

유전자 알고리즘을 이용한 최적 설계

박정선 · 류석규

한국항공대학교 항공우주공학과

Optimal Design by Genetic Algorithm

Jungsun Park and Suk-Kyu Ryu

I. 서 론

기술 개발 분야에서 제품의 경제성과 효율성을 극대화하기 위해서 여러 가지 방법들을 시도하고 있다. 그러한 방법들 중 하나가 최적화 방법(optimization method)이다. 그렇기 때문에 지난 수십 년 동안 최적화는 매우 중요시되어졌고, 여러 산업 분야(항공기, 자동차, 화학, 전기, 제조 산업 등)에 적용되어져 오고 있다.

그리고 점차 컴퓨터의 발전으로 최적화 방법을 사용하여 해결할 수 있는 문제들도 증가하게 되었다. 점차 복잡해져 가는 대규모 조합의 최적화 문제와 제한 조건이 많은 이산(discrete) 문제들을 해결하는 데에는 전통적 최적화 방법이 한계를 보였다. 전통적 최적화 방법으로는 목적함수가 복잡한 경우에 수렴하는데 어려움이 있고, 또한 여러 개의 국부적(local) 최적해가 존재할 때는 구해낸 최적해가 전역적(global) 최적해인지 신뢰하기 힘들다.

이러한 경우의 문제들을 극복하기 위해서 등장한 것이 Holland[1]가 제안한 새로운 개념의 최적화 방법인 유전자 알고리즘(genetic algorithm-GA)이다. 그리고 유전자 알고리즘을 구조물 설계에 적용한 것은 Goldberg 와 Santani[1]가 처음으로 시작하였다.

본 논문에서는 유전자 알고리즘의 기본적인 개념과 유전자 알고리즘을 이용한 연구 등에 대해서 고찰하고자 한다. 그리고 유전자 알고리즘 이외에 현재에 부각되는 다른 최적화 기법들에 대해서도 간단한 소개를 하겠다.

II. 유전자 알고리즘

유전자 알고리즘은 생물 진화의 원리와 유전학에 바탕을 두고 있다. 우수한 형질을 가진 개체가 자연계에 잘 적응하여 우수한 자손들을 생성한다는 자연계의 법칙을 그대로 응용한 알고리즘인 것이다. 이러한 유전자 알고리즘은 전통적 최적화 방법(classical optimization)들과는 상당히 다르다.

전통적 최적화 방법들은 도함수(gradient)를 이용하고 한정된 공간에서 모든 점들의 목적함수(objective function)를 한번에 하나씩 탐색을 하던가, 어떤 임의의 점을 선택하여 탐색을 시작하는 방법들을 사용하였다. 그렇기에 전통적 최적화 방법은 좁은 영역에서 효율적이었다.

그러나 유전자 알고리즘에서는 도함수의 개념을 전혀 사용하지 않고, 방향성 있는 탐색과 확률 탐색을 수행하기 때문에 연속-불연속의 혼합, 불연속, non-convex 영역 등을 포함하는 최적화 문제를 해결할 수 있는 장점이 있다. 유전자 알고리즘은 이진수의 조합으로 구성된 개체(individual=염색체)들의 집단(population)을 가지고, 선택(selection), 교배(crossover), 돌연변이(mutation)라는 세 가지 과정을 수행함으로써 최적화를 하는 알고리즘이다. 세 가지 요소는 유전자 알고리즘에서 가장 기본이 되는 연산자(operator)들이다.

첫 번째로, 선택(selection)이란 각각의 스트링(string)들이 적합도(fitness)에 따라 복제되어지는 과정이다. 즉, 다음 세대로 진화하는 과정에서 높은 적합도를 갖는 스트링들의 특징을 다음 세대로 전

달하는 역할을 한다. 선택의 방법에도 여러 가지 모델들이 도입되었다.

Dejong[1]은 최고의 적합도를 발생시키는 염색체를 유지한다는 엘리트 모델(elitism)을 소개하였고, Brindle[1]은 잔여 확률 표본 선택법, Baker [1]은 확률적 균등 표본 선택을 이용하였다. 그 이외에도 개체군 중에서 일정한 개수의 개체를 임의로 선택하여 그 중에 최고의 적합도를 가지는 개체를 다음 세대에 넘겨주는 경쟁 (tournament) 선택도 있다.

둘째로, 교배 (crossover)란 선택된 염색체들의 스트링 일부를 서로 교환함으로써 정보를 교환하는 과정이다. 이 과정을 통해서 염색체들은 높은 최적치를 가질 수 있는 확률적 기회를 가지며, 수렴 속도를 가속화 시켜주는 역할을 하게 된다. 교배에서도 수렴 속도를 제어하기 위한 여러 가지 방법들을 제시하였다. 대표적으로 단순(simple) 교배, 복수점(multi-point) 교배, 균일(uniform) 교배 등이 있다.

마지막으로, 돌연변이(mutation)란 선택과 교배 과정을 통해 얻을 수 없는 정보를 획득하는 과정이다. 적당한 확률에서 이 돌연변이 과정은 전역적인 탐색 효과를 나타내므로, 높은 적합도로의 수렴성을 매우 높게 해준다. 교차와 돌연변이와 같은 연산자들은 유전자 알고리즘에서 집단의 다양성을 높이는 역할을 하게된다.

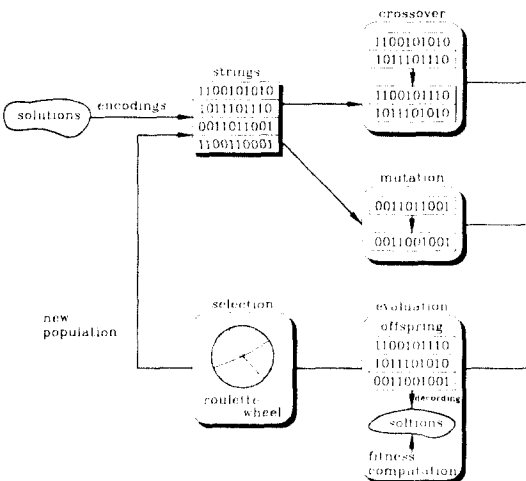


Fig. 1. 유전자 알고리즘 구조.

위와 같은 기본적인 연산자들로 구성된 유전자 알고리즘의 일반적인 구조를 형상화하면 Fig. 1과 같다.

III. 유전자 알고리즘의 개발

유전자 알고리즘도 여러 가지 조건의 최적 문제를 해결하는데 단점을 드러내었다. 그렇기에 이러한 단점을 보완하기 위한 방법들이 여러 연구자들에 의해서 오랫동안 개발되었는데, 그 중 하나가 돌연변이 확률에 관한 것이다.

유전자 알고리즘에서 적당한 돌연변이 확률은 수렴성을 높여주지만, 만약 너무 높은 돌연변이 확률을 사용하였을 경우는 진화의 후반부에 높은 적합도를 가진 개체를 모두 바꾸어 버리는 결과를 초래하기도 한다. 이같이 유전자 알고리즘의 중요한 요소인 돌연변이 확률을 선택하는데 있어서, 최적 시간의 효율성을 고려한 효과적인 변이가 발생하도록 개발한 것이 비균일 돌연변이(non-uniform mutation)이다. 비균일 돌연변이는 유전자 알고리즘이 정밀한 답을 제공하는데 필수적인 지역적 미세 조정 능력을 갖출 수 있도록 하기 위해서 사용된다. 돌연변이의 발생 위치에 따라 정해지는 전역적 정보를 돌연변이 과정 중 초기에는 전체 공간에 대해서 전역적 탐색을 수행하게 하고, 시간이(세대)이 지날수록 국소적 지역 탐색을 추구하게 하는 것이다.

그리고 개체집단의 크기(population size) 역시 유전자 알고리즘의 사용자가 결정해야 하는 가장 중요한 요소 중 하나이다. 만일 개체 집단의 크기가 너무 작으면 유전자 알고리즘이 너무 빨리 수렴하게 되므로, 전역적 최적해에 도달하지 못할 수 있다. 그리고 그 반대의 경우는 유전자 알고리즘이 수렴하는데 너무 많은 시간을 소모하므로 효율성이 저하된다. 이를 위해서 특정 염색체에 존재하는 세대수를 "나이"라는 개념으로 사용하여 자연적 소멸 현상을 표현하였다.

많은 연구자들이 각각 다른 측면에서 유전자 알고리즘의 개체 집단의 크기에 관한 연구를 하였다.

Ashish Ghosh[2]은 이러한 염색체들의 나이 개

념을 이용하여 non-station 환경에서의 함수를 최적화 하였고, Grefenstette[3]는 메타-유전자 알고리즘을 사용하여 다른 유전자 알고리즘의 매개변수를 제어하였다. Goldberg[3]는 최적 개체집단의 크기를 이론적으로 해석하였으며, 최근에는 Smith[3]가 선택 오차의 확률에 따라 개체 집단의 크기를 조정하는 알고리즘을 제안하였다.

유전자 알고리즘은 기본적으로 제한 조건이 없는 문제의 최대 값을 구한다. 그러므로 제한 조건이 있는 문제는 제한 조건이 없는 문제로 변환되어야 한다. 유전자 알고리즘에서는 제한 조건을 위반하는 개체를 처리하는 방법 중 하나로 제한 조건의 위반 정도에 따라 이들의 적합도(목적함수)를 감소시켜 주는 방법이 있다. 즉, 제한 조건의 위반 정도에 따라 벌점(penalty)을 부과하고 이 벌점을 유사 목적함수(pseudo objective function)에 반영함으로써 제한 조건이 없는 문제로 변환하는 것이다. 그리고 이 외에 Hajela[4]는 제한 조건을 다루기 위해서 전통적 범칙함수가 아닌 다른 방법을 제안하기도 하였다.

실제적인 구조물 최적설계에서는 하나 이상의 기준(criteria)이 존재하는 문제들이 있다. 즉 다중 목적함수로 표현되는 설계가 나타난다.

다목적함수 최적화 (multiobjective optimization)란 여러 개의 기준이 동시에 존재할 경우 이들을 하나의 기준으로 결합시킬 수 없는 경우의 문제를 최적화 하는 것을 말한다.

유전자 알고리즘은 도함수가 필요 없고, 단 한 개의 국부적 최적해 보다는 오히려 여러 개의 최적

해를 구하는 특성 때문에 다중목적함수 최적해를 구하는데 효과적이다.

Schaffer[1]는 VEGA(Vector Evaluated Genetic Algorithm)라는 다중목적함수 유전자 알고리즘을 개발하였다. 이는 simple-GA의 선택 과정에서 차례로 각각의 목적함수에서 균등의 선택을 수행함에 의해서 종속 세대(sub-population) 그룹을 생성하는 방법을 취하였다. Franklin Y. Cheng과 Dan Li[5]가 이 알고리즘을 응용하였다. Andrea M. Cook[6]은 다중목적함수 설계에 two-branch tournament 유전자 알고리즘을 사용하여 해석하기도 하였다.

그리고 이러한 다중목적함수나 다중모드를 위해서 niche와 같은 개념을 Cavicchio[1]의 논문에서 처음으로 유전자 알고리즘에 도입하였다. 유전자 알고리즘에서 niche는 각각의 염색체들이 가지는 적합도를 고려하여 생존률을 조정하는 방법이다. 이 niche 방법을 수행하기 위해서 Goldberg와 Richarson[1]은 Sharing Function을 도입하였다. Sharing 함수를 적용하면 가장 높은 최적해 뿐만 아니라 그밖에 다른 최적해의 분포도 알 수 있다. Fig. 2와 Fig. 3은 Sharing 함수의 작용에 대해 나타내 주고 있다.

Sharing function이란 각 세대에서 스트링에 대한 몫의 정도와 이웃을 결정하는 함수이다. S. Ramberger, Russenschuck[7]는 LHC(Large Hardron Collider)에 대한 초전도 자기 개념 설계(superconducting magnets conceptual design)를 하는데, niche 개념을 도입하였다.

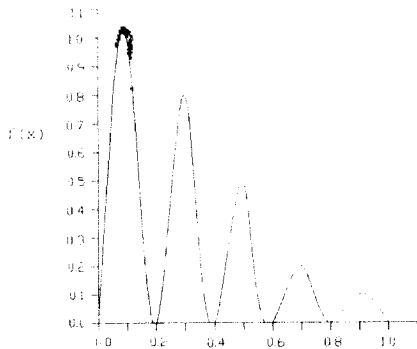


Fig. 2. sharing을 적용하지 않은 경우.

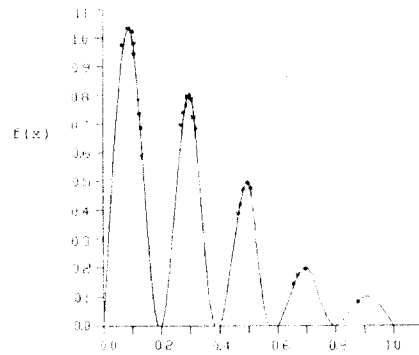


Fig. 3. sharing을 적용한 경우.

최적설계를 하는데 있어 수렴성은 매우 중요하다. 이 문제를 해결하기 위해서 많은 연구자들이 다른 측면에서 수행하였다. Goldgerg와 Sefrest[3]은 유전자 알고리즘을 유한 마르코프 체인(Markov chain)으로 분석하였고, Davis와 Principe[3]은 simulated annealing의 기존 이론을 토대로 마르코프 체인 유전자 알고리즘에 확장하는 가능성을 검토하였다. 그런가하면 Eiben, Aarts 그리고 Van Hee[3]는 유전자 알고리즘과 simulated annealing을 통일하는 추상 유전자 알고리즘(Abstract Genetic Algorithm)을 제안하기도 하였다. Banach 정점 정리를 이용한 수축 사상 유전자 알고리즘(Contractive Mapping Genetic Algorithm)도 이러한 방법들 중 하나이다.

IV. 유전자 알고리즘이 적용된 연구

유전자 알고리즘은 그 특성인 단순성과 일반성 때문에 여러 가지 문제에 적용되고 있다.

Goldberg[1]가 simple-GA로 mass spring dashpot 시스템의 설계를 시작으로 현재까지 많은 연구가 진행되어 왔다. 구체적으로 가스 파이프라인의 최적화, 순회 판매원 문제, 로봇의 행동 진화, 신경 망의 학습, fuzzy membership 함수의 최적화 등이 있으며 그 응용 범위의 한계를 지을 수 없는 실정이다. 유전자 알고리즘에 의한 연구들을 살펴 보면 다음과 같다.

Peter N. Harrison[8]은 composite stringer stiffened panel을 유전자 알고리즘을 이용하여 설계하였다. 그러나 이 문제에서는 유전자 알고리즘만으로는 매우 많은 해석을 요하기 때문에, 최적화 과정 중에 critical failure margin에 대하여 response surface approximations를 사용함으로써 효율성을 높여 주었다. 이 연구에서는 approximations가 유전자 알고리즘에 제약을 가해 더 좋은 수렴성을 가지도록 하는 역할을 하였다.

B.P. Wang[9]은 섬유로 강화된 복합재료가 냉각 되어질 때, 유도되는 잔류응력으로 인해서 나타나는 spring-in 현상을 최소화하는데 유전자 알고리즘을 도입하였다.

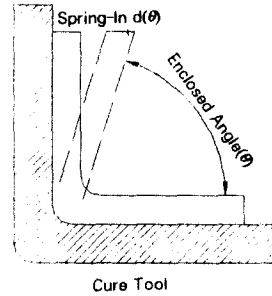


Fig. 4. Spring-In Phenomenon.

그리고 Hajela[10]는 설계변수가 non-convex한 multidisciplinary rotor blade를 유전자 알고리즘을 이용하여 설계하였는데, 이 연구에서 유전자 알고리즘의 특성을 확연히 보여 주었다.

팽창 가능한 구조물(inflatable structure)에 대해서도 연구가 이루어 졌다. 가벼운 무게와 낮은 비용 때문에 팽창 가능한 구조물은 우주 구조물용으로 커다란 잠재력(potential)을 지니고 있는데, 이 구조물에 대한 연구는 1960년대에 인공위성 Echo series에서 시작되었음에도 불구하고, 아직

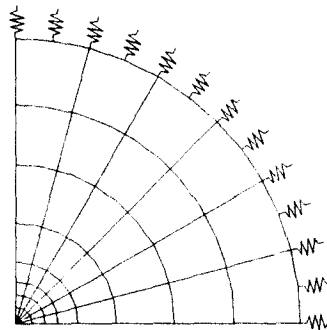


Fig. 5. Finite Element Model.

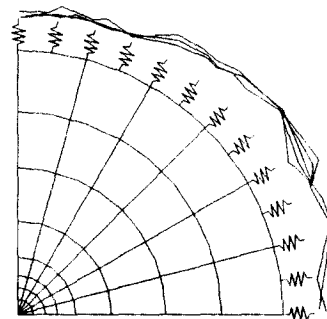


Fig. 6. Boundary Displacements for Best GA

까지는 그들의 사용이 보급되지 않았다. J.A. Bishop[11]은 이러한 팽창성 있는 구조물을 유전자 알고리즘을 이용하여 해석하였다. Fig. 5.와 Fig. 6.는 J.A. Bishop이 해석한 막(membrane) 구조물이다.

Rodolphe Le Riche[12]는 좌굴 하중(buckling load)에 대해 laminate stacking sequence를 최적화 하였다. 이 최적화에서는 유전자 알고리즘에 새로운 유전 연산자 (genetic operator) 인 permutation 을 제안하여 그 효율성을 입증하였다. permutation 연산자는 Boyang Liu[13]가 stacking sequence를 최적화 하는 데에도 사용하였는데, Liu는 이 연구에서 새로운 permutation 방법인 Gene-Rank GA 방법을 사용하여, 기존의 partially Mapped GA permutation 방법과 비교하였다.

IV. 그 밖의 최적화 방법

이 절에서는 유전자 알고리즘 이외의 새로운 최적화 방법들에 대해서 간단히 소개하겠다. Optimality Criteria Methods는 Prager[14]와 그 동료들에 의해서 개발되어 Venkayya, Knot[14]에 의해서 발전된 최적화 방법이다. 그러나 이들은 연속계에 대해서만 적용하였기에, 이를 Berke [14]가 불연속계에 대해서 발전시켰다. 특정한 설계 조건들에 대한 최적의 필요 조건을 사용하는 이 반복 기법은 특히 구조물에 관련된 최적화 문제에서 효과적이다.

Simulated annealing은 국소적 최소해로부터의 천이(transition)가 가능한 목적함수를 무작위로 평가하여 최적해를 구하는 조합형 최적화 방법이다. 비록 이 방법은 최적해를 구하기 위해서는 매우 많은 함수를 평가해야 되는 단점을 가지고 있으나, 나쁜 (ill-conditioned) 제한 조건의 함수에서도 높은 정확성을 가지고 전역적 최적해를 구해낼 수 있다는 강한 장점을 지니고 있다. 이 방법은 이름에서 알 수 있듯이 가열된 고체를 천천히 조절하여 냉각시킬 경우 매우 낮은 내부 에너지 상태의 결정 구조를 갖는다는 열역학적 개념을 응용한 것이다.

이러한 simulated annealing은 다음과 같은 몇 가지 특성을 나타낸다.

- 1) 함수나 제한 조건 평가에 있어 불연속적 성질을 도입하기 때문에, 수렴이나 천이 같은 특징들이 함수의 연속성에 영향을 받지 않는다.
- 2) 수렴은 feasible 영역의 convexity 상태에 영향을 받지 않는다.
- 3) 설계 변수가 반드시 양일 필요가 없다. simulated annealing은 유전자 알고리즘과 같은 확률적(stochastic) 방법의 하나이다.

최근 몇 년 동안 최적해를 구하기 위해서 neural computing 전략이 적용되었다. Neural Network이란 상호 연결된 단순 프로세서(neuron)의 거대한 병렬 네트워크(parallel network)이다.

각각의 뉴런은 다른 뉴런으로부터 정보를 받아 결과 절점(output node)으로 진행되는 결과를 산출한다. Neural Network는 뉴런, 뉴런간의 상호 연결성(connectivity) weight, 각 뉴런의 활동 함수로 구성되어져 있다. Fig. 7.은 한 개의 뉴런으로 구성된 입력값(input)과 결과값(output)의 관계를 나타내고 있다. 이러한 네트워크에 기반을 둔 최적화 방법이 Neutral-Network-Based Optimization이다.

그리고 실제적인 구조물에서는 목적함수나 제한 조건, 설계 데이터들이 모호한데, 이러한 문제를 해

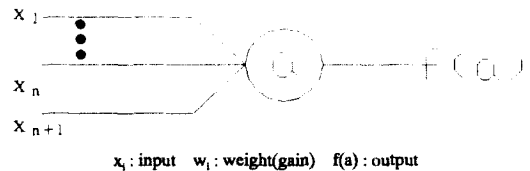


Fig. 7. Single Neuron and its Output.

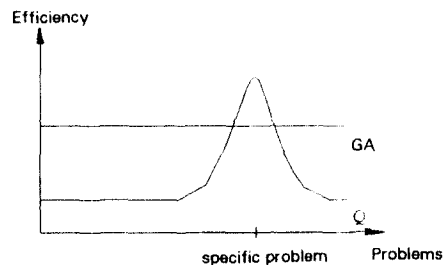


Fig. 8. 유전자 알고리즘 성능 비교.

결하는 방법으로는 Fuzzy Optimization Method 를 사용한다. 위에 소개된 최적화 방법들도 각각 나름대로의 장점들을 지니고 있기 때문에, 현재 많은 연구가 이루어지고 있다.

V. 맺은 말

지금까지 유전자 알고리즘에 대해 고찰하였고, 여러 연구 결과를 통해서 그 성능을 확인할 수 있었다. 전통적 최적화 방법(Q)과 유전자 알고리즘(GA)의 성능을 비교하면 Fig. 8과 같다.

아직까지 국내에서는 유전자 알고리즘에 대한 연구가 초기 단계에 머물러 있는 상태이나, 외국에서는 많은 연구에 적용되고 있다. 그리고 유전자 알고리즘이 전통적 최적화 방법보다는 많은 영역의 문제들을 다룰 수는 있으나, 제한 조건과 같은 여러 요소들에 의해서 성능이 저하되는 현상을 극복하기 위해서는 앞으로보다 많은 연구가 필요하다.

참 고 문 헌

1. David E. Goldberg, *Genetic Algorithms in Search, Operation, and Machine Learning*, Addison-Wesley Publishing Company, Inc., 1989.
2. Ashish Ghosh, Shigeyoshi Tsutsui and Hid-oo Tanaka, "Function Optimization in Nonstationary Environment using Steady State Genetic Algorithms with Aging of Individuals", IEEE, 1998, pp. 666-671.
3. Zbigniew Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 1996.
4. P. Hajela and J. Yoo, "Constraint Handling In Genetic Search - A Comparative Study", AIAA-95-1143-CP, 1995, pp. 2176-2186.
5. Franklin Y. Cheng and Dan Li, "Genetic Algorithm Development for Multiobjective Optimization of Structures", AIAA Journal, Vol. 36, No 6, June 1998, pp. 1105-1112.
6. Andrea M. Cook, "Using the Two-Branch Tournament Genetic Algorithm for Multiobjective Design", AIAA 98-1914, 1998, pp. 1752-1762.
7. S. Ramberger and S. Russenschuck, "Genetic algorithm with niching for conceptual design studies", COMPUMAG, 1997, pp. 509-510.
8. Peter N. Harrison and Rodolphe Le Riche, "Design of Stiffened Composite Panels By Genetic Algorithm and Response Surface Approximations", AIAA-95-1163-CP, 1995, pp. 58-68.
9. B.P. Wang, C.H. THo and M. Henson, "Minimization of Spring-In Composite Angle Components By Genetic Algorithm", AIAA-97-1338, 1997, pp. 2733-2738.
10. P. Hajela and J. Lee, "Genetic Algorithms In Multidisciplinary Rotor Blade Design", AIAA-95-1144-CP, 1995, pp. 2187-2197.
11. J. A. Bishop, "Shape Correction of Initially Flat Inflated Membranes by A Genetic Algorithm", AIAA-98-1984, 1998, pp.2309-2312.
12. Rodolphe Le Riche and Raphael T. Haftka, "Optimization of Laminate Stacking Sequence For Buckling Load Maximization By Genetic Algorithm", AIAA Journal, Vol. 31, No 5, May 1993, pp. 951-956.
13. Boyang Liu, Raphael T. Haftka and Mehmet A. Akgum, "Permutation Genetic Algorithm For Stacking Sequence Optimization", AIAA-98-1830, 1998, pp. 1141-1152.
14. Singiresu S. Rao, *Engineering Optimization*, John Wiley & Sons, Inc., 1996.
15. Mitsuo Gen and Runwei Cheng, *Genetic Algorithms and Engineering Design*, John Wiley & Sons, Inc., 1997.
16. Raphael T. Haftka and Zafer Gurdal, *Ele-*

ment of Structural Optimization, Kluwer Academic Publishers, 1992.

17. Garret N. Vanderplaats, *Numeriacal Optimization Techniques For Engineering Design*, MacGraw-Hill, Inc., 1984.

18. D. Barrios, L. Malumbres and J. Rios, "Convergence Conditions of Genetic Al-

gorithms", *Computer Math.*, Vol. 68, 1997, pp. 231-241.

19. Akira Todoroki, "Stacking Sequence Matching By Two-Stage Genetic Algorithm With Cansang-uineous Initial Population", AIAA-97-1228, 1997, pp. 1297-1302.

