

유전자 알고리즘을 이용한 소축척지도제작 편집 프로그램 개발 The Development of Editing Program in Small-Scale Mapping using a Genetic Algorithm

김현덕¹⁾ · 박경식²⁾ · 최석근³⁾ · 이재기⁴⁾

Kim, Hyun-Duck · Park, Kyung-Shicl · Choi, Seok-Keun · Lee, Jae-Kee

- 1) 충북대학교 대학원 토목공학과 석사과정(hyunduck0@git.co.kr)
- 2) 인하공업전문대학 항공지형정보시스템과 조교수(pks@inhac.ac.kr)
- 3) 충북대학교 공과대학 토목공학과 부교수(choi@chungbuk.ac.kr)
- 4) 충북대학교 공과대학 토목공학과 교수((Leejk@cbucc.chungbuk.ac.kr)

요지

소축척지도제작을 위한 자동화 처리 과정에서는 기하학적 및 논리적 오류가 발생하고, 기하학적 오류는 많은 부분에서 자동화 처리가 가능하나 논리적 오류는 여러 가지 경우에 대하여 자동 판단이 곤란한 경우가 많기 때문에 대부분 수작업으로 이루어 지고 있는 실정이다.

따라서, 본 연구는 수치지도를 이용한 일반화 처리 후의 지형도 제작시에서 나타나는 여러 가지 문제 중 도로 과장화로 인한 오류문제를 해결하기 위하여 도로와 건물 폴리곤간의 겹침위치를 자동 탐색하고, 이를 자동 처리하기 위한 프로그램을 유전자 알고리즘을 이용하여 개발하였다. 그 결과 지도제작 과정에서 발생하는 오류를 해결할 수 있었고, 지도제작 자동화율을 향상시킬 수 있었다.

1. 서 론

국가에서는 1/1,000, 1/5,000 및 1/25,000, 기타 소축척지도 등의 수치지도 및 종이지도와 각종 주제도 등을 제작하고 있다. 이와 같이, 축척이 다른 지도제작을 위해서는 막대한 예산과 시간이 소요되기 때문에 최신 정보제공과 시간·경제적인 면에서 많은 문제점들이 나타나고 있다.

국내·외에서의 소축척지도제작은 대축척지도를 이용하여 일반화처리과정으로 자동제작하자 하는 연구가 수행되고 있고, 컴퓨터 기술의 발달로 그 가능성이 높아졌으나, 표현 방법과 자동화 등 여러 가지 면에서 많은 문제점들이 나타나고 있다.

이러한 자동지도제작과정에서는 기존 수치지도나 지형도가 가지고 있었던 단순 도형 자료의 정보 부족과 자동화 처리 과정에서 발행하는 기하학적·논리적 오류가 발생하고 있고, 기하학적 오류는 이미 많은 부분에서 자동화 처리가 이루어 지고 있으나, 논리적 오류는 논리적으로 자동 판단이 곤란한 경우가 많기 때문에 대부분 수작업으로 이루어 지고 있어 자동화 처리가 요구되는 실정이다.

따라서, 본 연구는 소축척지도제작의 자료처리과정에서 도로과장화로 인해 도로와 건물이 겹치는 편집과정에서의 논리적 오류문제를 해결하기 위하여 도로와 건물 폴리곤간의 여러 가지 겹침오류를 자동 검색하고, 이를 처리할 수 있는 프로그램을 유전자 알고리즘(genetic algorithm)을 이용하여 개발하므로써 지도제작오류 최소화, 자동화율 향상, 지도제작비 절감 등에 기여할 수 있는 프로그램을 개발하는데 그 목적이 있다.

2. 유전자알고리즘 이론

유전자 알고리즘은 자연 진화 이론에 근거한 탐색 기법으로 최적화 문제를 해결하는 도구로 사용되고 있다. 자연 진화 법칙에 의하면 생존 확률이 큰 개체가 교배와 돌연변이의 과정을 통해 보다 적합한 방향으로 진화해 나가며, 부적합한 개체는 도태되어 사라지게 된다. 이러한 자연진화 법칙 기반으로 유전적 계승과 다윈의 생존경쟁이라는 자연 현상을 모델링한 확률적인 탐색 방법이다.

유전자 알고리즘은 복잡한 해 공간의 탐색 능력이 우수하여 변수와 제약이 많은 대형 공간문제를 푸는데 적합하고, 또한 모형에 대한 유연성 및 강건성이 높아 제약 첨가나 목적함수의 변경이 용이하다는 장점을 가지고 있다. 따라서, 지도 데이터의 경우 방대한 폴리곤 객체의 위치 예측 및 최적화 구성에 유전자 알고리즘이 적합하다.

문제 해결을 위한 임의의 해를 유전자 알고리즘을 이해하는 형태로 표현하여 염색체라 하고, 이 염색체상의 각 인자를 유전자라 한다. 생물학에서는 유전자가 염기 하나와 대응되지 않지만 유전 알고리즘에서는 인자 하나를 유전자라 한다. 생물학에서 유전자형(genotype)은 유전자의 조합이고, 표현형(phenotype)은 관찰되는 형질을 말한다. 전형적인 유전자 알고리즘은 이진수를 이용한 일차원적 표현을 사용하지만, 문제의 해를 단순 표현하기 위한 여러 표현법이 나타나고 있다. 따라서, 본 연구에서는 공간 탐색으로 인한 수행 능력을 고려하여 전형적인 유전자 표현 방법인 이진 표현을 그대로 사용하였다.

유전자 알고리즘에 적용되는 연산자는 선택(select), 돌연변이(mutation), 그리고 교배(crossover) 등 세가지로 자연계를 모방한 연산을 수행한다.

선택 연산자는 교배에 쓰이는 두개의 부모해를 고르기 위한 연산자로서 다양한 선택 방법이 있으나 공통적으로 우성 유전자가 선택될 확률이 높으며, 우성 유전자와 열성 유전자 사이의 적합도 차이를 조절함으로써 선택 확률을 조절할 수 있다. 이 차이의 정도를 선택압(select pressure)이라 한다. 선택압이 빠를수록 이른 수렴의 가능성이 높은 반면 낮으면 해집단의 평균 적합도가 떨어질 가능성이 많다.

이진 유전자를 사용하는 유전자 알고리즘에는 일반적으로 적합도에 비례하여 다음 세대에 유전인자를 복제하는데 이를 비례 선택이라 하고, 룰렛휠 선택, 토너먼트 선택 및 엘리트 보존 방법 등이 사용된다. 따라서, 지도 데이터의 경우 방대한 폴리곤 객체의 위치 예측 및 최적화 구성에 유전자 알고리즘이 적합하다. 그러나, 본 연구에서와 같은 논리적 오류를 해결하는 문제에 있어서는 기존 유전자 알고리즘과 비교해 보다 정확한 선별, 효과적인 탐색, 다양한 해 공간에 대한 검토가 가능하도록 검색시 대략적인 위치 이동 직선을 가진다.

유전자 알고리즘에 적용되는 연산자는 초기화, 돌연변이, 그리고 교배 등이다. 초기화 연산자는 유전자 알고리즘이 어떻게 시작되는지 결정하고, 돌연변이 연산자는 우수하지 않거나 전혀 다른 형질을 이용함으로써 근접할 수 없는 해를 찾거나 전혀 다른 개체를 생성하게 한다. 교배연산자는 두 부모유전자로부터 하나의 새로운 개체들을 생산하게 하고, 이 개체들은 부모 유전자로부터 유전된다.

목적함수는 단지 하나의 개체가 다른 개체들과 비교해 얼마나 더 좋은지에 대한 하나의 척도만을 제공한다. 일반적으로 적합도는 유전자 알고리즘의 교배 여부를 판단하여 개체들의 적합성을 결정하는데 사용된다. 이 적합도에 따라 유전자 알고리즘이 해집단을 수렴시키는데 큰 영향을 미친다.

3. 편집 알고리즘 개발

본 연구에서 개발된 자동처리 프로그램은 유전자 알고리즘을 이용하여 알고리즘 구현에 필요한 모델 선택 및 연산자로 토너먼트 선택, 일리티스트 모델, 일정 교배, 그리고 균등 돌연변이 등을 사용하였다.

유전자 알고리즘 적용은 도로 폴리곤 객체를 고정시키고, 겹침을 해결하기 위해 건물을 편위시키는 것에 중점을 두고 연구 수행하였다. 기본 구조는 지도 디스플레이 구조를 실험하고, 객체가 범위 내에서 이동할 수 있는 편위 개념을 도입하였다. 각 폴리곤 객체 n 은 x, y 좌표로 저장되는 점들의 집합으로 표현될 수 있다. 지도의 충돌을 제거하기 위하여 폴리곤 객체들은 충돌을 최소화하는 새로운 위치로 이동되어야 하고, 이동 위치는 전체 x, y 공간이지만 현실적으로 불필요하비만, 일정 반경의 인접한 공간이 각 폴리곤 객체의 이동 범위가 될 수 있으므로, 이를 객체의 이동 제약조건으로 둔다.

3.1 적합도 평가

지도 일반화 문제에서 해를 평가하는데 사용되는 목적함수는 각각의 정의가 필요하며, 이 정의에 의해 문제의 기본 구조가 모델링 된다. 본 연구에서 사용한 구조는 다음과 같다.

- $O : \{ o1..on \}$ 은 폴리곤 객체의 집합, n : 폴리곤 객체의 수
- do min : 폴리곤 객체의 최소 거리 임계값,
- 폴리곤 객체가 서로 충돌하면 DO_{ij} 는 1이 되고 없으면 0이 된다.
- 실제 폴리곤의 위치와 이동 후 폴리곤의 위치의 거리가 최소가 되어야 한다.

지도 일반화 문제에서 해를 평가하는데 사용되는 목적함수는 식(1)과 같다.

$$f = f_1 + f_2 + f_3 \quad \text{----- (1)}$$

f 는 다음에 해당하는 적합도 평가 함수의 총합은 식(2)가 된다.

$$f_1 = \sum_{i=1}^n \sum_{j=1}^n DO_{ij} \quad \text{----- (2)}$$

f_1 은 도로와 폴리곤 객체(o)의 관계이며, 두 객체가 충돌하면 DO_{ij} 는 1, 아니면 0의 값으로 계산된다.

$$f_2 = \sum_{i=1}^n \sum_{j=1}^n PO_{ij} \quad \text{----- (3)}$$

f_2 는 건물 폴리곤 간의 겹침 평가 함수로, 두 객체가 충돌하면 PO_{ij} 는 1, 아니면 0으로 계산된다.

$$f_3 = \sum_{i=1}^n \sqrt{(dx_i^2 + dy_i^2)} \quad \text{----- (4)}$$

f_3 는 객체의 초기 위치에서 이동후 위치 사이의 거리 관계이며, 디스플레이 상의 거리로 계산된다.

3.2 게놈 표현

게놈은 실제 좌표 값의 저장이 아니고, 디스플레이 위치의 실수 값 배열로 저장되며, 지도 디스플레이 상의 물리적 값과 일치한다. 게놈과 Phenotype은 그림 1과 같다.

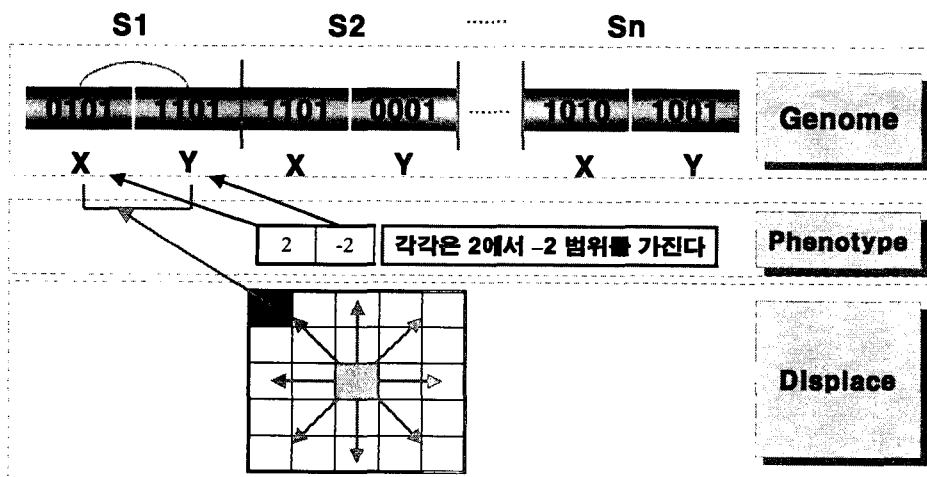


그림 1. Genome, phenotype and displace

지도 세그먼트(m), 개체군 크기(p_m), 그리고 최대 세대수(g_m)은 실험적으로 결정된 경험적(heuristic) 방법을 이용한다.

$$p_m = 4C(s_m) \quad \text{----- (5)}$$

$$g_m = 15s_m \quad \text{----- (6)}$$

경험적 값들은 실험적으로 결정되고 생성될 해들이 최적, 또는 최적에 가까운 해가 될 수 있게 한다.

3.3 교배와 돌연변이

교배 확률(probability of crossover or reproduction)은 각 세대에 얼마나 자주 교배가 발생할 것인가를 결정한다. 일점 교배(single point crossover)는 그림 2와 같이 한 지점만 부모세대의 유전자를 교환하는 교배 방법이다. 교배 확률이 0%으로 설정되면, 각 자식(offspring)은 부모(parent) 유전자를 그대로 복사하게 되고, 역으로 교배확률이 100% 설정되면 자식세대는 각 부모의 유전전달 부분이 교차되게 이루어지며, 본 연구에서는 교배 확률을 50%로 설정하였다.

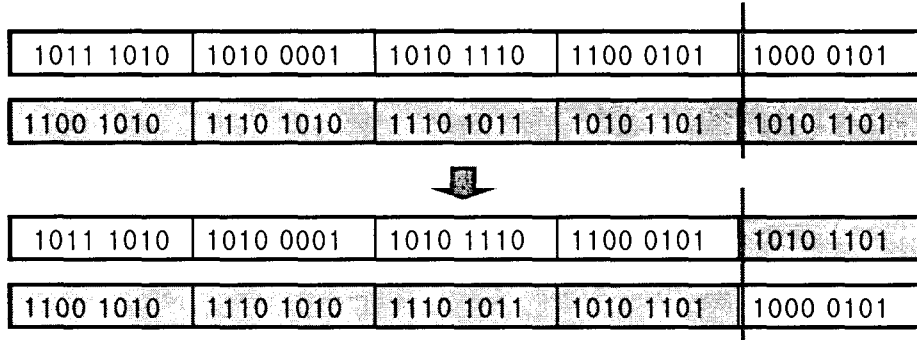


그림 2. 일점교배

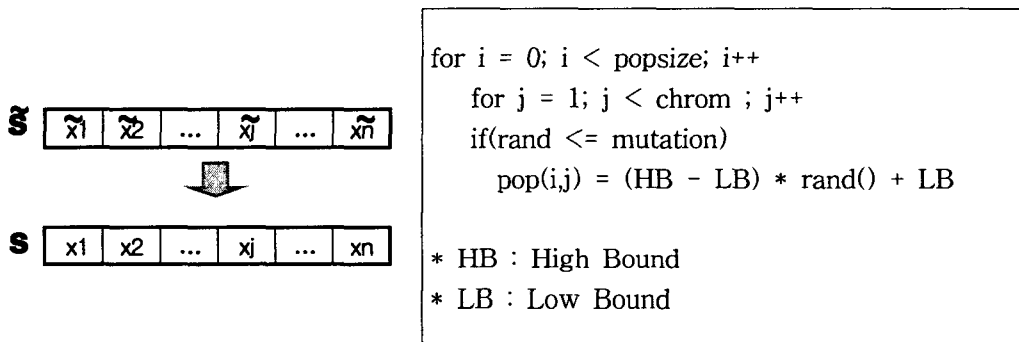


그림 3. 돌연변이와 프로그램 코드

3.4 검색 알고리즘 개발

도로 및 건물은 각각의 폴리곤 객체로 되어 있으므로 각 객체간의 논리적 오류를 판단해야 한다. 이 판단의 기준은 각 폴리곤 객체가 직선으로 되어 있으므로 도로 폴리곤 내의 한변에 해당하는 직선과 건물 폴리곤 내의 한변에 해당하는 직선을 가지고 두 직선의 교차 점을 판단하여 겹침을 확인 할 수 있다. 먼저 해당 폴리곤의 최소영역(MBR)을 검색후 교차점 검색을 통해 탐색하도록 하였다.

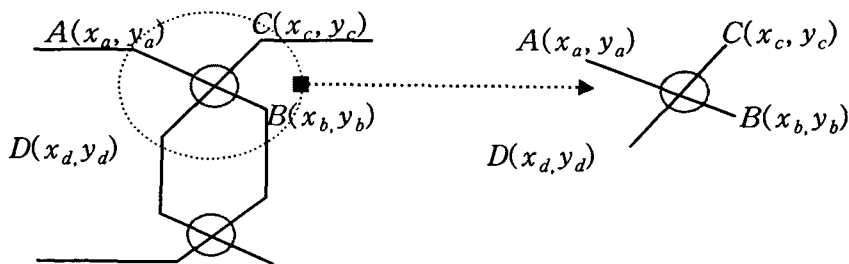


그림 4. 폴리곤 겹침

그림 4에서 폴리곤 점을 이용하여 각 점의 기울기를 구하고, 기울기가 같으면 평행이기 때문에 기울기가 같도록 선행조건을 고려하여야 한다.

각각을 X와 Y의 식으로 만들고, t와 s를 식 (7)를 이용하여 결정한다.

$$t = \frac{(x_c - x_a)(y_d - y_c) + (x_d - x_c)(y_a - y_c)}{(x_b - x_a)(y_d - y_c) - (x_d - x_c)(y_b - y_a)} \text{-----}(7)$$

$$s = \frac{(x_c - x_a)(y_b - y_a) + (x_b - x_a)(y_c - y_a)}{(x_b - x_a)(y_d - y_c) - (x_d - x_c)(y_b - y_a)}$$

4. 프로그램 설계 및 개발

4.1 프로그램 설계

일반적인 소축척 지도제작의 편집과정에서 발생하는 논리적 오류는 도로선형을 확대과정에서 많이 발생하며, 이로 인한 도로 및 건물은 각각의 폴리곤 객체로 되어 있으므로 먼저 각 객체간의 논리적 오류를 판단해야 한다. 이를 위해 도로 폴리곤 내의 한변에 해당하는 직선과 건물 폴리곤 내의 한변에 해당하는 두 직선의 교차 점을 판단하여 해당 폴리곤의 최소영역(MBR)을 검색한 후 교차점을 검색하도록 하였다. 한 점에 대해서 교차가 이루어 지면 두 폴리곤은 겹침이므로 더 이상 반복하지 않는다.

그림 5는 겹침 폴리곤 판단에 대한 전체적인 흐름도를 나타낸다.

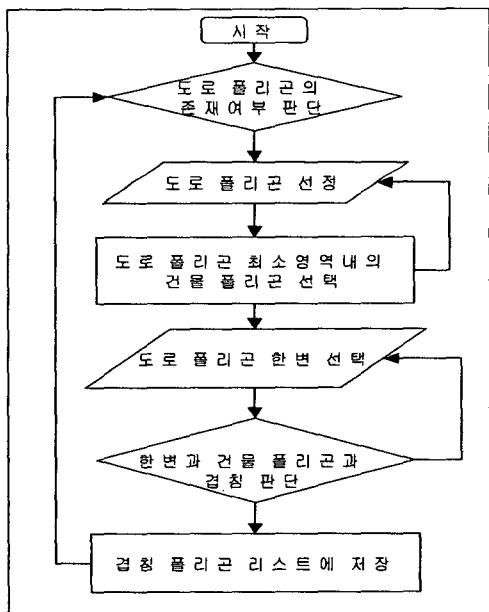


그림 5. 겹침 폴리곤 판단 흐름도

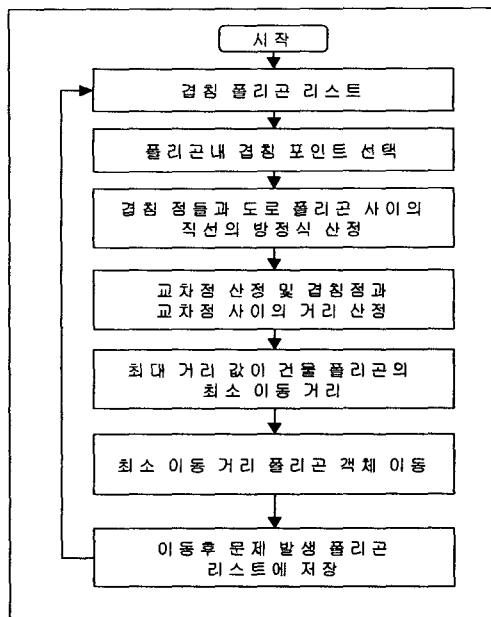


그림 6. 겹침 폴리곤의 처리 흐름도

4.2 프로그램 개발 및 처리결과

개발 프로그램은 검색과 자료처리 등 두과정으로 이루어진다. 검색 프로그램은 윈도우 2000을 기본 OS 환경으로 개발언어는 C/C++ 이고, 개발툴은 visual studio 6.0을 이용하였다. 검색 프로그램에 사용된 데이터는 shp 데이터로 도형 및 속성 정보까지 다룰 수 있어 향후 많은 정보를 추가하는데 용이하여 사용하였고, 검색 프로그램은 shp 데이터를 로드할 수 있다.

표 1. 프로그램 개발 환경

	최소 사양	운영 사양
하드웨어	CPU : Pentium 300Mhz 이상 Main Memory : 128MB 이상	CPU : Pentium 4 2.4Ghz Main Memory : 512MB
소프트웨어	Windows 2000 Windows 2000 Pro Windows XP Windows NT 4.0	Windows 2000 Pro MS Visual Studio (Visual C++ 6.0) MS Word 2002 및 한글 97

본 연구에서는 유전자 알고리즘에서 사용되는 제약변수를 모집단 100, 교배확률 80%, 돌연변이 확률 2%로 주었을때 가장 좋은 적합도를 가진 결과를 그림 7 및 8과 같이 얻을 수 있었다.

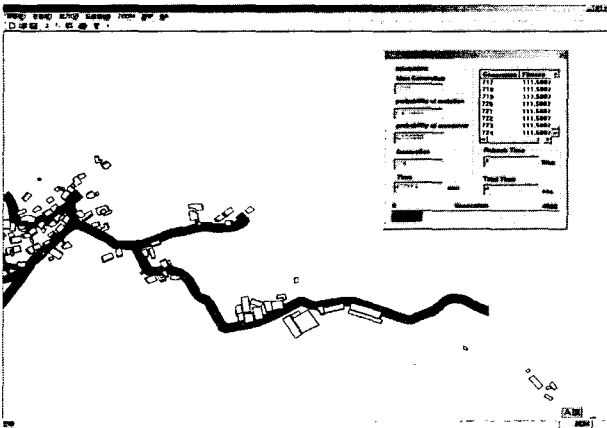


그림 7. 자동처리 프로그램 수행 과정

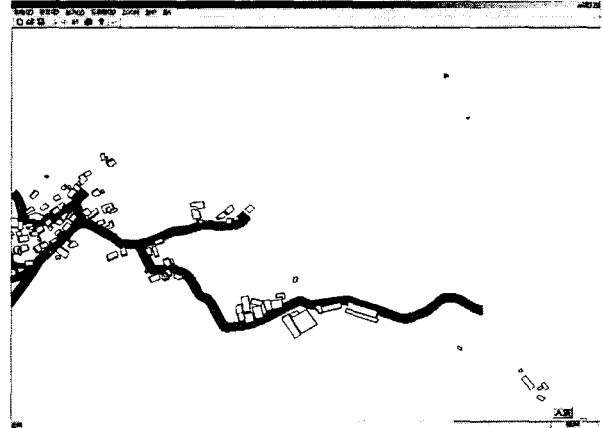


그림 8. 자동처리 프로그램 수행 결과

본 연구에서는 소축척 지형도를 제작하는 과정에서 발생하는 지도편집 오류를 본 연구에서 개발한 자동 처리 프로그램과 기존 이용되고 있는 수작업공정을 표 2와 같이 비교·분석했다.

표 2. 작업처리 결과분석

	오류탐색방법	처리과정	처리시간	탐색정확도	비 고
수 작업	육안 검사	숙련도	2	개인오차	검수시 반복작업이 필요하고, 오류처리시 완전처리 곤란
자동처리	자동 탐색	자동처리	16	100%	시간이 많이 소요

본 연구에서 개발된 자동처리 프로그램을 이용하여 작업을 처리한 경우 기존 수작업에 비해 지도편집 오류 검출과 개인적인 오차에 의한 오류를 줄이고 자동화율을 높일 수 있었으나, 자료처리 시간이 상대적으로 너무 많이 소요되는 결과를 나타냈으며, 향후 처리시간에 대한 문제 해결이 요구된다.

5. 결 론

본 연구는 지도 제작에 있어서 발생하는 편집 오류문제를 해결하기 위하여 유전자알고리즘을 적용한 결과 다음과 같은 결론을 얻을 수 있었다.

1. 소축척 지도제작시 발생하는 편집 오류에 대한 검색 및 처리 프로그램을 개발하므로써 지도제작 오

류문제 해결할 수 있었다.

2. 유전자 알고리즘을 이용한 지도제작 자동화처리 프로그램을 개발함으로써 지도제작 자동화의 가능성을 타진할 수 있었다.

참고문헌

1. Ware, J.M., Wilson, I.D. and Ware, J.A., "Reducing Graphic Conflict in scale Reduced Maps Using A Genetic Algorithm", Proceedings of GIS Research UK 2003 conference, City University, 2003
2. Steven Van Dijk., Dirk Thierens., Mark De Berg., "Using Genetic Algorithms for solving Hard Problems", GeoInformatica Vol 6, Issue 4, 2002, pp. 381~413
3. Ware, J.M., Wilson, I.D. and Ware, J.A., "A knowledge based genetic algorithm approach to automating cartographic generalisation", Proceedings of 22nd SGAI International conference on Knowledge based Systems and applied Artificial Intelligence, Cambridge, 2002. pp. 22~49.