

## 타원체상에서 경계획선을 위한 중간점계산의 정확도 Accuracy of Mid Point Computation for Boundary Delimitation on Ellipsoid

김병국\* · 이종기\*\* · 김정기\*\*\*

Kim, Byung-Guk · Lee, Jong-Ki · Kim, Jeong-Ki

### 요 旨

나라사이의 경계를 결정하기 위한 일반적인 방법은 등거리원칙이다. 등거리원칙은 각각의 기준선 또는 기준점의 경계에서 일정한 거리까지를 경계로 정하는 방법이다. 본 연구에서는 등거리 원칙을 기본으로 하는 알고리즘으로 폭넓게 사용되고 있는 Two-Point 알고리즘과 Three-Point 알고리즘을 정립한 후, 알고리즘을 적용한 결과와 오차를 확인할 수 있는 경계 획선 프로그램을 개발하였다. 이 프로그램은 인공지물이나 지형지물이 없어 경계를 결정하기 어려운 해양경계 획선시 특히 유용하게 사용될 수 있다. 그 중간점 계산결과 타원체상에서 등거리원칙에 기초한 알고리즘을 적용하면, 지도 투영에 의한 어떤 왜곡도 없이 적용할 수 있고 평면이나 구에서 적용한 것 보다 오차가 적었다. 경계 획선 프로그램 개발을 통하여 도면을 이용한 수작업에 의한 여러 오차를 제거 할 수 있고, 해양경계협상시 신속히 대처할 수 있다. 또한, 기준선의 기준점 위치에러는 계산된 중간선의 중간점으로 전파되는 것을 확인할 수 있었고 그 오차를 확인하여 적합한 기준면을 설정할 수 있었다.

### ABSTRACT

The general rule of boundary delimitation is a the principle of equidistant. The principle of equidistant is a method that determine boundary delimitation from fixed distant of baseline or basepoint. In this paper, study Two-Point Algorithm and Three-Point Algorithm that are widely used, and developed the Boundary Delimitation Program to verify the result and error. This program is specially useful for maritime boundary delimitation problem because there is no artificial and natural object in sea to determine boundary. As a result, The mid-points computed on Ellipsoid have small error rather than mid-points on plane or sphere without any distortion by map projection. Through developing boundary delimitation program, can eliminate the various manipulation errors using paper map, and quickly cope with maritime boundary delimitation negotiation. Also, verify that the error of basepoint in baseline is propagate the mid-point in mid-line, and determine suitable reference plane.

### 1. 서 론

경계획선은 1982년부터에서 시작된, 해양 국가들 사이의 협상 중 매우 중요한 쟁점이었다. 각각의 협상의 문제는 공평의 법적 원칙이 엄격하게 적용되는 국제적 경계를 결정하는 것이다. 하나의 공식이 각국의 경제적, 지형적 그리고 주위요소와의 전략적 경우에 모두 적용될 수 없기 때문에 어떠한 특별한 방법이 정해지지 않았다.

그러나 균등한 경계를 얻기 위한 몇 가지 지침이 1953년부터 International Law Commission(ILC)에 의해 제안되었다. 이 지침들 중에 하나는 등거리 원칙(the principle of equidistant)에 의거한 일반적인 규칙을 사용한다. 이 원칙을 응용한 방법은 경계 설정 협상을 위한 참고로 사용될 수 있는 충분한 가치가 있다. 이 등거리 원칙은 경계상의 모든 점들은 해안의 한 쪽에서 측정된 기준선(baseline)상의 최근접점에서 항상 등거리에 있어야 한다. 이 원칙을 구현하기 위한 여러 가지 기하학적 방법이 적용되어왔다.

이 논문에서는 등거리 원칙을 기본으로 하는 알고리즘

\*인하대학교 지리정보공학과 부교수

\*\*인하대학교 지리정보공학과 석사과정

\*\*\*국립해양조사원 측량과

중에 광범위하게 사용되고 있는 Two-Point 알고리즘과 Three-Point 알고리즘을 정립한 후, 이 알고리즘 결과를 확인할 수 있는 경계 획선 프로그램을 개발하였다. 이 프로그램은 인공지물이나 지형지물이 없어 경계를 결정하기 어려운 해양경계 획선시 특히 유용하게 사용될 수 있다. 이 프로그램을 이용하여 중간점을 계산한 후 오차 예측을 하였다. 또한, 3개의 기준면(plane, sphere and ellipsoid)에 따라 계산된 좌표의 오차예측을 통하여 적합한 알고리즘 및 기준면을 제안하였다.

## 2. 연구개요

### 2.1 연구배경 및 내용

단일 협정으로 모든 나라들이 적용할 수 있도록 대양의 경계를 결정하여 사용을 규제하기 위한 노력이 진행되어 왔다. 수년의 노력을 통하여 마침내 1982년 United Nations Convention on the Law of the Sea(이하 「UNCLOS」라 한다.)를 채택하게 되었다. UNCLOS는 처음으로 해양경계 획정, 어족자원관리, 대륙붕 등에 대한 보편적인 법적 제도를 제공한 것이다.

각각의 협상자들의 문제는 공평의 법적 원칙을 엄격하게 정의된 기하학적인 것을 국제적 경계(international boundary)로 변환하는 것이다. UNCLOS는 이 문제를 풀기 위하여 어떠한 특별한 방법을 정하지 않았다. 하나의 공식이 각각의 경제적, 지형적 그리고 주위요소와의 전략적 경우에 모두 적용될 수 없기 때문이다. 그러나 균등한 경계를 얻기 위한 몇 가지 지침 중에 하나는 등거리 원칙에 의거한 일반적인 규칙을 사용하는 것이다.

이 등거리 원칙(the principle of equidistant)은 경계상의 모든 점들은 해안의 한 쪽에서 측정된 기준선(baselines)상의 최근접점에서 항상 등거리에 있어야 한다. 이 원칙을 구현하기 위한 여러 가지 기하학적 방법이 적용되어왔다.

그러나 현존하는 방법들은 해도나 지도 위에서 유클리드 기하학을 사용하여 그래픽적으로 보여져 왔다. 이 방법들의 몇 가지 단점을 가지고 있다. 특별히 중요한 점은 임의의 투영법의 선택으로 인한 왜곡과 그래픽 수작업을 통한 정확도 상실로 인한 결정된 경계의 영향을 미치는 점이다. 또한, 경계획정시 사용되는 기하학적 획선 방식은 수작업에 의한 오차를 포함하고 시간의 많이 소요된다.

위의 문제점을 해결하고 각국에 주장에 따라 우리나라의 이해관계를 신속히 파악하기 위해 해양경계 획선 프로그램을 개발하였다. 기존의 연구를 통하여 개발된 해양경계 획선 프로그램이 지구를 구면으로 가정한 프로그램이었던 것에 비해, 이 연구에서는 중간선을 결정하는 거리에 따라 기준면을 달리하여 결과를 알아볼 수 있도록 3개의 기준면(plane, sphere and ellipsoid)에 따른 계산결과를 얻을 수 있도록 개발하였다.

또한, 중간점을 계산하기 위한 기준점의 좌표는 측량을 통해 좌표를 취득하는 방식이므로 오차를 포함하고 있다. 이 오차는 중간점을 구하는 과정을 통하여 계산된 중간점으로 오차가 전파하게 된다. 계산된 중간점의 오차를 파악할 수 있게 하였다.

계산된 중간점 결과와 중간점에서 각 점의 거리를 표준타원체상에서의 해양경계 획선 소프트웨어인 Delmar 1.0의 결과와 비교하여 그 타당성 및 정확도를 검증해보았고 그 결과 개발한 프로그램의 결과가 더욱 정확하였다.

## 3. 연구내용

### 3.1 직접문제 및 역문제

고려하고자 하는 평면은 측지 타원체상의 평면이다. 주요한 이점은 어떤 길이나 방위에서도 왜곡 없이 전세계의 경계를 설정하는데 쓰일 수 있다.

측지 데이터들을 사용하는 또 하나의 이점은 해도와는 달리 좀더 정확한 경계를 결정할 수 있다는 것이다. 이것은 기준선 점들을 지도나 해도 위에서 확인하는 것보다 측지 측량을 통하여 좀더 높은 정확도를 결정하는 할 수 있기 때문이다.

여기서 제시된 방법은 타원체상에서 측정된 커브로 정의되는 측지선을 사용한다. 이 측정된 측지 거리의 특성은 최소값이다. 즉, 어떠한 기하학적 모호성을 피하기 위하여 다음과 같이 정의한다.

$$S = \min \int_C dS \quad (3-1)$$

우리가 고려하는 타원체상의 직접문제(Direct Problem) 그리고 역문제(Inverse Problem), 즉, 측지선에 관련된 두 가지 전통적인 문제의 해법은 이 방법을 적용하기 위한 핵심이다.

주어진 측지좌표  $P_1$ 과 거리  $S_{12}$ , 그리고 측지 방위각  $a_{12}$ 를 이용하여 점  $P_2$ 의 측지좌표를 결정하기 위한 직접 문제는 다음과 같다.

$$\phi_2 = \phi(\phi_1, \lambda_1, S_{12}, \alpha_{12}) \quad (3-2)$$

그리고,

$$\lambda_2 = \lambda(\phi_1, \lambda_1, S_{12}, \alpha_{12}) \quad (3-3)$$

주어진 측지좌표  $P_1$ 과  $P_2$ 를 이용하여 측지거리  $S_{12}$ 와 측지방위각  $a_{12}$ 를 결정하기 위한 역문제는 다음과 같다.

$$S_{12} = S(\phi_1, \lambda_1, \phi_2, \lambda_2) \quad (3-4)$$

그리고,

$$a_{12} = \alpha(\phi_1, \lambda_1, \phi_2, \lambda_2) \quad (3-5)$$

위 두 문제를 풀기 위한 50여 가지의 알고리즘이 알려져 있지만, Vincenty(1975)에 의한 방법은 그 정확도와 효율성이 많이 알려져 있으므로 본 연구에서 직접문제와 역문제는 Vincenty 방법을 사용하기로 한다.

### 3.2 경계 획선 방법

타원체 상에서 등거리 원칙을 적용하기 위해서는 다음절부터 설명할 두 가지 알고리즘인 Two-Point 알고리즘과 Three-Point Algorithm이 많이 사용되며 등거리 원칙은 두 가지 알고리즘을 이용하여 적용된다. 첫번째 알고리즘은 해안을 따라서 인접한 한 쌍의 기준선 점으로부터 균등한 거리에 있는 경계의 중간점(mid-point)의 좌표를 찾는 것이다. 두번째 알고리즘은 인접한 세점으로부터 얻어진다는 것이 다를 뿐이다.

#### 3.2.1 Two-Point 알고리즘

Two-Point 알고리즘은 그림 1과 같이 해안선 기준선을 구성하는 점으로부터 각각 양쪽의 기준선에서 각각 한 점( $P_1, P_2$ )씩 선택한 후 등거리 원칙을 적용하여 중간점(mid-point)  $P_m$ 을 정하는 방법이다.

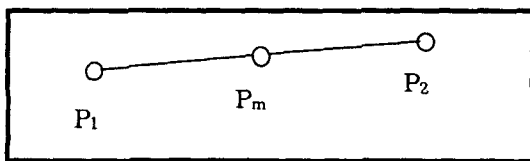


그림 1. Two-Point 알고리즘

#### 3.2.1.1 평면 또는 구(Plane or Sphere)

평면 또는 구 상에서 Two-Point 알고리즘은 마주보고 있는 두 점의 중간점을 계산함으로써 얻을 수 있다. 평면 또는 구 상에서 중간점  $P_m(X_m, Y_m)$ 의 좌표는 점  $P_1$ 과  $P_2$ 좌표의 평균값이다.

$$X_m = (X_1 + X_2)/2, Y_m = (Y_1 + Y_2)/2 \quad (3-6)$$

#### 3.2.1.2 타원체

타원체(Ellipsoid)상에서 Two-Point 알고리즘은 첫번째로 각각의 기준선에서 선택된 두 점  $P_i$ 와  $P_j$  사이의 거리  $S_{ij}$ 와 방위각  $a_{ij}$ 을 구한다.

$$S_{ij} = S(\phi_i, \lambda_i, \phi_j, \lambda_j) \quad (3-7)$$

$$\alpha_{ij} = \alpha(\phi_i, \lambda_i, \phi_j, \lambda_j) \quad (3-8)$$

식 (3-7)과 (3-8)에서 계산된 거리와 방위각을 이용하여 점  $P_i$ 를 기준으로 중간점  $P_m$ 의 측지 좌표를 결정한다.

$$\phi_m = \phi(\phi_i, \lambda_i, S_{ij}/2, \alpha_{ij}) \quad (3-9)$$

$$\lambda_m = \lambda(\phi_i, \lambda_i, S_{ij}/2, \alpha_{ij}) \quad (3-10)$$

마지막으로, 중간점  $P_m$ 이 유효한 경계 중간점으로서 만족해야 할 조건은 그 거리가  $P_i$  또는  $P_j$ 에 작거나 같아야 하고 모든 다른 점들에서도 마찬가지여야 한다.

$$\frac{S_{ij}}{2} \leq S_{mu}, \quad u = 1, 2, 3, \dots, p, \quad u \neq i, j \quad (3-11)$$

$$\frac{S_{ij}}{2} \leq 2_{mv}, \quad v = 1, 2, 3, \dots, p, \quad v \neq i, j \quad (3-12)$$

여기서  $u$ 와  $v$ 는 첫번째와 두번째 기준선에서 발견된 모든 점들을 나타낸다.

#### 3.2.2 Three-Point 알고리즘

$$Y_p = \frac{1}{2} \cdot \frac{(x_3^2 + y_3^2) \cdot (x_2 - x_1) + (x_2^2 + y_2^2) \cdot (x_1 - x_3) + (x_1^2 + y_1^2) \cdot (x_3 - x_2)}{y_3 \cdot (x_1 - x_2) + y_2 \cdot (x_3 - x_1) + y_1 \cdot (x_2 - x_3)} \quad (3-13)$$

$$X_p = \frac{1}{2} \cdot \frac{(x_3^2 + y_3^2) \cdot (y_2 - y_1) + (x_2^2 + y_2^2) \cdot (y_1 - y_3) + (x_1^2 + y_1^2) \cdot (y_3 - y_2)}{y_3 \cdot (x_1 - x_2) + y_2 \cdot (x_3 - x_1) + y_1 \cdot (x_2 - x_3)} \quad (3-14)$$

Three-Point 알고리즘은 그림 2와 같이 양쪽 기준선에서 세점( $P_1, P_2, P_3$ )을 선택한다.

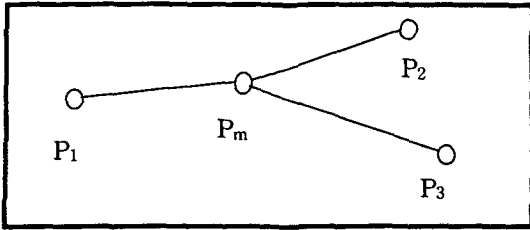


그림 2. Three-Point 알고리즘

한쪽 기준선에서 한점, 반대쪽 기준선에서 두점을 선택하여, 세점에서 등거리인 중간점( $P_m$ )을 찾는 방법이다.

### 3.2.2.1 평면 또는 구(Plane or Sphere)

평면 또는 구 상에서 Three-Point 알고리즘은 세 점에서 등거리에 있는 중간점을 계산함으로써 얻을 수 있다. 두 점( $P_1, P_2$ )이 한쪽 기준선에 있고 나머지 한점( $P_3$ )이 반대쪽 기준선에 있다고 가정하면 중간점  $P_m$ 는  $P_1$ 과  $P_2$  그리고  $P_2$ 와  $P_3$ 의 각각의 중간점에서의 수직인 두 선의 교차점이 되며 공식은 (3-13), (3-14)과 같다.

### 3.2.2.2 타원체

타원체(Ellipsoid)상의 세 점  $P_1(\phi_1, \lambda_1)$ ,  $P_2(\phi_2, \lambda_2)$  그리고  $P_3(\phi_3, \lambda_3)$ 에서 동일한 거리를 갖는 점을  $P_m(\phi_m, \lambda_m)$ 이라 하고,  $P_1$ 에서  $P_m$ 사이의 거리를  $s_1$ 이라 하면,  $s_1 = s_2 = s_3 = s$ 인 등거리 공식이 성립한다. 중간점  $P_m$ 의 초기 근사치를  $(\phi_m^0, \lambda_m^0)$ , 보정치(correction)를  $(\delta\phi_m, \delta\lambda_m)$ 라 하면,  $P_m$ 의 위치는 다음과 같다.

$$\phi_m = \phi_m^0 + \delta\phi_m \quad (3-15)$$

$$\lambda_m = \lambda_m^0 + \delta\lambda_m \quad (3-16)$$

일반적으로, 측지선(geodetic line)  $s_i$ 는  $\phi_m, \lambda_m$ 과  $\phi_i, \lambda_i$  ( $i = 1, 2, 3$ )의 함수이다.

$$s = s_i = f(\phi_m, \lambda_m, \phi_i, \lambda_i) = f(\phi_m^0 + \delta\phi_m, \lambda_m^0 + \delta\lambda_m, \phi_i, \lambda_i) \quad (3-17)$$

이것은 초기 근사치 좌표( $\phi_m^0, \lambda_m^0$ )을 사용하여 테일러 급수(Taylor series)로 확장할 수 있다.

$$s = s_i = f(\phi_m^0, \lambda_m^0, \phi_i, \lambda_i) + \frac{\partial f}{\partial \phi_m} \cdot \delta\phi_m + \frac{\partial f}{\partial \lambda_m} \cdot \delta\lambda_m + \frac{1}{2!} \frac{\partial^2 f}{\partial \phi^2} \cdot (\delta\phi_m)^2 + \dots \quad (3-18)$$

$P_m$ 의 초기좌표를 알고 있으므로, 근사 측지선  $s_i^0 = f(\phi_m^0, \lambda_m^0, \phi_i, \lambda_i)$ 는 타원체상에서 역문제가 된다. 비선형

(non-linear) 항목들을 모두 무시하면, 다음과 같다.

$$s = s_i^0 + a_i \cdot \delta\phi_m + b_i \cdot \delta\lambda_m, \quad i = 1, 2, 3 \quad (3-19)$$

여기서,  $a_i, b_i$ 는 다음과 같다.

$$a_i = -M_m^0 \cdot \cos \alpha_{mi}^0, \quad (3-20)$$

$$b_i = N_i \cdot \cos \phi_i \cdot \sin \alpha_{im}^0$$

식 (3-20)에서  $\alpha_{mi}^0, \alpha_{im}^0$ 은 각각  $P_m$ 에서  $P_1$  그리고  $P_i$ 에서  $P_m$ 까지의 방위각이고,  $(\phi_i, \lambda_i)$ 와  $(\phi_m^0, \lambda_m^0)$ 을 이용하여 계산한다.  $M_m^0$ 는 위도  $\phi_m^0$ 에서 자오선의 곡률반경이고,  $N_i$ 는 위도  $\phi_i$ 에서 묘유선의 곡률반경이다.

$$M_m^0 = \frac{a \cdot (1 - e^2)}{(1 - e^2 \cdot \sin^2 \phi_m^0)^{3/2}} \quad (3-21)$$

$$N_i = \frac{a}{\sqrt{1 - e^2 \cdot \sin^2 \phi_i}}$$

식 (3-19)의 첫 번째 방정식에서 두 번째와 세 번째 방정식을 빼면, 각각 다음과 같다.

$$\begin{bmatrix} a_2 - a_1 & b_2 - b_1 \\ a_3 - a_1 & b_3 - b_1 \end{bmatrix} \cdot \begin{bmatrix} \delta\phi_m \\ \delta\lambda_m \end{bmatrix} = \begin{bmatrix} s_1^0 - s_2^0 \\ s_1^0 - s_3^0 \end{bmatrix} \quad (3-22)$$

위 식의 첫 번째 행렬을 A라고 하면,  $(\delta\phi_m, \delta\lambda_m)$ 는 다음과 같이 계산될 수 있다.

$$[\delta\phi_m \quad \delta\lambda_m] = A^{-1} \cdot [s_1^0 - s_2^0 \quad s_1^0 - s_3^0] \quad (3-23)$$

$\delta\phi_m, \delta\lambda_m$ 가 계산된 후에  $P_m$ 의 값은 식 (3-15) 및 (3-16)에 의하여 구할 수 있다.

마지막으로, 중간점  $P_m$ 이 유효한 경계 중간점으로서 만족해야 할 조건은 그 거리가  $P_i$ 에 작거나 같아야 하고 모든 다른 점들에서도 마찬가지여야 한다.

$$S^{(n)} \leq S_{mu} \quad \begin{matrix} u = 1, 2, 3, \dots, p \\ u \neq i, j, k \end{matrix} \quad (3-24)$$

$$S^{(n)} \leq S_{mv} \quad \begin{matrix} v = 1, 2, 3, \dots, p \\ v \neq i, j, k \end{matrix} \quad (3-25)$$

## 3.3 중간점 오차 예측

### 3.3.1 오차예측 개요

미지값들(unknown values)은 종종 원하는 미지수에 기능적으로(functionally) 관련되어 있는 다른 양들의 직접 관측을 통하여 간접적으로 결정된다. 그리고 미지수는 계산된다. 거리 및 각도 관측을 통한 삼각점 좌표계산, 차분 레벨의 읽음으로서 삼각점 높이값 계산, 천문관측을

통한 선의 방위각을 결정하는 측량의 예가 있다. 그러나 모든 직접 관측치는 에러를 가지고 있으므로, 이 직접관측치로 계산한 값도 에러를 가지고 있다.

그러므로 이 에러의 간섭(intrusion) 또는 전파(propagation), 직접 관측치로부터 계산된 것으로부터 에러가 생기는 것을 오차전파(Error Propagation) 또는 오차예측(Error Estimation)이라 한다.

본 논문에서는 단지 임의에러(Random Error)만 직접 관측에 남아있다고 가정하기 위하여 모든 규칙적인 에러(systematic error)는 제거한다

### 3.3.2 Three-Point 알고리즘 오차예측

만약 3개의 기준점  $P_1, P_2, P_3$ 이 에러를 포함하고 있다면, 이 에러들은 자동적으로 위 식 (3-15)과 (3-16) 및 (3-23)에서 얻어진 중간점  $P_m$ 의 좌표( $\phi_m^0, \lambda_m^0$ )로 전파된다. 중간점  $P_m$ 을 위한 좌표 보정치 ( $\delta\phi_m, \delta\lambda_m$ )와 기준점  $P_i(i=1, 2, 3)$ 을 위한 좌표 보정치( $\delta\phi_i, \delta\lambda_i$ )의 관계는  $P_m$ 에서  $P_i$ 사이의 거리  $s_i$ 의 선형화를 통하여 구성할 수 있다.

$$s_i = f(\phi_m^0 + \delta\phi_m, \lambda_m^0 + \delta\lambda_m, \phi_i + \delta\phi_i, \lambda_i + \delta\lambda_i) \approx s_i^0 + a_i \cdot \delta\phi_m + b_i \cdot \delta\lambda_m + c_i \cdot \delta\phi_i + d_i \cdot \delta\lambda_i \quad (3-26)$$

여기서,  $s_i, a_i$  그리고  $b_i$ 는 이미 정의하였다. 그리고  $c_i, d_i$ 는 다음과 같다.

$$c_i = -M_i \cdot \cos\alpha_{im}^0 \\ d_i = -N_i = -N_i \cdot \cos\phi_i \cdot \sin\alpha_{im}^0 \quad (3-27)$$

식 (3-26)에서 두 번째와 세 번째 방정식을 첫 번째 방정식에서 빼면, 각각 다음과 같다

$$[\delta\phi_m, \delta\lambda_m]^T = A^{-1} \cdot [s_1^0 - s_2^0 \quad s_1^0 - s_3^0]^T + B \cdot [\delta\phi_1 \quad \delta\lambda_1 \quad \delta\phi_2 \quad \delta\lambda_2 \quad \delta\phi_3 \quad \delta\lambda_3]^T \quad (3-28)$$

여기서 A는 식 (3-23)과 같고 B는 다음과 같이 정의할 수 있다.

$$B = A^{-1} \cdot \begin{bmatrix} c_1 & d_1 & -c_2 & -d_2 & 0 & 0 \\ c_1 & d_1 & 0 & 0 & -c_3 & -d_3 \end{bmatrix} \quad (3-29)$$

식 (3-28)에 오차 전파 법칙을 적용하면  $P_m$  좌표에 대한 variance-covariance 행렬  $\Sigma_m$ 은 다음과 같이 얻어질 수 있다.

$$\Sigma_m = B \cdot \Sigma \cdot B^T \quad (3-30)$$

여기서  $\Sigma$ 는 세 기준점  $P_1, P_2, P_3$ 에 대한 좌표 벡터  $X_c$ 의 variance-covariance 행렬이다.

$$X_c = [\phi_1 \quad \lambda_1 \quad \phi_2 \quad \lambda_2 \quad \phi_3 \quad \lambda_3]^T \quad (3-31)$$

## 4. 연구결과

### 4.1 해양경계 획선 프로그램 개발

본 연구에서 개발한 경계획선 소프트웨어는 인접한 국가간의 경계를 결정하기 위한 컴퓨터 프로그램이다. 이 프로그램은 타원체상의 거리, 좌표변환 등의 측지학적 계산이 가능하며, 특히 해양경계 획선에 유용하게 사용될 수 있다. 이를 Ocean\_Master 2.0으로 명명하였다.

### 4.2 해양경계 획선 프로그램 서브모듈

#### 4.2.1 측지거리 및 좌표계산

직접/역문제를 통하여 두점간의 측지거리 및 한점에서 일정한 거리가 떨어진 점의 좌표를 계산한다(그림 3).

#### 4.2.2 좌표변환

타원체간의 좌표변환을 수행한다(그림 4).

#### 4.2.3 면적계산

폴리곤의 면적을 계산한다(그림 5).

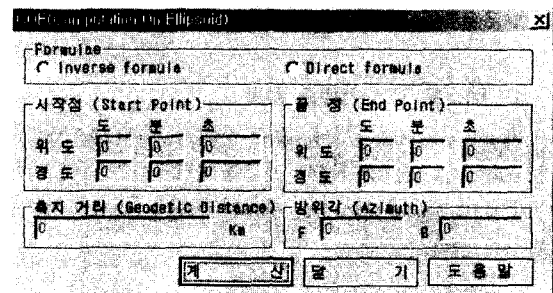


그림 3. 측지거리 및 좌표계산 모듈

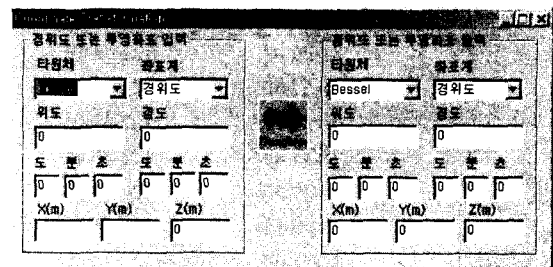


그림 4. 좌표변환 모듈

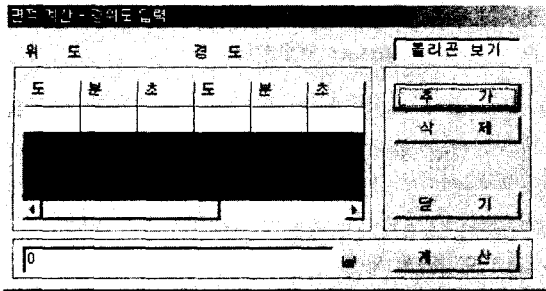


그림 5. 면적계산 모듈

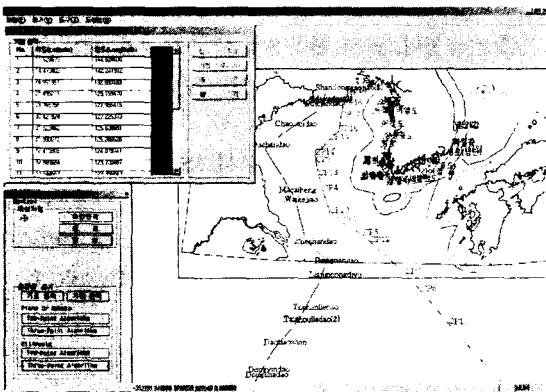


그림 6. 중간선 계산 모듈

#### 4.2.4 중간선 계산

알고리즘별 타원체별 중간선을 계산한다(그림 6).

#### 4.3 테스트 및 오차예측 결과

위 두 알고리즘 결과를 보기 위하여 서해(The Yellow Sea)를 중심으로 중국지역 기준점 8개와 우리나라 지역 기준점 9개를 선택하였고 선택된 기준점 좌표는 표 1, 표 2와 같다.

표 1. 테스트에 사용한 기준점(중국지역)

기준점 (BasePoint)	중 국(China)	
	위도(Latitude)	경도(Longitude)
1	37-24-00	122-42-18
2	36-57-48	122-34-12
3	36-53-42	122-31-06
4	36-44-48	122-15-48
5	35-53-30	120-53-06
6	35-00-12	119-54-12
7	33-21-48	121-20-48
8	33-00-54	121-38-24

표 2. 테스트에 사용한 기준점(대한민국지역)

기준점 (BasePoint)	대한민국(Korea)	
	위도(Latitude)	경도(Longitude)
1	36-58-38	125-45-02
2	36-36-36	125-32-30
3	36-07-05	125-58-11
4	35-53-10	126-04-15
5	35-39-30	126-06-16
6	35-20-03	125-59-14
7	34-43-03	125-11-25
8	34-40-18	125-10-25
9	34-06-51	125-04-42

측지거리 및 방위각 계산, 즉 역문제는 경계확선을 하기 위한 알고리즘의 핵심이다. 역문제 결과를 정확도를 인정받고 있는 Delmar 1.0과 비교하였다. 그 결과 본 연구에서 개발한 프로그램의 역문제 결과가 Delmar 1.0보다 정확하다는 것을 확인할 수 있었다. 표 1과 2의 기준점을 이용하여 계산한 결과 비교는 표 3과 같다. 여기서  $\alpha$ 는 방위각이고  $d$ 는 측지거리이며 단위는 m이다.

표 1, 표 2의 기준점을 두 알고리즘을 적용하여 중간점을 계산한 결과는 표 4와 표 5에 정리되어 있다. 그림 7을 보면 Two-point 알고리즘을 사용하였을 때 5개의 중간점이 계산되어 해안의 굴곡에 따른 중간선을 충분히 표현할 수 없었으나, 그림 8의 결과에서는 Three-point 알고리즘을 사용하였을 경우 8개의 중간점이 계산되어 양쪽 기준선간의 중간 지역을 충분히 표시할 수 있었다.

위 결과를 역문제에서 비교대상으로 삼았던 Delmar 1.0과 다시 비교하여 정확도를 검증하였다. 그 결과 본 연구에서 개발한 프로그램이 정확한 결과를 나타내었다. 아래의 표 6은 Three-Point 알고리즘을 사용하였을 때의

표 3. 역문제(Inverse Problem) 결과 비교

		Delmar 1.0	Ocean_Master 2.0
중국1~ 대한 민국1	$\alpha_{12}$	98-55-14.2996	<b>98-55-14.2996</b> 3474986
	$\alpha_{21}$	280-45-42.4282	<b>280-45-42.4281</b> 5377515
중국2~ 대한 민국2	d(m)	274413.471	<b>274413.470</b> 21515
	$\alpha_{12}$	97-31-1.9013	<b>97-31-01.9013</b> 342709
중국3~ 대한 민국3	$\alpha_{21}$	279-17-49.2179	<b>279-17-49.2178</b> 9871
	d(m)	268089.011	<b>268089.010</b> 614771
중국1~ 대한 민국1	$\alpha_{12}$	104-32-59.0198	<b>104-32-59.0197</b> 50716
	$\alpha_{21}$	286-36-12.4518	<b>286-36-12.4517</b> 669479
d(m)	320923.331	<b>320923.330</b> 917893	

표 4. Two-Point 알고리즘 결과

중간점 (Mid-Point)	Two-Point 알고리즘	
	위도(Latitude)	경도(Longitude)
1	37-00-48.5532	124-07-50.4192
2	36-47-45.4164	124-03-33.2820
3	35-48-49.5648	123-52-21.1476
4	33-34-35.5800	123-20-53.7756
5	34-49-25.4152	122-54-46.1801

표 5. Three-Point 알고리즘 결과

중간점 (Mid-Point)	Three-Point 알고리즘	
	위도(Latitude)	경도(Longitude)
1	37-17-40.6831	124-15-10.5452
2	35-10-33.9600	123-09-41.2848
3	34-37-42.5316	122-56-56.8104
4	36-53-13.4520	124-04-33.6612
5	36-12-35.7780	123-57-19.3284
6	35-31-52.9428	123-31-38.4492
7	34-54-25.7292	122-52-46.5708
8	34-07-23.7072	123-05-49.4376

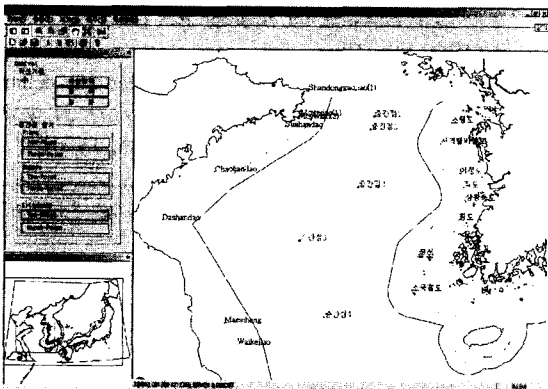


그림 7. Two-Point 알고리즘 결과

결과와 Delmar 1.0의 결과를 비교한 것으로 중국 및 한국의 기준점은 표 1과 2의 값을 이용하였다.

중간점은 해안선의 기준점에서 등거리 조건을 만족하는 경계이다. 중간점을 결합하여 그려진 경계는 내삽(interpolation)을 이용한 방법이므로 오차를 수반하게 된다.

Three-Point 알고리즘의 오차예측을 수행하기 위하여 다음 그림 4와 같이 임의의 세 점 P<sub>1</sub>, P<sub>2</sub> 그리고 P<sub>3</sub>를 선정한 후, P<sub>1</sub>과 P<sub>2</sub>의 거리를 10 km P<sub>2</sub>와 P<sub>3</sub>의 거리를

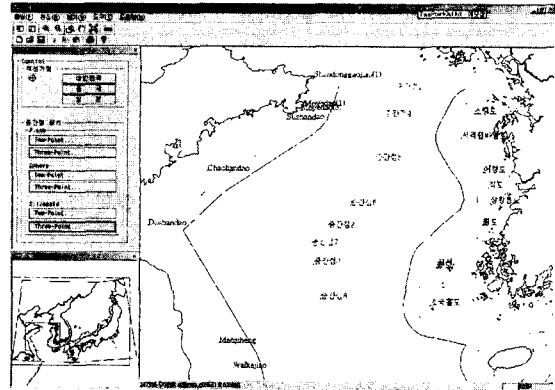


그림 8. Three-Point 알고리즘 결과

표 6. Delmar 1.0과 Ocean\_Master 2.0 결과 비교

Delmar 1.0		Ocean_Master 2.0	
중간점		중간점	
37-17-40.6831		37-17-40.6830306	
124-15-10.5452		124-15-10.5451788	
중국1	137634.11116123	중국1	137634.113
한국1	137634.11116123	한국1	137634.112
한국2	137634.11116123	한국2	137634.113
중간점		중간점	
35-13-18.3904		35-13-18.3936	
123-25-54.4249		123-25-54.4242	
중국2	146731.69917658	중국2	146731.695
한국1	146731.69917658	한국1	146731.707
한국2	146731.69917658	한국2	146731.707

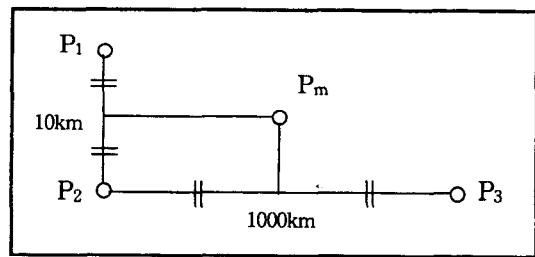


그림 9. 오차예측을 위한 기준점

1000 km로 두었다. 이 세 점을 이용하여 중간점 오차예측을 수행하였다.

표 7-9는 세 점의 표준편차가 각각 0.05 m일 때 오차 결과이다. 구와 타원체상에서의 결과는 큰 차이가 없는 이유는 타원체상에서 작은 이심률 때문이다. 북쪽방향의 에러가 동쪽방향보다 많은 이유는 P<sub>1</sub>-P<sub>2</sub>기준선이 P<sub>1</sub>-P<sub>3</sub>기

표 7. Three-Point 알고리즘 오차예측 결과 (1)

Point	Spherical solution	
	$\sigma_n$ [m]	$\sigma_e$ [m]
$P^m$	3.478	0.468

표 8. Three-Point 알고리즘 오차예측 결과 (2)

Point	Planar solution	
	$\sigma_n$ [m]	$\sigma_e$ [m]
$P_m$	3.535	0.035

표 9. Three-Point 알고리즘 오차예측 결과 (3)

Point	Ellipsoidal solution	
	$\sigma_n$ [m]	$\sigma_e$ [m]
$P_m$	3.476	0.468

준선보다 더 짧기 때문이다.

## 5. 결 론

경계 획선 알고리즘 정립과 프로그램개발을 통하여 중간점을 계산하고 오차예측한 결과 다음과 같은 결론을 얻었다.

1) 해양경계 획선 프로그램 개발을 통하여 도면을 이용한 수작업에 의한 여러 오차를 제거 할 수 있고, 해양 경계협상시 신속히 대처할 수 있다.

2) 기준선의 기준점 위치어러는 계산된 중간선의 중간

점으로 전파된다.

3) 오차전파는 알고리즘에 따라 다르며 기준면(평면, 구, 타원체)에 따라 다르다. 기준면에 따른 중간점 좌표의 차이는 몇 킬로미터이지만, 계산된 어러는 단지 몇 센티미터에 불과 하다. 그러므로 일정한 지역에 대한 정확도를 높이기 위해서는 지역에 맞는 기준면을 선택하는 것이 중요하다.

4) 타원체상에서 등거리원칙에 기초한 알고리즘을 정립하여 지도 투영에 의한 어떤 왜곡도 없이 적용할 수 있고 평면이나 구에서 적용한 것 보다 오차가 적으므로 중간점은 타원체상에서 계산해야 한다.

## 참고문헌

1. 해양수산부 해양법에 따른 국제연합협약, 1998 pp. 3-11.
2. 한국해양연구소, 주변해양 경계획선 방식에 관한 연구, 1992.
3. 국립해양조사원, 해양경계획선 프로그램개발, 1999.
4. Carrera G. A method for the delimitation of an equidistant boundary between coastal states on the surface of a geodetic ellipsoid. Int. hydrogr. Rev (Monaco), 1992, LXIV(1):147-159.
5. Sjoberg, Lars E. Error propagation in maritime delimitation. In: Proceedings: Geodetic Aspects of the Law of the Sea, Denpasar, Bali, Indonesia: 1996, 153-168.

(2001년 9월 14일 원고접수)