A Draft Guide
to the
Compleat ELEFAN

by

F.C. Gayanilo, Jr.
M. Soriano
and
D. Pauly

February
1988

INTERNATIONAL CENTER FOR LIVING AQUATIC RESOURCES MANAGEMENT
A draft guide to the Compleat ELEFAN

F.C. GAYANILLO, JR.
M. SORIANO
D. PAULY

February 1988

Published by the International Center for Living Aquatic Resources Management
MC P.O. Box 1501, Makati
Metro Manila, Philippines

A draft guide to the Compleat ELEFAN. ICLARM Software 2, 65 p.
International Center for Living Aquatic Resources Management, Manila, Philippines.

ISSN 0116-6964

ICLARM Contribution No. 435
A Draft Guide to the Compleat ELEFAN

by

F.C. Gayanilo, Jr. [1]
M.L. Soriano [2]
and
D. Pauly [3]

ABSTRACT

This draft guide introduces the 10-diskette, graphics-oriented Compleat ELEFAN software package (version 1.0, February 1988) for length-based fish stock assessment, implemented for IBM® PC and compatible microcomputers with at least 520 Kbytes of memory. Emphasis is given to the description of program operations, and to the linkages between the different routines, which cover most aspects of fish population dynamics (i.e. growth, mortality, gear selection, virtual population, yield-per-recruit analyses, etc). This guide does not – except in cursory manner and for new methods – discuss the models behind the various routines, but references are provided to key articles in the original literature. Readers are invited to provide suggestions towards improving the Compleat ELEFAN package and its future detailed documentation.

[1] version of February 1988; ICLARM Contribution No. 435
[2] International Center for Living Aquatic Resources Management, MC P.O. Box 1501, Makati, Metro Manila, PHILIPPINES.
[3] IBM is a registered trademark of International Business Machines Corp.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>v</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>vi</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>GETTING THE Compleat ELEFAN STARTED</td>
<td>4</td>
</tr>
<tr>
<td>Hardware requirements</td>
<td>5</td>
</tr>
<tr>
<td>Keyboard and software prompts</td>
<td>5</td>
</tr>
<tr>
<td>Starting the package</td>
<td>7</td>
</tr>
<tr>
<td>Diskcopy</td>
<td>8</td>
</tr>
<tr>
<td>DOS installation for a two-disk drives config</td>
<td>8</td>
</tr>
<tr>
<td>Hard disk installation</td>
<td>8</td>
</tr>
<tr>
<td>Peripheral installation</td>
<td>9</td>
</tr>
<tr>
<td>Serial port installation</td>
<td>9</td>
</tr>
<tr>
<td>Formatting a disk</td>
<td>9</td>
</tr>
<tr>
<td>WORKING WITH ELEFAN 0</td>
<td>11</td>
</tr>
<tr>
<td>Data entry</td>
<td>12</td>
</tr>
<tr>
<td>Printing of data</td>
<td>14</td>
</tr>
<tr>
<td>Editing a file</td>
<td>14</td>
</tr>
<tr>
<td>Probability entry</td>
<td>16</td>
</tr>
<tr>
<td>Adjusting class interval</td>
<td>16</td>
</tr>
<tr>
<td>Pooling of samples/files</td>
<td>16</td>
</tr>
<tr>
<td>Estimation of sample weight</td>
<td>17</td>
</tr>
<tr>
<td>Smoothing of L/F data</td>
<td>17</td>
</tr>
<tr>
<td>Deleting a file</td>
<td>17</td>
</tr>
<tr>
<td>WORKING WITH ELEFAN I</td>
<td>18</td>
</tr>
<tr>
<td>Data file access/restructuring</td>
<td>19</td>
</tr>
<tr>
<td>Submenu</td>
<td>20</td>
</tr>
<tr>
<td>Response surface analysis</td>
<td>21</td>
</tr>
<tr>
<td>Curves fitting by eye</td>
<td>22</td>
</tr>
<tr>
<td>Search for optimum base point</td>
<td>23</td>
</tr>
<tr>
<td>Graphic outputs</td>
<td>25</td>
</tr>
<tr>
<td>Working with ELEFAN II</td>
<td>27</td>
</tr>
<tr>
<td>Catch curve and resultant curve</td>
<td>28</td>
</tr>
<tr>
<td>Fitting of probability of capture</td>
<td>30</td>
</tr>
<tr>
<td>Recruitment pattern</td>
<td>31</td>
</tr>
<tr>
<td>Estimation of Loo and Z/K</td>
<td>32</td>
</tr>
<tr>
<td>Yield/recruit and biomass/recruit</td>
<td>33</td>
</tr>
</tbody>
</table>
WORKING WITH ELEFAN III ........................................ 35
  Preparing catch data ............................................ 39
  VPA I .......................................................... 41
  VPA II .......................................................... 43
  VPA III .......................................................... 44

WORKING WITH ELEFAN IV ........................................... 47
  Methodology .................................................... 48
  Computational options and variations ...................... 50

WORKING WITH ELEFAN V ........................................... 52
  Joint response surface for length-frequency
  and growth increment data .................................... 52
  Growth parameters estimations from
  growth increment data ....................................... 53
  A new method for estimating the
  parameters for a seasonally
  oscillating growth curve from
  growth increment data ...................................... 55

WORKING WITH MPA ................................................. 56
  Data file and sample access .................................. 58
  Bhattacharya method .......................................... 58
  Linking of the means ........................................ 60
  C.V. vs growth rate .......................................... 62

ACKNOWLEDGEMENTS ............................................... 63

REFERENCES ...................................................... 64
INTRODUCTION

The Compleat ELEFAN is a very comprehensive package incorporating numerous, rather sophisticated routines for length-based fish stock assessment - several of them new developments - which most colleagues, we believe will find very useful.

This brief draft guide, on the other hand, presents mainly the sequence of operations involved in running the various routines with little emphasis on the logic or equations behind these.

The reasons for this discrepancy are basically two: (a) while developing this package, from early 1986 to late 1987, we incorporated numerous routines based on new ideas which will still require proper scientific publications and (b) we want to write a self-sufficient, very comprehensive manual, covering all aspects of the program - yet we are aware that "bootleg" copies of the Compleat ELEFAN are already being passed around, and that colleagues are blaming us for not providing documentation for this package.

This guide is thus a mere stopgap. Until a comprehensive manual becomes available, users of the Compleat ELEFAN package are urged to not only read the contents of this document, but also of the papers cited therein, particularly those included in the recently published Proceedings of the Conference/Workshop held on February 1985 in Mazarra del Vallo, Italy, on "Theory and Application of Length-Based Methods in Stock Assessment".

This conference, which marked a crucial stage in the development of length-based methodologies for stock assessment, cannot be reviewed here. With regard to the package presented here, it must be mentioned, however that all suggestions made at the conferences concerning possible improvements of the ELEFAN approach package have been incorporated. Properly used, with well-sampled data, this

------------


1
package can thus be used to perform, reliably, most types of
assessments for which length-frequency data, or catch-at-
length-data are needed.

Some routines or programs which we intend to incorporate
into version 2.0 of the package (i.e. the one for which a
comprehensive manual will be made available), are among
others:

. ELEFAN VI, a program for multispecies yield-per-recruit
analysis which explicitly considers gradual mesh selection;

. a routine (to be included into ELEFAN II) for estimation
of values of von Bertalanffy parameters $L_\infty$ and $K$, pertaining to mean age-at-length (rather than to mean
length-at-age), as needed e.g. for unbiased estimates of $Z$
through length-converted catch curves; and

. a routine (to be included into ELEFAN I and V) for
estimation of an approximate joint confidence region for the
von Bertalanffy parameters $L_\infty$ and $K$ from the shape of the
response surface of "$R_n"", our new "goodness-of-fit index".

Because the Complete ELEFAN package is so large (10
diskettes in all), we will not supply listings of the
programs.

Rather, we would appreciate receiving comments from users of
this package which will help us improve it, notably by:

. removing bugs, misspelling, computational
errors, etc.

. replacing existing routines by more versatile or
powerful ones,

. adding new routines and subroutines, or

. making the presentation of routines or scripts
neater and/or more understandable.

Also, we would appreciate suggestions as to what should go
into the planned manual of version 2.0. To be most useful,
comments sent to us should be specific to a given program or
routine, and reports of bugs or errors should be supported
by the file upon which the results were based, detailed
description of the problem, and copies of any relevant
printouts. We are looking forward to such comments because
they will help us improve this package, so don't hesitate to
write us.
All comments shall be considered when we prepare version 2.0, which will be free for all those who have sent us usable comments and suggestions, and available for a small upgrade fee to all registered users of version 1.0 and its immediate successors.
GETTING THE COMPLEAT ELEFAN STARTED

Computer literacy is fast becoming an essential qualification for researchers in fisheries science. As more and more research tasks are supported by computers—from word processing to fisheries stock management decision making—professionals in this field are learning more about the powers and limitation of computers.

The Compleat ELEFAN software package is one of the many designed to alleviate the task of manually processing the data available to fisheries biologists. This document is designed for both beginners and experienced computer users.

Computer hardware refers to the physical entities in a computer system. These are the things that can be seen and touched, and is consists of electrical and electronic circuits, and mechanical devices. The basic configuration of a personal computer consists of a system unit and keyboard (see Fig. 1). The system unit is the central part of the computer and supports a variety of options. The keyboard, on the other hand, is the main input device which is used to send data to the system through a cable. The other pieces of hardware that must (or may) be added are the display (screen), the printer, the plotter, disk drives, etc.

![Fig. 1](image-url)
Hardware requirements of the Compleat ELEFAN

This software was developed for use on IBM PC microcomputers with graphic capabilities and their compatibles. If you are using a computer without this capability, you must add a graphic card (i.e. Hercules monochrome graphic card or color graphic adapter) to be able to run the software (note that a color monitor is not needed).

Except for one program (VPA III), all routines require a minimum of 520 Kbytes of memory. The VPA III program requires 640 Kbytes of on-board memory.

Output of results is generated via the screen and/or printer. Any printer (dot matrix or letter quality) can be used except for graphic outputs, which some printers cannot generate. A plotter can also be used to generate graphic outputs. The software is capable of generating plotter outputs through an IBM plotter (7371 model), but other similar plotters that are using the same GL (Graphic Language) may also be used (e.g. the HP 7470A plotter).

Two 5 1/4" disk drives or one 5 1/4" disk drive and a hard disk are recommended to run the software.

Keyboard and software prompts

The keyboard is used to enter data into the system or to respond to prompts generated internally. It consists of three sections: the typewriter area, the function keys, and the numeric keypad. These keys are "typematic", i.e. they repeat as long as they are held down.

The typewriter area is much like that found on a conventional typewriter except for some special keys found at the edge of the typewriter area (Fig.2).

![Keyboard Diagram](image)

**Fig. 2**

A prompt is a software request for additional information needed to carry out selected functions. There are three types of prompts which the software uses:
Menu – you are given a choice between several alternative functions or options, which you select by pressing the arrow keys to select and the <CR> key to access (<CR> here expresses "Carriage Return" or <ENTER>).

Line prompts – the software uses this to request a single item of information such as e.g. a mortality rate. This prompt is dealt with by typing the required characters and terminated by pressing the <CR>. If you realize that you have made a mistake but have not yet pressed <CR>, use the <BACKSPACE> key until the offending character is erased, then retype from there. The <BACKSPACE> key is the only editing key that can be used at this stage.

Confirmatory prompts – these are very similar to menu prompts except that you are given two or three options to choose from. Prompts which are answered with Y for yes and N for no belong to this category. Choices are given inside parentheses (e.g. "Proceed (Y/N)").

Selection prompts – these are very similar to the above, but you are given a list of function keys or number keys (e.g. <1>) with corresponding function or options.

Throughout the package, the function keys are attributed to the following functions.

[F1] – to redo a routine

[F2] – to move the "turtle" (or pointer) up or to the left

[F3] – to return to the previous menu

[F4] – to move the "turtle" down or to the right

[F9] – to proceed to the next routine

[F10] – to access more help messages (if applicable)

The function keys [F5], [F6], [F7] and [F8] are assigned to other functions (e.g. to dump graphic, mark points, etc.). As an alternative to [F2] and [F4], the arrow keys (left and right, respectively) may be used. [F6] is often used as the key to mark points for which the <SPACE BAR> may also be used.
Starting the package

To start the computer, it is advisable to begin with the peripherals connected to the system unit. The switches of the peripherals (screen, printer, etc.) are conveniently located outside of the hardware. There is no need to open any part of the peripherals.

The contrast and brightness of the PC's display can usually be adjusted using control knobs. Keyboards usually have adjustable tilt position for typing comfort. Hold the keyboard when changing position. Firmly push in the foot handles and turn them to the desired level.

Before proceeding, some preliminary operations are necessary and these maybe accessed from DOS or through the menu option called "Utilities" in the main menu of the package (see Fig. 3).

**Disk copy**: The first thing that you should do when you have unwrapped your set of ELEFAN diskettes is to make working copies. Any disk duplicator program can be used. The most common of these duplicators is the DOS DISKCOPY command. Put the original disks in a safe place and use the copies. If the copies are ever damaged or destroyed, you can make new copies from the original disk. For more informations on how to use the DISKCOPY command, please refer to the DOS manual.

```
NOTE : DO NOT write-protect the working copies of the ELEFAN package.
```

The Compleat ELEFAN contains a utility to create a backup copy of the diskettes. A minimum of two disk drives is needed for this routine. If you have only one drive, use the DISKCOPY command of MS DOS. Access the routine that will allow you to make a backup copy and have ready ten (10) diskettes for the working copies.

You should have the following diskettes:

- **DISK #1**: System and Utilities
- **DISK #2**: ELEFAN 0
- **DISK #3**: ELEFAN 0-A
- **DISK #4**: ELEFAN I
- **DISK #5**: ELEFAN II
- **DISK #6**: ELEFAN II-A
- **DISK #7**: ELEFAN III
- **DISK #8**: ELEFAN IV
- **DISK #9**: ELEFAN V
- **DISK #10**: MPA
DOS installation for a two-disk drive configuration: The disk operating system (DOS) is not included in the package. The DOS is prerequisite to some utilities and it should be installed only to DISK #1. To copy, insert a disk containing the DOS (note: the Compleat ELEFAN requires MS DOS 2.1 or higher) in A: drive. Boot the computer and wait for the DOS prompt (A>). Carefully insert DISK #1 (of the Compleat ELEFAN) in drive B:. Type SYS B: followed by <CR>. This will copy the DOS to DISK #1.

In addition to the DOS, the following DOS commands are needed by the package:

COMMAND.COM
GRAFTABL.COM
GRAPHICS.COM
ASSIGN.COM
FORMAT.COM
DISKCOPY.COM
MODE.COM

To copy these DOS commands, insert a DOS disk containing the files mentioned above in drive A:. With the DISK #1 still in drive B:, type COPY A:XX B: (where XX is the name of the DOS command that needs to be copied, e.g. FORMAT.COM) followed by a <CR>. There is no need to repeat the above procedure for the other disks of the package.

For a computer with a hard disk, boot the computer with either floppy disk drives empty and wait for a DOS prompt (e.g. C>). Carefully insert DISK #1 (of the Compleat ELEFAN) in drive A:. Type SYS A: followed by <CR>. [This can be bypassed if the package is to be installed to a hard disk; see below]

Hard disk installation: A utility (HDINSTAL.EXE) is provided in DISK #1 to properly install the package to your hard disk. Other methods of copying the set of programs to your hard disk may cause problems while using the package. Note also that a minimum of 2.7 megabytes of free storage space is needed.

The DOS path (sometimes called "directory", e.g. C:\DOS) should be known before running this hard disk installation program. The same six MS DOS utilities as listed above are needed to run the package, and this should be accessible through DOS path.

To run, insert DISK #1 in drive A:, then type HDINSTAL followed by <CR>. This utility will initialize any existing "ELEFAN" subdirectory.

Peripheral installation: A utility has been developed to allow the software to run on different system configurations. This utility may either be accessed through
the INSTALL.EXE which can be accessed directly from DOS (DISK #1) or through a routine in the utility menu. It will install the following:

1). the type of graphic card/adapter (CGA or Hercules monochrome graphic adapter),
2). the printer type,
3). the data drive specification and,
4). the program drive specification.

Installation is done only once, unless you wish to change peripherals or drive specifications.

Serial port installation: Graphic outputs may also be dumped via plotter; this requires the user to configure the communication ports (COM1:). Please refer to the plotter's manual for communication protocol parameters.

This installation may not be necessary if the plotter can be configured through the dip "switches" found at the back of the peripheral.

Serial printers may also be redirected through this utility. For more technical information on the requirements of this utility, please refer to computer manual, "Guide to Operations".

Formatting a disk: Data disks may either be created using the DOS command "FORMAT" or the utility to format a disk in one of the routines available within the package. Formatting, (1) initializes a disk to a recording format acceptable to DOS, (2) analyzes the entire disk for any defective tracks, and (3) prepares the disk to accept DOS files. For more information regarding the FORMAT command of MS DOS, please refer to the DOS manual.

CAUTION: Please note that formatting destroys all data on the disk. Because of this, you should be very careful before you decide to format a disk, particularly a hard disk.

For problems or deviations from the above, please refer to the manual (Guide to Operations) or ask an experienced user.

After installing the package properly, type ELEFAN after the DOS prompt (e.g. A>), then press <CR>.

Once the software package is accessed, the system menu will be loaded and executed (see Fig. 3). You may use the arrow keys (up and down) or function keys [F2] and [F4] respectively. To access a routine, position the highlighted
area (by using an arrow key) to the desired routine, then press <CR>.

[Diagram]

---

**Fig. 3**
WORKING WITH ELEFAN 0

Introduction

The Data Entry and File Management Routine referred to as ELEFAN 0 allows entry of data sets such as length-frequency data, weight-frequency data, commercial shrimp tail counts, tagging/recapture and/or length-at-age data and stores them in a format accessible to other ELEFAN routines.

The ELEFAN 0 menu contains nine routines appearing on computer screen as follows:

--- Data Entry ---
Print Data File
Edit Data File
Probability Entry
Adjust Class Interval
Pool Samples/Files
Sample Weight Estimation
Smooth Data File
Delete Files
Quit to main menu

This routine creates a new data file for any of the following data types:

a) length-frequency data
b) weight-frequency data
c) shrimp-tail counts
d) growth increment data
(e.g. tagging/recapture data)
e) length-at-age data

Items b) and c) are automatically converted to a length-frequency file and may then be used in any ELEFAN routine.

Files created via d) or e) may be accessed only through ELEFAN 0.

Fig. 4

Before going into the details of each of these routines, it is important to mention here the general screen layout of ELEFAN 0. The first four lines on the screen are reserved for the routine title, lines 5 to 18 are for the body of the routine and two separate boxes on the lower part of the screen are reserved for instructions (left box) and function keys (right box).
Attention should be given to the instruction box which is intended to guide the user. Attention should also be given to the function key box, which provides other options ensuring flexibility while a routine is being used. The function keys [F2] and [F4], which are not shown on the function key box, are assigned to move to previous field and next field, respectively.

Data entry

Unless foreign files are transferred to the ELEFAN system, the world of ELEFAN begins here. This routine allows for the entry of the following types of data sets:

1. Length-frequency data
2. Growth increment data
3. Length-at-age data

The length-frequency data may be accessed any time using any ELEFAN routine. Growth increment (e.g. tagging/recapture) data and length-at-age data once entered may only be accessed by ELEFAN 0, e.g. for editing, and by ELEFAN V for analysis.
The next version of the package will, additionally, allow weight-frequency and commercial shrimp tail count data to be stored on disk and analyzed.

Basically, the data entry routine is made up of three screens, arranged as follows:

**File specification screen**: This screen requires input providing a general description of the file such as the species name, number of samples, smallest and largest midlengths/midweights, class interval and unit of measurement used.

**Data entry screen**: The data entry screen requires input of the date of sampling (month, day and year) and frequency number for each class length or weight. (Tagging/recapture and length-at-age requires actual lengths). The lower right corner of the screen labelled "SUM" keeps track of the total frequencies entered. The data entry screen appears n times depending to the number of sampling dates entered in the file specification. Each sampling date is checked for duplicates and is used to sort the entries.
Save data file screen: This last screen requires input of a unique filename to save the data set onto a diskette. This filename is the only link to access a particular data set in any ELEFAN routine.

Printing of data

As needed, by using this routine, hard copies of any ELEFAN data file may be outputted to a printer. The only information required of the user is the filename where the data set is stored. Everything else follows.

Editing a file

Alteration of an existing ELEFAN data files is done using this routine. It has the same screen flow as that of Data Entry. The class interval of the file specification screen is the only field that can not be altered, (please refer to the Adjust Class Interval routine instead).

All fields may be edited and, in particular, a sample may be deleted by entering 99 for the month. Each sample appear on screen in chronological order. To select the desired sample for editing, simply enter its corresponding sample number while you are on the "SAMPLE" field or use function keys [F6] or [F9] to move from one step to another.

The "save edited file" screen allows the user to save the edited data set to a new filename or overwrite the old data set by using its old filename.
The "Edit Routine" may be invoked to RENAME a data set, by accessing an old file and saving it without alteration under a new filename. The old file may then be deleted using the "Delete Routine".

The "Edit Routine" may also be used to COPY an ELEFAN data file from one diskette to another by using function key [F9] – change data diskette, while on the save file screen.
Probability entry

This routine allows for the entry of probabilities of capture for each class length of a given data set. These probabilities may come from ELEFAN II (Catch Curve Routine), or previous studies of the species in question.

Each length frequency in the file is then divided (by length class) by its corresponding probability of capture to obtain the relative abundance for each class.

Adjusting class interval

Class intervals are altered using this routine, which was developed for the purpose of merging files with different class intervals. Basically it slices up the whole length data into the class interval the user desires.

The filename of the data set to be sliced-up is selected from the library and the desired change of class interval is entered. Succeeding screens will give a view of the sliced-up data file.

Note that a new class interval need not be a whole number, nor a whole multiple of the old class interval, i.e. data originally grouped into say 1 cm intervals may be regrouped into a 1.5 cm interval using this routine.

Pooling of sample/files

Pooling of samples or merging of two files is another data manipulation feature of the ELEFAN system. Two samples of one data set may be pooled together or samples of two files may be merged by any of the following options:

(1) Addition of frequencies; this option simply adds up frequencies of two or more samples and stores them under one sampling date.

(2) Weighting by the square root of sample size; this pooling option is a modification of (3) and is recommended for low frequency samples; the mathematical equation is as follows:

\[ f'ij = w_{ij} \cdot f_{ij} \]

where \( f'ij \) = pooled frequency for sample i, class j
\( f_{ij} \) = frequency number for sample i, class j
\( w_{ij} = f_{ij} / \sqrt{T_i} \) where \( T_i \) is total frequency for sample i.
(3) **Weighting by percentage**: weighting of samples by percentage gives each sampling date an equal weight, whatever its initial size.

Mathematically,

\[ f'_{ij} = w_{ij} \cdot f_{ij} \]

where \( f'_{ij} \) = pooled frequency for sample \( i \), class \( j \)
\( f_{ij} \) = frequency number for sample \( i \), class \( j \)
\( w_{ij} = f_{ij}/T_i \), where \( T_i \) is total frequency for sample \( i \).

**Estimation of sample weight**

The sample weight of a given file is estimated using the length-weight relationship \( (W = a \cdot L^b) \). Mean weight in each length class and of each sample are computed. The estimation of the mean weight of the fish of a given class is computed as suggested by Beyer (1987).

**Smoothing of L/F data**

This routine makes use of a running average over 3 or 5 classes to smooth the available length frequency data, as suggested by Laurec and Mesnil (1987) for cases where data are irregular.

**Deleting a file**

This routine deletes an ELEFAN data file not only from the diskette but also from the ELEFAN disk library. The user is cautioned from deleting an ELEFAN data file using the DOS utility primarily because this latter routine would not update the ELEFAN disk library.
WORKING WITH ELEFAN I

Introduction

ELEFAN I is a program which uses length frequency data to estimate the parameters K, Loo, C and WP of the von Bertalanffy equation, modified to express seasonal growth in length of fish or aquatic invertebrates (Pauly and David, 1981).

The program's key assumptions are:

- samples used are representative of the population;
- length differences can largely be attributed to differences in age;
- growth is similar from one year to the next, i.e. there are no factors inducing any strong changes in growth between years; and
- the seasonally oscillating von Bertalanffy growth equation provides an appropriate approximation to the growth of fish and aquatic invertebrates.

Although ELEFAN I could in principle be used in conjunction with any type of growth curve, only one growth curve was incorporated into this version. It is the seasonally oscillating version of the von Bertalanffy Growth Function (VBF) proposed by Pauly and Gaschütz (1979), which has the form

\[
L_t = L_\infty (1 - e^{-[K(t-t_0) + \frac{CK}{2\pi} \sin (2\pi (t-t_0))]})
\]

where ts=WP+.05 and all parameters are as defined in Pauly (1984).

There are three (3) routines available in the ELEFAN I program to help the user estimate growth parameters:

- Curve fitting by eye: allows a user to fit a growth curve through a (restructured) length-frequency data set, given a set of parameters;
- Response surface analysis: allows a user to vary two growth parameters in a 11 by 11 matrix. The 10+ points providing the best fit to the data are identified, allowing the user to locate combinations of parameters that are better than others; and
. Direct search for optimum parameter combination: allows the user to let the computer search for the "optimum" combination of growth parameter values, given a set of seeded parameters and step sizes by which the program can modify the seeded values.

These three routines use "Rn" as goodness-of-fit index; this index ("new R") is computed from $Rn=10^{\text{ESP/ASP}}/10$, with ESP/ASP being as defined in earlier versions of ELEFAN I (the use of Rn prevent the occurrence of negative values of goodness-of-fit).

A fourth routine is included which allows the user to output the results in graphic form through a printer and/or a plotter.

Data file access/restructuring

Immediately after ELEFAN I has been accessed from the main menu, as in any of the programs to be discussed later, the program will first search for the disk library created during data entry and manipulation.

![Diagram of file parameters and disc library](Fig. 6)
If no length-frequency file or data disk is found in the data drive, the computer will prompt for a change of data disk or offer the option to quit to the main menu by pressing [F3]. Once the disk library has been accessed, a file may be selected by moving the cursor, i.e. by pressing an arrow key followed by <CR>. The file accessed may still be replaced after the file parameters have been displayed (see Fig. 6).

Restructuring of data will commence immediately after confirmation, i.e. by pressing [F9]. Depending on the size of your file and the speed of the computer, this routine may take a minute or so.

An option to output the restructured data via the installed printer will also be given.

Submenu

This program has four subroutines (see Fig. 7) to analyze a given length-frequency file.

Fig. 7
Response surface analysis

The limits (lower and upper) for each of the four growth parameter will be required by the program. To leave a parameter constant, press <CR> when the program prompts for the upper limit for this parameter (see Fig. 8).

Two growth parameters have to be left variable; otherwise the limits for each of the parameters have to be re-entered.

![Fig. 8](image)

Default values for the starting points are provided. When all parameters have been entered, a prompt on whether or not a printer output is desired will appear. After answering the prompts, the user will be given the opportunity to change whatever needs changing before the computer starts generating the response surface.

In the final output, the best 10+ points will be boxed for easier identification of the best combination of parameters. (see Fig. 9). At any time, a user may abandon the routine by pressing [F3] or [F7] to rerun the response surface generator with another set of entries.
### Fig. 9

**Curve fitting by eye**

Given a set of parameters (i.e. $L_0$, $K$, $C$ and $W_{P}$), this routine will fit the curve over the restructured data plotted on screen. Aside from the response surface, a user will find this routine particularly helpful in defining, at least approximately, the values of these parameters and identifying a good starting point.

The starting point (starting sample and starting length, i.e. the coordinates of a point through which the curve must pass) will also be asked (see Fig. 10).
A ruler may be activated using [F5] to ease identification of a starting length (default is the class with the highest peak) once the starting length is prompted for input. Use the arrow keys to move the ruler to a desired location. Pressing the (CR) will change the default value of the starting length.
The values entered before a curve is drawn will be used as the next default value, i.e. a user may simply press <CR> if a former entry is to be kept unchanged.

Search for optimum base point

This routine will allow a user to "seed" growth parameter values (i.e. to enter first guesses) and the step size by which the guesses will vary. The program will vary the parameters up to 6 steps at a time. That is, if for example you entered a seeded value of Loo of 21, and a step size of .1, it will first vary the value for Loo to 21.1 and compute for any improvements in the Rn value. If the computed Rn value is equal or less than the previous value, it will check the Rn value for Loo equal to 20.9, then 21.2, then 20.8, and so on. It will continue to vary the parameter until a value is reached which leads to a higher Rn value, and this combination of parameters will be used as the new input values. The search will continue until no further improvement is reached.

Once the "seed" parameter values and step sizes have been entered (see Fig. 12), the user will be given the option, via [F5], to let the computer search for the coordinates of the best starting point given the seeded parameters entered earlier. Depending on the size of the file, this may take some time.

If [F6] is pressed, the starting point (i.e. starting length and starting sample) will have to be provided. Default values are available (i.e. the coordinates of the class with the highest "restructured" point value).

Optimal step size ranging from about 3-6% of a parameter value for the initial searches to 1% or less may be used to better identify a good set of parameter combinations. Depending on the size of your file, searching can be a long process.

The computed growth curve will be plotted over the restructured sample everytime a new base point is identified.
It is recommended to re-initiate a search several times, from different seed values before accepting an "optimum" as such.

Graphic outputs

After an optimum of growth parameters has been identified, this routine will allow users to output the results through either a plotter or a printer.

If a growth curve is desired, the parameters (see Fig. 13) will be required and the computer will compute the corresponding Rn value. Also available is an option to output the points of the growth curve on every 15th of the month. A sample plotter output is given in Fig. 14.
Fig. 13

Specie name: *Uponeus salinaeasis*
Other labels: RACAY GULF 1980
Filename: HUNGRI

Sample(s): 7
Class size: 1 cm
Largest ML: 10.5 cm
Smallest ML: 5.5 cm

GROWTH PARAMETERS

<table>
<thead>
<tr>
<th></th>
<th>(1)</th>
<th>(2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loo</td>
<td>20.9 cm</td>
<td>20.9 cm</td>
</tr>
<tr>
<td>K</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>C</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>WP</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>SS</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>SL</td>
<td>9 cm</td>
<td>11.5 cm</td>
</tr>
<tr>
<td>Rn</td>
<td>0.42</td>
<td>0.084</td>
</tr>
</tbody>
</table>

Fig. 14
WORKING WITH ELEFAN II

Introduction

ELEFAN II is a collection of auxiliary routines and a sequel to ELEFAN I. It covers major topics as follows:

1. Estimation of total mortality from a catch curve using VBGF parameters estimated by ELEFAN I or known from an independent source;

2. Smoothing of selection pattern given the probability of capture for each length class using a logistic transformation;

3. Estimation of Loo and Z/K using a modified version of the Wetherall method;

4. Derivation of seasonal patterns of recruitment; and

5. Relative yield and biomass per recruit analysis.

Below (Fig. 15) is a layout of ELEFAN II main menu.
Catch curve and resultant curves

Catch curves are constructed for the investigation of total mortality (Z) within a fish population. The catch curve routine requires the filename of the ELEFAN data set for analysis and growth parameters (Loo and K) to be entered through the keyboard.

The user is given the option to select particular samples to use in estimating Z; some transformation of the data set are allowed. Transformation options are as follows:

1. No transformation (original data),
2. Using the square root of sample total as weighting factor; and
3. Transforming all samples into % samples before pooling.

The selected/transformed samples are then added up by length class and each cumulated frequency (%N_L) is divided by the time needed for the fish to grow from the lower to the upper limit of its length class (△t_L).

The ln (%N_L/△t_L) values are then plotted vs age on the screen, as shown in Fig. 16.
The "turtle" (or pointer) on the graph requires the user to select points to use in estimating total mortality. The turtle may be moved from one point to the other using function keys ([F2] or [F4]) or arrow keys (left or right).

The points selected must include only fish which are fully recruited and vulnerable to the gear employed. A generally applicable rule is to use as a first point the one immediately to the right of the highest point of the catch curve.

Often, one or two points corresponding to very high "ages" must be excluded from the computation of Z because they pertain to length values very close to Loo, affecting Z dramatically. Jones (1987) and Laurec and Mesnil (1987) provided rules as to which of these points should be excluded from the computation.

A preliminary estimate of Z is made using the linear regression:

\[
\ln(\%N/\Delta t) = a+b* t, \text{ where } t = \text{relative age}
\]

An iteration procedure (P. Sparre, per. comm. to Pauly 1984, Chapter 5) corrects the Z = -b of the regression line for the non-linearity of the growth model and for the fact that some mortality occurs within each length class. The iteration equation is:

\[
\log(N/1\exp(-Z*\Delta t_1)) = a-Z*\Delta t_1;
\]

then natural mortality (M) is estimated from the empirical relationship:

\[
\log_{10} (M) = -0.0066 - 0.279 \log_{10} (Loo) + 0.6543 \log_{10} (K) + 0.4634 \log_{10} (T)
\]

where T is temperature in °Centigrade (Pauly 1980).

The user is not compelled to use the computer-estimated Z and M values, and the values for Z and M to be used must in fact be entered by the user.

Given Z and M, a resultant curve may be derived from the left, ascending arm of the catch curve. The fishing mortality (F = Z - M) ranges between M (=mortality in the highest length class not caught at all) and Z (=mortality from the first selected point of the catch curve). Thus,
$Z_i = M + F_i$, where $Z_i$ and $F_i$ are the total mortality and fishing mortality for class $i$, respectively. The number of available fish in the population is then estimated using:

$$N_i = N_{i-1} \times \frac{1 - \exp(Z_{i-1} \Delta t)}{1 - \exp(Z_i \Delta t)}$$

The screen plot of the catch curve with estimates for $Z$, $M$ and $N_i$'s is shown in Fig. 17.

The probability of selection $P_i$ is $N_i(\text{caught})/N_i(\text{available})$. The resultant curve is fitted using a logistic transformation ($x = \log(p/q)$) or moving average over on the selection probabilities. Fig. 18 shows an example of a resultant curve.

![Catch Curve Plot](image)

**Fig. 17**

**Fitting of probability of capture**

This routine estimates $L_{20}$, $L_{50}$ and $L_{70}$ given a set of probability of capture over different length classes. This is one of the few routines in the ELEFAN package that may be used without the need to access an ELEFAN data file. The basic information needed to run this routine are the smallest midlength, largest midlength, class interval and the probability of capture for each length class. The algorithm and screen output is similar to the resultant curve section of the catch curve routine.
Recruitment pattern

A recruitment pattern is obtained by projecting a set of length-frequency data backward onto a one-year time axis. Recruitment patterns can be used to infer the number of recruitment pulses occurring per year (see Fig. 19). However, when the parameter $t_a$ is not given, the exact time of recruitment cannot be determined.

A routine (Soriano 1987) is built into the program which automatically fits a recruitment pattern with one or two normal distributions when the recruitment explained by one or two normal curves ranges between 90% and 110% of the observed annual recruitment.

In the same manner as in the catch curve routine, an ELEFAN data file and its growth parameters ($L_{oo}$, $K$, $C$, and $WP$) must be provided. The user is also given the option to select samples for analysis and the type of data transformation to use for computation.
Estimation of Loo and Z/K

This routine estimates Loo and Z/K using the method of Wetherall (1986), as modified by Pauly (1986).

In the same manner as with the catch curve and recruitment pattern routines, the user has the option to select samples and transform the available length-frequency data before pooling.

The transformed samples are then cumulated by length class and plotted on the left box of the screen (see Fig. 20) with midlength on the X-axis. The right box gives a plot of \((L_x(\text{mean}) - L'_x)\) vs \(L'_x\).

The user is then required, using the plot on the left box, to select a point representative of animals fully selected and recruited, from which Loo and Z/K will be estimated.

A regression line is then fitted to the points with each point weighted by its cumulative frequency. The Loo and Z/K estimates are given by (Pauly 1986):

\[
(Y_i - X_i) \times N_i = a + b \times X_i \times N_i
\]

\[
Loo = a / -b
\]

\[
Z/K = (1 + b) / -b
\]
Yield/recruit and Biomass/recruit

This routine estimates the optimum level of exploitation rate (E) using the relative yield/recruit (Y'/R) model of Beverton and Holt (1964) as modified by Pauly and Soriano (1986). This routine may be accessed without using an ELEFAN data file. Y'/R may be computed in two different manners:

1. assuming knife-edge selection

   wherein Y'/R is a function of M/K, E and c (mean length at first capture/Loo), as in Beverton and Holt (1964).

2. using probabilities of capture that are gradually changing

   the input required here are the smallest midlength, largest midlength, class interval, Loo, M/K and the probability of capture for each midlength (a maximum of five data sets can be entered at a time).
Also included in this routine is a plot of relative biomass/recruit ($B'/R$) vs. exploitation rate ($E$). Fig. 21 is a sample screen output of $Y'/R$ and $B'/R$ with logistic selection.

**Fig. 21**
WORKING WITH ELEFAN III

Introduction

ELEFAN III differs from ELEFAN I and II in that more is needed to run the program than just length-frequency data. The added data which ELEFAN III requires are catch data usually by month. Also, as is characteristic of most approaches based on Virtual Population Analysis (VPA), it is the entire catch from the whole stock which must be used, not only a part thereof.

ELEFAN III incorporates three types of VPA, i.e. VPA I, VPA II, and VPA III.

VPA I estimates for any given cohort, the standing stock (in numbers) and fishing mortalities by time interval (month, quarter, year, etc.). VPA II is used to estimate mean standing stock for a stable-age distribution, as can be simulated by combining data for several years. VPA III provides estimates of standing stock and fishing mortality by month and by length, which is achieved by "slicing" (pseudo-) cohorts through catch-at-length data by means of a set of growth parameters. This approach assumes that little exchange occurs between the monthly "cohort", which applies mainly in short-lived animals, such as anchovies or penaeid shrimps, for which the VPA III routine has been specifically designed.

Beverton and Holt (1957) showed that the catch (Ci) from a population during a unit time period (i) is equal to the product of the population size at the beginning of the time period (Ni) times the fraction of the deaths caused by fishing, times the fraction of total deaths, or

\[ C_i = \frac{F_i}{Z_i} \left(1 - e^{-Z_i}\right) N_i \]

where \( F_i \) is the fishing mortality in the ith period,
\( Z_i = F_i + M \), and M is the natural mortality, generally assumed constant for all period,

The version of Beverton and Holt's catch equation which has become most widely used for stock assessment purposes, however, is

\[ \frac{N_i+1}{C_i} = \frac{Z_i \cdot e^{-Z_i}}{F_i (1-e^{-Z_i})} \]
also written

\[ \frac{C_i}{N_{i+1}} = \frac{F_i}{Z_i} (e^{Z_i} - 1) \]  

(3)

which is the equation in Gulland's (1965) Virtual Population Analysis and which can be derived from (1) by substituting for \( N_i \) the relationship

\[ N_i = N_{i+1} \cdot e^{Z_i} \]  

(4)

Equation (2) is used with catch-at-age data from the whole of a fishery, and covering most of the life span of a given cohort (thus VPA is used to estimate retroactively the size of a past cohort), an estimate of \( M \) and a (guessed) value of the fishing mortality that affected the oldest age group of a given cohort \( (F_t) \). Terminal fishing mortality \( (F_t) \) and the terminal catch \( (C_t) \) are used to estimate the size of the terminal population \( (N_t) \), either from

\[ N_t = \frac{C_t \cdot Z_t}{F_t (1-e^{-Z_t})} \]  

(5)

or from

\[ N_t = \frac{C_t \cdot Z_t}{F_t} \]  

(6)

Generally, equation (5) is used when the cohort is not extinct past \( N_t \) (and \( C_t \)), while equation (6) is used when \( C_t \) includes the last remnants of a cohort (ELEFFAN III incorporates equation 5). Then, using \( N_e \) as initial value of \( N_{i+1} \), \( F_t \) and \( N_t \) values are estimated sequentially from older to younger age group be repeatedly solving equations (2) and (4), respectively.

The discussion that follows is limited to the principles behind these various routines and the reasons they were incorporated into ELEFFAN III.

VPA I: The principle behind VPA hinges around the concept of the "cohort", i.e., a group of fish or invertebrates born or hatched and recruited at more or less the same time and sharing throughout their lives a common pattern of exploitation by their predators and a fishery.
In most applications of VPA I, the "exploitation" by predators is expressed by a single value of the rate of natural mortality M, which is assumed to apply throughout most of the life of the cohort (i.e., from the time the animals enter the fishery to the time the cohort is decimated).

VPA, then, is a method to reconstruct a cohort (i.e., to estimate the number of animals that were present at each age (of a cohort's life) by using a value of M and the catch by the fishery (in terms of numbers) of each group in the cohort. The cohort is always reconstructed backward in time, starting with the last caught animals of a cohort (the "terminal catch"), which is used to obtain an approximation of the "terminal population" using a guessed value of the "terminal fishing mortality" exerted upon the "terminal population". Then, successive estimates of the population sizes are obtained for each age group by adding the catch of the fishery to a previous estimate of population size, to which the number of fish caught by predators has also been "added".

The method, which was formulated by Gulland (1965) has been reviewed by Pope (1972), Mesnil (1980) and Pauly (1984). It is incorporated in ELEFAN III as "VPA I" to allow users of the Complete ELEFAN system to acquaint themselves with VPA in its original form and to perform age-structured VPAs whenever the data for such analyses are available to them.

VPA II: Catch-at-length data differs from catch-at-age data (such as used in age-structured VPA or VPA I) in that they do not pertain — nor can readily be made to pertain — to a given cohort as defined above.

Still, versions of VPA can be run with catch-at-length data. Two very different versions of VPA are incorporated in ELEFAN III which uses catch-at-length data. The first of these (VPA II) is conceived such that while it is not meant to help in reconstructing any given cohort, it can, given catch data covering the life span of several cohort, reconstruct an average cohort. Thus, the results of VPA II are not structured in time; instead the population sizes and fishing mortalities which are typical outputs of VPAs are related to sizes. These results, therefore cannot be used to manage a fishery in real time, nor to study temporal fluctuations of recruitment.

VPA III: VPA III was devised to combine the advantages of both VPA I and II, to allow use of catch-at-length data as inputs and still to run an "age"-structured VPA, and thus to obtain results that are structured both in size and in time (see Fig. 22 and 23).
Fig. 22. Fascimile of an ELEFAN III output (via plotter) showing catch-at-length data (not to scale) and one of the many monthly cohorts which can be superimposed on these data. The upper two panels show the population and fishing mortality estimates pertaining to this cohort, along with its biomass (rightmost panel). The population estimate pertaining to month 1 is an expression of recruitment of fish of length $L_r$.

Fig. 23. Showing the summary result of a VPA III pertaining to a given month (here May 1969), and showing the catch data, the cohort "passing through that month (box with narrowing lines), the size structure of the population and other statistics."
This was achieved by assuming that all fish in the population under investigation have the same growth parameters, as is also assumed in the other ELEFAN programs.

In reality, not all fish in a given cohort have the same growth parameters, however, and it can be expected that some fish will "leave their cohort" because they grow either faster or slower than predicted by the mean cohort growth curve. Such differences in growth rate here have the effect of artificially increasing the autocorrelation between monthly estimates of recruitment.

Preparing catch data

A routine is available to allow users to store monthly catches and coefficients of the length-weight relationship. For managing the data, three routines are available: (a) data entry, (b) data editing, and (c) viewing or printing of data (see Fig.24).

![CCC CATCH FILE MANIPULATION ROUTINE](image)

**Fig. 24**

**Data entry.** This subroutine will allow monthly catch data (in tonnes) and the constants of the length-weight relationship to be stored onto the data disk.

Monthly catches (always in metric tonnes), and the 'a' and the 'b' constants in the length-weight relationship will be asked sequentially. Note that the "a" value must link length to weight in grams (g). The values of the previous entry
will serve as a default value for the coefficients (a and b). However, the monthly catches will not have a default value. To go back to a previous entry (date), press [F2] (see Fig. 25).

**FILE PARAMETERS**

- **Filename**
  - ANCHOV

- **Beginning period (MM/YY)**
  - 1/53

- **Ending period (MM/YY)**
  - 12/92

- **Month - Year**
  - 1 - 1953

- **Catches (in tonnes)**
  - 2000

- **'a' conversion factor**
  - 9.65

- **'b' exponent**
  - 9.65

**INSTRUCTIONS**

Enter the catch and conversion factors for the length-weight relationship. Press [F2] to go to previous entry, [F3] to quit to ELIPAN III menu or [CD] to obtain default values (conf. only).

![Fig. 25](image)

**Editing routine.** This routine will allow a catch file saved earlier to be edited. To edit, a user may either go through

**FILE PARAMETERS**

- **Filename**
  - ANCHOV

- **Beginning period (MM/YY)**
  - 1/53

- **Ending period (MM/YY)**
  - 12/92

- **Enter the date to edit (MM/YY)**
  - 1/55

- **Catch (in tonnes)**
  - 500

- **'a' factor**
  - 6.56

- **'b' exponent**
  - 3

**INSTRUCTIONS**

Press [F3] to edit, else press any other key.

![Fig. 26](image)
the file sequentially or in any order by entering the date (MM/YY) to go to a particular record. To save the edited file, press [F9]; otherwise, press [F3] to return to the previous menu without changing the file (see Fig. 26).

Print/View routine. This subroutine will allow data stored and/or edited to be displayed on screen or printed. Once a file has been accessed, pressing [F9] will print results to printer, while [F8] will only display results in the screen (see Fig. 27).

![Diagram of Print/View Routine for Stored Catch Data]

**Fig. 27**

**VPA I**

The following input is necessary to run this program: (1) a file identifier independent of the files stored in disk; (2) the number of periods for which catches are available; (3) M and Fε estimates (on an annual basis); and (4) catches by ages starting with the youngest fish (see Fig. 28).

Given the necessary input (above), the results generated by this routine will be displayed graphically (see Fig. 29) and an option of either printing the results to a printer or dumping the graphic result via a plotter or a printer is also given. The user also has the option to repeat the procedure (changing the values of M and/or Fε).
Fig. 28

Fig. 29
VPA II

VPA II makes use of either catch-at-length data or length-frequency data saved on disk through ELEFAN 0. In either type of data, the mean annual catch data (in tonnes) represented by the samples and the constants for the length-weight relationship will be required by the program. For length-frequency data, in addition to the input requirements mentioned earlier, monthly catch (in tonnes) will be asked to allow conversion of the data from length-frequency to catch-at-length type of data. The growth parameters (Loo and K), as well as natural and terminal fishing mortalities will also have to entered (see Fig. 30) to complete the necessary inputs before a length-structured VPA can be performed.

![Fig. 30](image-url)

After the requirements for VPA II have been satisfied, the computer will estimate the steady-state biomass for each length class. These results may be output via a printer. The other VPA II results (estimated $F_1$, catches and population) may be displayed graphically on screen. As with VPA I the user has the option to print the numerical results or dump the graphic results via the printer or plotter (see Fig. 31).
VPA III

VPA III, as mentioned earlier, incorporates the features of both VPA I and II. A rather comprehensive set of monthly length-frequency and catch data is needed to successfully apply the routine. Data that were stored as a "length-frequency" data file may be transformed to a catch-at-length data file using a file of monthly catches in tonnes (see Fig. 31)

Fig. 31

![Diagram showing file parameters and graphics output of VPA III]

<table>
<thead>
<tr>
<th>FILE PARAMETERS</th>
<th>DISC LIBRARY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filename</td>
<td>DISK002</td>
</tr>
<tr>
<td>Specie name</td>
<td></td>
</tr>
<tr>
<td>Other labels</td>
<td></td>
</tr>
<tr>
<td>No. of samples</td>
<td></td>
</tr>
<tr>
<td>Class interval</td>
<td></td>
</tr>
<tr>
<td>Range of length</td>
<td></td>
</tr>
</tbody>
</table>

**INSTRUCTIONS**

Choose file from disc library. Press [F10] for help

Fig. 32
above). The user will be asked to select a file from the library (see Fig. 32) of catch files. The catch-at-length data file may be printed to a printer. Because of the way this routine is programmed, continuous computer paper is needed for printing.

The user is also given the opportunity to smooth the computed catch-at-length data. The growth parameters (Loo, K, C and WP), the natural fishing mortality and the terminal fishing mortality are required input. This routine may take up to 10 minutes, depending on the size of the file and the processing speed of the available computer. Using the parameters given, it will also initialize the array that stores the results for computed monthly summaries (see Fig. 34).

Once the above routine has been completed, the user will be given the opportunity to reanalyze separate cohorts, one after the other (see Fig. 33). The results generated in each cohort analysis will update the array that stores the monthly summaries. Computed results will be shown graphically and these results may either be printed or dumped to a plotter or printer (see Fig. 22). To proceed to the monthly summary routine, enter [F9] when a cohort number is prompted.

![Virtual Population Analysis III: Length-converted VFA](image)

Fig. 33
For the monthly summaries, only the date (MM/YY) will be asked. The summaries will be shown graphically. These results, as in the case of the above routine, may either be printed to a printer or plotted via a plotter and/or printer (see Fig. 34 and Fig 23).

**Fig. 34**
WORKING WITH ELEFAN IV

Introduction (adapted from Munro 1984).

In conventional catch curves, the logarithms of the relative abundances of successive age groups are plotted against age, in order to obtain an estimate of the total instantaneous mortality rate, \( Z \), from the slope of the descending right arm of the plot. Values of the left of the descending arm of the graph, reflecting catches of incompletely retained fishes are ignored.

Pauly (1982, 1983) has described how length-frequency distributions can be converted to age-structured catch curves in which the observed frequencies in successive length groups are divided by the time required for a fish to grow through each length group and plotted against the estimated ages of the mid-points of successive length groups. From the slope of the resulting length-converted catch curve, an estimate of \( Z \) is derived. This value is composed of the fishing mortality rate (\( F \)) generated by the gear in question plus the natural mortality rate (\( M \)). Usually, this method is applied only to length groups which are larger than the size (\( L' \)) at which the fishes become fully retainable by the gear and is inapplicable to gillnets and other gears which exhibit a bell-shaped selection curve or to length groups which are within the selection ogive of gear such as trawls.

The method now proposed is based around the premise that for any fishing gear, the catch (\( N \)) of each successive catch is a function of the probability of retention (\( P \)) of that size group in the fishing gear and the relative abundance (\( A \)) of that size group in the population. Also, the total mortality coefficient (\( Z \)) observable between successive length groups in an exploited stock is the sum of the natural mortality coefficient plus the product of the probability of retention, \( P_1 \), and the prevailing fishing mortality coefficient (\( F \)).

Thus, for the \( i \)th length group

\[
Z_i = M + P_1 F
\]

Thus, a regression of estimates of \( Z_i \), against \( P_1 \) should yield a regression of slope \( F \) and \( Y \)-axis intercept equal to \( M \).
Methodology

The requirements for implementing the method are as follows:

a. Unbiased estimates of the length composition of the catch averaged over a year which will be entered through ELEFAN 0 and can readily be accessed through the displayed disk library once ELEFAN IV is accessed through the main menu of the package;

b. Estimates of parameters of the von Bertalanffy growth function (Loo and K); and

c. Estimates of the probability of retention (P) at successive lengths (L), acquired independently of the estimates of the mean annual length composition. (See Hamley (1975) and Pope (1975) for reviews of various methods for estimating selection curves).

![Graph showing ln(N/Δt) vs. Relative age (year-t0)]

Fig. 35. Length-converted catch curve for fishes exploited by a single size of gill net in which values of log N = A_m/t_{i+1} are plotted against the relative ages (t_m - t_o) attained at the mid-points of successive length groups. Note the changes in slope resulting from mesh selectivity. Data from Table 1.
Fig. 36. Regression of coefficients of mortality, \( Z_i \), between successive length groups against the probabilities of retention, \( P_x \), at the median points between those successive length groups. Data from Table 1.

Table 1. Hypothetical example \((K = 0.5, L_{100} = 30, M = 1.00, F = 2.00)\) showing steps in derivation of estimates of total mortality rate \((Z_i)\) from catches \((N_m)\) in successive length groups \((L_m)\) by calculating apparent abundance \((A_m = C_m/P_m)\) and true relative abundance \((R_m = A_m/\Delta t_m)\). Survival rate \(S_t = R_{m+1}/R_m\) and \(Z_i = \log_e S_i/\Delta t_{m,m+1}\). See text for details. The ages, \(t\), of successive lengths are the relative ages which are calculated by assigning a zero value to the parameter, \(t_0\), of the von Bertalanffy growth function. The subscripts i and m refer to the beginning and midpoint of a length group.

<table>
<thead>
<tr>
<th>(L_m)</th>
<th>(L_i)</th>
<th>(P)</th>
<th>(N_m)</th>
<th>(A_m)</th>
<th>(\Delta t_{i,i+1})</th>
<th>(R_m)</th>
<th>(S_i)</th>
<th>(\Delta t_{m,m+1})</th>
<th>(Z_i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.5</td>
<td>16.0</td>
<td>.04</td>
<td>12</td>
<td>89</td>
<td>742</td>
<td>( 1.597 )</td>
<td>.148</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>17.0</td>
<td>17.0</td>
<td>.28</td>
<td>12</td>
<td>69</td>
<td>43</td>
<td>1.672</td>
<td>.130</td>
<td>1.524</td>
<td>5.011</td>
</tr>
<tr>
<td>18.0</td>
<td>18.0</td>
<td>.62</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>( 1.751 )</td>
<td>.174</td>
<td>.904</td>
<td>2.27</td>
</tr>
<tr>
<td>18.5</td>
<td>18.5</td>
<td>.85</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>1.833</td>
<td>1.74</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>19.0</td>
<td>19.0</td>
<td>.90</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.007</td>
<td>1.74</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>19.5</td>
<td>19.5</td>
<td>.97</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>20.0</td>
<td>20.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>20.5</td>
<td>20.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>21.0</td>
<td>21.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>21.5</td>
<td>21.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>22.0</td>
<td>22.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>22.5</td>
<td>22.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>23.0</td>
<td>23.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>23.5</td>
<td>23.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>24.0</td>
<td>24.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>24.5</td>
<td>24.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>25.0</td>
<td>25.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>25.5</td>
<td>25.5</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
<tr>
<td>26.0</td>
<td>26.0</td>
<td>1.00</td>
<td>43</td>
<td>272</td>
<td>633</td>
<td>2.197</td>
<td>1.91</td>
<td>1.524</td>
<td>5.004</td>
</tr>
</tbody>
</table>

a) Values in brackets are \(t_m\)'s
The analytical procedure is illustrated in Table 1 and Figs. 35 and 36 and by a hypothetical example from a gill net fishery. The procedure is as follows:

a. Tabulate the annual average length frequencies \(N_m\) of the catch and the probabilities of retention \(P_m\) of successive length groups \(L_m\) (the subscript \(m\) refers to the mid-point of a length group);

b. Calculate the apparent relative abundances \(A\) of successive length groups within the selection curve as \(A_m = N_m/P\);

c. Calculate the ages (in years) at the start \(t_k\), mid-point \(t_m\), and end \(t_{k+1}\) of successive length groups and the time required \(t_{k+1-k}\) for a fish to grow through each length group;

d. Divide \(A_m\) by \(t_{k+1-k}\) to get the true relative abundance \(R_m\) of the fish in each length group (Fig. 1 shows a length-converted catch curve in which loge \(R_m\) is plotted against relative age \((t-t_0)\). Note the sigmoid shape generated by mesh selectivity);

e. Calculate the survival rate between successive length groups, \(S = R_{m+1}/R_m\) and the annual coefficients of total mortality, \(Z_m = \loge S_m\), \(t_m, m+1\) between successive midlengths; and

f. Plot \(Z_m\) against \(P_k\), to obtain estimates of \(F\) and \(M\), using the equation \(Z_m = P_k(F+M)\), and obtain an estimate of \(M\) (Fig. 36).

Computational options and variations

A variation of the above is implemented in this package wherein it is assumed that the values of \(N_m\), \(A_m\) and \(R_m\) are derived from sampling gear (Gear 1) with a retention range smaller than is used commercially, and where the \(Z_m\)'s are computed from the above values and are plotted against the probabilities of retention, \(P_k\), of successive lengths, \(L_k\), in the commercial gear (Gear 2). In such a case, estimates of \(Z\) pertaining to lengths below the retention range of the commercial gear (where \(P_k=0\)) are direct estimates of natural mortality \((M)\). In other words, the probabilities of capture of Gear 1 are used to compute the \(Z\) values, which are then plotted against the probabilities of capture of Gear 2 to estimate \(M\) as the \(Y\)-intercept of the regression.
Putting the foregoing in a different form, if estimates of \( R_m \), and thus \( Z_k \), can be obtained for length groups outside of (usually smaller than) the retention range of the commercially utilized fishing gear, estimates of the natural mortality rate, \( M \), of the unexploited size groups can be obtained from the sample length-frequency distributions.

Probability of capture for Gear 1 may be entered and saved through ELEFAN 0.

Another computational option as suggested by J. Moreau (pers. comm.) is to interpolate the computed values of \( Z_k \) and to plot these values against the values of \( P \) entered (Gear 2). This would differ from the methodology suggested by J. Munro (1984) in which, \( Z_k \) between successive length groups is plotted against \( P_k \) values interpolated between successive length groups.

**Fig. 37. J. Moreau’s modified approach. Data from Table 1.**

For either of the two options, the regression equation and standard error is estimated.
WORKING WITH ELEFAN V

Introduction

ELEFAN V is based on an approach suggested by Morgan (1987), and is used to estimate growth parameters through simultaneously analyzing length-frequency and growth increment data.

Growth increment data can be of three different types:

- tagging/recapture data;
- growth increments derived from length-at-age data;
- growth increments obtained from model class progression analysis (MPA).

The approach used in ELEFAN V for the simultaneous analysis of length-frequency data and growth increment data involves presentation of a mean goodness-of-fit index (Rm) for different data types in the form of a response surface.

This program contains two major routines:

1. Response surface for analysis of length-frequency and growth increment data and
2. Growth parameters estimation from growth increment data alone.

Response surface for length-frequency and growth increment data

The response surface presents the goodness-of-fit index (Rn) for the length-frequency data, computed as:

\[ Rn = \frac{10^{-\text{esp}}}{10} \]

and of the coefficient of determination \( r^2 \) of the actual and predicted lengths at the end of the growth increments; thus, \( Rm = \frac{(Rn + r^2)}{2} \).

The response surface is a two-way table with one growth parameter varying on the X-axis and another parameter on the Y-axis. More particularly, for two parameters K and Loo (leaving C and wp constant), the response surface provides Rm (x1000) values (see Fig. 3B).
This routine also provides an assessment of the compatibility of length-frequency to growth increment data (see Fig. 39)

Growth parameters estimation from growth increment data

The estimation of growth parameters from growth increment data begins with a Gulland and Holt plot (ΔL/Δt vs L, see Gulland and Holt 1959) giving the user a preliminary view of the data structure. Using the G-H plot allows the user to detect and delete outliers.
After samples have been selected, a regression line is fitted on the data for preliminary estimates for K and Loo. The plot of deviations from the line over time is used to identify seasonal patterns in the data, and provides initial estimates of C and WP (see Fig. 40).
The preliminary estimates of Loo and K from the G-H plot may be improved using either of three methods:

1. Munro's method (Munro 1982);
2. Fabens' method (Fabens 1965); or
3. A newly developed method for the analysis of seasonally oscillating growth using increment data (see below).

The selected points may then be saved in another file to be analyzed jointly with length-frequency data. It is recommended to try to estimate growth parameters from available growth increment data (and to identify which part of the available data can be used for such purpose) before proceeding to the simultaneous analysis of \( L/F \) and growth increment data.

A new method for estimating the parameters of a seasonally oscillating growth curve from growth increment data

When growth increments are available, each of which has an initial date \((t_1)\) and length \((L_1)\) and a final date \((t_2)\) and length \((L_2)\), the growth parameters \(Loo, K, ts, \) and \(C\) of a seasonally oscillating version of the VBGF can be computed using a multiple regression derived as a further development of the model of Appeldoorn (1987), of the form:

\[
Y = -Kd + b_1 \sin 2\pi t_2 - \sin 2\pi t_1 + b_2 \cos 2\pi t_2 - \cos 2\pi t_1
\]

\[
\text{where } Y = \log_e \frac{(L_0 - L_2)}{(L_0 - L_1)}
\]

\[
b_1 = \frac{CK \cos 2\pi t_2}{2\pi}
\]

\[
b_2 = \frac{CK \sin 2\pi t_2}{2\pi}
\]

\[
t_s = \tan^{-1}(b_3/b_2)/2\pi
\]

\[
d = t_2 - t_1
\]

Thus, one can compute \(R^2\) values for different values of \(Loo\) until \(R^2\) is maximized. Therefore, estimating the four parameters needed \((Loo, K, ts\) and \(C\)) involves searching in only one dimension.

[Note that "\(t_s\)" is related to the Winter point (WP) used in the ELEFAN programs through \(t_s + 0.5 = WP\)]
WORKING WITH MODAL CLASS PROGRESSION ANALYSIS (MPA)

Introduction

Prior to the discovery of daily and seasonal rings in fish otoliths and scales, the detailed analysis of length-frequency data was, besides marking-recapture studies and direct observation of captive fish, the only method available to draw inferences on the growth of tropical fishes, and indeed the only method that could be applied routinely (Pauly 1987).

A great deal of confused terminology surrounds the use of length-frequency data in growth analysis and before discussing Modal Class Progression (MPA), it is necessary to clarify the terms we shall use.

We believe the term "Petersen's method", which is often very loosely used, should be restricted to the analysis of one length-frequency sample at a time. This analysis usually consists of identification of class modes or means in the sample in question using either a purely visual approach, graphical techniques, or computer-based methods of class separation. The next step in Petersen's technique is the subjective attribution of relative ages to the class modal or mean lengths thus identified. This then enables the estimation of growth parameters using the identified size groups, and the relative ages attributed to them.

Petersen's method for length-frequency analysis is illustrated in Fig. 41 below, which also illustrates what happens if the subjective attribution of relative ages goes wrong (which happens quite frequently, and is behind many statements that the VBGF is not suitable as a model for growth of this or that fish).

Fig. 41. Application of the 'Petersen Method' (sensus stricto) to a hypothetical length-frequency sample. Note that the time separating various peaks must be assumed, a difficult task in animals which may, or may not, spawn several times a year.
We shall use the term "Modal Progression Analysis" (MPA) for the method in which several length-frequency samples are plotted sequentially, and where the apparent shift of modes is used to infer growth. In MPA, the identification of class modal or mean lengths is done as in Petersen's method, after which subjective identification of modes perceived to belong to the same cohort of fish enables the tracing of growth curves, or computation of growth increments between modes. Thus, in MPA, the critical issue is not the attribution of "ages" to the various groups — as in the Petersen method — but the linking of means perceived to belong to a same cohort (see Fig. 42).

![Diagram of Modal Progression Analysis](image)

**Fig. 42.** Application of 'modal progression analysis' to a set of two samples obtained at known times \( t_1, t_2 \). Note that the problem here is the proper identification of peaks to be interconnected, and not the question as to how the much time separates the peaks (as was the case in Fig. 41).

The subjective judgement required in the second step of both of these methods and the large potential errors thus induced have prompted a search for ways of reformulating these two traditional methods so that improved solutions are obtained.

This was achieved in this package in two ways:

1. by constraining possible solutions through the use of pre-selected growth functions, usually the VBGF and especially its seasonally oscillating formulation (ELEFAN I); and

2. by combining the length-frequency data with other information, such as growth increment data from tagging-recapture experiments or length-at-age data obtained, e.g., from otolith analysis (ELEFAN V).

However, many researchers prefer to stick to modal progression analysis (MPA) despite its shortcomings. For these, we have included here a computerized version of MPA as part of the Compleat ELEFAN package.
This routine is subdivided into four parts: (a) separation of length-frequency samples into normal distributions using the method of Bhattacharya (1967) or entry of means (and standard deviations if available), (b) linking of the means, (c) computation of the C.V. of Loo and (d) saving of the linked means (which represent growth increment data, having the same characteristic as tagging/recapture data) which can then be analyzed using ELEFAN V.

Data files and sample access

As in the case of other routines of the Compleat ELEFAN, the computer will first search for a disk library in the data disk when this routine is accessed. Once the library is read, the directory will be shown in the right portion of the screen. The user will be asked to select a file from the list displayed.

Once a file has been read, its parameters will be shown in the display. The computer will read the samples after [F9] has been pressed. Next, the computer will ask for the number of the sample to work with (a value entered that is greater than the total number of samples or less than 1 will not be accepted). Once the sample is accessed, the program will chain to Part I of MPA, which is the separation of normal distribution using the Bhattacharya method.

Bhattacharya method

There are four boxes in the left side of the screen (Fig. 43). The first box contains the filename and sample number accessed. The next box contains the parameter of the turtle, i.e., the coordinates for the X and Y axes (after transformation) and the population (N). The third box contains the parameters of a normally distributed component that has been identified, and the last box is for the messages. In this particular instance (see Fig. 43) the user is asked to mark the first point by moving the turtle using the arrow keys and then pressing the [F6] or <SPACE BAR> when the turtle is properly positioned. Later the user will be asked to identify the second point.

[F10], [F3] and [F8] are interrupt keys whose functions (help, quit to main menu and graphic printer dump respectively) may be accessed at any time, as needed.

The population in each normally distributed component will be computed and the regression line and the normal curve will be drawn (see Fig. 44). The user will be given the opportunity to select new points if the computed parameters of the component generated is not satisfactory.
The estimated population of the defined component will be subtracted from the sample and a display of the new transformation will be displayed.
The procedure is repeated until most or all visible components are separated, in which case a user can press [F9] to end. A summary of results will be displayed and may be printed.

The program will then compute expected frequencies for the whole sample (see Fig. 45); when sufficient degrees of freedom are available, a chi-square test will be performed, indicating how well the expected frequencies fit the observed frequencies. These results may be plotted on a per-sample basis on a plotter (see Fig. 46). The program will automatically store the computed means for later analysis. These means, however, will be lost if you immediately return to the main menu (by pressing [F3]) and it will also be overwritten if the same sample is reanalyzed.

Linking of the means

When all (or enough) samples have been analyzed, enter 0 to proceed to the routine allowing linking of the mean lengths (i.e., derivation of growth increment data). This routine uses the four arrow keys to move the turtle (see Fig. 47).

Position the turtle at a point representing the start of a growth increment, and mark this point by pressing [F6] or <SPACE BAR>, then move the turtle to a point representing the end of the growth increment in question and mark it (note that negative growth increments will be accepted).
Continue until all points representing end points of growth increments have been marked.

Press [F7] to redo the routine. The C.V.'s, means and $\Delta L/\Delta t$ value for each pair of means are saved for later analysis.
C.V. vs growth rate

In this newly developed routine, a plot of the $\Delta L/\Delta t$ values vs the mean C.V. of the linked means is generated (see Fig. 48). This plot may be used to identify the most likely of two alternative hypotheses concerning the distribution of lengths about age:

(i) constant C.V.; or
(ii) C.V. declining with $\Delta L/\Delta t$ and hence with length.

In case the first alternative applies, an overall mean C.V. is computed from the individual C.V. estimates. For the second alternative to be likely, the correlation coefficient between the growth increments and the means of the paired C.V. values must be significant and positive; also the intercept of the (GM) regression (i.e. the C.V. of Loo) must be positive.

Estimates of the C.V. of Loo may be used, when available, as an input to Sainsbury's (1980) method of computing population growth parameters from growth increments derived from fish with individually variable growth parameters. The linked points (i.e. growth increments) may be saved on disk as an increment data file by pressing [F9].
A re-linking of the means may be done by pressing [F7] if the above plot gives an unsatisfactory result.

ACKNOWLEDGEMENTS

The authors of the Compleat ELEFAN package wish to extend their gratitude to various fisheries scientists who have generously contributed their time and effort to the development and testing of various routines. Although we have received a lot of suggestions for improvements, we were not able to incorporate all of them in this version. However, we will consider them when we start developing the next version.
REFERENCES


The ICLARM Software Project

ICLARM has initiated, in early 1987, a new activity, the "ICLARM Software Project", involving the dissemination of software for calculators and microcomputers, along with their supporting documentation (e.g., manuals or scientific paper). The material now available for distribution, presently consists of software produced at ICLARM, but will in the near future include public domain programs, as well as software made available by their authors to ICLARM for free worldwide distribution. This software will include the areas of fish population dynamics, fisheries and aquaculture economics, fish genetics and other fields covering ICLARM’s areas of interest.

All software will be made available at cost, i.e., including only material, mailing and handling costs. Copyrighted material will not be distributed. The availability of new programs will be made public through notices in NAGA, the ICLARM Quarterly, brochures and the ICLARM publication catalogue. Authors and potential contributors to this scheme are invited to write to the Director, Resources Assessment and Management Program, ICLARM, MC P.O. Box 1501, Makati, Metro Manila, Philippines. Note that checks payments to ICLARM must be made to a US-based bank.