

# 네트워크 문제를 위한 새로운 진화 알고리즘에 대하여

석 상 문\*

## On a New Evolutionary Algorithm for Network Optimization Problems

Sang-Moon Soak\*

### ■ Abstract ■

This paper focuses on algorithms based on the evolution, which is applied to various optimization problems. Especially, among these algorithms based on the evolution, we investigate the simple genetic algorithm based on Darwin's evolution, the Lamarckian algorithm based on Lamarck's evolution and the Baldwin algorithm based on the Baldwin effect and also investigate the difference among them in the biological and engineering aspects. Finally, through this comparison, we suggest a new algorithm to find more various solutions changing the genotype or phenotype search space and show the performance of the proposed method.

Conclusively, the proposed method showed superior performance to the previous method which was applied to the constrained minimum spanning tree problem and known as the best algorithm.

Keyword : Evolutionary Algorithms, The Fixed Charge Transportation Problem, Adaptive Evolutionary Algorithm

## 1. 서 론

일반적으로 진화 알고리즘(Evolutionary Algorithms : EAs)<sup>1)</sup>이라고 하면 유전 알고리즘(Genetic

algorithm), 진화 전략(Evolutionary strategy), 진화 프로그래밍(Evolutionary programming), 유전 프로그래밍(Genetic programming) 등을 가리키는 용어이다. 이들을 다른 알고리즘들과 구분하는 가장 기

본적인 특징은 이러한 알고리즘들이 자연계의 “자연선택(natural selection)”과 “적자생존(the survival of the fittest)”의 개념을 배경으로 하는 알고리즘이라는 것이다[3, 4]. 즉, 개체들 간의 짝짓기를 통해 우수한 형질을 가진 유전자가 다음 세대에 더 많은 후손을 생성한 다음 탐험(exploration)과 탐색(exploitation)을 담당하는 연산자들을 이용해, 우수한 해를 찾아 나간다.

이러한 적자생존이라는 개념은 다윈(Darwin)의 진화론을 바탕으로 한다. 하지만, 이렇게 고안된 진화 알고리즘이 적자생존이라는 개념을 통해 해 집단을 새롭게 구성하면서 우수한 해를 찾아가는 좋은 성능을 지니고 있음에도 불구하고 이러한 메카니즘 때문에 해를 개선 시켜 나갈 때 마다 해 집단에 동일한 해가 기하급수적으로 증가하여 조기 수렴의 위험이 있다는 단점을 지니고 있다. 그리고 Soak et al.[20]은 순회판매원문제에서 진화를 거듭할수록 교차 연산자(crossover operator) 적용 시 동일한 해가 부모해로 선택되는 횟수를 측정하였는데, 실험한 4가지 문제 모두에서 세대수가 200 이상이 되면 약 30% 이상의 교차 연산자 과정에서 동일한 해가 부모해로 선택된다는 것을 확인하였다. 또한 Eckert and Gottlieb[2]는 고정비용수송문제(the fixed charge transportation problem)에서 사용되는 교차 연산자들에 의해 생성된 자손이 해집단에 이미 존재하는지를 나타내는 복제율(duplication rate)을 분석하고 해 집단 내에 동일한 해가 존재하지 않도록 자손을 생성하는 방법을 제안하였다.

조기 수렴의 문제와 함께 또 다른 단점은 우수한 해 주변을 정밀하게 탐색하는 메카니즘이 이러한 진화 알고리즘들에 존재하지 않기 때문에 전역해(global optimum)를 찾아나가는데 한계가 있다는 것이다[3, 4]. 이는 결국 진화 알고리즘의 기본 개념인 적자생존의 개념 및 기타 연산자들이 지니고 있는 단점이다.

최근에는 또 다른 진화 이론의 하나인 라마르크(Lamarck)의 진화론을 바탕으로 하는 새로운 진화 알고리즘들이 제안되고 있다[12, 15]. 라마르크의 이론은 일생동안 획득한 어떤 조직체의 특성들이 그것의 후손들에게 유전될 수 있다는 사실에 기초한다. 하지만, 일반적으로 개별 조직체의 경험이 그 조직체의 유전자형(genotype)을 바꾸는 메카니즘은 현재까지 그 실 예를 찾아 볼 수 없기 때문에 생물학적인 관점에서는 신뢰받지 못하고 있다. 그러나 문화적 진화(cultural evolution)라는 관점에서 리처드 도킨스가 “이기적 유전자”에서 “멤(meme)”이라는 개념을 이용해 라마르크의 이론이 가능하다는 것을 설명한 이후 라마르크의 이론을 공학적으로 응용한 논문들이 등장하기 시작했다[5, 10]. 그리고 Moscato는 이를 Memetic 알고리즘(Memetic Algorithm)이라는 새로운 형태의 알고리즘으로 정형화하였다[12, 13]. 특히 이러한 라마르크의 진화론에 기반하는 알고리즘들은 학습(learning)이라는 개념을 통해 기존의 진화 알고리즘들이 지니고 있던 세밀한 지역탐색의 단점을 해결할 수 있는 특징을 지니고 있다.

또한, 라마르크의 진화론과 유사한 볼드윈 효과(Baldwin Effect)라는 것이 있는데 이는 생물학적으로 일생동안 어떤 개체에 의해 학습된 내용들이 종의 수준(species level)에서 진화 학습을 강화한다는 개념으로, 이 또한 진화 알고리즘 내에서 해의 개선을 위한 목적으로 사용되고 있다[7, 10].

본 논문에서는 기존의 다윈의 진화론에 기반을 둔 위에 언급한 일반적인 진화 알고리즘들(진화 전략, 진화 프로그래밍, 유전 프로그래밍)에 초점을 맞추기 보다는 다양한 진화론에 기인하는 다윈 알고리즘(단순 유전 알고리즘), 라마르크 알고리즘, 볼드윈 알고리즘에 대해 알아보고 이러한 방법들이 공학적으로 어떻게 구현이 되고 어떠한 차이가 있는지 알아본다. 그리고 이러한 알고리즘들이 기존의 연구들에서 어떠한 결과들을 보여 왔는지 알아보고, 이러한 방법들의 비교를 통해 새로운 진화에 기반을 둔 알고리즘을 제안하고 간단한 실험을 통해 제안하는 방법의 유효성을 입증한다.

1) 본 논문에서 사용되는 진화 알고리즘은 단순 유전 알고리즘 또는 순수하게 진화에 기반을 둔 알고리즘이라는 의미로 사용된다.

본 논문의 구성은 다음과 같다. 제 2장에서 다양한 진화론에 기반을 둔 알고리즘들을 생물학적 및 공학적을 알아보고 제 3장에서는 기존의 이러한 진화에 기반을 둔 알고리즘들이 어떠한 문제들에 적용되었으며, 어떠한 성능을 보였는지에 대해 알아본다. 그리고 제 4장에서는 기존의 진화에 기반을 둔 알고리즘들의 비교를 통해 새로운 알고리즘을 제안하고 제 5장에서 실험을 통해 제안하는 알고리즘의 성능을 보이며 제 6장에서 결론을 맺는다.

## 2. Darwinian, Lamarckian과 Baldwinian

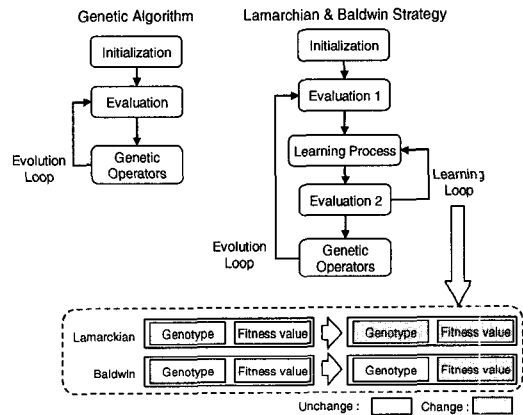
우선 다윈의 진화론과 라마르크의 진화론, 볼드윈 효과가 생물학적으로 어떠한 차이가 있는지 알아보자.

다윈의 진화론은 서론에서 언급한 것처럼 생물학적인 “적자생존”과 “자연선택”이 기반이 된다. 따라서 우수한 형질을 지닌 개체가 더 많은 자손을 다음 세대에 형성함으로써 개체를 번식시켜 나가고, 라마르크의 진화론은 일생동안 어떤 개체에 의해 획득된 특징이 직접적으로 자손에게 전달될 수 있다는 것에 기반을 두고 있다. 즉, 어떤 생물체가 높은 나무 위에 붙어 있는 나뭇잎을 따 먹기 위해 목을 길게 빼는 동작을 반복해서 할 때 이로 인해 다른 생물체에 비해 목이 길어져 더 높이 있는 나뭇잎을 잘 따 먹을 수 있게 되고 이렇게 길어진 목과 관련된 유전인자가 자손에게 직접적으로 전달되어 자손 또한 목이 긴 체로 태어날 수 있다는 것이다. 반면에, 볼드윈 효과는 일생동안 어떤 개체에 의해 획득된 특징이 간접적으로 자손에게 전달될 수 있다는 것에 기반을 둔 것으로 일단 학습된 행동은 결국 본능이 된다는 개념에 기반을 두고 있다.

이렇게 볼드윈 효과는 언뜻 보면 라마르크의 진화론과 서로 유사해 보이지만 차이가 있다. 볼드윈 효과에서는 표현형(phenotype)의 경험에 의한 유전형(genotype)의 직접적인 변형은 이루어지지 않는다는 것이다. 따라서 앞의 예를 기초로 설명하면 라마르크의 진화론에서는 목을 길게 빼는 동작으로 인해

목이 길어지게 한 유전인자(phenotype)가 자손에게 직접적으로 전달되는 반면에 볼드윈 효과에서는 이러한 행동이 유전적인 형질로 전달되는 것이 아니라 단순히 본능으로써 자손의 무의식중에 숨어 있게 된다는 것이다.

이제 위에서 언급한 생물학적인 관점에서의 세 가지 방법, 다윈 진화론, 라마르크 진화론 그리고 볼드윈 효과가 공학적인 관점에서 진화 알고리즘에 어떻게 응용이 되는지 알아보도록 하자.



[그림 1] 유전 알고리즘, 라마르크형 알고리즘 그리고 볼드윈 알고리즘

[그림 1]은 이 세 가지 방법을 비교 설명한 것이다. 다윈의 진화론에 바탕을 둔 진화 알고리즘들 중 대표적인 알고리즘인 단순 유전 알고리즘(simple genetic algorithm)의 경우 간단하게 초기화 단계를 거친 다음 각 해들을 평가를 하게 되고 평가된 적합도 값에 따라, “적자생존”과 “자연선택”이라는 진화론의 기본 개념들을 구현하기 위해 몇 가지 연산자를 사용한다. 이들을 통해서 적자생존의 개념을 구현하여 새로운 해집단을 형성하고 형성된 해집단에 변화를 주기 위해 연산자들을 적용하는 진화과정(evolution loop)을 반복수행하게 된다.

이에 반해 라마르크의 진화론과 볼드윈 효과에 바탕을 둔 진화 알고리즘은 다윈의 진화론에 기반을 둔 유전 알고리즘이 지니고 있는 구성에 새로운 구성이 추가된 형태로 구현이 되는데, [그림 1]에서

라마르크형 알고리즘과 볼드윈 알고리즘을 동일한 그림으로 표현한 것은 실질적으로 이 두 방법이 프로세스상에서의 차이는 없기 때문이다. 즉, 두 방법 모두 유전 알고리즘과 동일한 과정을 포함하고 단순 유전 알고리즘의 평가와 유전 연산자의 수행단계 사이에 각각의 개별 개체(chromosome)들에 학습이라는 개념을 적용하게 된다. 이런 학습이라는 개념은 유전 알고리즘에 사용된 진화(evolution)라는 개념과는 다른 개념으로, 간단하게 진화라는 개념은 해 집단의 여러 해들 간의 유전 정보의 교환을 통해서 이루어지는 것을 의미하며, 학습이라는 개념은 해 집단의 각 개별 개체(유전 알고리즘의 chromosome)가 학습을 통해 더 나은 개체로 변화해 가는 것을 의미한다. 따라서 진화와 학습의 구분은 개체의 변화가 개별 개체 단위에서 이루어지느냐 아니면 해 집단의 단위에서 이루어지느냐에 따라 구분이 된다.

어쨌든 라마르크 알고리즘과 볼드윈 알고리즘은 유전 연산자를 통해서 새롭게 형성된 해들에 학습 과정을 적용하고 학습에 의해 변화된 해들을 평가하는 과정이 추가가 된다. 이렇게 각각의 개별 개체들을 학습 과정을 통해 변화시켜 나가는 과정이 학습 루프(learning loop)이다. 그리고 일반적으로 최적화 문제에서 이러한 학습이라는 과정은 지역탐색알고리즘(local search algorithm)들에 의해 구현이 된다. 따라서 일반적으로 라마르크 전략과 볼드윈 효과를 적용하는 진화 알고리즘의 성능은 학습과 진화를 어떻게 조화를 이루게 하느냐에 달려있다. 다시 말해 만약 학습에 중점을 둔다면 지역해(local optimum)쪽으로는 강한 수렴으로 인해 초기에 적자생존으로 인한 해 집단의 다양성 상실을 초래하게 되고 이는 지역탐색 알고리즘을 반복해서 사용하는 것과 별 차이가 없는 결과를 초래할 수 있으며, 진화에 중점을 두면 지역해로의 수렴 전에 지역탐색이 끝남으로 해서 기존의 진화 알고리즘을 단독으로 적용하는 것과 성능에서의 별 차이가 없을 수도 있다. 사실 후자는 단순 유전 알고리즘이 지역탐색을 위한 메카니즘을 가지고 있지 않기 때문에 생기는

단점으로 이를 보완하기 위해 학습 알고리즘 즉, 지역 탐색 알고리즘을 사용하는 형태로 연구자의 관심을 끌어난 이유이기도 하다.

라마르크 알고리즘과 볼드윈 알고리즘의 차이는 이러한 학습과정이 끝난 이후에 발생하게 되는데, 만약 각 개별 개체에 대한 반복적인 학습과정이 끝난 이후에 학습과정 중 가장 우수했던 해를 가지고 해당 개체 및 해당 개체의 적합도 값을 모두 대체한다면 이는 라마르크 알고리즘이 되고 만약 개체는 그대로 두고 적합도 값만 대체한다면 이는 볼드윈 알고리즘이 된다. [그림 1]의 아래 그림은 학습을 통해 변화하는 부분의 차이를 보여준다.

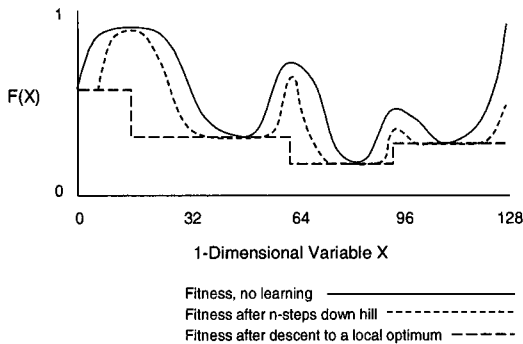
이러한 두 전략이 가지는 특징은 최적화 알고리즘의 측면에서 보면 각 개체에 지역 탐색 알고리즘을 적용하여 지역탐색을 통해 찾은 근사해들 또는 근사해가 가지는 적합도 값을 가지고 해 집단을 형성하기 때문에 대체로 수렴속도가 아주 빠르다는 것이다. 이는 단순히 컴퓨터에 의해 계산되는 계산시간을 의미하는 것이 아니라 적은 횟수의 진화과정을 통해서도 빨리 진화를 한다는 것이다. 이는 이미 Hinton and Nowlan의 논문에서도 언급한 것처럼 학습이 진화를 촉진시킬 수도 있음을 의미한다[7].

### 3. 기존 연구 결과들

본 장에서는 각 알고리즘들이 기존의 연구들에서 어떠한 성과와 결과를 보여주었는지에 대해 알아본다.

우선 각 알고리즘들에 대한 기존의 연구 결과들을 알아보기 전에 학습이라는 과정이 탐색에 어떠한 영향을 미치는지 알아보도록 하자. 학습의 효과에 대한 설명은 Gruau and Whitley[6]의 논문에 잘 설명이 되어있다. [그림 2]는 그들의 논문에서 인용한 것으로, 이는 1차원 함수의 적합도 값이 지역탐색법 즉, 학습 과정을 통해서 더 효과적으로 탐색될 수 있음을 보여주는 좋은 예이다.

그림에서 확인할 수 있는 바와 같이 최소화 문제에서 학습과정을 수행하지 않을 경우와  $N$ -step 학습과정을 수행하는 경우 그리고 완전한 지역탐색



[그림 2] 1차원 함수의 적합도 탐색공간(fitness landscape)상의 학습의 효과. 최소화문제의 경우 해의 개선은 적합도 탐색공간의 아래쪽 방향으로 이동하게 된다. 이 그림은 학습 과정을 가지고 있지 않은 경우와  $N$ -단계 지역 탐색 방법 및 지역 최적해까지 완전 탐색(full descent)을 수행한 경우의 적합도 탐색공간 비교를 보여준다[6].

(full-step)을 수행하는 경우의 결과를 보여 주고 있다. 만약 어떤 문제에서 이웃해를 모두 탐색하는 학습과정을 수행하게 된다면 함수의 변곡점에서 적합도 함수의 값이 바뀌게 되고 결국에 이는 그림에서 점선으로 나타난 것처럼 적합도(fitness) 탐색 공간을 바꾸는 효과를 가지게 되고 변경된 탐색 공간은 원래의 탐색 공간에 비해 훨씬 단순화되어 있음을 확인할 수 있다. 결국 이를 통해서 알 수 있는 것처럼 라마크 알고리즘과 볼드윈 알고리즘처럼 학습과

정을 적용할 경우, 문제를 보다 빨리 또는 쉽게 해결할 수 있는 가능성을 보여주고 있다. 물론 모든 문제에서 이러한 학습과정이 진화에 도움을 주는 것은 아니다. 다음에서는 각 알고리즘들이 그동안 어떠한 문제들에서 어떤 성능을 보여왔는지에 대해 알아보도록 하자.

<표 1>은 기존 연구 결과들을 정리한 것으로 이를 통해 알 수 있는 분명한 사실은 Hinton and Nowlan의 연구를 비롯하여 다양한 연구들에서 보여준 것처럼 학습 과정을 진화 과정과 함께 사용하는 것이 도움이 될 수도 있다는 것이다. 물론 학습 과정 또한 Turney[16]와 Mayley[11]가 지적한 것처럼 많은 계산 시간을 필요로 한다든지, 지역해에 빠져 최적해(global optimum)를 찾아가지 못하고 조기 수렴을 한다든지 또는 유전형과 표현형이 높은 상관관계를 가지지 않을 경우 오히려 학습이라는 개념이 진화의 방향성을 잃게 만들어 임의 탐색의 결과를 초래할 수도 있다는 등과 같은 단점을 지니고 있지만 기존의 실험 결과들을 통해 확인할 수 있는 것처럼 성능적인 측면에서는 더 우수한 결과를 보여 주고 있다.

또한, 앞에서 볼드윈 전략을 사용하는 경우가 적합도 공간(fitness landscape)을 바꿈으로써 해서 훨씬 문제를 쉽게 해결할 가능성이 있다는 것을 기존의 연구를 바탕으로 설명을 하였다[7]. 이는 또 다른 변형에 대한 가능성의 여지를 제공한다.

<표 1> 기존의 연구 결과들

	Applications	Performance
Gruau and Whitley[6]	Boolean 뉴럴 네트워크	볼드윈 > 라마크 > 단순 유전
Whitley et al.[20]	함수 최적화	볼드윈 > 라마크 ≈ 단순 유전
Houck et al.[9]	<ul style="list-style-type: none"> <li>◦ 함수최적화,</li> <li>◦ 위치 재할당(location-allocation problem)</li> <li>◦ 생산 셀 구성 문제(the manufacturing cell formation)</li> </ul>	라마르크 > 볼드윈 > 단순 유전
Julstrom[10]	4-cycle 문제	라마르크 > 단순 유전 > 볼드윈
Ross[15]	순회판매원 문제	라마르크 > 단순 유전
Ong and Keane(2004)	함수 최적화 문제	라마르크 > 단순 유전

그럼 다른 변형은 가능하지 않을까? 일반적인 유전 알고리즘의 경우 유전형 탐색공간(genotype search space), 표현형 탐색공간(phenotype search space) 및 적합도 탐색공간(fitness search space)과 같은 세가지 탐색공간을 탐색한다. 그 중 볼드윈 알고리즘은 적합도 공간(fitness landscape)을 변형시키고 라마크 알고리즘은 표현형 공간(phenotype landscape)과 적합도 공간(fitness landscape)을 모두 변형시킴으로써 더 우수한 성능을 보여준다. 그럼 유전형 공간(genotype landscape) 또는 표현형 공간(phenotype landscape) 만을 바꿀 수는 없을까? 그리고 만약 이러한 알고리즘이 있다면 볼드윈 전략을 사용하는 방법처럼 탐색공간을 더 부드럽게 만듬으로써 또는 탐색공간을 바꿈으로써 더 효율적으로 해를 찾을 수 있지 않을까?

따라서 다음 장에서는 유전형 공간(genotype landscape) 또는 표현형 공간(phenotype landscape)을 바꿀 수 있을 뿐만 아니라 이미 찾은 해를 다시 찾는 불필요한 과정을 해소하고 이러한 과정을 통해 보다 더 많은 영역의 탐색이 가능하며 항상 우수한 해가 더 많이 해를 생성함으로써 발생하는 조기 수렴의 문제를 보완할 수 있는 새로운 진화 알고리즘을 제안한다. 이러한 생각의 출발은 앞에서 설명한 세 가지 방법들을 좀 더 구체적으로 비교하면서 시작된다.

## 4. 새로운 진화 알고리즘

### 4.1 기존 알고리즘들의 비교

본 장에서는 앞에서 살펴본 세 가지 방법들을 좀 더 구체적으로 비교해 보면서 새로운 방법의 가능성에 대해 알아보도록 한다.

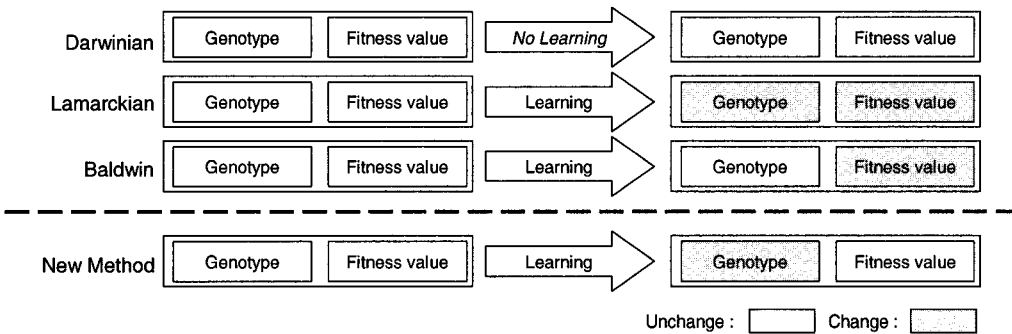
[그림 1]에서 세 방법의 차이를 확인할 수 있었다. 여기서 이 모든 전략들의 공통점은 적자생존의 개념에는 변함이 없다는 것이다. 하지만 기존의 진화에 기반을 둔 알고리즘들에서 선택전략에 의해 구현되어 온 적자생존의 개념은 실질적인 생물학적

진화에는 다양한 변화의 가능성이 존재함에도 불구하고 일정한 확률로 적자생존의 개념을 구현함으로써 어떻게 보면 생물학적인 진화에 어울리지 않는다. 물론 이러한 문제를 해결하기 위해 토너먼트 선택전략과 같이 기존에 다양한 선택전략들이 제안되어 왔다. 하지만 본 논문에서는 선택전략에 변화를 주는 것이 아니라 해의 평가와 선택전략 사이에 바뀐 해에 대한 평가과정을 수행하지 않는 학습과정을 돕으로써 선택전략에 의해 새롭게 생성된 해에 일정한 노이즈가 포함되도록 하는 방식을 사용하여 이러한 문제를 해결한다.

[그림 3]은 앞서 살펴본 다윈 알고리즘, 라마크 알고리즘과 볼드윈 알고리즘이 각각 학습이라는 과정이 적용된 이후에 해 표현법과 적합도 함수값에 어떠한 변화를 초래 하는지를 보여주고 있다. 다윈 알고리즘의 경우는 학습과정이 없기 때문에 아무 변화도 일으키지 않는 반면, 라마크 알고리즘과 볼드윈 알고리즘의 경우에는 학습과정이 각각 해 표현법과 적합도 값에 영향을 미친다. [그림 3]에서 알 수 있듯이 기존의 방법들을 통해서 가능한 또 하나의 방법이 존재한다. 즉, 학습을 각 해들에 적용한 이후 적합도 함수값에는 변화를 주지 않고 단순히 해 표현(genotype 일수도 있고 phenotype 일수도 있음)에만 변화를 일으키는 것이다. 이는 2가지 측면에서 기존의 방법과 다른 특징을 가진다.

첫째, 볼드윈 알고리즘이 적합도 공간(fitness landscape)을 바꾸는 것과 유사하게 유전형 또는 표현형 공간(genotype or phenotype landscape)을 바꾸는 효과를 지닌다.

둘째, 학습과정 이후에 일어나는 진화단계에서 선택전략(selection strategy)에 의해 구현되는 적자생존의 개념에 불확실성을 부여함으로써 해서 보다 다양한 해 공간의 탐색이 가능할 뿐만 아니라 조기 수렴의 위험을 방지할 수 있다. 즉, 일반적으로  $n+1$  세대의 해 집단은  $n$  세대의 해의 적합도 값에 의해 결정이 된다. 하지만 제안하는 방법의 경우  $n$  세대의 해의 적합도 값에는 변화가 없이 유전형 또는 표현형에 변화를 줌으로써 실질적인  $n+1$  세대의



[그림 3] 학습 과정을 사용하는 새로운 진화 알고리즘

해 집단을 구성하는 해의 분포가 달라지게 된다. 즉, n세대에서 가장 우수했던 해가 유전형 또는 표현형의 변화로 인해 n+1세대의 구성단계에서 가장 좋은 적합도 값을 가지지 않을 수도 있다는 것이다. 하지만 이렇게 변형된 해는 여전히 우수한 특징들을 보유하고 있기 때문에 오히려 약간의 변형을 통해 항상 우수한 해에 의해 해집단이 빠르게 지배되는 것을 방지할 수 있을 뿐만 아니라 이미 찾은 해를 다시 찾을 필요 없이 해 주변의 탐색을 가능하게 하는 효과를 보일 수 있다.

그리고 위에서 언급한 특징을 가진 알고리즘이 고려해야 할 또 다른 점은 기존의 학습에 기반을 둔 알고리즘들이 과도한 학습단계로 인해 많은 계산 시간을 필요로 했다는 것이다. 즉, 이는 각각의 학습단계 이후에 변화된 해를 평가하는 과정에서 발생하는 것으로 이렇게 소요된 계산시간에 비해 개선된 해의 수준은 그렇게 만족할 만한 수준이 아니었다.

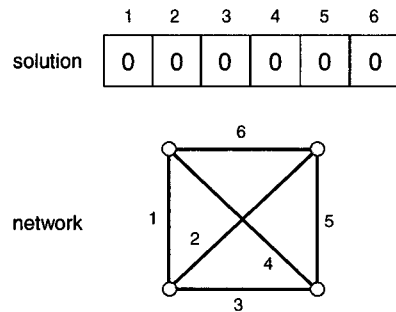
하지만, [그림 3]을 보면 새로운 방법에서는 그럴 필요가 없다. 왜냐하면, 학습단계를 한번 수행하든 여러 번 수행하든 유전형 공간(genotype landscape) 또는 표현형 공간(phenotype landscape)을 바꾸는 효과는 동일하기 때문이다. 오히려 여러 번의 학습 과정을 수행할 경우 과도한 탐색 공간(landscape) 변형으로 인해 학습과정이 진화과정을 왜곡시킬 수 있다[6, 11, 19, 20]. 따라서 새로운 방법에서는 학습 과정이 진화과정을 왜곡하지 않는 수준에서 학습을

수행함이 옳을 것이다.

아무튼, 이러한 방식으로 알고리즘이 구현된다면, 앞에서 언급한 적자생존으로 인한 기존의 방법들의 단점을 보완할 수 있을 뿐만 아니라 기존의 학습이라는 개념을 사용하는 방법들에 비해 과도한 학습 시간으로 인한 계산 시간을 줄이는 효과를 기대할 수 있을 것이다. 다음에서는 위에서 제시한 특징들을 지니는 한 가지 방법을 제안한다.

#### 4.2 제안하는 알고리즘

제안하는 알고리즘을 좀 더 단순화하고 구체화하기 위해 네트워크 문제에 한정하여 개발하였다.



[그림 4] 사용된 해 표현법과 네트워크와의 관계

여기서 네트워크의 각 에지 상의 번호는 해 표현법의 각각의 유전인자(gene)상에 있는 번호와 일치한다.

우선 제안하는 알고리즘은 기존의 진화 알고리즘들과 해 표현법 자체가 아주 다르다. 각각의 유전인자(gene)는 네트워크 상에 존재하는 모든 가능한 에지들에 각각 할당되고, 초기에 모든 유전인자(gene)의 값이 0으로 초기화된다. [그림 4]는 사용된 해 표현법과 네트워크와의 관계를 보여주고 있다. 이러한 각각의 유전인자의 값은 진화 과정 동안에 해당 유전인자가 의미하는 에지가 우수한 형질을 보유하고 있는지 아닌지에 대한 측도를 나타내게 된다. 그리고 제안하는 알고리즘은 기존의 라마크 알고리즘과 볼드윈 알고리즘처럼 학습이라는 방법을 알고리즘 내에 포함한다. 하지만, 앞에서 지적한 것처럼 기존의 두 방법이 이러한 학습 과정 때문에 과도한 계산시간을 필요로 한다는 단점을 지니고 있기 때문에 이를 줄이기 위해 각 해들에 단 한 번의 학습과정을 수행한다. 이는 또한 진화의 과정을 왜곡시키지 않는 범위에서 행할 수 있는 최소한의 방법일 수 있다. 또한 이러한 효과가 결국 앞에서 지적한 진화 알고리즘들의 적자생존 개념으로 인해 해 집단의 다양성을 급속하게 잃어가는 것을 방지하고 또한 이미 한번 찾았던 해를 다시 찾을 필요가 없어 이미 찾은 해 주변의 해를 탐색할 수 있도록 학습과정을 구현하였다. 이렇게 구현된 학습과정은 제안하는 방법에서  $\alpha$ 와  $\beta$ 과정으로 표현이 된다. 결국 이러한  $\alpha$ 와  $\beta$ 과정은 각 해에 대해 벌(penalty) 또는 상(reward)을 주는 의미를 지닌다.

이와 같이 구현된 제안하는 알고리즘의 절차는 [그림 5]와 같다. 좀 더 구체적으로 [그림 5]에 나타난 제안하는 방법을 설명하면,  $\alpha$ 과정에서 현재까지 찾았던 가장 우수한 해에 포함된 에지들에 벌을 주는 것은 만약 상을 주게 될 경우 해의 변화가 없기 때문에 다음 진화과정에서도 동일한 해가 가장 우수한 해로 선택된 확률이 높을 뿐만 아니라 우수한 해에 상을 줌으로써 해 집단이 이 해로 인해 급격하게 지배될 것이기 때문이다. 하지만, 벌을 줌으로써 비록 우수한 형질을 가진 유전인자(즉, 에지)이지만 근소한 값으로 인해 표현형(phenotype)에 포함되지 못한 유전인자들과 비록 표현형에 포

함되어 있지만 포함되지 못한 유전인자와 근소한 차이밖에 나지 않는 에지들 간에 경쟁을 유발할 수 있게 된다.  $\beta$ 과정 또한  $\alpha$ 과정과 동일한 특성을 가진다. 하지만  $\beta$ 과정에서는 모든 에지를 대상으로 하는 것이 아니라  $\theta$ 값을 통해 일부의 에지만을 고려함으로써 매 단계에서 동일한 에지들 간의 경쟁 관계를 유발시키는 것이 아니라 여러 가능성을 고려할 수 있도록 구현하였다.

<p>1. 초기화 단계 모든 유전인자 <math>gene[i][j]=0</math>으로 초기화.</p> <p>2. 네트워크 생성 및 평가 네트워크를 형성하는 에지 집합 <math>T \leftarrow \phi</math> ; <math>A \leftarrow E</math> (<math>E</math> : 모든 가능한 에지) ; for <math>i = 1</math> to <math>pop\_size</math> do</p>
<p>2.1 네트워크 생성 while 네트워크가 형성되지 않으면 do 가장 낮은 <math>gene[i][j]</math> 값을 가지는 에지 <math>j</math> 를 선택 ; <math>A \leftarrow A - j</math> ; if <math>j</math>가 네트워크에 포함이 가능하면 then <math>T \leftarrow T \cup j</math> ; return <math>T</math> ;</p>
<p>2.2 방정식 (1)을 이용해서 적합도 평가 if <math>Best \geq Sol[i]</math> then <b>/** <math>\alpha</math>process **/**</b> <math>Best \leftarrow Sol[i]</math> ; for <math>j = 1</math> to <math>L</math> do if <math>j \in T</math> then <math>gene[i][j] \leftarrow gene[i][j] + \alpha</math> ; else <b>/** <math>\beta</math> process **/**</b> for <math>j = 1</math> to <math>L</math> do <math>r = random\{0, 100\}</math> ; if <math>((r &lt; \theta) \ \&amp;\&amp; \ (j \in T))</math> then <math>gene[i][j] \leftarrow gene[i][j] - \beta</math> ;</p>
<p>3. 선택과정(Selection) 4. 교차연산자(Crossover operator) 5. 변이연산자(Mutation operator) 6. 종료조건을 만족하면 종료, 그렇지 않으면 2단계로.</p>
<p><math>pop\_size</math> : 해집단의 크기, <math>L</math> : 해의 길이</p>

[그림 5] 제안하는 알고리즘의 절차

그리고 제안하는 방법은 다양한 네트워크 문제에 적용이 가능하다. 예를 들어 [그림 5]에서 2.1의 네



트위크 형성 단계를 다루는 문제의 특성에 맞게 수정하면 다양한 네트워크 문제에 사용이 가능한 특징을 가진다. 본 논문에서는 걸침나무를 대상으로 하고 있기 때문에 걸침나무를 형성하는 방법을 간단히 설명하면, 우선 [그림 4]에서 처럼 다루는 네트워크 문제에서 모든 가능한 에지들을 각각의 유전인자로 표현을 한다. 그리고 각각의 유전인자들을 유전인자의 값에 따라 오름차순으로 정렬하고, 정렬된 유전인자를 순서대로 트리에 포함될 수 있는지를 고려한다. 만약 고려되는 유전인자 즉, 에지가 트리에 포함될 수 있다면(사이클을 형성하지 않는다면) 트리( $T$ )에 포함시키고 그렇지 않을 경우 고려중인 유전인자를 건너뛰고 다음 유전인자를 고려한다. 결국 이러한 방식으로 모든 가능한 유전인자를 고려하면 최종적으로 트리를 얻을 수 있다. 하지만 트리문제의 경우  $n$ 개의 노드가 있을 경우 모든 노드를 연결하는 트리는  $n-1$ 개의 에지로 구성되기 때문에 모든 유전인자 즉, 에지를 고려함이 없이 단지  $n-1$ 개의 에지가 모이면 결국 트리를 형성하게 된다.

만약 다른 네트워크 문제를 다루고자 한다면 동일한 방법으로 2.1 부분을 다루는 문제에 맞게 표현을 하면 간단하게 적용이 가능하다.

## 5. 실험

### 5.1 Quadratic 최소걸침나무문제(Quadratic Minimum Spanning Tree Problem : QMST)

Quadratic 최소걸침나무문제(QMST)는 Assad and Xu[1]에 의해 처음으로 소개된 문제로 전통적인 최소걸침나무문제의 변형이다. QMST는 네트워크에 존재하는 에지들의 쌍이나 그룹의 비용에 대한 Quadratic 함수를 비용함수로 가지는 환경에서 최소걸침나무를 탐색하는 문제이다. QMST 문제는 이미 Assad and Xu에 의해 NP-hard 문제로 증명되었으며, 따라서 기존의 Prim 알고리즘이나 Kruskal 알고리즘과 같이 단순한 최소걸침나무문제를 위한

알고리즘들을 이용해서 해결할 수 없는 문제로 다음과 같이 정형화된다.

$$\text{Min } f(x) = \left[ \sum_{i=1}^M \sum_{j=1, j \neq i}^M a_{ij} x_i x_j + \sum_{i=1}^M b_i x_i \mid x \in T \right]$$

여기서,  $a_{ij}$ 는 에지  $i$ 와 에지  $j$ 사이의 상호비용(intercoast) 즉, 두 에지가 모두 트리에 존재할 경우 발생하게 되는 비용이고,  $x_i$ 는 의사 결정 변수로 만약, 에지  $i$ 가 트리에 존재하면 1의 값을 가지고 그렇지 않으면 0의 값을 가진다. 그리고  $b_i$ 는 에지  $i$ 가 지니고 있는 비용값(cost)를 의미한다. 따라서, 결국 QMST 문제는 기존의 단순 최소걸침나무문제에 에지들 상호간의 상호비용이 존재하는 경우를 고려하는 형식으로 구현된다.

Assad와 Xu는 QMST를 해결하기 위한 3가지 휴리스틱 알고리즘을 제안하였지만, 그 후 Zhou and Gen[22]은 Prüfer수를 사용하는 진화 알고리즘이 Assad와 Xu가 제안한 3가지 휴리스틱 알고리즘보다 우수한 성능을 보여준다는 것을 실험을 통해 입증을 하였다. 따라서, 본 논문에서는 제안하는 알고리즘의 유효성을 보이기 위해 QMST문제에서 Prüfer수를 사용하는 진화 알고리즘과의 비교 실험을 수행한다. 그리고 Prüfer수를 사용하는 진화 알고리즘에 대한 좀 더 구체적인 내용은 Zhou and Gen의 논문 [22]을 참조하기 바란다.

### 5.2 비교 실험

본 논문에서는 비교실험을 위해 4가지 문제를 생성하였다. 각 문제들에서 비용 매트릭스는  $500 \times 500$  좌표 상에서 임의로 생성한 노드들 상호간의 거리를 이용하였으며, 상호비용 매트릭스는  $[1, 20]$  범위 내에서 임의로 생성하였다. 또한 QMST 문제의 경우 최적해를 찾는 방법이 현재까지 알려져 있지 않기 때문에 각 문제들에서 찾은 가장 우수한 해를 기준으로 각 알고리즘의 성능을 평가하였다. 따라서 <표 3>에 나와 있는 값들은 다음의 식을 이용해서 계산되었다.

$$\text{Gain} = (\text{Fitness} - \text{Best Solution}) / \text{Best Solution} \\ \times 100 (\%)$$

여기서, Fitness는 각 알고리즘들이 찾아낸 해를 의미하며, Best Solution은 각 알고리즘들이 찾아낸 해들 중에서 가장 우수한 해를 의미한다.

〈표 2〉 실험 환경

	New Algorithm	Previous Algorithms
Population size	100	100
Crossover	Uniform	Uniform
Crossover Rate(%)	60	60
Mutation	Swap	Random Perturbation
Mutation Rate(%)	60	60
Termination(generation)	1000	1000

또한 QMST 문제의 경우 현재까지 효율적인 지역 탐색 알고리즘이 알려져 있지 않다. 따라서 Prüfer 수를 사용하여 구현된 볼드윈 알고리즘과 라마크 알고리즘의 학습과정을 위해 진화 알고리즘의 변이 연산자(random perturbation)를 반복 실행하는 방법을 사용하였으며, 학습횟수는 QMST50과

QMST60에서는 각각 노드 개수 즉, 50과 60회씩으로 하였으며, QMST90과 QMST100에서는 과도한 계산시간의 소요 때문에 50회로 설정하여 실험을 수행하였다. <표 2>는 본 논문에서 사용된 각 알고리즘의 파라메타 값들을 보여준다. 그리고 새로운 알고리즘의 경우  $\alpha$ 와  $\beta$ 는 각각 1을 사용하였고,  $\theta$ 는 60%를 사용하였다.

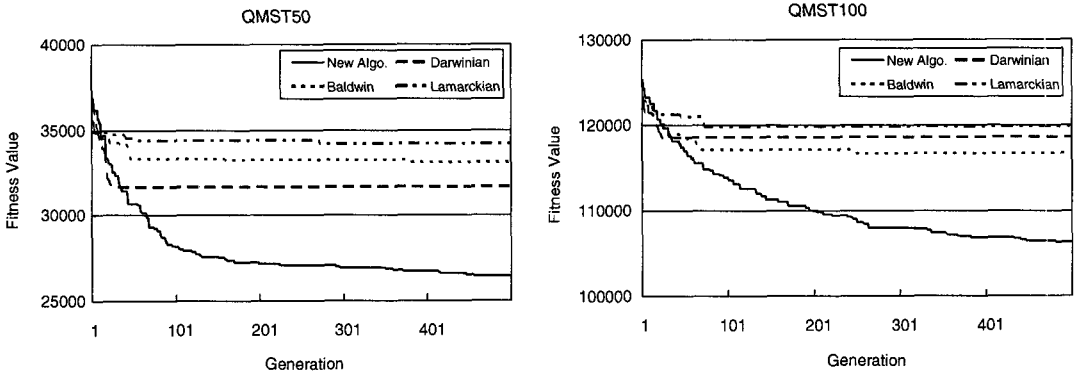
<표 3>은 각 알고리즘들을 30회씩 수행한 결과를 보여준다. 네 가지 실험 문제 모두에서 제안하는 알고리즘이 가장 우수한 해를 찾았으며, 평균 성능에서도 가장 우수한 해와 비교해서 2.5%를 넘지 않는 범위 내에 있음을 확인할 수 있다. 또한 가장 열등한 해(Max)의 비교에서도 다른 세 가지 알고리즘들에 비해 월등히 우수한 결과를 보여준다는 것을 확인할 수 있다. 뿐만아니라 표준편차(STD)에서도 다윈 알고리즘과 볼드윈 알고리즘보다 우수한 결과를 보여주고 있으며, 라마크 알고리즘과 비교해 볼 때 많은 차이가 없음을 확인할 수 있다. 이는 라마크 알고리즘이 가지는 반복해서 우수한 해를 찾아나가는 학습과정의 특성 때문이다. 그리고 제안하는 알고리즘을 제외한 다른 세 가지 알고리즘들 간의 비교에서는 볼드윈 알고리즘이 가장 우수한 결과를 보여주었으며, 다음은 라마크 알고리즘,

〈표 3〉 실험 결과 비교

	New Algorithm					Darwinian Algorithm				
	Min	Avg	Max	STD	CPU	Min	Avg	Max	STD	CPU
QMST50	0	<b>2.47</b>	4.80	1.05	44.7	21.05	30.23	35.82	3.21	19.03
QMST60	0	<b>1.96</b>	3.78	0.87	65.5	19.38	23.61	27.09	2.00	33.9
QMST90	0	<b>0.65</b>	1.55	0.46	174.3	15.20	17.57	19.58	1.23	139.7
QMST100	0	<b>0.80</b>	2.00	0.53	218.4	13.70	15.33	16.85	1.05	186.1

	Baldwinian Algorithm					Lamarckian Algorithm				
	Min	Avg	Max	STD	CPU	Min	Avg	Max	STD	CPU
QMST50	24.70	27.37	29.64	1.39	1300.2	27.79	29.03	30.32	0.70	1379.8
QMST60	17.83	20.67	22.78	1.22	2659.2	20.50	22.22	23.18	0.64	2890.0
QMST90	13.84	14.75	15.82	0.76	7652	16.50	17.27	17.80	0.38	8873.4
QMST100	12.01	13.17	14.22	0.63	10525.9	14.96	15.54	16.00	0.34	12201.4



[그림 6] 각 알고리즘들의 수렴과정

다윈 알고리즘의 순이었다. 계산시간 측면에서는 비록 제안하는 알고리즘의 인코딩 방법 및 학습과정을 사용하는 이유로 다윈 알고리즘에 비해 많은 계산시간을 필요로 하였으나, 볼드윈 알고리즘과 라마르크 알고리즘에 비해서는 월등히 적은 계산시간을 소요하였다. 이는 앞에서도 언급한 바와 같이 이 두 알고리즘은 다수의 학습과정을 사용하고 일반적으로 이러한 알고리즘들에서 소요되는 계산시간의 대부분이 학습과정 동안 계산하는 적합도 함수 평가에서 소요되기 때문이다. 반면에 제안하는 알고리즘은 찾아낸 해의 특징을 고려하여 한 번의 학습과정을 사용할 뿐만 아니라 이렇게 변형된 해에 대한 적합도 값을 다시 계산하지 않기 때문에 이 두 알고리즘에 비해 적은 계산시간이 소요되었다.

[그림 6]은 각 알고리즘들 중에서 가장 우수한 성능을 보인 결과의 수렴과정을 보여주고 있다. 그림에서 확인할 수 있는 바와 같이 제안하는 알고리즘의 경우 진화과정(generation)이 반복됨에 따라 해 또한 계속해서 개선되어 나가는 반면에 다른 세 가지 알고리즘의 경우는 초기에 해의 개선을 보였을 뿐 그 이후로는 해를 개선 시켜 나가지 못하고 있음을 확인할 수 있다.

결론적으로 제안하는 새로운 알고리즘이 다른 기존의 알고리즘들보다 더욱 우수한 성능을 보여 줄 뿐만 아니라 학습과정을 사용하는 다른 두 알고리즘에 비해서 계산시간도 상당히 빠름을 확인할 수

있다. 특히 기존에 QMST 문제에서 우수한 성능을 주는 것으로 알려진 Prüfer수를 사용하는 진화 알고리즘에 비해서도 월등히 우수한 결과를 준다는 것은 고무적인 사실이다.

## 6. 결 론

본 논문에서는 기존의 진화에 기반을 둔 세 가지 알고리즘, 다윈 알고리즘(단순 유전 알고리즘), 라마르크 알고리즘 및 볼드윈 알고리즘이 지니고 있는 특징을 생물학적 및 공학적으로 알아보고 상호간의 비교를 통해서 새로운 진화에 기반을 둔 알고리즘을 제안하였다.

특히, 기존 방법들은 모두 동일한 적자 생존의 개념을 보유함으로써 해서 조기 수렴이나 지역하로의 수렴의 우려가 있음을 지적하였으며 또한 기존의 진화에 학습과정을 추가한 라마르크 알고리즘과 볼드윈 알고리즘이 학습으로 인해 비록 우수한 성능을 보일지라도 과도한 계산 시간을 필요로 한다는 단점이 있음을 지적하였다.

제안하는 새로운 방법은 이러한 기존의 방법들의 단점을 보완하는 형식으로 구현이 되었으며, QMST 문제에 적용하여 기존의 우수한 성능을 보인 Prüfer수를 사용하는 유전 알고리즘, 라마르크 알고리즘 및 볼드윈 알고리즘보다 더 우수한 성능 및 조기 수렴 없이 진화과정과 함께 해를 개선시켜 나간다는 것

을 수렴과정을 통해 보여주었다. 또한 제안하는 방법은 학습을 이용하는 기존 방법과 달리 단 한 번의 학습과정을 수행하고 이에 대한 평가 단계를 거치지 않음으로 해서 라마크 및 볼드윈 알고리즘에 비해 상당한 계산시간을 절약할 수 있다는 것도 보여주었다.

추후 과제로는 제안하는 알고리즘을 보다 정교하게 만들 필요가 있으며 다양한 파라메타 값들에 대한 분석 및 다양한 문제에 적용하여 제안하는 방법의 보편성을 보일 필요가 있다고 판단된다.

## 참 고 문 헌

- [1] Assad, A. and W. Xu, "The Quadratic Minimum Spanning Tree Problem," *Naval Research Logistics*, Vol.39(1992), pp.399-417.
- [2] Eckert, C. and J. Gottlieb, "Direct Representation and Variation Operators for the Fixed Charge Transportation Problem," *Lecture Notes in Computer Science*, Vol. 2439(2002), pp.77-87.
- [3] Gen, M. and R. Chen, *Genetic Algorithms and Engineering Design*, Wiley, 1997.
- [4] Goldberg, D., *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, MA, 1989.
- [5] Grefenstette, J.J., "Lamarckian Learning in Multi-agent Environments," *In Proceedings of 4th International Conference of Genetic Algorithms*, (1991), pp.303-310.
- [6] Gruau, F. and D. Whitley, "Adding learning to the cellular development of neural networks : evolution and the Baldwin effect," *Evolutionary Computation*, Vol.1, No.3(1993), pp.213-233.
- [7] Hinton, G.E and S.J. Nowlan, "How Learning Can Guide Evolution," *Complex Systems*, Vol.1(1997), pp.495-502.
- [8] Holland, J.H., *Adaption in Natural and Artificial Systems*, MIT Press, 1992.
- [9] Houck, C.R., J.A. Joines, and M.G. Kay, "Utilizing lamarckian evolution and the Baldwin effect in hybrid genetic algorithms," *Meta-Heuristic Res. Appl. Group, Dept. Ind. Eng., North Carolina State Univ., NCSU-IE Tech. Rep. 96-1*, 1996.
- [10] Julstroml, B.A., "Comparing darwinian, baldwinian, and lamarckian search in a genetic algorithm for the 4-cycle problem," *Late Breaking Paper at the 1997 Genetic and Evolutionary Computation Conference*, 1999.
- [11] Mayley, G., "Landscapes, Learning Costs and Genetic Assimilation," *Evolutionary Computation*, Vol.4, No.3(1996), pp.213-234.
- [12] Merz, P and B. Freisleben, "Fitness Landscapes, Memetic Algorithms, and Greedy Operators for Graph Bipartitioning," *Evolutionary Computation*, Vol.8, No.1(2000), pp. 61-91.
- [13] Moscato, P., "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts Toward memetic Algorithms," *Tech. Rep. No.790, California Institute of Technology*, 1989.
- [14] Raidl, G.R. and J. Gottlieb, "Empirical Analysis of Locality, Heritability and Heuristic Bias in Evolutionary Algorithms : A Case Study for the Multidimensional Knapsack Problem," *Evolutionary Computation*, Vol.13, No.4(2005), pp.441-475.
- [15] Ross, B.J., "A Lamarckian Evolution Strategy for Genetic Algorithms," in *Practical Handbook of Genetic Algorithms : Complex Coding Systems*, 1999.
- [16] Turney, P., "Myths and legends of the

- Baldwin Effect," *In proceedings of the Workshop on Evolutionary Computing and Machine Learning at the 13th International Conference on Machine Learning*, (1996), pp.135-142.
- [17] Sendhoff, B., M. Kreutz, and W.V. Seelen, "A condition for the genotype-phenotype mapping : Casualty," *Proceedings of the Seventh International Conference on Genetic Algorithms*, Morgan Kaufman, (1997), pp.73-80.
- [18] Soak, S.M., H.G. Lee, and S.C. Byun, "Subtour Preservation Crossover Operator for the Symmetric TSP," *Journal of the Korea Institute of Industrial Engineers*, accepted, 2006.
- [19] Whitley, D., "A genetic algorithm tutorial," *Statics and Computing*, Vol.4(1994), pp.65-85.
- [20] Whitley, D., V.S. Gordon, and K.E. Mathias, "Lamarckian Evolution, The Baldwin Effect and Function Optimization," *Parallel Problem Solving from Nature III*, (1994), pp.6-15.
- [21] Xu, W., "On the quadratic minimum spanning tree problem," *Proc. of the Third Int. Conf. on Genetic algorithms*, ed. J. Schaffer. Morgan Kaufmann Publishers, San Mateo, California, (1995), pp.141-148.
- [22] Zhou, G. and M. Gen, "An Effective GA Approach to The Quadratic Minimum Spanning Tree Problem," *Computers & Operations Research*, Vol.25, No.3(1998), pp.229-237.