

論文97-34C-5-7

병렬 CES를 이용한 QAP 해법

(Solution of Quadratic Assignment Problem Using Parallel Combinatorial Variant of Evolution Strategy)

朴來正*, 李賢**, 朴哲薰*

(Lae-Jeong Park, Hyun Lee, and Cheol Hoon Park)

요 약

본 논문에서는 잘 알려진 조합 최적화 문제인 배치 최적화 문제(QAP)를 해결하기 위한 Evolution Strategy의 조합용 변형(CES)의 병렬화를 제안한다. PCES는 분산 유전자 집단을 통하여 국소 최적화에 빠질 가능성을 줄임으로써, CES보다 더 효과적이다. benchmark 문제에서의 실험을 통하여 PCES가 CES와 유전 알고리즘(GA)보다 우수함을 보인다.

Abstract

This paper presents a parallel combinatorial variant of Evolution Strategy(PCES) to solve well-known combinatorial optimization problems, Quadratic Assignment Problems(QAPs). The PCES reduces the possibility of getting stuck in local minima due to maintenance of subpopulation and thus it is more effective than the CES. Experiment results on two benchmark problems show that the PCES is better than the CES and the genetic algorithm(GA).

I. 서 론

Quadratic assignment problem(QAP)는 잘 알려진 전형적인 NP-hard 조합 최적화 문제^[1]로서, 병원 설계(layout), 키보드 디자인, 공장 장치(facility) 배치 문제 등 많은 다양한 실제 문제에 대한 적절한 모델로 사용되어 왔다^[2]. QAP는 Koopman과 Beckman^[3]에 의해 정형화된 문제로서 배치 및 물류이동 비용을 최소화하도록 N개의 각기 다른 장치를 N개나 그 이상의 위치에 배치하는 문제이다. 다양한 분야의 적용이 가능하다는 실제적인 중요성으로 인해 QAP를 해결하기 위한 많은 방법들의 개발을 위한 많은 연구가 이루

어져 왔다. 그러나, 일반적으로 branch-and-bound와 같은 exact한 최적화 기법으로는 작은 크기의 문제만을 해결할 수 있으며, 크기가 큰 문제는 계산적으로 불가능하다. 따라서 최근의 연구는 적절한 계산 시간 내에 상당한 크기의 문제에서 좋은 해를 찾을 수 있는 효과적인 heuristic의 개발을 목표로 하고 있다. 일반적으로 heuristic은 construction 방법과 improvement 방법으로 나누어진다. construction 방법에서는 비용을 줄이는 방향으로 한 번에 한 장치씩 선택된 위치에 배치하는 방법이다. 이에 비해 improvement 방법은 random하게 한 해를 선택하고, 비용이 줄도록 그 해를 조금씩 변형해 가는 방법이다. 최근에는 improvement 방법으로 간주할 수 있는 기법으로 stochastic search 알고리즘을 사용하여 QAP를 해결하려는 시도가 진행되었다. simulated annealing, TABU search, 유전 알고리즘^[4] 등을 이용하여 기존의 heuristic 보다 향상된 성능을 보였다^[5,6,7].

본 논문에서는 ES(Evolution Strategy)를 최적화

* 正會員, 韓國科學技術院 電氣 및 電子工學科
(Department of Electrical Engineering Korea Advanced Institute of Science and Technology)

** 正會員, 韓國通信技術研究所
(Korea Telecom International)

接受日字:1996年9月13日, 수정완료일:1997年5月2日

문제인 QAP에 적용시킨 CES를 설명하고, 보다 효과적인 성능 향상을 위하여 병렬 CES(PCES)^[8]를 제안한다. PCES는 여러 개의 소수전자 집단을 이용하여 여러 국소적인 탐색 영역에서 동시에 해를 구하고, 일정한 시간마다 서로 해를 교환함으로써 성능을 향상할 수 있다^[9]. benchmark 문제에서의 실험을 통하여 CES와 유전 알고리즘과 비교 분석하고 그 성능의 우수성을 보인다.

II. CES를 이용한 QAP 해법

1. QAP

QAP는 장치 배치 비용과 장치 사이의 재료 운반 비용이 최소가 되도록 N개의 장치를 N개의 지정된 위치에 배치하는 문제이다. QAP는 TSP 등의 여러 다른 문제의 일반화된 형태이며, 일반적으로 TSP보다 해결하기 어려운 것으로 알려져 있다. 즉, TSP는 큰 크기의 문제라도 제한된 enumeration을 통해서도 해결가능 하지만, QAP는 다소 더 많은 enumeration을 통해서 최적의 해를 찾을 수 있다. QAP는 다음의 3 matrix에 의해 표현된다.

$D = \{d_{ij}\}$, 위치 i와 j간의 거리.

$F = \{f_{hk}\}$, 장치 h와 k간의 재료의 흐름.

$C = \{c_{hi}\}$, 장치 h를 위치 i에 배치할 때의 비용.

보통 D와 F는 정수값을 가지는 대칭 matrix이고 C는 고려하지 않는다. 해의 형태는 하나의 순열 p로 나타낼 수 있다. 즉, $h = p(i)$ 는 위치 i에 장치 h를 배치한다는 것을 의미한다. 따라서, 문제는 다음의 비용 함수를 최소화하는 순열을 찾는 것이다.

$$\min_{\mu(\lambda)} \left\{ \sum_{i=1}^N \sum_{j=1}^N d_{ij} f_{\mu(i)\mu(j)} + \sum_{i=1}^N c_{i\mu(i)} \right\}$$

2. CES

CES는 Volker Nissen이 제안한 알고리즘으로, 일반적으로 수치 최적화 문제에 사용하는 ES를 조합형 최적화 문제에 적용한 것이다^[7]. CES의 기본적인 구조는 (1, λ)-ES 형태를 가지고 있다. 자세한 동작 절차는 다음과 같다. 각 해를 순열 형태로 표현하고, 하나의 해를 부모로 하여 변이를 통하여 λ 개의 자손을 만들고, 이 자손들 중 제일 좋은 해를 다음 세대의 부모로 한다. 이러한 동작을 반복적으로 일정한 시간만큼

수행하거나, 만족한 결과가 얻어질 때까지 수행한다. 주요 연산자는 다양한 perturbation중에서 transposition 변이를 사용한다. 자세히 설명하면, ^[1, 2] 사이의 임의의 랜덤 정수값을 발생시킨 다음, 그 값에 해당하는 수만큼 2개씩의 위치를 임의로 선택하여 그 두 위치의 값들을 서로 교환한다. 이러한 변이를 통하여 해를 찾아나가는 도중에 종종 국소 최소점에 빠질 가능성이 높다. 따라서 이러한 문제를 극복하기 위하여 destabilization이라는 과정을 도입하였다. 이 과정에서는 ^[3, 10] 사이의 랜덤 정수값을 발생시켜 그 수만큼 교환이 일어나기 때문에 원래의 부모와는 아주 상이한 자손을 발생시킴으로써 국소점을 벗어날 수 있다. 기본적인 알고리즘을 정리하면 아래와 같다.

단계 1 : 1개의 부모를 생성한다.

단계 2 : λ 개의 자손을 부모로부터 변이를 통하여 생성한다. (변이 방법 : ^[1, 2]의 임의의 랜덤 값을 발생시킨 다음, 부모를 그 수만큼 변형한다.)

단계 3 : 생성된 자손들의 fitness를 계산한다.

단계 4 : 좋은 자손을 연속적으로 특정 횟수 내에 얻지 못할 경우 destabilization을 수행한다.

단계 5 : 좋은 fitness를 가지는 자손을 다음 세대의 부모로 하여, 종료 기준을 만족할 때 까지 단계 1로 가서 과정을 계속 한다.

Volker Nissen이 제안한 CES는 (1, λ)-ES의 형태를 가지고 있기 때문에, 진화 연산의 일반적인 장점 중의 하나인 병렬성을 효과적으로 사용하지 못한다. 즉, 하나의 부모로부터 해를 찾아가기 때문에, genetic hill-climber 처럼 특정 값에 빨리 수렴한다. 이 문제를 해결하기 위하여 destabilization을 사용하는데, 이는 현재까지 찾은 탐색 영역과 다른 새로운 영역에서 탐색을 계속하고자 하는 접근 방식이다. 이와 다른 접근 방식으로 초기에 넓은 탐색 공간에서 여러 영역의 탐색을 허용하여 올바른 탐색 공간을 찾아가도록 하는 방법으로서, (μ , λ)-ES을 사용하여 시뮬레이션 하여 보았다. (μ , λ)-ES 방식은 μ 개의 부모에서 자손을 생성하므로, (1, λ)-ES에 비해 초기에 넓은 영역을 탐색할 수 있다. 또 다른 형태의 진화 방식인 ($\mu + \lambda$)-ES는 μ 개의 부모와 λ 개의 자손 중에서 제일 좋은 해를 가지는 μ 개를 다음 세대의 부모로 하는 방식이다. 이

방식의 특징은 제일 좋은 해는 항상 다음 세대에 유지한다는 것이다.

III. PCES와 GA

1. PCES

단순히 부모의 개수를 늘리는 것보다 더 큰 탐색의 병렬성을 확보하기 위한 방법으로서, 소유전자 집단을 두어 각각 병렬적으로 연산하고, 각 소유전자집단에서 제일 좋은 해를 서로 교환하는 방식인 island model^{19,101}을 채택하여 CES의 성능을 향상시킬 수 있을 것이다. 즉, 이는 CES를 병렬로 여러개를 사용하는 parallel CES(PCES)인 방식을 사용한다. 이 기법은 기존의 유전 알고리즘에서 성능 향상 및 병렬 컴퓨터 상에서의 수행을 통한 계산 시간 단축을 위하여 많이 사용하는 방식이다. 각 CES 단위내에서만 경쟁 (competition) 하기 때문에 단순히 부모의 개수를 늘린 (μ, λ) -ES 보다 PCES는 탐색 도중의 다양성을 유지하기 용이하며, 따라서 상대적으로 국소점에 빠지지 않고 더 넓은 영역의 탐색이 가능해진다. PCES는 CES보다 넓게 탐색을 수행할 수 있기 때문에, 국소적으로 탐색하는 CES에 비해 우수한 해를 찾기 위해 소요되는 수행 시간을 상대적으로 줄일 수 있을 것이다. 이는 하나의 국소적인 탐색 기법을 사용하여 promising 탐색 영역(좋은 해를 포함하고 있는 탐색 영역)을 탐색하는 것보다, 동시에 여러 개의 국소적인 탐색 기법을 수행하면서, 각각의 탐색 도중에서 서로 간의 정보 교환을 통하여 promising 영역을 찾아가는 방식이 좋은 해를 상대적으로 빠른 시간에 찾을 가능성이 높을 수 있기 때문이다. PCES도 최적해 근처의 탐색 영역으로 수렴하게 될 때는 각 소유전자 집단의 해들도 서로 유사해지게 된다. 이러한 경우에는 여러 개의 소유전자 집단을 유지할 필요가 없기 때문에, 하나의 소유전자 집단을 사용한다.

2. GA

기본적인 유전 알고리즘(GA)을 사용하였고, 기존의 GA에서 사용하는 교차 방식과 약간 다른 방식을 적용한 점이 기존의 GA와 다른 점이다. 표현 방식은 순열 형태를 채택하였으며, 이러한 표현 방식에서는 위치 정보가 성능에 큰 영향을 준다. 따라서 GA의 주요 연산자인 교차(crossover)는 2개의 부모로부터 좋은 인자

를 가지는 자손을 발생시킴으로 각 부모의 좋은 인자를 추출할 수 있는 확률이 높은 방법을 사용하여야 한다. 따라서, 그림 1과 같이 새로운 교체 연산자를 제안하였다. 이 방식의 주요 특징은 2개의 부모가 fitness가 높은 경우, 위치와 순서 정보가 같은 인자들은 성능에 좋은 영향을 가지는 block을 형성한다. 이 block은 위치와 순서를 고려한 fitness에 좋은 영향을 가지는 공통적인 인자이므로 다음 세대에 이 인자를 포함시켜야 하므로, 두 부모의 같은 위치에 같은 값을 가지는 경우에는 그대로 자식에 전달하고, 나머지 부분에 대하여는 position-based 교체 연산자(PBX)¹¹¹를 변형한 방식을 적용하였다. PBX만을 사용하였을 경우, 위치와 순서를 동시에 고려할 수 없기 때문에 위와 같은 방식을 제안하였다.

block										
		↓	↓	↓	↓					
부모 1	e	f	h	i	j	c	d	a	b	g
부모 2	b	d	h	i	j	c	a	g	f	e
교체 점		*						*		
자 손	b	f	h	i	j	c	d	a	g	e

그림 1. QAP를 위한 새로운 교체 연산자

Fig. 1. A new crossover operator tailored to QAP.

참고로 PBX는 하나의 부모에서 랜덤하게(0.5의 확률) 바뀌어야 할 인자들을 선택하고, 선택되지 않은 인자들은 그대로 자손에 전달한다. 그리고 선택된 인자들은, 다른 한 부모에서의 그 인자들이 배열해 있는 순서와 동일하게 자손의 비어있는 부분에 차례로 채운다. 즉 한 부모에서는 위치 정보를 다른 부모에서는 순서 정보를 대물림하는 방식이다. 변이 방법은 (μ, λ) -ES에서 사용한 변이를 그대로 사용하였다.

IV. 시뮬레이션 결과

C.E. Nugent가 사용한 2개의 benchmark 문제 (NUG20, NUG30)¹¹²¹에 대하여 각각의 방법들에 대하여 비교하였다. 표 1은 N이 20일 때이고, 표 2는 N이 30일 때의 문제에 적용시킨 결과이다. 표 1에서의 CPU시간은 SUN SPARC 10에서의 최적해를 찾을 때까지의 시간이다. 대체로 ES의 경우에는 부모가 1개일 때보다는 10개인 경우가 좋은 결과를 보였다. 결국 탐색 공간의 증가로 성능의 향상을 가져온 것이다.

표 1. NUG20 문제에서의 여러 방법들의 성능 비교(알려진 최소값은 2570)

Table 1. Experimental results on NUG20 of ESs, GA, and PCES with several different population sizes(best known cost = 2570).

	실행 횟수	최소값	최대값	평균값	평균CPU 시간
(1,100)-ES	5	2570	2570	2570	96.6 sec
(10,100)-ES	5	2570	2570	2570	81.8 sec
(1+100)-ES	5	2570	2570	2570	97.1 sec
(10+100)-ES	5	2570	2570	2570	82.2 sec
GA	5	2570	2621	2580.2	111 sec
PCES (10,100)-ES*5	5	2570	2570	2570	71.8 sec

표 2. NUG30 문제에서의 여러 방법들의 성능 비교(알려진 최소값은 6124, 모두 evaluation 횟수를 같게 하여 수행)

Table 2. Experimental results on NUG30 of ESs, GA, and PCES with several different population sizes(best known cost = 6124, the total number of evaluations is the same).

	실행 횟수	최소값	최대값	평균값
(1,100)-ES	5	6132	6158	6148.0
(10,100)-ES	5	6124	6170	6145.6
(1+100)-ES	5	6186	6404	6278.8
(10+100)-ES	5	6174	6226	6195.2
GA	5	6196	6243	6214.4
(10, 500)-ES	5	6128	6222	6155.2
PCES1 (10,100)-ES*5	5	6124	6138	6130.2
PCES2 (10,20)-ES*5	5	6132	6202	6165.0

(μ, λ)-ES가 ($\mu + \lambda$)-ES 비해 좋은 결과를 가지는데, 이는 ($\mu + \lambda$)-ES는 항상 좋은 해를 다음 세대에 유지시킴으로써 국소 최소점을 벗어날 기회가 현저히 감소하게 되는데서 기인한다. GA는 유전자 집단의 크기를 100으로 하였다. GA의 경우에는 결과가 좋지 않았는데, GA의 교체 연산자가 ES의 transposition 연산보다 초기에 많은 perturbation을 일으키기 때문에 초기에는 빨리 수렴하지만, 유전자 집단내의 다양성이 줄어들고, 따라서 점차 탐색 능력이 현저하게 줄면서 fine tuning이 되지 않은 특성의 결과이다. 결국 trans-

position을 사용하는 것이 효과적인 연산자로 볼 수 있었다. NUG30에서 PCES1의 경우는 본 실험에서 가장 좋은 결과를 가지며, (10,100)-ES 보다 높은 evaluation마다 비용 감소율을 보였다. PCES1은 소유전자집단을 5개로 하고 각 소유전자집단 내에서 (10,100)-ES를 100회 수행하였으며, 이러한 전체 과정을 20번 수행하였다. PCES2는 (10,20)-ES를 100회 수행하고 전체적으로 100번 수행하였다. PCES2는 소유전자집단내의 자손의 감소로 탐색 공간이 감소하여서 국소 최소점에 빠지기 쉬웠다. (10,500)-ES는 PCES와의 비교를 위한 실험으로 유전자 집단의 크기를 PCES와 같게 하였는데, (10,500)-ES가 PCES보다 나쁜 결과를 가졌다. 결국 이것은 PCES의 병렬성에 의하여 성능이 좋게 나왔다는 것을 알 수 있다.

V. 결 론

본 논문에서 제안된 방식과 Volker Nissen이 제안한 CES의 성능 비교 실험에서 CES의 적은 탐색 공간의 문제점을 PCES를 통하여 어느 정도 좋은 결과를 얻을 수 있었다. GA의 경우에는 진화 연산자를 선택하는데 있어서, 위치 정보와 순서 정보가 동시에 고려되어질 수 있는 교차의 적절한 연산법을 발견하지 못하여 좋은 성능을 찾기 어려웠다. CES는 초기에 빠르게 비용 함수가 감소하는 경향이 보였고 초기 수렴 현상을 보이다가 destabilization에 의해 아주 천천히 감소하였다. 결국 CES는 genetic-hill climber처럼 초기에는 빠른 감소를 보이나 이내 국소 최소점에 빠지고 destabilization의 랜덤 확률 탐색에 의하여 다른 탐색 공간을 찾으므로, 크기가 큰 QAP문제에서는 좋은 성능을 내기 어려운 반면, PCES는 병렬적인 탐색 특성으로 인하여 좋은 결과를 보였다. CES와 PCES의 경우에서 부모의 개수와 변이율을 진화 상황에 따라 적응적으로 변하게 함으로써 성능을 향상하려는 연구가 진행되어야 할 것이다. 또한, PCES에 대한 일반적인 성능 평가를 위하여 다양한 최적화 문제에서의 성능 비교 분석이 요구된다. simulated annealing과의 비교 실험이 진행 중이다.

참 고 문 헌

[1] M. Garey and D. Johnson, *Computers and*

- Intractability : A guide to the theory of NP-completeness*, W. H. Freeman, 1979.
- [2] R. E. Burkard, "Quadratic assignment problems," *European Journal of Operational Research*, vol. 15, pp. 283-289, 1984.
- [3] T. C. Koopmans and M. J. Beckmann, "Assignment problems and the location of economic activities," *Econometrica*, vol. 25, pp. 53-76, 1957.
- [4] D. E. Goldberg, *Genetic Algorithms in search, optimization, and machine learning*, Addison-Wesley, 1989.
- [5] D. T. Connolly, "An improved annealing scheme for the QAP," *European Journal of Operational Research*, vol. 46, pp. 93-100, 1990.
- [6] P. Yip and Y.-H. Pao, "A guided evolutionary simulated annealing approach to the quadratic assignment problem," *IEEE Trans. on System, Man, and Cybernetics*, vol. 24, pp. 1383-1387, 1994.
- [7] V. Nissen, "Solving the quadratic assignment problem with clue from nature," *IEEE Trans. on Neural Networks*, vol. 5, no. 1, pp. 66-72, 1994.
- [8] 이 현, 박 래정, 박 철훈, "병렬 CES를 이용한 QAP 해법," 1995년도 퍼지 및 지능시스템 춘계 학술대회 논문집, pp. 102-105, 1995
- [9] 박 철훈, 박 래정, "Genetic algorithm을 이용한 Traveling salesman problem 해법," 1992년도 신호처리 종합학술대회 논문집, pp. 590-593, 1992
- [10] H. Muhlenbein, M. Schomisch, and J. Born, "The parallel genetic algorithm as function optimizer," *Proc. of The Fourth Int. Conf. on Genetic Algorithms*, pp. 271-278 1991.
- [11] T. Starkweather, S. McDaniel, K. Mathias, D. Whitley, and C. Whitley, "A comparison of genetic sequencing operators," *Proc. of The Fourth Int. Conf. on Genetic Algorithms*, pp. 69-76, 1991.
- [12] R. E. Burkard, S. E. Karisch, and F. Rendl, "QAPLIB-A quadratic assignment problem library," *European Journal of Operational Research*, vol. 55, pp. 115-119, 1991.

 저 자 소 개

朴來正(正會員)

1968년 12월 29일생. 1991년 2월 서울대학교 전기공학과 졸업. 1993년 2월 한국과학기술원 전기및전자공학과 석사 졸업. 1997년 현재 한국과학기술원 전기및전자공학과 박사과정 재학중. 관심분야는 패턴인식, 신경회로망, 유전알고리즘, 최적화 알고리즘등임.

朴 哲 薰(正會員)

1961년 4월 14일생. 1984년 2월 서울대학교 전자공학과 졸업. 1990년 미국 California Institute of Technology 공학석사 및 박사학위 취득. 1997년 현재 한국과학기술원 전기및전자공학과 부교수. 관심분야는 패턴인식, 퍼지 논리, 신경망 제어, 유전알고리즘등임.

李 賢(正會員)

1970년 8월 9일생. 1993년 2월 경북대학교 전자공학과 졸업. 1996년 2월 한국과학기술원 전기및전자공학과 석사 졸업. 1997년 현재 한국통신 기술(주) 연구소 근무중.