} F_{10}(0,v) \\ F_{10}(1,v) \end{matrix} \right\} = \left\{ \begin{matrix} DF(0,v)/Du \\ DF(1,v)/Dv \end{matrix} \right\} \quad (8)$$

denote the cross boundary derivatives along the edges.

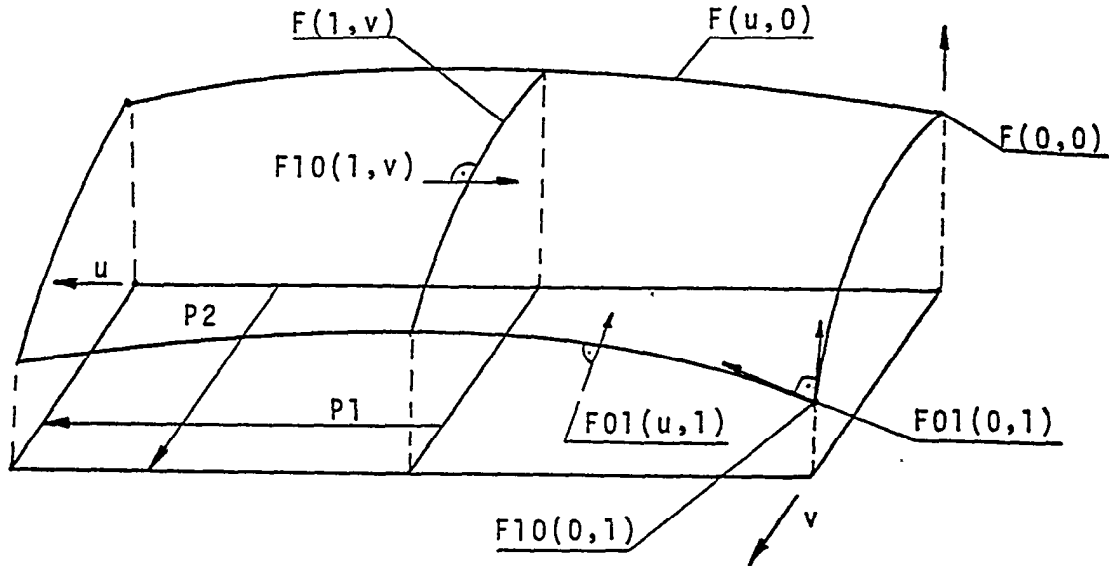


Fig 9: Bicubically blended Coons patch

The edge functions are also interpolated by cubic Hermite functions (see (6)) using the C - and C -data provided as input for each patch (see chap. 1.1):

$$F(u,1) = \begin{bmatrix} h_0(u) & h_1(u) & h_0(u) & h_1(u) \end{bmatrix} \begin{bmatrix} F(0,1) \\ F(1,1) \\ F_{10}(0,1) \\ F_{10}(1,1) \end{bmatrix} \quad (9)$$

$$F(1,v) = \begin{bmatrix} h_0(v) & h_1(v) & h_0(v) & h_1(v) \end{bmatrix} \begin{bmatrix} F(1,0) \\ F(1,1) \\ F_{01}(1,0) \\ F_{01}(1,1) \end{bmatrix} \quad (10)$$

The cross boundary derivatives are linearly blended together:

$$F_{01}(u,1) = (1-u) F_{01}(0,1) + u F_{01}(1,1) \quad (11)$$

$$F_{10}(1,v) = (1-v) F_{10}(1,0) + v F_{10}(1,1) \quad (12)$$

The use of the cubically blended Coons patch generally suffers from the approximation of the twist partition in the matrix B:

$$T = \begin{bmatrix} F_{11}(0,0) & F_{11}(0,1) \\ F_{11}(1,0) & F_{11}(1,1) \end{bmatrix}. \quad (13)$$

In order to keep the problem simple, the twist partition T is approximated by zero. But it normally does not result in a sufficient surface interpolation. One of several different techniques for approximating the twist terms is the so called "Gregory's Square" which is a 12-Parameter scheme (see /2/ for more details).

Since adjacent patches have the same positions and gradients at the patch vertices as obtained from the ship lines graph and since their common edge functions are interpolated by the same cubic Hermite functions and the cross boundary derivatives by the same linear blends, the surface represented by rectangular patches must be a C1-surface.

2.2 Interpolation Schemes over Triangular Patches

Barnhill, Birkhoff and Gordon (BBG) initiated in 1973 /9/ the "triangular" Coons patch by applying the boolean sum to the three projectors of the standard triangle (see Fig. 10):

$$(P_i \oplus P_j)F = P_i F + P_j F - P_i P_j F \quad \begin{matrix} i, j=1,2,3 \\ i \neq j \end{matrix} \quad (14)$$

with

$$P_1 F = \begin{bmatrix} h_0(r) & h_1(r) & h_0(r) & h_1(r) \end{bmatrix} \begin{bmatrix} F(0, \alpha) \\ F(1-\alpha, \alpha) \\ F_{10}(0, \alpha) \\ F_{10}(1-\alpha, \alpha) \end{bmatrix} \quad \text{and} \quad (15)$$

$$r = P/(1-P) \quad (15)$$

P2F and P3F are defined analogously. This interpolant interpolates to the values at the corners and edges of the standard triangle.

In order to apply the BBG-interpolant to an arbitrary triangle affine transformations on the standard triangle need to be performed. Positions and directions of the tangential derivatives are preserved under affine transformation but not the directions of the cross boundary derivatives. Therefore they need special treatment as described in more details in /2/ and /4/.

The "Brown - Little" triangle, however, is a C¹-interpolation scheme for arbitrary triangles that interpolates to the values at the corners and edges of the triangular patch:

$$(BL)F = A*P_1 + B*P_2 + C*P_3 \quad (17)$$

with the weight functions

$$A = b_2^2 * b_3^2, D \quad B = b_1 * b_3^2, n \quad C = b_1 * b_2 / D \quad \text{and}$$

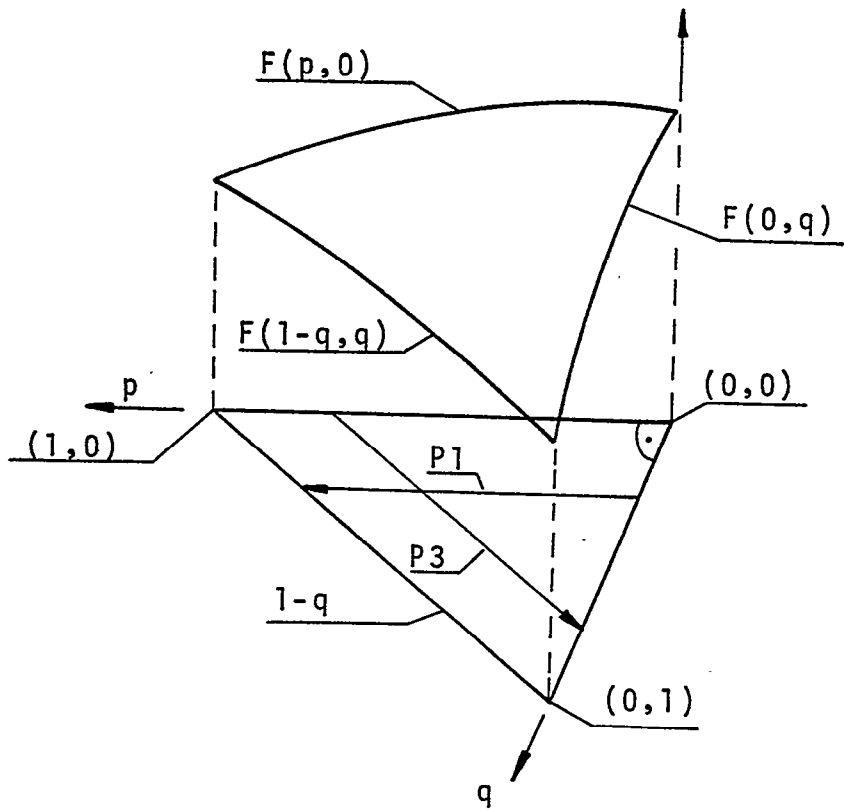


Fig. 10: BBG - interpolant on standard triangle

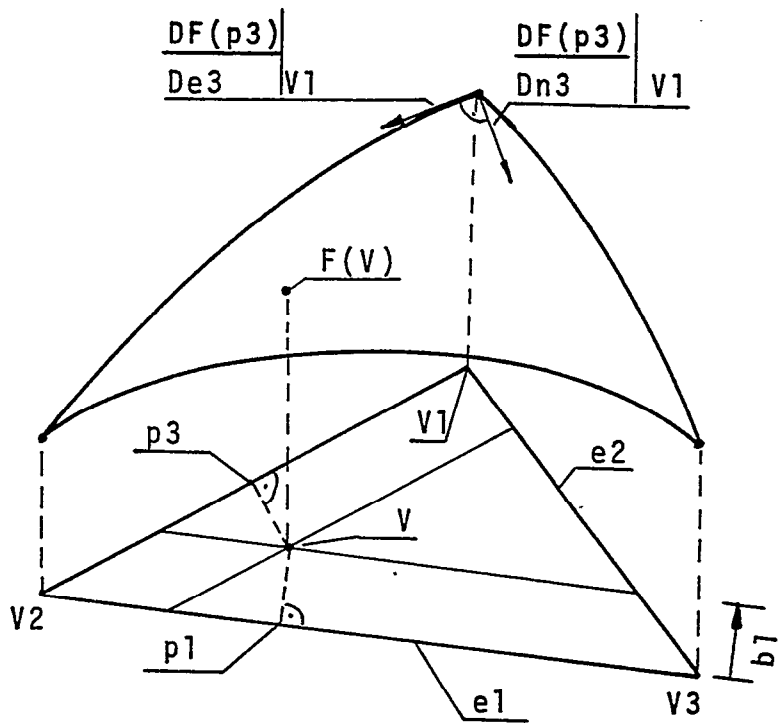


Fig. 11: Brown - Little triangle

$$D = b_1^2 * b_2^2 + b_1^2 * b_3^2 + b_2^2 * b_3^2 \quad (13)$$

The b_i , $i=1,2,3$ are the barycentric coordinates at the point V with the property:

$$b_1 + b_2 + b_3 = 1 \quad (19)$$

The physical coordinates X , Y can be computed by:

$$X = \sum_{i=1}^3 b_i * X_i, \quad Y = \sum_{i=1}^3 b_i * Y_i \quad (20)$$

The projectors P_iF are defined as follows:

$$P_iF(X,Y) = [(F(p_i) + b_i * 2AR) / (V_{i+1} - V_{i+2}) * DF(p_i) / Dn_i] \quad (21)$$

where AR denotes the area of the projection of the triangular patch onto the XY -plane and $DF(p_i)/Dn_i$ the cross boundary derivative along edge i . The V_i 's are the patch vertices and the p_i 's are the parameters which vary between the values of the vertex coordinates, which enclose the same edge. They are the projections of an arbitrary chosen interpolation point V within the patch onto the single edges (see Fig. 11).

In the case of the triangular patches the edge functions are also interpolated by cubic Hermite functions using the C^0 and C^1 -data from the graph. Since adjacent patches need to have at least one edge in common, the two corresponding vertices need to have the same positional and slope data at those points, which are computed from the gradients. Therefore the edge functions of adjacent triangular but also adjacent triangular and rectangular patches will be identical and the surface constructed from the triangular patches will be C^1 .

3. RESULTS

3.1 Computational Problems

In the first stage of designing the programming system SHIPDS-SHIPLO, most of the concern was dedicated to the program organisation (see chap. 1.3) rather than to the perfection of the interpolation programs. Therefore those interpolation programs, which were already available, have been adopted accordingly to the requirements of the program organisation. Thus the bicubically blended Coons patch with the zero twist partition (see (1) and (5)) for the rectangular patches and the Brown-Little triangle (see (16)) for triangular patches have been implemented. The results are shown in Fig. 12. They are surely not satisfactory yet for most "real world" applications. The poor quality of the interpolation is partly caused by the inaccuracy of the input data, which have been manually read off the graph shown in Fig. 1 and

Fig. 2.

The cubic Hermite functions for interpolating the edge curves of the surface patches have been used in nonparametric form so far. This also creates problems, since their interpolation accuracy decreases very fast with the increase of the slopes of the tangents at the patch corners.

* The tangential and cross boundary derivatives needed for both the triangular and rectangular patches are computed from the gradients at the vertices. In case of the rectangular patches, in fact, they are identical with the gradient components and this part of the ship's hull surface indeed is a C^1 - surface and interpolates correctly **over** its entire region.

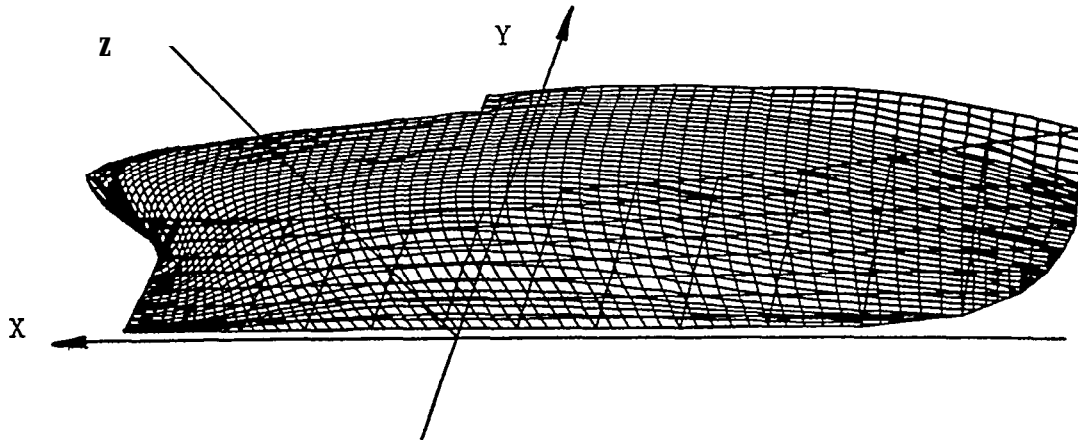
The triangular interpolation scheme also interpolates a C^* **surface as** expected. But the local, (i.e. in the interior of the patch) interpolation accuracy, especially in parts of the surface of high curvature as the stern of the ship, is very low. Noncollinear edges of triangular patches along the surface boundaries, on the other hand, create an inconsistency for the local interpolation along the boundaries. This problem has not yet been solved theoretically and results in inaccurate interpolation along the surface boundaries. It will depend on the type of the surface as well as on the particular application whether the final interpolation will be sufficient or if other correction steps have to be taken (see Fig. 12 and Fig.13).

Discontinuities in a smooth surface as they might be intended and **occur** in the case of the harbor tug Huntington (see Fig. 1 and Fig. 2), can successfully be treated. But they have to be restricted to follow exactly the edges of just the rectangular patches.

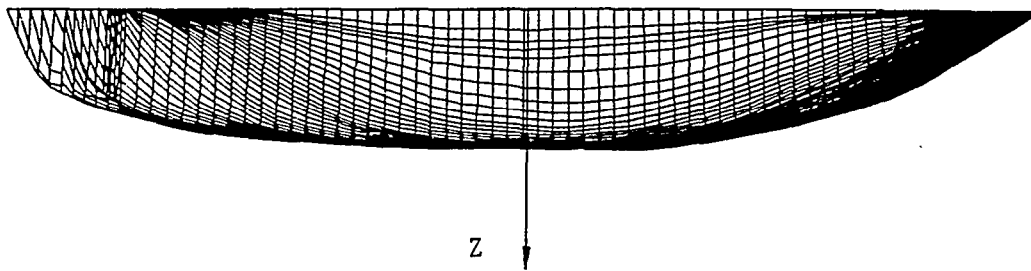
3.2 Graphical Problems

As example for the output device for the "second phase" of the system a TEKTRONIX - storage tube has been chosen, that was equipped with a transformation package to create 3-D images.

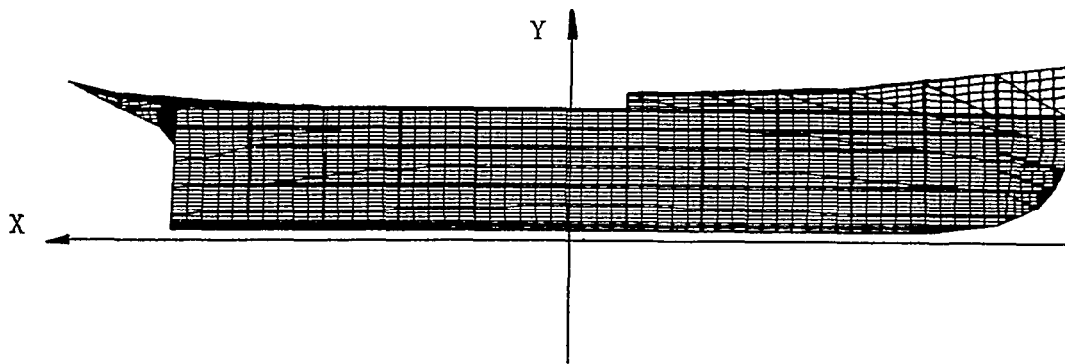
In order to obtain a reasonable impression of a 3-D image on a 2-D TEKTRONIX - screen a regular texture on the surface patches is needed. This can be accomplished easily for rectangular and triangular patches separately. But since the interpolation points of the triangular and rectangular patches are computed on different grids (see chap. 1.2 and Fig. 7) it is not easy to achieve a completely homogeneous texture on this kind of graphical device and thus avoid the considerable distortion of the principally correct results.



Harbor tug Huntington (3-D front view)



Harbor tug Huntington (top view)



Harbor tug Huntington (side view)

Fig. 12: First results from the programming system SHIPDS - SHIPL0

4. NECESSARY AND POSSIBLE IMPROVEMENTS

In order to obtain results that are acceptable for 'real world' applications of the system some improvements have to be achieved. Thus the rectangular interpolation scheme with the 'twist approximation equal to zero has to be replaced by a better method. The triangular interpolant, the Brown-Little triangle, is known for its poor approximation of the interior of the patch. It also will be substituted by an acknowledged better method; Because of the modular structure of the program SHIPDS those changes can be performed easily.

The nonparametric interpolation used for the edges of the triangular and rectangular patches will be sufficient for many problems. It is desirable, however, to implement a parametric version in order to be more general and to avoid problems with extrem slope values as they occur at the bottom and the stern of a ships hull.

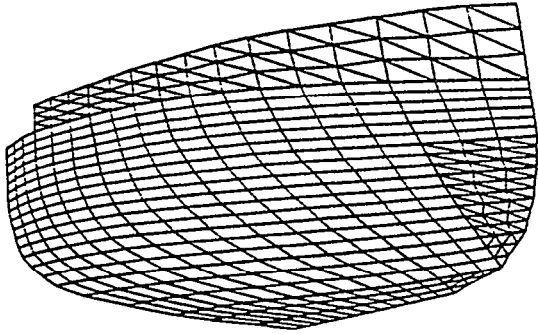
As Fig. 13 can prove those "better" interpolation schemes are available as part of the surface editor SURFED /8/ of the CAGD group at the University of Utah. The triangular and rectangular methods can be combined in order to represent a complex surface efficiently like the one of a ships hull, For the particular instance of the harbor tug Huntington (see Fig. 13) the more advanced methods, which have been applied, are a modified version of the "Gregory's Square" - method that can even interpolate quadrilateral patches as a superset of rectangular patches and the BBG - method for arbitrary triangles.

CONCLUSION

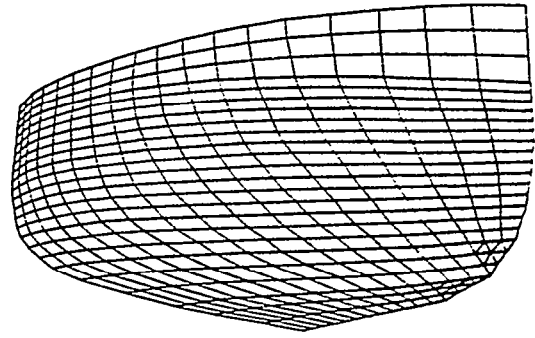
The current version of the two phase programming system has proven that the program organisation can meet the expectations for a production oriented application. The objectives of portability, generality, flexibility and efficiency can be achieved.

The quality of the interpolation or, better, of the representation of a complex shaped surface depends very much on the methods chosen. There are several methods available that can be incorporated into the system and that allow to combine triangular and rectangular patches and thus exploit the efficiency of the rectangular interpolation schemes with the accuracy of global shape approximation of the triangular patches.

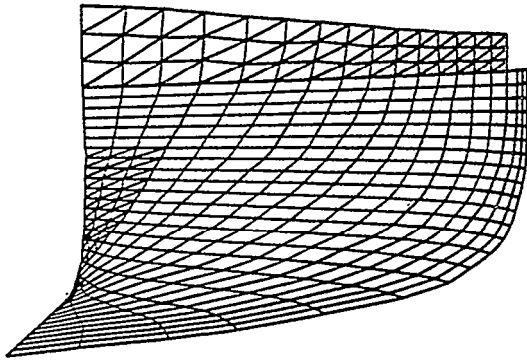
The second phase of the current version of the programming system was designed and performed for the purpose of illustration rather than for final application. It will have to be changed for a particular application of graphical or production device.



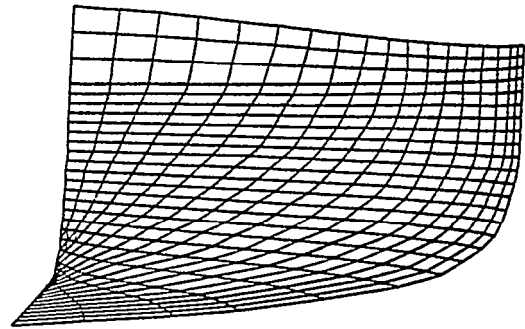
**triangular and rectangular
patches (front view)**



**triangular and quadrilateral
patches (front view)**



**triangular and rectangular
patches
(front view from the inside)**



**triangular and quadrilateral
patches
(front view from the inside)**

Fig. 13: Results from improved interpolation methods illustrated on harbor tug Huntington

Since the output data of the first phase are provided in the most general form and since the interpolation points are computed on a regular grid, the second phase of the programming **system** is not limited only to the pure interfacing task. It also could include simple numerical integration methods in order to provide additional features, such as the computation of volumes, centers of gravity etc. of various parts of the ship.

The system needs only very little core size (10K words), so that it can be installed even on smaller computers and thus offer its benefits also to the small shipbuilding and engineering companies.

ACKNOWLEDGEMENTS

This research project was supported in part by the National Science Foundation Grant MCS78-01966 and in part by the "Deutscher Akademischen Austauschdienst" in Bonn, Germany. The author greatly appreciates the help and advice from the CAGD-group at the University of Utah in general and the moral support from Robert E. Barnhill and Richard F. Riesenfeld in particular.

REFERENCES

- /1/ Coons, S. A. Surfaces for Computer Aided Design of Space Forms, Project MAC, Massachusetts Institute of Technology, March 1967
- /2/ Barnhill, R. E. Representation and Approximation of Surfaces, Reprinted from Mathematical Software III, Academic Press, 69-120, 1977
- /3/ Barnhill, R. E. A **New** Twist in Computer Aided Geometric Design, Brown, J. H. Computer Graphics and **Image** Processing 8, Klucewicz, M. I. 78-91, 1978
- /4/ Klucewicz, M. I. A Piecewise C^1 Interpolant to Arbitrarily Spaced Data, M.s. thesis, Mathematics Department, University of Utah, 1977

- /5/ Faux, I. D. Computational Geometry for Design and
Pratt, M J. Manufacture,
Ellis Horwood Publishers, 1979
- /6/ Barnhill, R. E. Computer Aided Geometric Design,
Riesenfeld, R. F. Academic Press, 1974
- /7/ Baker, B. S. Introduction to Steel Shipbuilding,
McGraw-Hill Books Company, 1943
- /8/ Dube, R. SURFED - An Interactive Editor for
Herron, G. J. Free-Form Surfaces,
Little, F. F. Computer Aided Design, volume 10
Riesenfeld, R. F. number 2, 1978
- /9/ Barnhill, R. E. Smooth Interpolation in Triangles,
Birkhoff, G. J. Approx. Theory 8, 114-128, 1973
Gordon, W. J
- /10/ Nowacki, H. Computer Aided Ship Design, Lecture Notes
The Department of Naval Architecture and
Marine Engineering, College of Engineering,
University of Michigan, Nr. 018, 1970

Additional copies of this report can be obtained from the
National Shipbuilding Research and Documentation Center:

<http://www.nsnet.com/docctr/>

Documentation Center
The University of Michigan
Transportation Research Institute
Marine Systems Division
2901 Baxter Road
Ann Arbor, MI 48109-2150

Phone: 734-763-2465
Fax: 734-763-4862
E-mail: Doc.Center@umich.edu