진화 알고리즘에서 휴리스틱 연산

류정우°, 김명원
숭실대학교 컴퓨터학과
ryu0914@orgio.net, mkim@computing.ssu.ac.kr

Heuristic Operation in Evolutionary Algorithms

Joung Woo Ryu°, Myung Won Kim
School of Computing, Soongsil University

요 약

진화 알고리즘에서 고려할 사항 중 하나는 문제와 관련 있는 진화연산 즉, 교배 연산과 돌연변이 연산을 정의하는 것이다. 일반적으로 교배 연산은 두 개체의 정보를 교환하는 재조합 연산으로써 진화의 속도를 촉진시키는 역할을 하고 돌연변이 연산은 개체집단의 다양성을 유지시키는 역할을 한다. 그러나 이러한 진화연산자는 확률에 근거하여 모든 개체에 적용되는 맹목적인 연산이 가질 수 있는 진화 시간 지연의 문제점을 갖는다.

본 논문에서는 맹목적 진화연산에 의한 진화 시간 지연을 해결하기 위해 휴리스틱 연산을 제안한다. 휴리스틱 연산은 문제의 특성에 맞지 않는 개체에만 적용되는 연산으로 진화 시간을 단축시킬 수 있다. 따라서 이러한 휴리스틱 연산의 타당성을 확인하기 위해 본 논문에서는 진화 알고리즘을 이용하여 최적의 클러스터 위치와 개수를 자동으로 찾아주는 문제에 클러스터의 특성을 고려한 휴리스틱 연산인 합병연산과 분할연산 그리고 K-means연산을 정의하여 다차원 실험데이터로 실험한 결과를 보이고 있다.

1. 서론

유전자 알고리즘(Genetic Algorithm)은 1975년 존 홀랜드(John Holland)에 의해 제안된 전역적 탐색기법이며 이 기법은 자연현상의 자연도태와 진화의 메카니즘에 기반을 둔 확률적인 탐색 알고리즘으로서 특히, 최적화 문제에 효율적인 알고리즘이다. 하지만 유전자 알고리즘은 일정한 길이의 이진 스트링과 단지 두 개의 기본적인 유전자 연산(교배 연산, 돌연변이 연산)들을 사용하기 때문에 문제를 유전자 알고리즘에 적합한 형태로 변경할 필요가 있으며 이는 많은 응용분야에 적용될 수 없는 문제점이 된다. 이러한 문제점을 해결하기 위해 개체의 데이터구조와 특정한 유전자 연산에 문제고유의 지식을 포함시킨 진화 알고리즘(Evolutionary Algorithm)이 제안되었으며 이는 더 많은 분야에 사용되고 있다.[1]

따라서, 진화 알고리즘에서 고려할 사항 중 하나는 문제와 관련 있는 진화연산 즉, 교배연산과 돌연변이연산을 정의하는 것이다. 일반적으로 교배연산은 두 개체의 정보를 교환하는 재조합 연산으로써 진화의 속도를 촉진시키는 역할을 하고 돌연변이 연산은 개체집단의 다양성을 유지시키는 역할을 한다. [2]에서는 유전자 알고리즘에서는 교배연산의 역할을 중요하고 진화 알고리즘에서는 돌연변이연산의 역할이 중요하다고 말하고 있다. 특히, 개체의 데이터구조가 가변적 길이인 진화알고리즘에서는 두 개체를 재조합시키는 것보다 문제의 특성에 맞지 않는 개체를 특성에 맞게 변화시켜주는 접근방식이 일반적으로 문제해결에 대한 바람직한 방향으로 보인다.

본 논문에서는 진화 알고리즘을 이용하여 최적의 클러스터 위치와 개수를 찾아주는 문제에 있어 특성에 맞는 휴리스틱 연산을 정의한다. 정의된 휴리스틱 연산을 사용한 경우와 교배연산을 사용한 경우를 비교하여 전자의 경우가 보다 진화 지연 시간을 단축시키는 것을 확인함으로써 진화 알고리즘에서 휴리스틱 연산의 타당성을 보여준다.

본 논문의 구성은 다음과 같다. 우선 2장에서는 휴리스틱 연산의 타당성을 검증하기 위한 문제 영역으로 클러스터링에 대해 살펴본다. 3장은 제안한 휴리스틱 연산을 살펴보고 4장은 제안한 휴리스틱 연산을 사용한 진화 알고리즘을 클러스터링 문제를 해결할 수 있도록 설계 한 것에 대해 기술하고 있다. 5장에서 다차원 실험데이터를 클러스터링 하는데 있어 진화연산과 휴리스틱 연산을 비교 분석한다. 마지막으로 5장에서 결론을 맺는다.

2. 클러스터링

클러스터링이란 주어진 데이터를 군집화 하는 것으로 , 한 군집 내에 있는 데이터들은 유사성(similarity)이 높은 반면 다른 군집에 속하는 데이터들과는 차별성(dissimilarity)이 높도록 데이터를 분류하는 것이다.

본 연구는 한국과학재단 특정기초연구과제
(과제번호 : 98-0102-01-01-3)의 지원을 받았음

클러스터링은 특별한 정보나 배경지식 없이 데이터들 간의 주어진 척도를 이용하여 결과를 이끌어 내므로 비 교사 학습(unsupervised learning)에 속하는 패턴분류 방법으로써 현재 패턴인식, 영상처리 등의 공학분야에 널리 적용되고 있을 뿐 아니라, 최근 많은 관심의 대상이 되고 있는 데이터마이닝 분야에서 핵심기술로 활발히 연구되고 있다.

클러스터링 알고리즘은 크게 분할적 클러스터링(partitional clustering)과 계층적 클러스터링(hierarchical clustering)으로 나눌 수 있다.

특히, 분할적 클러스터링은 임의의 데이터가 단지 하나의 클러스터에 포함되는 단순 클러스터링(hard clustering)과 두 개 이상의 클러스터에 동시에 속하는 것을 허용하는 퍼지 클러스터링(fuzzy clustering)으로 나뉘어 진다. 이와 같은 알고리즘들은 초기 값에 따라 지역적 최적해로 수렴될 수 있는 문제점과 사전에 클러스터 개수를 결정해야 하는 문제점, 그리고 잡음에 민감한 문제점을 가지고 있다.

최근 이러한 클러스터링 문제를 해결하기 위해 진화 알고리즘을 이용하려는 연구가 활발히 진행되고 있다.[3][4][5]

3. 휴리스틱 연산

휴리스틱 연산은 맹목적 진화연산과는 다르게 문제의 특성에 맞지 않는 개체에만 적용되는 연산이다. 따라서 문제가 정의되어야만 그에 따른 휴리스틱 연산을 설계할 수 있다.

본 논문에서는 진화알고리즘을 이용한 최적의 클러스터 개수를 찾는 문제에 휴리스틱 연산을 설계하여 적용한다. 따라서 클러스터링의 특성은 클러스터 내의 모든 데이터들은 높은 유사성을 가져야 하며, 반면 다른 클러스터에 속하는 데이터들은 높은 차별성을 가져야 한다. 그러므로 설계된 휴리스틱 연산은 다음과 같이 세 개의 연산으로 정의 할 수 있다.

3.1. 합병 연산 (merge operation)

합병 연산은 다른 클러스터에 속하는 데이터들은 높은 차별성을 가져야 하는 특성을 고려한 연산이다. 즉, 가장 근접한 두 클러스터를 합병하는 연산이다.

두 클러스터 중심간의 거리가 임계값 θ_M보다 작으면 합병 연산을 적용하여 (그림1.왼쪽)과 같이 두 클러스터의 중심 v_1, v_2을 합하여 한 개의 클러스터로 만든다. 생성된 클러스터 중심은 식(1)과 같이 계산된다.

$$v = \frac{v_1 + v_2}{2} \quad (1)$$

여기서 임계값 는 평균 중심거리에 상수 α를 곱한 것으로서 식(2)과 같다., c는 클러스터 개수를 의미한다.

$$\theta_M = \alpha\left[\frac{2}{c(c-1)}\sum_{i=1}^{c-1}\sum_{j=i+1}^{c} D_E(v_i, v_j)\right],\ 0 < \alpha \le 1 \quad (2)$$

$$\text{단, } D_E(v_i, v_j) = \sqrt{\sum_{k=1}^{d}(v_{ik} - v_{jk})^2}$$

만약, 한 세대에 두 중심간의 거리가 임계값 보다 작은 것이 2개 이상

일 경우에는 그 중 가장 가까운 중심 두 개만을 합병한다.

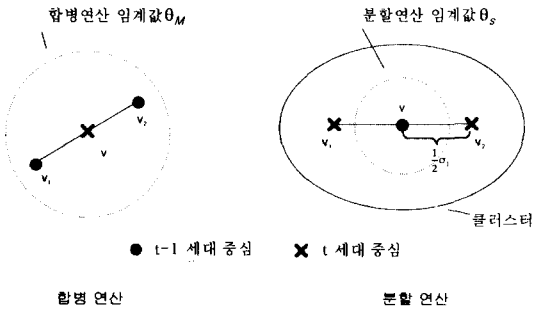

그림 1. 합병 / 분할 연산

3.2. 분할 연산 (split operation)

분할 연산은 클러스터 내의 모든 데이터들은 높은 유사성을 가져야 하는 특성을 고려한 연산이다. 즉 클러스터내의 데이터가 넓게 분산된 경우 두 개의 클러스터로 분할하는 연산이다.

클러스터 내의 분산을 클러스터의 표준편차 벡터 요소들 중 임계값 θ_S보다 크면 분할 연산을 적용하여(그림1. 오른쪽)와 같이 중심v를 두 중심 v_1, v_2 로 분할한다. 분할된 중심좌표는 식(3)과 같이 표준편차가 가장 큰 차원만 고려한다 여기서 σ_k 은 k차원에서의 표준편차를 의미한다.

$$\begin{cases} v_{1k} = v_k + \frac{1}{2}\sigma_k, \quad v_{2k} = v_k - \frac{1}{2}\sigma_k, \quad (\sigma_k = \max_i [\sigma_i]) \\ v_{1i} = v_{2i} = v_i \quad (i \neq k) \end{cases} \tag{3}$$

이때 임계값 θ_S는 모든 클러스터의 표준편차 벡터 요소 합의 평균에 상수 β를 곱한 값으로서 식(4)과 같다.

$$\theta_S = \beta \left[\frac{1}{c} \sum_{i=1}^{c} SD(v_i) \right], \quad 1 \le \beta \tag{4}$$

$$\text{단, } SD(v_i) = \frac{1}{d} \sum_{k=1}^{d} \left(\sqrt{\frac{1}{N_i} \sum_{x_j \in S_i} (x_{jk} - v_{ik})^2} \right)$$

만약, 한 세대에 표준편차 벡터 요소가 임계값 보다 큰 것이 2개 이상일 경우에는 그 중 가장 큰 벡터요소를 가지는 클러스터 하나만 분할한다.

3.3. K-means 연산 (K-means operation)

앞에서 정의한 합병 연산, 분할 연산에 의해 생성된 중심은 클러스터 내에서 정확한 중심이 아닌 대략적인 중심에 위치하고 있어 정확한 중심을 찾기 위해서 더 진화를 시켜야할 필요가 있다. 이러한 문제를 개선하기 위해 본 논문은 [3]에서 제안한 K-means 알고리즘을 한 단계 적용한 연산을 사용한다. K-means 연산은 앞에서 설명한 연산들을 사용하여 생성된 중심들을 가지고 각 데이터를 가장 가까운 중심으로 재 할당하여 새로운 클러스터 중심을 계산한다. K-means 연산을 통해 매 세대마다 클러스터 중심들을 데이터 분포에 가장 적합한 중심으로 교정함으로써 진화 속도를 개선한다.

4. 진화알고리즘을 이용한 클러스터링

클러스터링 문제는 입력데이터를 포함하는 입력공간에서 유사한 데이터들을 그룹화시키는 문제로 생각할 수 있다. N개의 데이터를 c개의 클러스터로 그룹화 할 수 있는 경우의 수는 식(5)과 같이 표현할 수 있다.

$$\frac{1}{c!} \sum_{i=0}^{c} \binom{c}{i} (-1)^{c-i} i^N \tag{5}$$

이처럼 클러스터링 문제에 있어서 최적의 클러스터를 찾는 것은 NP-complete문제로 알려져 있으며 또한 어떤 클러스터링이 최적이냐에 대한 수학적 모델이 아직 알려지지 않았다.[4]

최근 이러한 클러스터링 문제를 자연도태와 진화의 메카니즘에 기반을 둔 확률적인 병렬탐색 알고리즘인 진화알고리즘을 사용하여 해결하려는 연구가 활발히 진행되고 있다.[3][4][5]

본 논문에서는 (그림2)과 같이 진화알고리즘에 휴리스틱 연산을 사용한다.

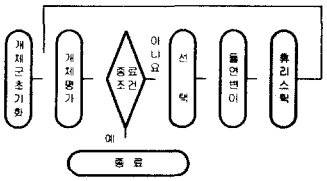

그림 2. 진화 과정

진화알고리즘을 이용하여 최적의 클러스터 개수와 위치를 찾기 위해 개체를 실수로 표현된 가변적 길이를 가지도록 인코딩 한다. 또한 개체 평가에 있어 클러스터 특성인 클러스터 내의 유사성과 클러스터간의 차별성을 같이 고려한 적합도 함수를 정의한다. [5]에서는 클러스터 내의 유사성은 데이터들이 각 차원별로 클러스터 중심으로부터 얼마나 분산되었는가를 표준편차를 사용하여 정의한 분산도로서 나타낸다. 또한 클러스터간의 차별성은 각 클러스터내의 데이터들이 갖는 평균 소속정도의 합으로 정의한 분리도로서 나타낸다. 소속정도란 임의의 데이터가 클러스터에 포함될 가능성을 의미한다. 이와 같이 정의된 적합도 함수는 각 개체들의 특성을 평가하여 다음 세대의 개체집단을 선택하기 위한 척도가 된다. 다음 세대의 개체집단을 선택하기 위한 방법으로 룰렛휠(roulette wheel)방법과 최상의 개체를 보존하는 엘리트 방법(elitist method)을 사용한다.

본 논문에서는 앞서 설명한 휴리스틱 연산을 적용하기 때문에 진화된 클러스터 중심의 위치가 최적의 위치일 가능성이 높다. 하지만 보장은 할 수 없다. 따라서 가우시안 분포함수를 이용한 돌연변이 연산을 이용하여 현재 클러스터 중심으로부터 가까운 곳에서 새로운 중심이 선택될 확률을 높여 개체의 다양성을 유지한다.

표1.은 한 세대에서 돌연변이 연산과 휴리스틱 연산이 적용되는 순서를 보여주고 있다.

표1. 연산 수행 순서

```
돌연변이 ( );
if ( 두 중심간의 거리 < θ_M ) then
    합병연산 ( 가장 가까운 두 중심 );
if ( 클러스터 표준편차 벡터 요소 > θ_S ) then
    분할연산 ( 가장 큰 표준편차 벡터요소 );
K-means연산 ( );
```

5. 실 험

본 논문에서는 최적의 클러스터 개수를 찾는 문제에 있어 정의한 휴리스틱 연산의 타당성을 검증하기 위해 [5]에서 사용한 다차원 실험데이터에 교배연산을 적용하였을 때와 그 대신 휴리스틱 연산을 적용하였을 때 진화 시간을 비교하며 또한 휴리스틱 연산만 사용하여 문제를 해결하는 경우와 진화알고리즘에 휴리스틱 연산을 사용하여 문제를 해결하는 경우를 비교한다.

다차원 실험데이터는 가우시안 분포 데이터로써 10개의 원 중심으로부터 가우시안 분포를 갖도록 각각 50개씩 데이터를 생성한 것이다. 따라서 총 500개의 데이터를 가지고 있으며, 본 실험을 위해 차원만을 2차원, 10차원 20차원으로 확장하여 세 개의 다차원 실험데이터를 생성한다.

본 실험에서 실험 데이터 변수로 표2와 같이 선언한다.

표 2. 실험 데이터 변수

개체집단크기	30	α	0.5
돌연변이확률	0.2	β	1.5

5.1 교배연산과 휴리스틱 연산

본 실험에서 생성된 클러스터가 적합한지 알아보기 위해 생성된 클러스터 중심과 가우시안 분포의 원 중심간의 오차를 나타내고 있다. 오차는 데이터의 원 중심과 가장 근접한 클러스터 중심간의 거리를 데이터 영역의 대각선 길이(데이터 공간의 최대거리)로 나눈 비율로 나타낸다.

표3. 각각의 데이터에 대한 오차를 보여주고 있다.

표 3 . 실험데이터에 대한 오차

데이터	각 차원의 영역	오차(%)	중심개수
2차원 실험데이터	[0,10]	0.84	10
10차원 실험데이터	[0,1]	0.99	10
20차원 실험데이터	[0,10]	0.46	10

본 논문에서는 탐색공간이 클 때 교배연산을 사용하는 경우 보다 휴리스틱 연산을 사용할 경우 보다 진화 시간이 단축되는 것을 확인하기 위해 휴리스틱 연산 대신 교배연산을 사용하였다.

그림3 그림4은 교배연산과 돌연변이 연산을 적용했을 때와 교배연산 대신 휴리스틱 연산과 돌연변이 연산을 적용했을 때 각각의 성능을 나타내고 있으며 교배연산 대신 휴리스틱 연산을 적용한 경우가 더 빨리 진화되는 것을 알 수 있다.

이는 클러스터링에서 클러스터간의 상관성이 강하기 때문이다. 즉, 일부의 클러스터 중심위치 또는 클러스터 개수가 바뀌면 다른 클러스터도 바뀌게 된다. 따라서 부모세대로부터 가능성이 높은 두 개체들을 선택하여 교배연산을 적용하게 되면 정보가 교환되어 일부의 클러스터 중심위치 또는 클러스터 개수가 바뀌게 된다. 이는 교배연산을 적용하기 전보다 더 우수한 개체가 생성될 가능성을 낮게 한다.

그러므로 제안한 알고리즘은 선택방법에 의해 선택된 개체들 사이에서 맹목적으로 선택된 두 개체의 정보를 교환하는 교배연산 대신 개체 각각에 대해 클러스터내의 유사성과 클러스터간의 차별성을 고려하여 휴리스틱 연산을 적용하는 것이 보다 효율적이라고 할 수 있다.

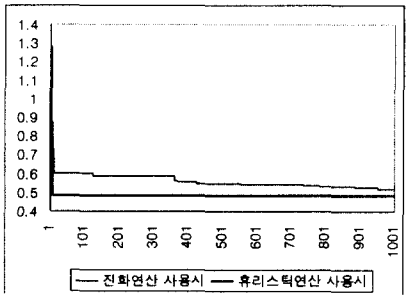

그림 3 1000세대 진화된 적합도값의 변화

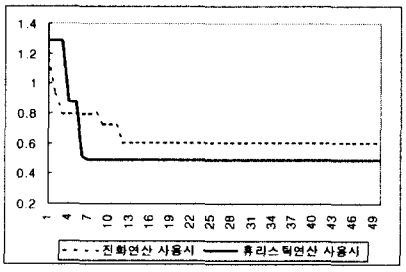

그림 4. 20세대 진화된 적합도값의 변화

5.2. 휴리스틱 연산만 사용하는 경우

앞서 실험을 통해 문제의 특성상 교배연산 보다 휴리스틱 연산을 사용한 경우가 진화 시간을 단축시켰음을 알 수 있다. 그렇다면 본 실험은 휴리스틱 연산만 적용하여 가장 좋은 결과를 취할 경우와 진화 알고리즘에 휴리스틱연산을 사용하는 경우 어떤 결과의 차이가 있는지 알아보기 위해 아래와 같은 실험을 통해 살펴본다.

초기해를 임의로 생성한 크기가 30인 개체집단을 20세대 수행하면 최적의 클러스터 개수와 중심의 위치를 찾는다는 것을 앞의 실험을 통해 알 수 있었다. 본 실험에서는 진화알고리즘을 이용한 최적의 클러스터 개수를 찾는 시간만큼 휴리스틱 연산만을 이용하여 반복 수행한 다음 결과를 비교한다.

휴리스틱 연산만을 사용하는 경우 적합도 평가 단계와 개체 선택 단계가 제외됨으로 진화알고리즘을 사용하는 경우보다 약 2배가 빠르다는 것을 실험을 통해 알 수 있었다. 그러므로 실험에서는 비교의 공정성을 위하여 크기가 제안한 알고리즘의 2배인 60인 개체집단을 휴리스틱 연산만을 사용하여 20회 반복 수행한다. 이것을 초기 클러스터 중심과 클러스터 개수를 각각 다르게 하여 10회 실시한 결과, 수렴된 평균 적합도는 표.4와 같다. 여기서 ( )는 10회 실험 중 적합한 클러스터 개수와 중심을 찾은 횟수를 나타낸다. 따라서 제안한 알고리즘은 10회 모두 옳게 찾았으나 휴리스틱 연산만을 사용한 경우에는 다차원일수록 옳게 찾는 횟수가 적고 제안한 알고리즘과의 적합도 오차도 커지는 것을 알 수 있다. 추가로 크기가 120인 개체집단을 실험한 결과 역시 제안한 알고리즘 보다는 못하지만, 개체집단이 60인 경우보다는 좋은 결과를 보이고 있다.

따라서 휴리스틱 연산만을 적용하여 최적의 해를 찾는다는 것을 보장할 수 없으며, 또한 있다 하더라도 진화알고리즘에 휴리스틱 연산을 사용하는 경우보다 수렴 시간이 오래 걸린다. 더욱이 탐색공간이 커질수록 그 가능성은 더욱 적어진다. 반면 진화알고리즘에 휴리스틱 연산을 사용하는 경우 매 세대 전 세대에서 우수한 성질의 개체를 선택하여 연산을 수행하기 때문에 최적의 해를 찾을 가능성이 보다 높다는 것을 알 수 있다.

표 4. 휴리스틱 연산만 사용한 경우와 진화알고리즘에 휴리스틱 연산을 사용한 경우의 성능 비교

방법 / 개체크기 / 데이터	제안한 알고리즘	휴리스틱 연산만 사용한 경우	
	30	60	120
2차원 데이터	4.3676 (10)	4.5043 (8)	4.4763 (8)
10차원 데이터	2.4364 (10)	3.1607 (3)	3.0449 (5)
20차원 데이터	23.5227 (10)	33.5304 (3)	30.3321 (5)

6. 결 론

본 논문은 진화 알고리즘에서 문제의 특성을 고려한 휴리스틱 연산에 대한 타당성을 보여주기 위해 최근 많이 연구되고 있는 진화알고리즘을 이용한 최적의 클러스터의 위치와 개수를 찾는 문제에 적용하여 교배연산을 적용하였을 때와 그 결과를 비교하였다. 그 결과 진화 알고리즘에서 교배연산과 같은 재조합 연산은 개체들 간의 정보를 교환함으로써 진화 속도를 향상시키지만 클러스터링 문제와 같이 유전인자간의 상관성이 강할 경우 교배의 역할을 보장할 수 없다. 따라서 맹목적으로 개체에 적용하는 진화연산보다 문제의 특성을 고려하여 특성에 맞지 않은 개체에만 적용하는 휴리스틱 연산을 사용하여 접근하는 것이 보다 효율적이다.

지금까지는 진화연산에 의해 살펴보았으나, 향후 계획으로는 휴리스틱연산과 개체집단의 크기와의 관계에 대해 연구할 계획이다. [6]에서는 개체집단의 크기가 작을 때 교배보다 돌연변이가 더 유용하다는 것을 보였으며, 반면 [7]에서는 개체집단의 크기가 클 때 돌연변이보다 교배가 더 유용하다는 것을 증명하였다. 따라서 휴리스틱 연산과 개체집단의 크기와의 관계를 살펴보고 실험 데이터뿐만 아니라 벤치마크 데이터나 더 나가 영상인식 또는 데이터마이닝 같은 응용분야에 활용할 계획이다.

참 고 문 헌

[1] Z. Michalewicz, "Genetic Algorithm + Data Structures = Evolution Programs", Third, Extended Edition, Springer-Verlag, 1995

[2] Spears, William M. "Adapting Crossover in Evolutionary Algorithms", Proceedings of the Evolutionary Programming Conference, 367-384 ,1995

[3] K.Krishna and M. Narasimha Murty, "Genetic K-Means Algorithm", IEEE Trans. Syst., Man, Cybern., VOL. 29, No. 3, pp. 433-439,1999

[4] YoungJa Park and ManSuk Song, "A Genetic Algorithm for Clustering Problems",In Symposium on Genetic Algorhim-98 568-575, July, 1998

[5] 김명원,강명구,류정우, "휴리스틱 진화 알고리즘을 이용한 클러스터링 알고리즘", 2000가을 학술발표논문집(B) 제 27권 2호 -, 2000

[6] Eshelman, L. Personal communication, Philips Laboratories, BriarcliffManor, NY

[7] Spears,W.M., and V.A. Anand., "A Study of Crossover Operators in Genetic Programming",Proceedings of the International Symposium on Methodologies for Intelligent Systems,409-418,1991