GOMADS – AN INTERACTIVE EDITING SYSTEM FOR HYDROGRAPHIC CHARTS

J.L. Kestner and S.R. Oraas
Canadian Hydrographic Service

Abstract

GOMADS is a PDP-11, storage display, FORTRAN based graphic editing system developed as part of the automated cartography project of the Canadian Hydrographic Service. Cartographic applications usually require a data set containing thousands of inches of curvy lines. GOMADS uses a three level data structure in order to minimize the amount of data searching required in windowing and in locating features.

Unlike many other graphic editing systems which use a cursor following technique for entering curvy lines, in GOMADS a cubic spline curve fitting routine is used to fit a line to a series of points.

Commands follow a prompt-response-verification format. Changes are never made to the data until the complete change has been displayed and verified by the user.

A text editing facility is available to edit the information describing the graphic data.

Future developments include the addition of an on-line digitizing table.

GOMADS – SYSTÈME D'ÉDITION INTERACTIF POUR LES CARTES HYDROGRAPHIQUES

Résumé

GOMADS est un système d'édition graphique à affichage à mémoire utilisant le Fortran et fonctionnant sur ordinateur PDP-11; il a été développé dans le cadre du projet de cartographie automatisée du Service hydrographique du Canada. Les applications cartographiques font habituellement appel à un ensemble de données comportant des milliers de pouces de lignes courbes. GOMADS utilise une structure de données à trois niveaux visant à réduire au minimum la recherche nécessaire pour cerner et localiser des caractéristiques.

À l'encontre de nombreux autres systèmes d'édition graphique qui utilisent un curseur pour l'introduction des courbes, le système GOMADS utilise un programme d'ajustement de courbes à fonction cubique.

Les commandes s'effectuent selon un format suggestion-réaction-vérification. Des modifications ne peuvent être apportées aux données qu'après affichage et vérification par l'utilisateur du changement complet.

Un dispositif d'édition du texte permet d'éditer l'information décrivant les données graphiques.

On prévoit ajouter au système une table de numérisation en ligne.

GOMADS - An Interactive Editing System for Hydrographic Charts

1. INTRODUCTION

GOMADS, an acronym for Graphical On-line Manipulation and Display System, is an interactive editing system designed to meet the needs of the Canadian Hydrographic Service. The system was developed during the period from June, 1975 to August 1976 by the Automated Cartography unit at C.H.S. headquarters, Ottawa. This document gives a brief overview of the system, and describes in some detail those features that make GOMADS unique.

The subject of this paper is limited to interactive editing of digital graphics within a comprehensive "automated" cartographic system. The complete system is not described except to note that it includes digitizing and plotting facilities. It also features software capable of doing many "off-line" or fully automated processes. These processes are done only once per chart and require little or no interaction with the user. They include:

1. Projection and scale change.
2. Selection of a subset (window) by specifying a polygon.
3. Sounding unit conversion.
4. Merging several digital graphics to form a mosaic.
5. Distortion correction.
6. Symbolization.

2. PURPOSE OF THE SYSTEM

The GOMADS system was designed to enable a cartographer within the C.H.S. to manipulate a digital chart file in a manner similar to current manual techniques. The result is basically an interactive graphics editing system, but with the design optimized to meet the particular needs of the C.H.S. The following criteria were used for the design:

1. The system should be able to handle the following data:
   a) 100,000 soundings, stored as X,Y, depth and flag word.
   b) 5000 inches of "curvy" line, at 500 points per inch.
   c) 1000 symbols of the type that normally appear on hydrographic chart.
   d) 2000 names - place names, notes, titles, etc.
   e) some point-to-point lines.

Note that the system is based on the 16-bit word; a coordinate is represented as a pair of 16-bit integers.

2. The user should be able to add, delete, move, and change features in a manner convenient to him using a storage display as the interaction device.
3. Response times should be under five seconds for all disc searches and modifications. Since re-drawing the picture on the display will take longer, it must be possible to make many changes with-

out re-drawing the picture after each one.

4. The system should be easy to use, and as foolproof as possible in operation.

3. IMPLEMENTATION

3.1 Hardware Implementation

GOMADS was implemented on the following equipment:

1. PDP-11/40 CPU with 28K core memory and hardware floating point and extended instruction set.
2. Two RK05 cartridge disc drives, with 2.4 Mbyte capacity each.
3. Two TU10 nine track, 800 bpi magnetic tape drives.
4. Tektronix 4014-1 storage display terminal, with enhanced graphics option and high speed interface (40 Kbaud). Also included is a joystick for controlling the crosshair cursor and a hard copy unit.
5. DEC VT05 alphanumeric CRT terminal, 9.6 Kbaud.

3.2 Software Implementation

GOMADS was implemented using the DOS/BATCH operating system supplied by Digital Equipment. This is a disc-based operating system which permits only one user to operate at any one time. It includes a Fortran IV compiler, a linker which permits complex overlay structures, and a device independent file manipulation system.

The GOMADS software is written almost entirely in Fortran, and the remainder is in PDP-11 assembler code. The assembler routines include the driver for the Tektronix 4014 and a few higher level routines which critically affect the drawing speed on the 4014. All disc input/output is handled by standard Fortran random access statements. The code itself is very modular in design, the system containing about 250 subroutines. It is felt that this modularity greatly enhances the readability and maintainability of the code, as well as permitting modifications to be made easily.

The GOMADS software was found to be too large to fit on a PDP-11 with 28K of core, and hence the overlay capability of the linker had to be used. Currently GOMADS is broken into 22 segments, forming an overlay structure up to 4 levels deep. This structure was arranged so that a minimal number of core swaps occurs during program execution, and the whole procedure is quite transparent to the user.

4. COMMANDS

4.1 Command Overview

The general philosophy behind the design of the command structure was to try to make the commands relatively self-explanatory and as foolproof as possible. To achieve these ends, the following techniques were adopted:

1. Commands follow a "prompt-response" format. If the operator is required to perform some action during a command sequence, the computer prints a message telling him to do so. Most commands can be

executed successfully by an inexperienced user without referring to the user's guide.

2. Whenever the user has to enter a command directing GOMADS what to do next, he can type "HELP" to obtain a listing of the commands available at that particular time.

3. Whenever a feature must be located in the data file for possible modifications, GOMADS will always brighten the feature on the 4014 and print descriptive information on the VT05 terminal. The user must then verify that this is indeed the correct feature. If it is the wrong feature, the user can direct GOMADS to continue searching for the correct one. This technique eliminates problems with overplotted features.

4. No change is ever made to the data file until after the complete change has been specified, drawn on the 4014 (in write-through mode if possible), and verified by the user.

Commands are specified using a four character string, grouped into two sets of two character strings. For editing commands, the first two characters indicate the type of feature to be edited, and the last two characters give the function to be performed. There are four main types of features-lines ("LI"), soundings ("SO"), symbols ("SY") and names ("NA"). The types of functions which can be performed depend on the feature, but generally include deleting, adding, moving, and changing. For example, to delete a sounding the user types "SODE", to delete a line he types "LIDE", and to move a sounding he types "SOMO".

Some commands require only a two character sequence. These include the drawing commands, the file manipulating commands, and miscellaneous others. In general, the commands do not allow any optional parameters to follow them, as it is felt that this greatly increases the complexity of the command. If any parameters are required, GOMADS will ask for them. Often the user will be able to respond to a question by hitting the carriage return key only, and in that case a default value will be substituted. All user responses are checked for validity before being used; if the response is invalid, either the question will be asked again or the command will be aborted.

4.2 Line Additions

The problem of adding "curvy" lines interactively is not handled in the usual manner. Most interactive graphics editors allow the user to input a curvy line by moving the device which controls the cursor on the display, and the resultant line is the path followed by the cursor. This technique was found to be impractical with the standard crosshair cursor hardware built into the Tektronix 4014 display, since the cycle time for reading the crosshair location was too high (i.e. the data input rate was too low). Hence a curve fitting technique was adopted. In this method, the user inputs a series of points on the display and GOMADS then fits a smooth curve through them. The curve used is of the cubic spline type, with continuous first derivatives at all data points. The algorithm is exceptionally stable and produces, for any reasonable set of input points, an excellent approximation of the curve a human draftsman would draw.

In practice this technique has proven to be quite satisfactory, and in fact superior to the normal cursor-following method. The curve fitting method is best at producing long smooth curves such as contours, and worst at rough curves such as shoreline. For hydrographic charts, there is no requirement to add shoreline on the display. High-accuracy lines cannot be input on the display, since there is no way for the user to know where the line should go exactly; to edit these the user must go back to the digitizing table. The major line editing requirement on the display is for lines where the accuracy is not as important as the presentation, such as contours. For this application the curve fitting technique excels. The GOMADS evaluation tests showed the line addition and modification techniques to be very effective.

4.3 Command Summary

Command	Description
DR	draw a new picture on the 4014
CE	define a center for the next picture
MA	define a magnification for the next picture
DP	define new display parameters
LI	list all available line edit commands
LIAD	add a line
LIDE	delete an entire line
LISE	delete a segment of a line
LICH	change a segment of a line
LICL	clip the end off a line
LIEX	extend a line
LIMP	move the points in a point-to-point line
SO	list all available sounding edit commands
SOAD	add a sounding
SODE	delete a sounding
SOCH	change the value of a sounding
SOFL	change the flag word of a sounding
SOMO	move a sounding
SOSE	select a sounding
SOUN	unselect a sounding
SY	list all available symbol edit commands
SYAD	add a symbol
SYDE	delete a symbol
SYCH	change a symbol
SYMO	move a symbol
NA	list all available name edit commands
NAAD	add a name
NADE	delete a name
NAMO	move a name
NARS	rotate and stretch a name
NAMC	move characters in a name
DE	list all available descriptor edit commands
DEFC	modify the feature code in a descriptor
DESN	modify the source number in a descriptor
HE	list all available commands
CO	concatenate another NTX file
WR	write the current file onto mag tape

CU	print out the XY coordinates of the cursor
QU	quit this GOMADS session

5. DISPLAY ROUTINES

One of the major concerns during the design of GOMADS was the time required to draw a picture on the 4014. Hydrographic data files tend to be very large, since they contain large quantities of lines digitized at 500 points to the inch. A typical chart can easily contain one million points in the line work alone. The Tektronix 4014 operates on a 40 Kbaud line to the PDP-11, which is relatively slow by today's standards, and the commands it accepts must appear as ASCII character strings. To position the beam prior to drawing a line requires a sequence of up to seven characters; thereafter each point on the line requires up to five characters when operating in the standard vector drawing mode. There is however another mode which requires only one character for each point on the line, but this can only be used when the points are encoded as Freeman vectors. In general the fastest mode was used in the GOMADS drawing routines.

The main driver for the 4014 was written in assembler, using the interrupt structure of the PDP-11. It has a 1024 character circular buffer which it uses for both standard output and for refreshed output. The driver was not written as a standard DOS/BATCH device driver, since monitor constraints would not allow effective utilization of the interrupt structure. The driver also handles input from the 4014 on a non-interrupt basis.

The remainder of the display routines were written partially in Fortran and partially in assembler. Initially all the routines were written in Fortran, and those that proved to be too slow were converted to assembler. All the standard drawing functions are handled, such as scaling, centering, and windowing the picture. Scaling is handled by shifting, a faster operation than multiplying. Windowing is done in Fortran using real arithmetic. Other functions handled by the drawing package include enabling the crosshair cursor, reading the cursor position, and clearing the screen. All the routines have been written so that they can either output directly on the screen or they can enter commands and coordinates into the refresh buffer for drawing in either write-through mode or bright mode.

A certain amount of sophistication was also built into the data file handling routines for the specific purpose of speeding up the drawing. Section 6 of this document explains this. The final result is that for magnifications of 0 or greater, GOMADS' drawing speed is now limited by the speed of the line. A typical data file being drawn at a working magnification (2 or 3) takes about 15-30 seconds.

6. GOMADS DATA STRUCTURE

The fundamental requirement of the GOMADS data structure is to provide a capability of quickly and efficiently accessing data by its position. To access data by position can be defined in terms of a window. A window is a rectangle whose sides are parallel to the X,Y axis and delimited by two coordinates specifying the rectangle's lower left-hand corner (min X, min Y) and its upper right-hand corner (max X, max Y). A PDAM (Positional Direct Access Method) can now be defined as a method which will return all the information wholly or partially contained in a given window. The GOMADS PDAM is used by the drawing routines in determining which information will appear on the screen, where the window is defined by the screen's lower left and upper right-hand corners. It is also used by the feature location routines, where the window is defined by a small tolerance about the cursor position.

The GOMADS data structure is composed of three inter-connected levels. Each level becomes a file containing references to the file below it. The cell file is the highest level, the descriptor file is the middle and the data file is the bottom level.

6.1 Cell File

The design of GOMADS' PDAM is greatly simplified by the restrictions imposed by the size of available digitizing and automatic plotting tables. The maximum size of these devices is less than 64 inches by 64 inches. This means that both the source and output graphic must fit within a 64 X 64 inch square. In GOMADS the 64 X 64 inch square is subdivided into half inch by half inch squares producing a maximum of 64 X 64 X 4 = 16,384 source cells. The cell file consists of records (directly accessible by cell number) containing the numbers of the features which intersect each square of the imaginary ½ X ½ inch grid superimposed on the source document. This level of the data structure enables the system to quickly obtain a list of all the features intersecting a given window.

The cell record itself consists of eight integer pointers. The first seven of these are the record numbers in the descriptor file of descriptors whose data intersect the cell. If less than seven DAD's (Descriptor and Data) intersect the cell, one or more of the record numbers will be zero. The eighth pointer is non-zero only when more than one record is associated with a cell. For example, if nine DAD's happen to intersect cell 752, then the descriptor record numbers of the first seven descriptors will be stored in record number 752 of the cell file. The eighth word of record 752 will contain the record number of another record. This other record contains the descriptor record numbers of the last two DAD's which intersect cell 752.

6.2 Descriptor File

The second level of the data structure is the descriptor file in which each record is an Interchange file descriptor with four additional one-word fields. Two of these fields are used as a pointer into the data file to the data described by the descriptor. The first of these words gives the record number in the data file; the second word gives the offset within the record in which the data begins.

The other two additional fields contain forward and backward link pointers. These are simply descriptor file record numbers of descriptors which are linked together. The flag word in the descriptor permits descriptors to be linked in two different ways. Descriptors are graphically linked when they are really all part of the same feature. For example, code 1 lines are broken into segments of approximately one half inch with separate, but graphically linked DAD's for each segment. The other type of link, the logical link is used to group different data types which may be related in an "information" sense.

6.3 Data File

The last level of the data structure is the data file itself. It is implemented as linked list of records. The first word of the record contains the record number of the continuation (if any) of the data under a particular descriptor. The second word contains a count of the number of words in the record which are in use. The rest of the record is packed with data. Data for a descriptor does not necessarily begin with a new record.

Besides retrieving data efficiently by position, the data structure must be capable of being modified in four important ways: inserting strings, deleting strings, inserting descriptors and deleting descriptors. Repeated insertions or deletions should not be wasteful of disc space nor degrade the response time of the system. All the files in the GOMADS data structure are either linked list files, pure direct access, or a combination of these. This means GOMADS should suffer minimal degradation from extensive modifications to a single file.

7. INTERCHANGE FORMAT

Early in the design phase of GOMADS it became apparent that the existing data formats were inadequate. The major deficiency arose from the desirability of editing both symbolized and unsymbolized data, since at that time unsymbolized data existed only in acquisition format and symbolized data in drawing format. Hence a new format, called Interchange (NTX) format, was designed to overcome this deficiency as well as other ones known at that time.

NTX format consists of a main header followed by a number of DAD's, and finishes with a special end-of-file indicator. The main header contains information which applies to the entire file, for example, the projection, scale, chart limits, sounding units, and digitization date. The DAD's which follow contain the actual data. A DAD (standing for Descriptor and Data) consists of a fixed length descriptor followed by a variable length data string. The descriptor contains such things as the number of words in the DAD, a 10 character feature code describing the kind of feature being represented, and a data code describing the format of the data string. Currently there are eight valid data codes:

1. Freeman vector lines, used for all "curvy" lines
3. point-to-point lines
4. dashed lines, with each dash a straight line segment
5. ASCII text, information only
6. names - position with text

7. names - each character has a position and rotation angle
8. symbols
10. soundings

All coordinates are represented as a pair of 16-bit cartesian coordinates on a .002 inch grid at chart scale. Freeman vector lines allow a great deal of data compression, since each point on the line takes 3 bits rather than 32 bits.

NTX format can accommodate both symbolized and unsymbolized data. Unsymbolized data is data as it is digitized or collected automatically; e.g., a sandy shoreline would appear as a solid line in code 1 (Freeman vector) format. Symbolized data takes the form it has on the finished chart, e.g., the same sandy shoreline would appear as three rows of dots in code 8 (symbols) format. Since it is desirable to be able to edit both types of data, GOMADS is designed to handle both and to work in conjunction with the symbolization program STAR1. A bit in the flag word of each descriptor indicates whether that DAD is symbolized or not.

In order to facilitate processing on a mini computer, each DAD in NTX format has been limited in length. Freeman vector lines (code 1) have a maximum length of 72 16-bit words, and all the other code types are limited to 270 words. This means that it must be possible to link two or more DAD's together to accommodate large features such as shorelines and contours, and this is accomplished by means of another bit in the flag word of the descriptor. Two types of links are possible. One type indicates that two DAD's are graphically connected, e.g., two parts of the same line. The other type indicates that two DAD's are linked in an "information" context, e.g., a bridge might consist of a curvy line, some point-to-point line, and a label describing it. Another bit in the flag word of the descriptor indicates this type of link.