

다중제약 배낭문제를 위한 새로운 유전 알고리즘 (A Novel Genetic Algorithm for Multiconstrained Knapsack Problem)

이상욱, 석상문, 이주상, 장석철, 안병하
광주 북 오룡 1 광주과학기술원 기전공학과
{yashin96, soakbong, jusang, stniron, bayhay@gist.ac.kr}

Abstract

The knapsack problem (KP) is one of the traditional optimization problems. Specially, multiconstrained knapsack problem (MKP) is well-known NP-hard problem. Many heuristic algorithms and evolutionary algorithms have tackled this problem and shown good performance. This paper presents a novel genetic algorithm for the multiconstrained knapsack problem. The proposed algorithm is called 'Adaptive Link Adjustment'. It is based on integer random key representation and uses additional α and β -process as well as selection, crossover and mutation. The experiment results show that it can be archive good performance.

1. 서론

다중제약 0-1 배낭 문제 (이하 MKP)는 잘 알려진 NP-hard 문제로써 다음과 같이 표현된다.

$$\text{최대화 } f(x_1, \dots, x_n) = \sum_{j=1}^n p_j x_j \quad (1)$$

$$\text{제약조건 } C_i: \sum_{j=1}^n w_{i,j} x_j \leq b_i \quad (2)$$
$$i = 1, \dots, m$$
$$x_j \in \{0, 1\}, \quad j = 1, \dots, n$$

여기서 $p_j > 0, r_{i,j} \geq 0, b_i \geq 0$

목적함수 $f(x_1, \dots, x_n)$ 는 m 개의 제약조건 C_i 를 잘 고려되어 최대화되어야하는 문제이다. 여기서, x_j 는 0또는 1의 값만 가질 수 있다.

MKP는 다양한 분야에 적용되고 있다. 일례로 경제 분야에 사용된 경우를 살펴보면: n 개의 프로젝트 ($j = 1, \dots, n$)와 m 개의 자원 ($i = 1, \dots, m$)을 고려해보자. 각 프로젝트는 자원 $w_{i,j}$ 을 소비하여 이득 p_j 을 올린다. 이때 문제는 어떤 프로젝트 조합을 선택하여 자원제약 b_i 을 초과하지 않으면서 이득을 최대화 할 수 있는지가 된다.

많은 연구자들은 MKP의 최적해를 구하기 위해 노력하였다. Gavish와 Pirkul은 작은 크기의 MKP의 최적해를 구할 수 있는 방법을 제안하였다 [1].

Balas와 Martin은 정수 제약을 완화 시켜 선형계획법 (linear programming)을 적용하기도 하였다 [2]. 즉, $x_j \in \{0, 1\}$ 를 $0 \leq x_j \leq 1$ 로 완화시켜 심플렉스 (simplex)방법을 이용하여 x_j 를 구했다. 이렇게 완화된 문제는 NP-hard 문제가 아니며 쉽게 최적해를 구할 수 있다. 실수표현으로 나온 x_j 는 적절한 heuristic을 사용하여 정수로 변환한다.

최근에는 유전 알고리즘을 이용하여 MKP를 해결하려는 연구가 많이 진행되었다. MKP에 적용된 기존의 유전 알고리즘은 표현법에 따라 크게 3가지로 나눌 수 있다. 비트 스트링 표현법 (Bit String Representation) [3], 순차 정렬 표현법 (Order Based Representation) [4] 그리고 가중치 코드 표현법 (Weight Code Representation) [5]. 본 논문에서 제안하는 방법은 순차 정렬 표현법과 유사한 방법이다.

본 논문은 다음과 같은 순서로 전개된다. 2장에서는 제안하는 유전 알고리즘에 대해 설명한다. 벤치마크 문제에 제안한 유전 알고리즘을 적용하여 실험한 내용이 3장에 전개된다. 마지막으로 4장에서 결론을 맺고 미래 연구에 대한 방향을 제시한다.

2. 제안한 유전 알고리즘

본 논문에서 제안하는 방법은 ALA (Adaptive Link Adjustment)로써 Soak에 의해 처음 소개되었다 [6]. ALA는 PR (Permutation Representation)과 랜덤 키 표현법 (Random Key Representation) [7]과 유사한 방법으로 다음과 같이 적용된다.

step 1 초기해 생성.

모든 초기 유전자에 0을 할당.

step 2 유전인자 정렬.

각 해에 있는 유전인자들을 값에 따라 내림차순 정렬.

step 3 프로젝트 계획 수립.

정렬된 순서에 따라 $x_j = 1$ 을 할당하여

$C_i; \sum_{j=1}^n w_{i,j}x_j \leq b_i$ 를 검사한다. 만약, 제약 조건을 만족하면 그대로 두고, 만족하지 못하면 $x_j=0$ 으로 할당한다.

step 4 평가 단계.

step 3에서 나온 결과를 바탕으로

$f(x_1, \dots, x_n) = \sum_{j=1}^n p_j x_j$ 를 이용하여 적합도 함수를 구한다.

step 5 α -과정, β -과정.

step 5.1 α -과정.

최적해를 갱신한 해에 대해서만 α -과정을 수행한다. 따라서 반복 초기를 제외하고는 거의 나타나지 않는 과정이다. 방법은 최적해를 갱신한 해에서 1로 선택된 유전자에 대해 α 만큼 가산준다.

step 5.2 β -과정.

최적해를 갱신하지 못한 모든 해에 대해 β -과정을 수행한다. 따라서 제안한 알고리즘의 주요 연산이라 할 수 있다. 방법은 최적해를 갱신하지 못한 모든 해에서 0으로 선택된 유전자에 대해 θ 의 확률로 β 만큼 감산한다.

step 6 선택전략.

실세계 토너먼트 선택전략 (real world tournament selection, RWTS) [6] 적용.

step 7 교차 연산.

교차확률에 따라 Uniform 교차연산 적용.

step 8 돌연변이 연산.

돌연변이확률에 따라 상호교환 변이연산 적용

step 9) 종료.

정해진 종료 조건을 만족하면 종료, 아니면 step 2로 이동.

3. 실험

제안한 유전 알고리즘의 성능을 확인하기 위해 OL-library에서 제공한 MKP 벤치마크 문제에 적용하였다 [8]. 1차적으로 제안한 유전 알고리즘의 가능성을 살펴보기 위해 벤치마크 문제 중 최적해가 알려진 문제에 적용하였다. 모집단 수 (Population Size)와 반복 회수 (Iteration Number)는 $n*m$, 교차확률과 돌연변이확률은 0.6 그리고 α 와 β 는 1로 설정하고 실험하였다. 실험 환경은 Pentium IV 2.4GHz, Visual C++ 6.0에서 실행하였다. 그 결과는 table 3.1 과 같다.

n	m	최적해 빈도	평균오차	시간(s)
6	10	10/10	0.0	0.016
10	10	10/10	0.0	0.125
15	10	10/10	0.0	0.672
20	10	10/10	0.0	1.547
28	10	10/10	0.0	6.25
39	5	10/10	0.0	3.828
50	5	10/10	0.0	9.578

<table 3.1> 실험 결과

실험 결과는 같은 환경 및 변수 설정에서 총 10번의 실험을 한 결과이다. 실험 결과, 제안한 알고리즘은 7개의 문제에서 10번 모두 최적해를 찾았다.

4. 결과

이 논문은 다중제약 배낭문제에 새로운 유전 알고리즘을 적용하였다. 비록 최적화가 알려진 크기가 작은 문제일지라도 모든 문제에 대해 최적해를 찾은 것은 매우 고무적인 결과라 할 수 있다. 심화 연구로 크기가 큰 문제에 적용하면 알고리즘의 정확성 성능을 평가할 수 있으리라 생각된다.

5. 참고 문헌

- Gavish B., Pirkul H., "Efficient Algorithms for Solving Multiconstraint Zero-One Knapsack Problems to Optimality," *Mathematical Programming*, Vol31, pp.78-105,(1985).
- Balas E., Martin C.H., "Pivot and Complement - a Heuristic for 0-1 Programming," *Management Science* 26(1), (1980).
- Olsen A. L., "Penalty Functions and the Knapsack Problem," *Pro. of the 1st International Conference on Evolutionary Computation 1994*, Orlando, FL, edited by D. B. Fogel, pp.559-564,(1994).
- Hinterding R., "Mapping, Order-independent Genes and the Knapsack Problem," *Proc. of the 1st IEEE International Conference on Evolutionary Computation 1994*, Orlando, FL, edited by D. B. Fogel, pp.13-17,(1994).
- Raidl G. R., "Weight-Codings in a Genetic Algorithm for the Multiconstraint Knapsack Problem," *Proc. of the 1999 IEEE Congress on Evolutionary Computation*, Washington DC, pp.596-603.(1999).
- Soak S. M., Corne D. Ahn B. H., "'Adaptive Link Adjustment' Applied to The Fixed Charge Transportation Problem," Submitted to *IEEE Transaction on Evolutionary Computation*.
- Bean J., "Genetic Algorithm and Random Keys for Sequencing and Optimization," *ORSA Journal on Computing*, Vol 6, pp.154-160.(1994).
- <http://people.brunel.ac.uk/~mastjbjeb/orlib/mknapiinfo.html>.