BSPE 00388-665-5

심해저 광물자원탐사에서의 수심자료획득 및 자료분석에 관한 연구

A Study on the Analysis of the Bathymetric Data for the Deep Seabed Mining Exploration

1994. 3

한 국 해 양 연 구 소

한국해양연구소 도서실
장서
1997. 4. 11.
44409

제 출 문

한국해양연구소장 귀하

본 보고서를 "심해저 광물자원탐사에서의 수심자료획득 및 자료분석에 관한 연구"사업의 최종보고서로 제출합니다.

1994년 3월

연구책임자 : 문 재 운
연 구 원 : 강 호 석
장 태 헌
김 기 현

요 약 문

I. 제 목

심해저 광물자원탐사에서의 수심자료획득 및 자료분석에 관한 연구

II. 연구내용 및 결과

해저지형 분석시 대부분 객관적인 자료없이 프로파일상에 나타나는 지형 모습을 상대적으로 비교하는 정성적인 방법을 사용하고 있으며, 정량적분석으로는 경향면 분석과 같은 통계적 방법이 사용되고 있다. 그러나 이같은 방법으로는 지역간의 대략적인 지형경향성의 유사성 유무만을 알 수 있을뿐 지형상태의 특성을 지시하는 요인들에 관한 자료를 얻을 수 없으므로 지역별 지형특성 정량화 및 이를 통한 징역간의 지형비교라는 관점에서 볼때 적용의 한계가 있다. 따라서 본 연구에서는 수심측정기를 통해 입수되는 수심자료를 사용하여 조사지역의 지형특성을 구성하는 요인들을 정량화하는 프로그램을 작성하고자 하였다.

지형특성을 구성하는 요인들로 1) 평균수심, 2) 해저면의 평균경사도, 3) 해저면의 경사도별 분포, 4) 경사구간별 평균수심, 5) 경사구간별 평균수평거리, 6) 해저면 기복의 평균표고차, 및 7) 해저면 기복점의 수를 사용하였으며, 실제 적용시 해저지형을 정량화하는데 이들 요인들이 적절히 사용되어질 수 있음이 밝혀졌다.

그러나 본 프로그램을 통한 지형분석은 분석조건에 따라 결과가 달라질 수 있으므로 분석지역에 대하여 사전에 충분한 검토작업이 선행되어야 하는 제약이 있으며, 아울러 분석조건을 달리하여 얻은 결과를 상호 비교함으로서 보다 정확한 분석을 수행할 수 있다.

SUMMARY

I. Title of Study

A Study on the Analysis of the Bathymetric Data for the Deep Seabed Mining Exploration.

II. Abstract

Most of topographic analysis has been based on the qualitive comparasions of topographic characteristics in the analog profiles of echosounder. In the view of quantitative analysis, trend surface analysis has been used for defining topographic trends. However, these method do not provide sufficient information to figure out each topographic characteristics in quantitative terms. In this study, we tried to define several factors which provide topographic characteristics of sea bottom features and designed a computer program for topographic analysis using such factors.

To define topographic characteristics, seven factors such as 1) average water depth, 2) average slope angle of sea bottom, 3) frequency of each slope interval, 4) average water depth of each slope interval, 5) average horizontal distance of each slope interval, 6) average vertical difference of relief, 7) number of extreme (i.e., top and bottom) points of relief in given distance were determined. These topographic factors are proved to be effactive and useful for quantifying sea bottom configurations.

Howerer, this method also has limitations. Above all, you must divide areas with similar characteristics before you apply this method, then determine analytical conditions, such as interval of slope angle, minimum vertical difference to be determined as relief, unit interval in horizontal distance etc. Therefore, the accuracy of analysis depends on how well you prepared such pre-analysis work.

CONTENTS

List of Tables 11
List of Figures 12

I. Introduction 13

II. Data used for topographic analyaia 15

III. Processing of bathymetric data 17

1. Interval of bathymetric data 17
2. Filtering of bathymetric data 18

IV. Analytical factors for defining sea bottom topography 22

1. Average water depth 22
2. Slopes 23
   (1) Average slope angle, % of each slope interval and its average water depth 23
   (2) Average horizontal distance of each slope interval 24
3. Relief 26
   (1) Average vertical difference 26
   (2) Level of horizontal development 28

V. Analytical procecure 30

1. Limitations of analysis 30
2. Example of analysis 31

VI. Results 36

References

Appendix 1. Program for filtering of bathymetric data
Appendix 2. Program for topographic analysis
Appendix 3. Out-put data print-outs of example area (A), (B), and (C)

목 차

표목차 11
그림목차 12

I. 서 언 13

II. 지형분석에 사용되는 자료 15

III. 수심자료의 처리 17

1. 측심위치 간격 17
2. 이상치 제거를 위한 수심자료 검정 (filtering) 18

IV. 분석에 고려된 지형요인 22

1. 평균수심 22
2. 해저면의 경사각 23
(1) 해저면의 평균경사도, 경사각도별 분포 및 평균수심 23
(2) 경사구간별 평균수평거리 24
3. 해저면의 기복 26
(1) 해저면 기복의 평균표고차 26
(2) 해저면 기복의 발달정도 28

V. 분석방법 30

1. 분석상의 제약 30
2. 분석의 예 31

VI. 결 론 36

참고문헌

Appendix 1. Program for filtering of bathymetric data
Appendix 2. Program for topographic analysis
Appendix 3. Out-put data print-outs of example area (A), (B), and (C)

표 목 차
(List of Tables)

Table 1. Interval of data points relating to the ship speed and bathymetric data receiving interval. 17

Table 2. Results of analysis from three example areas. Out-put data print-outs of Area (A), (B), and (C) are listed in Appendix 3. 32

그 림 목 차

(List of Figures)

Fig. 1. Example of input data. 16

Fig. 2. Filtering method for eleminating mechanical error data of water depth. 20

Fig. 3. Example of bathymetric profiles showing (a) before filtering and (b) after filtering. 21

Fig. 4. Bathymetric profiles of three example areas. Results of analysis of Area (A), (B), and (C) are in Table 2. 35

I. 서 언

해양탐사를 수행함에 있어 대상지역에서 수집되는 수심자료는 해저지형을 파악하는데 필요한 가장 기본적인 자료라할 수 있다. 따라서 정확한 해저지형의 상태를 규명하기 위하여 많은 정밀한 수심측정기기들이 개발되어 왔다.

한편, 수심측정기기의 발전에도 불구하고 수집된 수심자료들을 사용하여 조사지역의 해저지형을 분석하는데는 단지 기록장치(recoder)에 기록되는 해저면의 프로파일(profile)상에 나타나는 지형의 모습을 정성적으로만 분류, 평가하는데 그치고 있다. 따라서 지역간의 해저지형 비교시에도 객관적인 자료의 비교없이 프로파일상에서 보여지는 자료만을 사용하는 정성적 방법에 의존하고 있다. 그러나 정량적 비교를 위한 분석기법 개발의 필요성이 대두됨에따라 지역의 경향면 분석을 통해 얻은 상관계수(correlation coefficient)를 상호 비교함으로서 유사성의 유무를 판정하는 통계적인 방법(Merriam et al., 1966; Davis, 1973)이 개발되었다. 그러나 이와같은 방법으로는 단순히 상대적으로 비슷한 크기를 갖는 지역간의 대략적인 지형경향성의 유사성 유무만을 밝힐 수 있을뿐 해저면의 경사각, 표교차, 해저분지의 크기 등 지형상태의 특성을 지시하는 요인들에 관한 자료를 얻을 수 없으므로 지역별 지형특성의 정량화 및 이를통한 지역간의 지형비교라는 관점에서 볼때 적용의 한계를 보여주고 있다.

본 연구에서는 현재 흔히 사용되는 수심측정기를 통해 입수되는 수심자료를 이용하여 조사지역의 지형특성을 구성하는 요인들을 정량화하는 프로그램 개발을 시도하였다. 따라서 본 프로그램에서 얻은 지형구성 요인들의

정량화된 값들로서 조사지역의 지형상태를 파악할 수 있도록 하였으며 아울러 비교대상지역에서 산출된 값과의 비교를 통해 차이점을 분석할 수 있도록 하고자 하였다. 그러나 본 프로그램을 통해 얻어지는 지형구성요인들의 정량값은 분석에 사용된 전체 자료지역의 평균값을 나타내는 것이므로 아무런 사전작업 없이 단순히 임의로 설정한 지역의 자료를 사용하여 분석을 실시하였을 경우 실제의 지형상태와는 상이한 결과를 얻을 수 있다. 따라서 본 연구결과를 적용하기 위해서는 먼저 분석대상지역의 전반적인 지형분포를 기록장치를 통해 기록되는 해저면의 프로파일을 이용하여 지형상태에 따른 대략적인 구역을 설정한 후 본 프로그램을 통해 각 구역별 지형특성을 정량화시켜야하며 또한 대략적인 구역설정 과정에서 발생될 수 있는 오차는 시행착오의 과정(try and err method)을 통해 줄여야하는 분석기법상의 제약을 가지고 있다. 참고로, 본 프로그램은 C-language를 사용하여 작성하였으며 컴파일러로는 Turbo C 2.0을 사용하였음을 밝혀둔다.

II. 지형분석에 사용되는 자료

수심측정에 일반적으로 사용되는 장비로는 한방향의 음파를 사용하여 수심을 측정하는 단일빔음향측심기(single-beam echosounder)가 있다. 그러나 이와같은 음향측심기로는 조사측선을 따라 측정된 2차원적인 수심자료만을 얻을 수 있는 관계로 조사해역의 수심도 작성시 측선간의 간격에따라 정밀도가 달라진다. 따라서 정밀도를 높이기 위해서는 측선간의 간격을 좁혀야 하므로 광역해역조사에는 많은 시간이 소요된다. 이와같은 단점을 보완하기 위하여 다중빔음향측심기(multi-beam echosounder)가 개발되었다. 최근 개발되어 사용되는 Seabeam과 같은 다중빔음향측심기는 측선상의 수심만을 측정하는 단일빔음향측심기의 단점을 보완하여 측선을 따라 일정한 폭(width)내에 포함되는 지역의 수심을 동시에 연속적으로 측정할 수 있도록 함으로서 신속하게 넓은지역을 조사할 수 있으며 관련자료들을 디지탈형태로 수신할 뿐아니라 수심자료의 경우 등고선으로 표시된 3차원적인 도면상태로 표시하여준다. 이와같은 결과는 단일빔음향측심기를 사용하였을 때에 비해 보다 정밀하게 지형의 분포상태를 보여주는 관계로 비교적 넓은지역의 지형특성을 쉽게 이해할 수 있도록 한다. 그러나 이와같은 잇점에도 불구하고 고가의 장비가격으로 인하여 일반화되지 않고 있다.

본 연구에서 사용되는 지형분석용 프로그램은 현재 사용되고있는 단일빔 또는 다중빔음향측심기를 통해 입수되는 디지탈 수심자료 및 위치정보를 사용하여 지형특성을 이루는 요인(factor)들의 값을 계산하도록 구성되었다(Fig. 1). 한편, Seabeam과 같은 다중빔음향측심기를 통해 입수되는 수심자료는 단일빔음향측심기와는 달리 여러개의 음파수신장치를 통해 복수의 수

심자료들이 동시에 입력되므로 본 프로그램을 사용하기 위해서는 이들 가운데서 측선을 따라 수신되는 자료만을 후처리(post-processing)하여 따로 분리해서 사용하여야 한다.

① ② ③ ④ ⑤ ⑥

```
1993/183.000058 15.54907 -132.28479 5089
1993/183.000059 15.54907 -132.28479 5089
1993/183.000100 15.54907 -132.28479 5089
1993/183.000101 15.54907 -132.28479 5089
1993/183.000102 15.54907 -132.28478 5088
1993/183.000103 15.54907 -132.28478 5088
1993/183.000104 15.54906 -132.28478 5088
1993/183.000105 15.54906 -132.28478 5088
1993/183.000106 15.54906 -132.28478 5088
1993/183.000107 15.54906 -132.28478 5088
1993/183.000108 15.54905 -132.28478 5088
1993/183.000109 15.54905 -132.28478 5088
1993/183.000110 15.54904 -132.28477 5088
1993/183.000111 15.54904 -132.28477 5088
1993/183.000112 15.54903 -132.28477 5088
1993/183.000113 15.54903 -132.28477 5088
1993/183.000114 15.54903 -132.28477 5088
1993/183.000115 15.54903 -132.28477 5088
1993/183.000116 15.54903 -132.28477 5088
1993/183.000117 15.54903 -132.28477 5088
```

시 분 초

Fig. 1. Example of input data. ① year, ② Julian day, ③ time, ④ latitude, ⑤ longitude (west represented in ⊖ value), ⑥ water depth (m).

III. 수심자료의 처리

해저지형에 관한 정보는 연속적으로 측정된 수심자료 및 측정위치에 관한 자료를 종합함으로서 알 수 있다. 따라서 지형정보의 정확성은 측심위치간격과 수심자료의 정밀도에 따라 차이를 보이게 된다.

1. 측심위치 간격

측심위치간격은 Table 1 에서 보여지는 것처럼 측정기기의 자료수신 간격과 선박의 속도에 따라 달라지므로 조사지역의 범위 및 조사목적에 따라 적절히 조절할 수 있다. 그러나 일반적으로 수심측정기는 기기의 상태 및 조사선박의 상태를 고려하여 자료수신간격을 일정하게 고정시켜 사용하고 있으며 조사시 선박의 속도 역시 조사비용 및 수심측정기의 상태를 고려해야 함으로 일정한 범위이내에서만 속도조절이 가능하다.

Table 1. Interval of data points relating to the ship speed and bathymetric data receving interval.

Ship Speed	Bathymetric Data Receiving Interval						
	1 sec	2 sec	5 sec	10 sec	15 sec	30 sec	60 sec
5 knot	2.57 m	5.14 m	12.86 m	25.72 m	38.58 m	77.17 m	154.33 m
6 knot	3.09 m	6.17 m	15.43 m	30.87 m	46.30 m	92.60 m	185.20 m
7 knot	3.60 m	7.20 m	18.01 m	36.01 m	54.02 m	108.03 m	216.07 m
8 knot	4.12 m	8.23 m	20.58 m	41.16 m	61.73 m	123.47 m	246.93 m
9 knot	4.63 m	9.26m	23.15 m	46.43 m	69.45 m	138.90 m	277.80 m
10 knot	5.14 m	10.29 m	25.72 m	51.44 m	77.17 m	154.33 m	308.67 m

2. 이상치 제거를 위한 수심자료 검정(filtering)

일반적으로 측정장치를 통해 입수되는 수신신호는 기기내에서 제어되나 때로 선체의 흔들림 또는 해황상태 등 수신상태의 불량에 기인하여 수심측정치에 이상값(error data)이 포함되는 경우가 있으므로 정확한 분석을 위해서는 먼저 이와같은 이상값를 제거하는 단계를 거쳐야 한다.

수심측정지점(data point)간의 거리는 Table 1에서 보여지듯이 조사선의 속도 및 자료선택간격에 따라 달라진다. 따라서 먼저 이상값 제거를 위한 자료검색후 조사지역의 특징 및 조사목적에 따라 측정지점간의 거리을 지정하여 이를 기준으로 등간격으로 자료를 취하여 분석에 사용할 수 있도록 프로그램을 작성하였다 (Appendix 1).

Fig. 2는 프로그램의 알고리즘을 보여주는 예로 연속적으로 측정된 수심자료로 부터 일정한 시간간격으로 자료를 취합하는 과정을 보여주고 있다. 즉, 이상치 제거를 위하여 일정한 수(홀수개)의 연속된 수심값을 취하여 이들값중 중앙값(median)을 대표값으로 설정함으로서 이상치를 제거하고자 하였다. 본 프로그램(Appendix 1)에서는 7개의 연속된 수심값에서 중앙값을 취하도록한바, 이는 자료수신간격이 1초, 선속이 8노트인 경우를 가정할때 첫번째 자료가 수신된 지점에서 부터 일곱번째 자료가 수신된 지점까지의 거리는 약 30 m 이며 각각의 자료간격은 4.12 m 이다. 그러나 중앙값을 얻기위하여 사용하는 연속된 수심자료의 갯수는 앞서 언급한 바와같이 측정기기의 자료수신 간격과 선박의 속도, 조사지역의 범위 및 조사목적에 따라 달라질 수 있으므로 이를 조정하기 위해서는 실행프로그램에서 다음과 같이 이값을 바꿔주어야 한다.

Input the parameter of filtering?(odd)

7 <- - - - - - - - - - - - - - - - - - - 7개인 경우

일반적으로 해황 또는 수심측정기기의 이상상태에 기인한 값은 Fig. 3a에서 보여지듯이 주변의 수심과는 전혀 상이한 수심을 보이며 이같은 좁은 범위에서의 급격한 수심변화는 자연상태에서는 기대할 수 없으므로 이와같은 수심값은 위에서 설명한 중앙값을 선택하는 과정에서 제거(Fig. 3b)되어 질 수 있다. 한편, 이상의 과정에서 얻은 중앙값은 연속된 홀수개의 수심값중 가운데 지점의 수심값으로 설정하였다. 즉, 7개의 연속된 수심값을 사용하였을 경우 중앙값은 4번째 자료를 측정한 지점의 수심값으로 사용하게 된다 (Fig. 2).

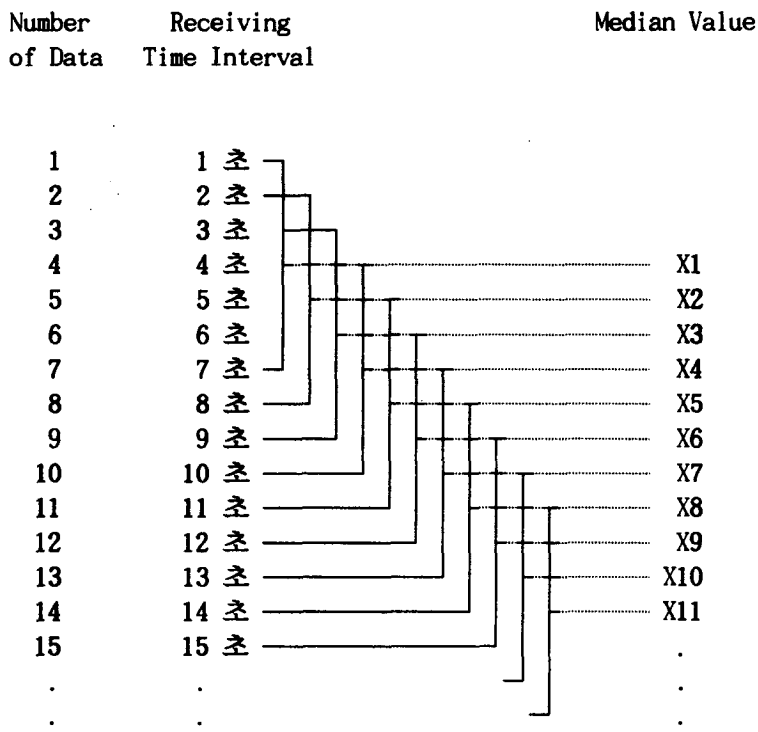


Fig. 2. Filtering method for eleminating mechanical error data of water depth.

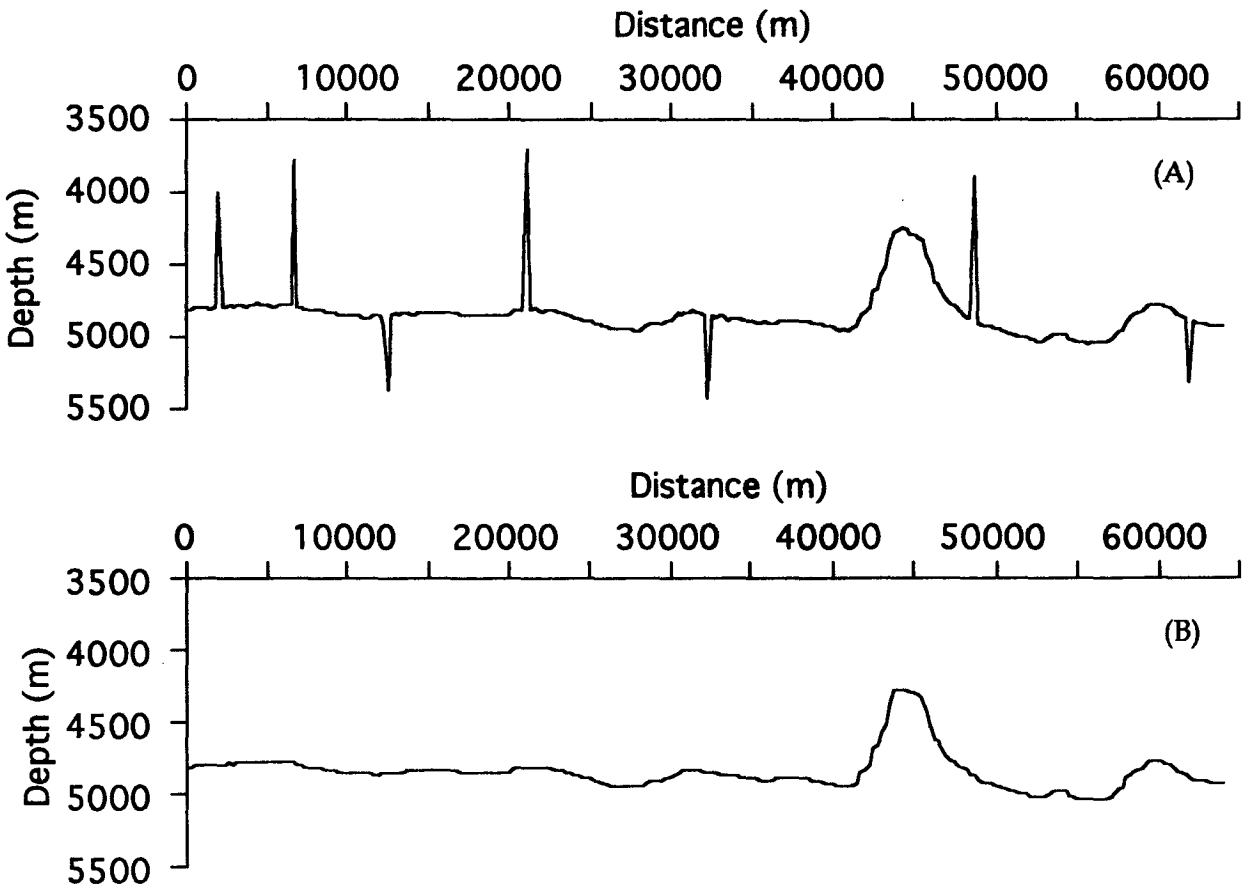


Fig. 3. Example of bathymetric profiles showing (a) before filtering and (b) after filtering.

IV. 분석에 고려된 지형요인

해저면상에서 발달하는 해저평원, 해저능, 해저산, 해구 등 해저지형의 특징은 수심, 해저면의 경사도 및 이들의 연속된 상태, 표고차 등 공간적 배열상태를 지시해주는 요인들을 종합함으로서 구분되어질 수 있다. 따라서 해저지형의 정량화를 위하여 지형의 특성을 이루는 요인으로 다음의 7 가지 항목을 설정하여 각 항목별 값을 수심자료로 부터 계산하기위한 프로그램을 작성하였다 (Appendix 2).

1) 평균수심
2) 해저면의 평균경사도
3) 해저면의 경사각도별 분포
4) 경사구간별 평균수심
5) 경사구간별 평균수평거리
6) 해저면 기복의 평균표고차
7) 해저면 기복의 발달정도

1. 평균수심

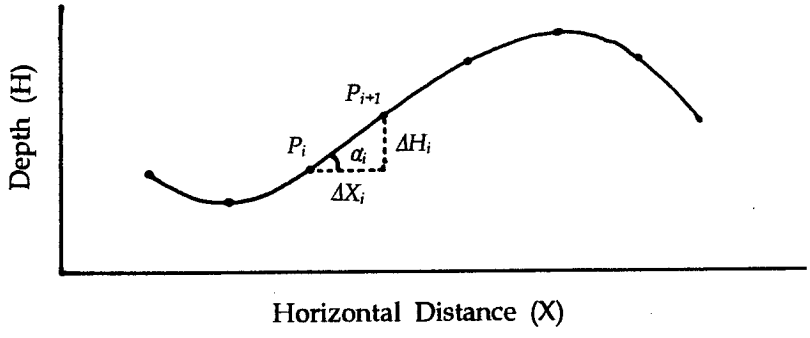
수심은 해저지형 판독에 가장 기본이 되는 자료로 분석하고자 하는 지역에서 측정된 모든 수심자료의 평균한 값을 사용하였다. 평균수심값은 지형의 요철정도, 해저산의 분포, 대상지역의 크기 등 여러가지 주변요인에 따라 계산된 값의 의미정도가 달라질 수 있으므로 지형비교에 있어서 가장 초보적인 자료로 사용된다.

2. 해저면의 경사각

해저면상에서 발달하는 해저지형의 특징은 해저면의 경사도 및 이들의 연속된 상태로서 판단할 수 있다. 따라서 이같은 요인들을 측정하기 위하여 연속적으로 측정된 측심위치 및 수심자료를 사용하여 다음과 같은 지형요인들을 계산하였다.

(1) 해저면의 평균경사도, 경사각도별 분포 및 평균수심

해저면의 경사구배는 해저지형을 판별하는 기본적인 요인이므로 경사각도별 분포로서 해저면의 상태를 파악할 수 있다. 해저면 경사각계산에는 앞서 설명한 수심자료 검정을 거친 자료를 사용하였으며, 두지점간의 거리는 두지점의 위치좌표(경위도 좌표)로 부터 Mercator 도법을 사용하여 계산하였다. 따라서 두지점간의 거리 및 수심차이를 알 수 있으므로 이로부터 평균경사각 및 경사구간별 평균수심을 계산하였으며, 이를 도식화하면 다음과 같다.



$$\Delta H_i = H_{i+1} - H_i$$

$$\Delta X_i = X_{i+1} - X_i$$

$$\alpha_i = \arctan \frac{\Delta H_i}{\Delta X_i}$$

여기서, ΔH_i 는 두지점간의 수평거리, ΔX_i 는 두지점간의 수심차이, α_i 는 경사각을 나타낸다.

계산된 경사각자료들은 경사각의 구간간격(step of angle: a)별로 구분(0-a. a-2a. 2a-3a. 3a-4a. .....) 취합하여 이들의 분포를 백분률로 표시하였다. 한편, 경사각의 구간간격 a는 목적에따라 아래와 같이 실행프로그램에서 선택할 수 있도록 하였다.

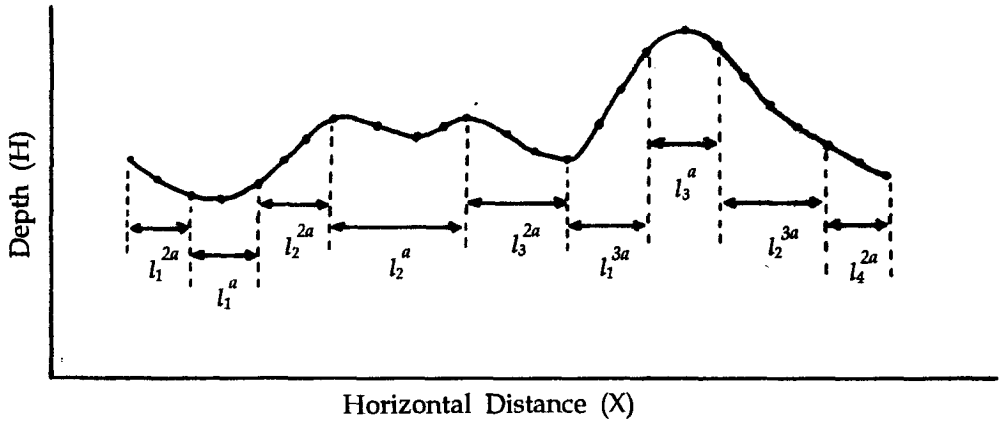
Q> Input interval degree(int)?

3 <- - - - - - - - - - - - - - - 3° 인 경우

한편, 평균경사도 및 경사구간별 평균수심은 구간별 경사도 계산시 사용된 수심측정지점에서의 경사도 및 수심값을 사용하여 계산하였다.

(2) 경사구간별 평균수평거리

경사구간별 평균수평거리는 Fig. 6에서 보여지는 것처럼 연속된 동일한 구간간격에 해당하는 지점간의 거리(l)를 모두 합하여 이를 빈도수로 나누어 평균값을 산출한 것이다. 이를 도식화하면 다음과 같다.



0° - a° 구간의 평균거리, $l^{a}=\left(\sum_{i=1}^{n} l_i^{a}\right)\frac{1}{n^{a}}$

a° - 2a° 구간의 평균거리, $l^{2a}=\left(\sum_{i=1}^{n} l_i^{2a}\right)\frac{1}{n^{2a}}$

2a° - 3a° 구간의 평균거리, $l^{3a}=\left(\sum_{i=1}^{n} l_i^{3a}\right)\frac{1}{n^{3a}}$

⋮ ⋮

이와같이 계산된 경사구간별 평균수평거리는 해저지형별로 이들의 크기를 짐작할 수 있다. 예를들면 해저평원의 경우 경사각이 일반적으로 3° 이하이므로 0° - 3° 구간의 평균수평거리는 분석하고자하는 측선상의 지역내에 분포하는 해저평원의 평균크기를 나타내어준다. 한편, 심해저에 분포하

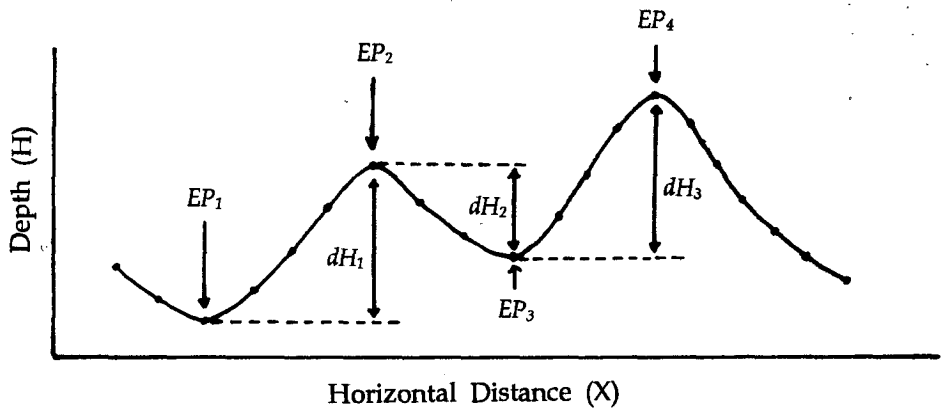
는 망간단괴와 관련된 해저지형자료의 경우 일반적으로 경제성이 있는 즉, 채광가능한 지역은 평균경사면이 6° 이내인 지역으로 제한되어 있으므로 이 같은 지역의 분포를 파악하기 위해서도 이상의 자료는 사용되어질 수 있다.

3. 해저면의 기복

해저면의 기복정도(relief)는 해저면의 경사정도와 함께 해저면의 특징을 규정하는 또다른 요인이다. 이와같은 해저면의 기복상태는 일정지역에 분포하는 해저면기복의 수직 및 수평방향으로의 발달정도 즉, 해저면기복의 표고차 및 이들의 발달정도로 특징지워진다. 따라서 본 프로그램에서는 이와같은 해저면기복을 특징지우는 두가지 요인을 아래와 같이 계량화하여 분석에 사용하고자 하였다.

(1) 해저면 기복의 평균표고차

연속적으로 측정된 수심자료에서 보여지는 해저면기복의 정점과 곡점은 수심측정지점 사이에서 계산되는 경사각의 변화에서 찾을 수 있다. 즉, 곡점의 경우 경사각이 ⊖ 값에서 ⊕ 값으로 변환되는 지점이며, 반대로 정점의 경우 경사각이 ⊕ 값에서 ⊖ 값으로 변하는 지점이다. 따라서 기복의 표고차는 이와같은 곡점과 정점에서 측정된 수심의 차이로서 얻을 수 있으며 이를 도식화 하면 다음과 같다.



$$dH_i = \left| H_{EPi+1} - H_{EPi} \right|$$

$$VR = \left(\sum_{i=1}^{n-1} dHi \right) \frac{1}{n-1}$$

여기서, dHi는 기복의 표고차이며, H_{EPi}는 첫번째 기복점의 수심, H_{EPi+1} 은 다음번 기복점에서의 수심, VR은 평균표고차, n은 기복점의 갯수를 나타낸다.

그러나 해저평원이라 할지라도 실제 해저면은 정도의 차이는 있으나 해저능지역에서처럼 굴곡을 보여주고 있다. 따라서 이같은 점을 고려하여 본 프로그램에서는 일정한 표고차 이상을 보여주는 기복만을 계산에 사용하도

록 작성하였으며 이값은 사용하고자 하는 목적에 따라 달리 아래와 같이 실행프로그램에서 설정할 수 있도록 하였다. 아래 예의 경우 10 m 이하의 표고차를 보여주는 요철면은 기복으로 간주하지 않으므로 계산에 포함시키지 않음을 뜻한다.

Q> Input the parameter of extreme points(ex 10 m)?

10 <- - - - - - - - - - - - - - - - 10 m인 경우

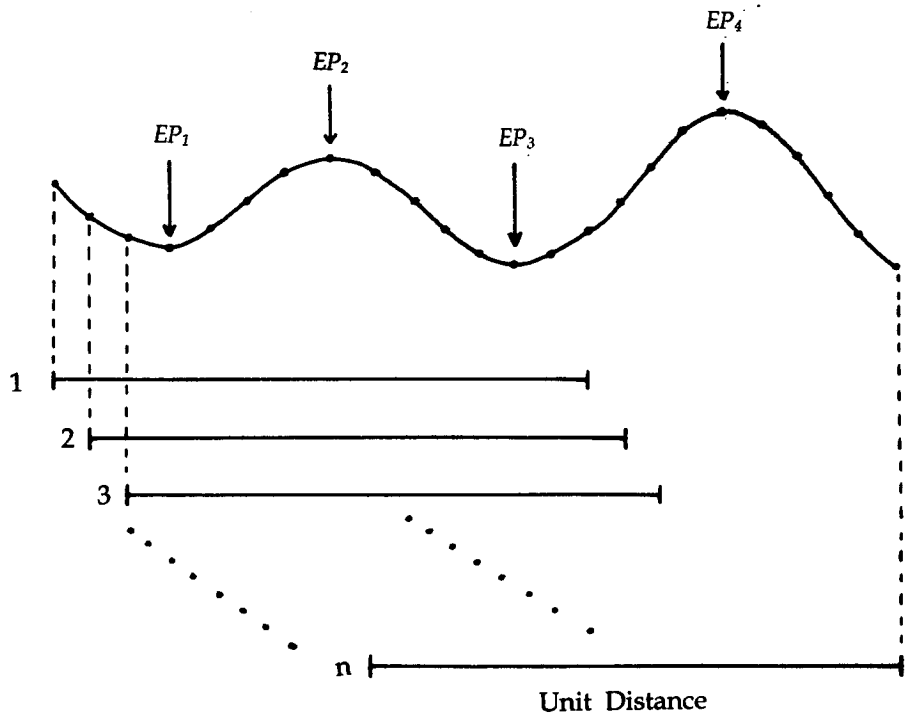
(2) 해저면 기복의 발달정도

해저면 기복의 발달정도는 일정한 단위거리내에 분포하는 기복점의 수로서 측정되어 질 수 있으며, 단위거리는 분석하고자 하는 지역의 크기 및 목적에 따라 달라질 수 있으므로 이 역시 실행프로그램에서 아래와 같이 달리 설정할 수 있도록 하였다.

Q> Input the parameter of horizontal difference(ex 10000 m)?

10000 <- - - - - - - - - - - - - - - - 10 km인 경우

한편, 기복의 발달정도를 계산하기 위하여 사용방법을 도식화하면 다음과 같다.



$$HR = \left(\sum_{i=0}^{n} N_i \right) \frac{1}{n}$$

여기서, HR은 단위거리내에 분포하는 기복점의 평균수, n은 계산에 사용된 단위거리의 갯수, Ni는 각각의 단위거리내에 분포하는 기복점의 수를 나타낸다.

V. 분석방법

1. 분석상의 제약

조사지역의 해저지형 분석에 본 프로그램을 적용하는데는 앞서 서언부분에서 언급한 바와 같이 몇가지 제약이 있다.

첫째, 본 프로그램은 먼저 조사지역의 지형을 해저면 프로파일 자료로서 지형상의 특징별로 구분(예: 해저평원, 해저산, 해저능 지역)한 후 각각의 구분된 지역별로 지형특성을 규정하는데만 사용될 수 있도록 작성되었다. 그러나, 지형에 따른 지역구분은 전체지역의 크기, 자료의 수, 분석목적에 따라 달라질 수 있으므로 본 프로그램을 적용하여 분석하기전에 이와같은 점을 충분히 고려하여 지역을 구분함으로서 분석의 효율성을 높일 수 있다.

둘째, 본 프로그램에 사용한 지형분석에는 동일한 방향의 측선에서 입수한 수심자료들을 사용해야 한다. 예를들면, 분석하고자 하는 지역내에 동서방향 및 남북방향의 측선에서 입수한 자료들이 함께있는 경우, 측선방향별로 동서방향, 남북방향의 자료들을 구분하여 분석해야 하며 이렇게 함으로서 분석의 일관성을 유지할 수 있다. 한편, 이와같이 분석하고자 하는 지역내에 두방향 이상의 측선자료가 있는 경우, 분석결과로서 해저지형의 방향성, 해저분지의 평균크기 등의 지형특성을 알 수 있다. 이점은 지역간 비교시에도 동일하게 적용된다.

2. 분석의 예

태평양 심해저면에 발달하는 해저평원 및 해저능지역에서 획득한 자료를 사용하여 얻은 결과(Table 2, Appendix 3)와 이들이 프로파일상에서 나타나는 지형모습(Fig. 4)과의 비교를 통해 실제모습에서의 차이점이 본 프로그램을 사용하여 얻은 지형자료값에 어떻게 반영되는지 보고자 하였다. 참고로 분석에 사용된 수심값은 1분간격으로 측정된 것으로 측심지점간의 간격은 약 250 m 이며, 분석조건으로 경사는 3도 간격, 최저기복값은 10 m, 기복점의 수는 10 km 를 기준하여 산출하였다.

Table 2에서 나타나는 바와같이 각기다른 3 지역에서 얻은 지형값으로부터 서로의 유사성 및 상이점을 유추할 수 있다. (A) 지역과 (C) 지역의 경우 평균경사각이 매우 낮고, 0 - 3 도의 경사면의 분포율이 매우 높은며, 아울러 평균수심이 비슷한점으로 보아 이지역은 해저평원에 해당하는 지역으로 분류할 수 있다.

그러나, 이와같은 유사성에도 불구하고 지형상의 차이점을 몇몇 다른항목에서 찾을 수 있다. 즉, 0 - 3 도의 경사면지역의 평균연장거리, 해저면 기복의 평균표고차, 기복점의 평균갯수에서 보여지는 차이점은 비록 이들지역이 해저평원지역임에도 불구하고 지형의 발달양상에 차이가 있음을 나타내어 준다. (A) 지역의 경우 (C) 지역에 비해 가파른 경사면을 갖는 지역이 분포하고 있으며, 이로 인하여 평균표고차의 값이 높게 나타나는 반면, 기복점의 수는 (C) 지역에 비해 적게 나타난다. 이같은 점으로 미루어 (A) 지역의 경우 대부분이 평탄한 지역이나 부분적으로 몇개의 소규모 해저능이 발달한 지역임을 알 수 있다.

Table 2. Results of analysis from three example areas. Out-put data print-outs of Area (A), (B), and (C) are listed in Appendix 3.

Area	(A)	(B)	(C)
Avg. slope angle	2.1 °	7.0 °	1.5 °
% of slopes with 0-3 ° Avg. length of slopes with 0-3° Avg. depth	81.0 % 1,795 m 4,950 m	42.4 % 549 m 4,725 m	79.7 % 774 m 4,944 m
% of slopes with 3-6° Avg. length of slopes with 3-6° Avg. depth	13.0 % 331 m 4,947 m	17.7 % 284 m 4,584 m	16.5 % 205 m 4,917 m
% of slopes with 6-9° Avg. length of slopes with 6-9° Avg. depth	4.2 % 282 m 4,913 m	10.1 % 312 m 4,461 m	3.8 % 179 m 4,984 m
% of slopes with 9-12° Avg. length of slopes with 9-12° Avg. depth	0.5 % 233 m 4,873 m	7.1 % 265 m 4,326 m	- - -
% of slopes with 12-15° Avg. length of slopes with 12-15° Avg. depth	0.5 % 260 m 4,912 m	9.1 % 295 m 4,635 m	- - -
% of slopes with 15-18° Avg. length of slopes with 15-18° Avg. depth	0.5 % 260 m 4,851 m	3.0 % 305 m 4,851 m	- - -
% of slopes with 18-21° Avg. length of slopes with 18-21° Avg. depth	- - -	4.0 % 216 m 4,738 m	- - -
% of slopes with 21-24° Avg. length of slopes with 21-24° Avg. depth	- - -	2.0 % 219 m 4,389 m	- - -
% of slopes with 24-27° Avg. length of slopes with 24-27° Avg. depth	0.5 % 248 m 4,866 m	1.5 % 263 m 4,058 m	- - -
% of slopes with 27-30° Avg. length of slopes with 27-30° Avg. depth	- - -	1.5 % 214 m 4,539 m	- - -
% of slopes with 30-33° Avg. length of slopes with 30-33° Avg. depth	- - -	- - -	- - -
% of slopes with 33-36° Avg. length of slopes with 33-36° Avg. depth	- - -	1.0 % 265 m 4,609 m	- - -
Avg. vertical difference	60 m	99 m	18 m
No. of hills and valleys in 10 km	5.47	11.14	7.42
Avg. depth	4,947 m	4,621 m	4,940 m
Length of lines	50.2 km	45.5 km	28.0 km

이와는 달리 (C) 지역은 가파른 경사면을 갖는 지역이 거의 분포하지 않고 있으며, 또한 평균표고차의 값이 낮은 사실로 보아 전반적으로 (A) 지역에 비해 평탄한 지역임을 알 수 있다. 그러나 (C) 지역에서 분포하는 기복점의 수가 (A) 지역에 비해 많은점으로 보아 (C) 지역에 발달하는 해저평원은 (A) 지역에 비해 단속적으로 분포하며, 평원과 평원사이는 소규모의 언덕과 같은 경사가 있는 지역으로 연결되어 있음을 알 수 있다. 이같은 점은 0 - 3도 경사지역의 평균거리에서도 알 수 있다. 즉, (C) 지역의 경우 평균거리가 약 774 m 인 반면, (A) 지역의 경우 1,795 m 인 점으로 볼때 (C) 지역에서 분포하는 분지의 크기가 (A) 지역에 비해 훨씬 작다는 사실을 알 수 있다.

한편, 각 경사구간별 평균수심값을 보면 일반적으로 (A), (B) 지역 모두 평균수심값과 큰 차이를 보여 주지 않으므로 이는 두지역 모두 해저평원지역임을 다시한번 확인해 준다. 그러나 (A) 지역의 경우 경사도가 낮은구간이 높은구간에 비해 수심이 깊으며 또한 평균수심값과 거의 비슷한 점은 앞서 언급한 바와같이 소규모의 해저능이 부분적으로 분포하기 때문이라는 해석과 일치하는 결과이다.

(B) 지역의 경우는 위에서 언급한 (A), (C) 지역과는 완전히 다른 지형상태를 보여주고 있다. 즉, 평균경사도, 경사구간별 분포, 평균표고차, 기복점의 수 등 거의 모든 부분에서 (A), (C) 지역에 비해 높은 값을 보여주고 있으며, 이는 (B) 지역이 매우 굴곡이 심한 해저면 즉, 해저능지역임을 나타내어 준다. 한편, (B) 지역의 경사구간별 평균수심값은 전체평균수심값과 비교시 차이가 매우 크며 이런 결과는 앞의 해석과 일치하고 있다.

이상의 결과를 종합하면, (A) 지역은 부분적으로 소규모의 해저능이 분포하는 해저평원지역이며, (C) 지역은 해저능이 거의 전무한 해저평원지역인 반면, (B) 지역은 전반적으로 해저능이 발달한 지역으로 해석된다. 이같은 분석결과는 Fig. 4의 해저면 프로파일과 비교해볼때 서로 잘 일치하고 있다.

이와같이 동일한 분석조건(경사면 분류간격, 최적기복값 등)에서 실시한 분석결과로서도 지형분류가 가능하나, 동일한 범주에 속하는 지형지역간의 비교시 지형특성에 따라 앞서의 분석조건을 달리 설정함으로서 새로운 상이점을 찾을 수 있다.

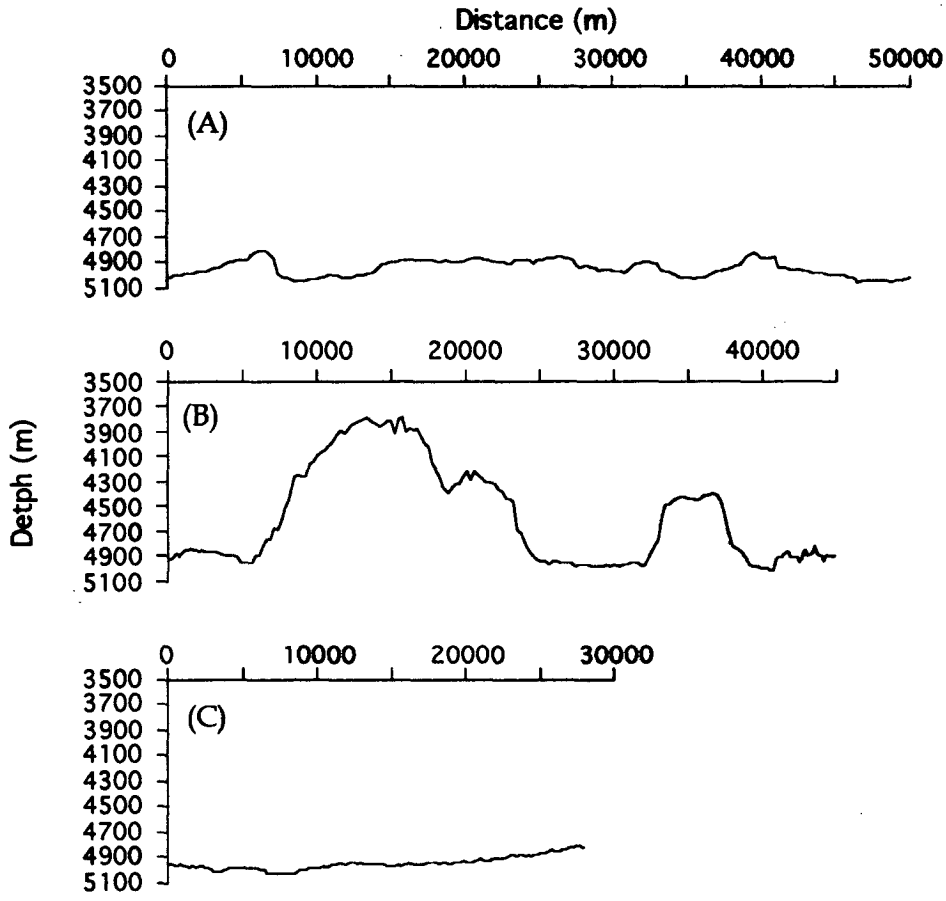


Fig. 4. Bathymetric profiles of three example areas; (A) abyssal plain with hills, (B) abyssal hills, and (C) abyssal plain. Results of analysis of Area (A), (B), and (C) are in Table 2.

VI. 결 론

본 프로그램을 사용하여 해저지형을 분석한 결과, 지형특성을 구성하는 요인으로 설정한 1) 평균수심, 2) 해저면의 평균경사도, 3) 해저면의 경사각도별 분포, 4) 경사구간별 평균수심, 5) 경사구간별 평균수평거리, 6) 해저면 기복의 평균표고차, 7) 해저면 기복점의 수 등이 해저지형을 정량화하는데 적절히 사용되어 질 수 있음이 밝혀졌다.

그러나 본 지형분석 프로그램은 적용시 많은 제약을 가지고 있다. 특히 분석전에 지형에 따른 대략의 지역구분을 한후 각 지역별로 분석을 실시해야 하며, 분석하고자 하는 지역의 크기 및 사용하는 수심자료의 분포에 따라 분석조건을 달리해야한다. 한편, 프로그램을 통해 산출된 지형요인값은 각각으로는 별의미가 없으며, 모든값들을 종합하여 상호관계를 비교함으로서 정확한 분석결과를 얻을 수 있다. 더구나 최종적으로 산출되는 지형요인값들은 분석조건에 따라 값이 달라질 수 있으므로 분석을 실시하기 전에 이점을 고려하여 분석조건을 설정해야하며, 아울러 분석조건을 달리 하여 얻을 결과를 상호 비교함으로서 보다 정확한 분석을 실시할 수 있다.

따라서 대상지역에 대하여 사전에 충분한 검토작업을 거친후 분석을 실행함으로서 본 프로그램이 가지고 있는 취약점을 보완할 수 있으며, 아울러 보다 정확한 분석결과를 얻을 수 있다.

참 고 문 헌

유 환 영, 정 성 민, 1993, Turbo C를 이용한 자료구조론. 상조사: 1-496.

전 웅 섭, 1992, Turbo C 2.0: user's 편. 크라운출판사: 1-353.

황 희 웅, 1992, Turbo C 2.0 Library Manual. 교학사: 1-1077.

Davis, J. C., 1973, Statistics and Data Analysis in Geology. Wiley International Edition. 298-411.

Kernighan, B. W. and D. M. Ritchie, 1978, The C Programming Language. Prentice-Hall: 1-272.

Merriam, D. F. and P. H. A. Sneath, 1966, Quantitative comparision of contour maps. J. Geophys. Res. 71: 1105-1115.

SeaBeam Instrument Inc., 1992, SeaBeam 2000 Bathymetric Swath Survey System Operation Manual (Prepared for KORDI). 1-92.

Simrad Subsea, 1992, SIMRAD EA500 Hydrographic Echo Sounder Instruction Manual. 1.1-1.14, 2.1-2.89.

Tyce, R. C., 1987, Deep Seafloor Mapping System - A Review. MTS Journal 20(4): 4-16.

Appendix 1. Program for bathymetric data filtering

```
/* Data filtering program for topographical analysis data */
/* 1994. 3. 10. by Hos. Kang. */

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

struct seabeam {
                char day[20];
                char lat[15];
                char longi[15];
                int depth;
};

struct seabeam lib;
struct seabeam *queue;

int rear, front, n, flag=0;

int main(argc,argv)
int argc;
char *argv[];
{
 void add_queue();
 struct seabeam delete_queue();
 struct seabeam del;
 FILE *in,*out;
 char waste[100];
 int *maxmin, head, mean;
 int i, j, temp, incre=0;

 front = rear = -1;

 if ( argc != 3 ) {
        printf( "Usage: filter Inputfile Outfile" );
        exit(1);
 }

 printf("Input the parameter of filtering?(odd)\n");
 scanf("%d", &n);
 maxmin = (int *)malloc( sizeof(int)*n );
 queue = (struct seabeam *)malloc( sizeof(struct seabeam)*n);

        out = fopen (argv[2], "w");
        in  = fopen (argv[1], "r");

 for ( i= 0; i < n; i++ ) {
```

```
                fscanf ( in, "%s %s %s %d", lib.day, lib.lat, lib.longi, &lib.depth);
                fgets(waste, 100, in);
                add_queue( &lib );
 }

while(fscanf( in, "%s %s %s %d", lib.day, lib.lat, lib.longi, &lib.depth) != EOF)
{
        fgets(waste, 100, in);
        for ( i=0; i<n; i++)
                maxmin[i]=queue[i].depth;

        for( i=0; i<n-1; i++) {
                for ( j=i+1; j<n; j++) {
                                if ( maxmin[i] < maxmin[j] ) {
                                                temp = maxmin[i];
                                                maxmin[i] = maxmin[j];
                                                maxmin[j] = temp;
                                }
                }
        }

        mean = maxmin[n/2];
        head = ( n/2 + incre ) % n;
        fprintf(out, "%s %s %s %d\n", queue[head].day, queue[head].lat, queue[head].longi, mean);
        incre++;
        del = delete_queue();
        add_queue( &lib );

 }/* end while */

        fclose(in);
        fclose(out);
        free(maxmin);
 return(0);
}

void add_queue( data )
struct seabeam *data;
{
 if( flag == 1 ) {
                printf("queue_full\n");
                exit(0);
 }
 rear=( rear + 1 ) % n;
 if ( front == rear ) flag = 1;
 strcpy(queue[rear].day, data->day);
 strcpy(queue[rear].lat, data->lat);
```

```
 strcpy(queue[rear].longi, data->longi);
 queue[rear].depth = data->depth;
}

struct seabeam delete_queue( )
{
 struct seabeam data;

 if( front == rear  && flag == 0 ) {
            printf("queue_empty \n");
            exit(0);
 }

 if( front == rear ) flag = 0;
 front = ( front + 1 ) % n;
 data = queue[front];
 return(data);
}
```

Appendix 2. Program for topographic analysis

```
/* Topographical Analysis program  */
/* 1994. 3. 12.  by Hos. Kang.     */


#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#define  MAXPART   90

typedef  struct {
     double x_dis;
     double x_depth;
     double o_depth;
     int x_mount;
} KODOS;

KODOS *list;

int x_count, x_first, deg_cnt[MAXPART], deg_part[MAXPART];
int p_sign, pb_sign, bong, b_first, p_first, verfirst, vercount;
float para;
long MAXDIS;
double deg_len[MAXPART], deg_dep[MAXPART], bangle, bver, finalver;

double Absf (value)
double value;
{
 if ( value < 0. ) value = -value;
 return (value);
}

void Merell (y_refer, x_refer, y_lat, x_long, y_dis, x_dis)
double y_refer, x_refer, y_lat, x_long, *y_dis, *x_dis;
{
 double a, phi, e, y_re, x_re, y_la, x_lo;
 double rad, cen, y_disre, y_disla;
 void Degtorad();

 a = 6378206.4;        /* in m */
 phi = 3.141592654;
 e = 0.0822719;
 rad = 0.017453292;
 cen = 10.;            /* in degree, the latitute at which the projection */
                       /* is true */

 y_refer  += 0.1e-10;   /* for truncation */
 x_refer  -= 0.1e-10;
```

```
y_lat   += 0.1e-10;
x_long  -= 0.1e-10;
Degtorad (y_refer, &y_re);      /* min to degree and degree to radian */
Degtorad (x_refer, &x_re);
Degtorad (y_lat, &y_la);
Degtorad (x_long, &x_lo);

*x_dis = a*(x_lo-x_re)*cos(cen*rad);      /* in m */

                                                    /* distance from reference */
y_disre = a*log(tan(phi/4+y_re/2)* \
                                pow( (1-e*sin(y_re))/(1+e*sin(y_re)), e/2. ))*cos(cen*rad);

                                                    /* distance from point */
y_disla = a*log(tan(phi/4+y_la/2)* \
                                pow( (1-e*sin(y_la))/(1+e*sin(y_la)), e/2. ))*cos(cen*rad);

*y_dis = y_disla-y_disre;             /* in m */
}

void Degtorad (deg, rad)
double deg, *rad;
{
 double min, radian;

 radian = 0.017453293;
 min = deg-(int)deg;
 *rad = ((int)deg+min*100/60) * radian;
}

void Calangle (cal_y, cal_x, bdepth, cal_ag)
double cal_y, cal_x, bdepth, *cal_ag;
{
 double angle = 57.29577951;

 if (cal_y == 0 )
   *cal_ag = 0.;
 else
   *cal_ag = atan2 (cal_y, cal_x) * angle;

 list[x_count+1].x_dis = cal_x;

 if (bangle != 0. || *cal_ag != 0.) {
 if ( bangle * (*cal_ag) <= 0. && x_first != 1) {
    list[x_count].x_mount = 1;
    list[x_count].x_depth = bdepth;

  if ( b_first != 1 && Absf(list[x_count].x_depth-list[bong].x_depth)
```

```
                                < para ) {
    list[x_count].x_mount = 0;
    list[x_count].x_depth = 0.;
  } else
    bong = x_count;

  if (b_first) {
    b_first = 0;
    bong = x_count;
  }

}  else  {
          list[x_count].x_mount = 0;
          list[x_count].x_depth = 0.;
    }
}  /* end if */

  /* calculate vertical difference */
if ( list[x_count].x_mount == 1 ) {
          if ( verfirst == 1 ) {
                                bver = list[x_count].x_depth;
                                verfirst = 0;
          } else {
                                finalver += Absf( bver-list[x_count].x_depth );
                                bver = list[x_count].x_depth;
                                vercount++;
                    }
}
x_first = 0;
x_count++;
bangle = *cal_ag;

}

void Hori_diff (num, totcount, tothori, totmount, endsign)
int num, totcount, *tothori, *totmount, *endsign;
{
 int i, cmount = 0;
 double h_total = 0.;

 for (i = num; i < totcount; i++) {
    h_total += list[i+1].x_dis;
                    if (list[i].x_mount == 1)   cmount++;
                    if (h_total >= MAXDIS ) {
        *totmount += cmount;
        *tothori += 1;
        return;
                    } /* end if */
```

```
 } /* end for */
 if ( h_total < MAXDIS)  *endsign=1;
}

void Total_angle (cal_ag, totangle)
double cal_ag, *totangle;
{
 *totangle += Absf (cal_ag);
}

void Part_degree (cal_ag, cal_x, inter, factor, bdepth)
int inter, factor;
double cal_ag, cal_x, bdepth;
{
 int i, j;

 for ( i = 0, j = 0; j < factor; i += inter, j++) {
    if( Absf(cal_ag) >= i && Absf(cal_ag) < inter+i ) {
       deg_cnt[j]++;          /* each degree count */
                               deg_dep[j] += bdepth;      /* each depth total */
                               deg_len[j] += cal_x;     /* each degree length */
       p_sign = j;
       if ( p_first == 1 ) pb_sign = p_sign;
       if ( p_sign != pb_sign )
       deg_part[pb_sign]++;
       pb_sign = j;
       p_first = 0;
       return;
    } /* end if */
 } /* end for */
}

void Total_length (cal_x, totlen)
double cal_x, *totlen;
{
 *totlen += cal_x;          /* total line depth */
}

void Caldis_y (bdepth, depth, cal_y)
double bdepth, depth, *cal_y;
{
                    /* calculate depth length */
 *cal_y = Absf (depth) - Absf (bdepth);
}

void Caldis_x (blat, blongi, lat, longi, cal_x)
double blat, blongi, lat, longi, *cal_x;
{
```

```
double merc_y, merc_x;

Merell (blat, blongi, lat, longi, &merc_y, &merc_x);
                    /* calculate x-distance */
*cal_x = sqrt ( pow (merc_y, 2.) + pow (merc_x, 2.) );
}

void Tot_depth (depth, totdepth)
double depth, *totdepth;
{
*totdepth += Absf (depth);
}

int Filecnt (infile)
char *infile;
{
        FILE *in;
        int count = 0;
        char string[200];

if ( (in = fopen (infile,"r") )  == NULL ) {
                        printf(" %s input file not exist!\n", infile);
                        exit(1);
        }

while ( fgets (string, 200, in) != NULL )
                                count+=1;
        fclose (in);
        return (count);
}

main ()
{
FILE *in, *info, *out, *out2;
static double f_inter[MAXPART], f_depth[MAXPART], f_avlede[MAXPART];
static int f_avnum[MAXPART];
char infofile[15],  infile[15], outfile[15], outs[15];
char time[20], ans, waste;
double x_refer, y_refer, x_long, y_lat, y_dis, x_dis;
double cal_x, cal_y, totdepth, blat, blongi, bdepth;
double totlen, cal_ag, slodepth, totver, totangle;
double f_verdif = 0., f_hordif = 0., f_adepth = 0., f_totlen = 0.;
double f_avang = 0., lat, longi, depth;
int i, j, count, inter, factor, endsign, f_count, number;
long tothori, totmount;

printf ( "Q>Input infofile name?\n" );
scanf ( "%s", infofile );
```

```
info = fopen (infofile, "r" );

printf ( "Q>Input interval degree(int)?\n" );
scanf ( "%d", &inter );
factor = 90 / inter;
printf ( "Q>Input the parameter of extreme points(ex 10 m)?\n" );
scanf ( "%f", &para );
printf ( "Q>Input the parameter of horizontal difference(ex 10000 m)?\n" );
scanf ( "%ld", &MAXDIS);

printf ( "Q>Input outfile name?\n" );
scanf ( "%s", outs );

if ( (out2 = fopen (outs,"w") )  == NULL ) {
                printf(" %s cannot open!!!\n", outs);
                exit(1);
}
fprintf(out2,"##############################################################\n");
fprintf (out2,"## Line files:");
f_count = 0;

while ( fscanf (info, "%s", infile) != EOF ) {
  f_count++;
  number = Filecnt (infile);
  list = ( (KODOS *) malloc (sizeof(KODOS)*(number+1)) );

  if ( ( in = fopen (infile,"r"))  == NULL ) {
     printf (" %s input file not exist!\n", infile);
     exit (1);
  }
        /* parameter initiation */
  totdepth = 0.;      totlen = 0.;      totangle = 0.;
  bangle = 1.;        tothori = 0;      finalver = 0.;
  endsign = 0;        totmount = 0;     x_count = 0;
  x_first = 1;        p_first = 1;      b_first = 1;
  bong = 0;           verfirst = 1;     vercount = 0;


  for ( i = 0; i < number; i++)  {
     list[i].x_dis    = 0.;
     list[i].x_depth = 0.;
     list[i].o_depth = 0.;
     list[i].x_mount = 0;
  }

  for ( i = 0; i < MAXPART; i++) {
     deg_cnt[i]  = 0;
     deg_len[i]  = 0.;
```

```
                deg_dep[i]  = 0.;
                deg_part[i] = 0;
}

        /* read first data and delete error depth */
for (;;) {
                fscanf (in,"%s %lf %lf %lf",time, &lat, &longi, &depth);
   if ( depth != 0. )  break;
}

Tot_depth (depth, &totdepth);
blat = lat;          blongi = longi;          bdepth = depth;
count = 1;           list[0].o_depth = depth;

              while ( fscanf (in,"%s %lf %lf %lf", time, &lat,

                        &longi, &depth) != EOF )
              {
                        list[count].o_depth = depth;
                        count += 1;

                        Tot_depth (depth, &totdepth);
                        Caldis_x (blat, blongi, lat, longi, &cal_x);
                        Total_length (cal_x, &totlen);
                        Caldis_y (bdepth, depth, &cal_y);
                        Calangle (cal_y, cal_x, bdepth, &cal_ag);
                        Total_angle (cal_ag, &totangle);
                        Part_degree (cal_ag, cal_x, inter, factor, bdepth);

                        blat  = lat;
                        blongi = longi;
                        bdepth = depth;

              }/* end sub while */

deg_part[pb_sign]++;       /* for last data  */

for (i = 0; i < count; i++) {
                Hori_diff (i, count, &tothori, &totmount, &endsign);
   if ( endsign == 1 )  break;
}

        for (i = 0, j = 0; j < factor; i += inter, j++) {
                f_inter[j] += (double)deg_cnt[j]/(count-1);
                if ( deg_cnt[j] == 0 )
                                f_depth[j] = 0.;
                else
                                f_depth[j] +=  deg_dep[j]/(double)deg_cnt[j];
```

```
        }

    for (i = 0, j = 0; j < factor; i += inter, j++) {
      if( deg_part[j] != 0 ) {
        f_avlede[j] += deg_len[j] / deg_part[j];
        f_avnum[j] += 1;
      }
    }

    f_avang += totangle / (count-1);
    f_verdif += finalver / vercount;
    f_hordif += (double)totmount / tothori;
    f_adepth += totdepth / count;
    f_totlen += totlen;

    fprintf ( out2, " %s ", infile );
    if ( f_count%4 == 0 )  fprintf(out2,"\n\t\t");
    fclose (in);
    free (list);
  } /* end main while */

  fprintf (out2, "\n\n\nInterval degree      %    average depth(m)\n" );
fprintf(out2,"-----------------------------------------------------------\n");
  for (i = 0, j = 0; j < factor; i += inter, j++)
                    fprintf(out2," %2d --- %2d    %10.3lf   %10.2lf \n", i, inter+i,
                                                f_inter[j]/f_count*100., f_depth[j]/f_count);
  fprintf (out2,"\n### Average angle of slope degree %lf\n\n",
            f_avang/f_count);
  for (i = 0, j = 0; j < factor; i += inter, j++) {
     if  ( f_avnum[j] != 0 )
     fprintf (out2, "## Av. length of bottom part with ang. from\
 %2d to %2d degree %lf m\n", i, inter+i, f_avlede[j] / f_avnum[j]);
  }

  if ( f_verdif == 0.0 )
    fprintf(out2,"\n### Average veritical difference : not find slope\n");
  else
    fprintf(out2,"\n### Average veritical difference %lf m\n",
                                                f_verdif / f_count);

  fprintf (out2, "\n### Average horizontal difference %lf extr/10km\n",
                                                f_hordif / f_count);
  fprintf (out2, "### Average depth %lf m\n", f_adepth / f_count);
  fprintf (out2, "### Length of lines %lf m\n", f_totlen);
fprintf(out2,"##############################################################\n");
  fclose (info);
} /* end main */
```

Appendix 3. Out-put data print-outs of example area (A), (B), and (C)

1. Area (A)

```
##############################################################
## Line files: rline.024

Interval degree    %
-------------------------------------------------------------
 0 --- 3   81.018519 %
 3 --- 6   12.962963 %
 6 --- 9   4.166667 %
 9 --- 12   0.462963 %
12 --- 15   0.462963 %
15 --- 18   0.462963 %
18 --- 21   0.000000 %
21 --- 24   0.000000 %
24 --- 27   0.462963 %
27 --- 30   0.000000 %
30 --- 33   0.000000 %
33 --- 36   0.000000 %
36 --- 39   0.000000 %
39 --- 42   0.000000 %
42 --- 45   0.000000 %
45 --- 48   0.000000 %
48 --- 51   0.000000 %
51 --- 54   0.000000 %
54 --- 57   0.000000 %
57 --- 60   0.000000 %
60 --- 63   0.000000 %
63 --- 66   0.000000 %
66 --- 69   0.000000 %
69 --- 72   0.000000 %
72 --- 75   0.000000 %
75 --- 78   0.000000 %
78 --- 81   0.000000 %
81 --- 84   0.000000 %
84 --- 87   0.000000 %
87 --- 90   0.000000 %

### Average angle of slope degree 2.104869

## Av. length of bottom part with ang. from  0 to  3 degree 1794.699669 m
## Av. length of bottom part with ang. from  3 to  6 degree 330.886036 m
## Av. length of bottom part with ang. from  6 to  9 degree 281.863426 m
## Av. length of bottom part with ang. from  9 to 12 degree 232.667961 m
## Av. length of bottom part with ang. from 12 to 15 degree 259.565906 m
## Av. length of bottom part with ang. from 15 to 18 degree 259.627444 m
## Av. length of bottom part with ang. from 24 to 27 degree 247.700570 m

### Average veritical difference 60.344828 m

### Average horizontal difference 5.470046 extr/10km
### Average depth 4946.875576 m
### Length of lines 51150.282406 m
##############################################################
```

2. Area (B)

```
##############################################################
## Line files: rline.038

Interval degree    %
-------------------------------------------------------------
 0 --- 3   42.424242 %
 3 --- 6   17.676768 %
 6 --- 9   10.101010 %
 9 --- 12   7.070707 %
12 --- 15   9.090909 %
15 --- 18   3.030303 %
18 --- 21   4.040404 %
21 --- 24   2.020202 %
24 --- 27   1.515152 %
27 --- 30   1.515152 %
30 --- 33   0.000000 %
33 --- 36   1.010101 %
36 --- 39   0.000000 %
39 --- 42   0.505051 %
42 --- 45   0.000000 %
45 --- 48   0.000000 %
48 --- 51   0.000000 %
51 --- 54   0.000000 %
54 --- 57   0.000000 %
57 --- 60   0.000000 %
60 --- 63   0.000000 %
63 --- 66   0.000000 %
66 --- 69   0.000000 %
69 --- 72   0.000000 %
72 --- 75   0.000000 %
75 --- 78   0.000000 %
78 --- 81   0.000000 %
81 --- 84   0.000000 %
84 --- 87   0.000000 %
87 --- 90   0.000000 %

### Average angle of slope degree 7.007518

## Av. length of bottom part with ang. from  0 to  3 degree 548.829396 m
## Av. length of bottom part with ang. from  3 to  6 degree 284.164781 m
## Av. length of bottom part with ang. from  6 to  9 degree 311.945134 m
## Av. length of bottom part with ang. from  9 to 12 degree 265.325713 m
## Av. length of bottom part with ang. from 12 to 15 degree 295.493844 m
## Av. length of bottom part with ang. from 15 to 18 degree 305.226620 m
## Av. length of bottom part with ang. from 18 to 21 degree 215.987117 m
## Av. length of bottom part with ang. from 21 to 24 degree 218.924655 m
## Av. length of bottom part with ang. from 24 to 27 degree 262.719230 m
## Av. length of bottom part with ang. from 27 to 30 degree 214.020731 m
## Av. length of bottom part with ang. from 33 to 36 degree 265.209928 m
## Av. length of bottom part with ang. from 39 to 42 degree 270.524771 m

### Average veritical difference 98.807018 m

### Average horizontal difference 11.139860 extr/10km
### Average depth 4620.502513 m
### Length of lines 45513.753632 m
##############################################################
```

3. Area (C)

```
##############################################################
## Line files: rline.032

Interval degree    %
-------------------------------------------------------------
 0 --- 3   79.746835 %
 3 --- 6   16.455696 %
 6 --- 9   3.797468 %
 9 --- 12   0.000000 %
12 --- 15   0.000000 %
15 --- 18   0.000000 %
18 --- 21   0.000000 %
21 --- 24   0.000000 %
24 --- 27   0.000000 %
27 --- 30   0.000000 %
30 --- 33   0.000000 %
33 --- 36   0.000000 %
36 --- 39   0.000000 %
39 --- 42   0.000000 %
42 --- 45   0.000000 %
45 --- 48   0.000000 %
48 --- 51   0.000000 %
51 --- 54   0.000000 %
54 --- 57   0.000000 %
57 --- 60   0.000000 %
60 --- 63   0.000000 %
63 --- 66   0.000000 %
66 --- 69   0.000000 %
69 --- 72   0.000000 %
72 --- 75   0.000000 %
75 --- 78   0.000000 %
78 --- 81   0.000000 %
81 --- 84   0.000000 %
84 --- 87   0.000000 %
87 --- 90   0.000000 %

### Average angle of slope degree 1.462800

## Av. length of bottom part with ang. from  0 to  3 degree 774.374787 m
## Av. length of bottom part with ang. from  3 to  6 degree 205.174468 m
## Av. length of bottom part with ang. from  6 to  9 degree 179.439481 m

### Average veritical difference 18.464286 m

### Average horizontal difference 7.421384 extr/10km
### Average depth 4940.459119 m
### Length of lines 28047.344000 m
##############################################################
```