

유전자 알고리즘을 이용한 토큰버스 네트워크의 타이머 선정

(Selection of Timer for Token Bus Automation Networks with Genetic Algorithm)

(*)이 상 호, 이 경 창, 김 인 준, 이 석

부산대 대학원 생산기계공학과, 부산대 공대 생산기계공학과

ABSTRACT

This paper focuses on development of a timer selection algorithm for IEEE802.4 token bus networks to serve large-scale integrated manufacturing systems. Such factory automation networks have to satisfy delay constraints imposed on time-critical messages while maintaining as much network capacity as possible for non-time-critical messages. Therefore, the object of this paper is to develop timer selection algorithm to minimize a certain penalty function. This paper presents an algorithm based on Genetic Algorithm. The efficacy of the algorithm has been demonstrated by a series of simulation experiments.

1. 서론

컴퓨터는 생산의 자동화와 관리를 포함한 산업의 거의 모든 분야에서 필수적인 요소가 되었지만, 이와 같이 적용된 자동화가 전체적인 생산 과정을 총체적으로 고려한 것이 아니라, 독립적인 생산 과정에 초점을 두어 구성되었기 때문에 생산성의 향상에도 한계가 있었다. 따라서, 대규모 생산시스템에서 여러가지 독립된 형태의 생산자동화를 유기적으로 연결시켜 주기 위한 노력이 진행되었으며, 이것이 컴퓨터 네트워크(computer networking)에 의한 컴퓨터 통합생산(Computer Integrated Manufacturing, CIM)[1]의 일환이다. 즉, 제한된 공간 내에 분산되어 있는 여러 컴퓨터와 생산 장비들을 공통된 전송 매체로 연결하여 상호 간의 정보 교환을 가능하게 하는 것이다.

대규모 생산 시스템에서는 Fig.1과 같이 다양한 종류의 데이터가 존재한다[2-3]. 즉, 작업장 레벨에서는 여러 생산장비들의 상태를 알리는 센서신호나 제어신호 등과 같은 짧은 메시지들이 주기적으로 빠른 통신을 요구하는 특성이 있으며, 상위레벨에서는 CAD파일이나 문서파일과 같은 비교적 긴 메시지들이 비주기적으로 전송되는 특성이 있다.

따라서, 대규모 통합 생산시스템에 컴퓨터 네트워크를 설계하려고 할 때, 시간에 대한 긴급성을 요구하는 메시지들은 전송지연

에 관한 요구조건을 만족하도록 설계하는 것이 중요하다. 컴퓨터 네트워크의 설계시에는 토폴로지(topology)와 전송매체, 통신규약(communication protocol)을 선정할 후, 프로토콜 변수를 적절히 설정하게 된다.

이러한 변수의 선정과정에서 네트워크의 통신부하에 대한 통계적인 예측과 전송속도, 통신규약의 특징 등이 고려되며 선정된 값에 따라 전송지연과 throughput과 같은 중요한 성능지표가 직접적인 영향을 받는다[4].

그러나, 프로토콜 변수와 네트워크 성능사이의 일반적인 관계가 잘 알려져 있지 않기 때문에, 네트워크 전문가가 그의 경험을 바탕으로 이러한 변수를 선정하는 것이 일반적이었다.

본 연구의 목표는 자연계에서 관찰되는 적자생존의 유전법칙에 바탕을 두고 있는 유전자 알고리즘(Genetic Algorithm)을 이용하여 네트워크의 환경에 적합한 프로토콜 변수를 선정할 수 있는 체계를 확립하는데 있다. 이러한 유전자 알고리즘은 전체의 탐색 공간상에서 여러개의 염색체(chromosome)를 가지고 동시에 탐색을 수행하기 때문에, 전역 최적값으로 수렴할 가능성이 크다. 또한, 미분 불가능이나 멀티모달(multi-modal), 비선형과 같은 어려운 문제에도 강인한(robust) 특성을 발휘한다[5].

2. IEEE 802.4 토큰버스 프로토콜

공장 자동화용 표준 통신 프로토콜인 MAP(Manufacturing Automation Protocol)에서는 매체 접속 제어(Medium Access Control, MAC)의 표준으로 IEEE 802.4 토큰버스 프로토콜을 선정하였다.

토큰 버스 방식은 버스(Bus) 구조상에 부착된 스테이션들이 논리적 링(Logical Ring)을 형성하여 상호 통신하는 방식으로 토큰을 획득한 스테이션은 규정된 시간동안 매체의 제어권을 갖는다. 또한, 각 스테이션은 Fig.2와 같이 네개의 카운트-다운 타이머, 즉 Token Holding Timer(THT), Token Rotation Timer(TRT4, TRT2, TRT0)와 네개의 큐를 갖는다. 이와 같은 것은 Fig.3과 같이 우선 순위6의 메시지가 THT의 시간동안 전송될 수 있는 반면, 제 4

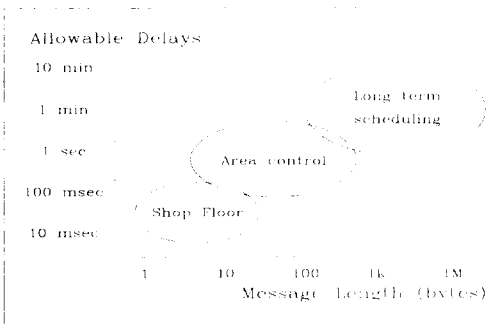


Fig.1 Disparate Data Groups

지의 하위 순위의 메시지는 해당 TRT에서 토큰이 논리적 링을 순환하는데 걸린 시간을 삭감한 시간동안 전송할 수 있다. 또한 IEEE802.4 표준에서는 $TRT4 \geq TRT2 \geq TRT0$ 이 만족되는 타이머 선정을 요구한다.

이와 같은 우선순위 도구의 목적은 한정된 네트워크 용량을 상위 우선순위를 갖는 메시지에 우선적으로 할당하고, 하위 우선 순위의 메시지는 여분의 용량이 있을 때만 전송하도록 하는 것으로, 메시지의 전송 지연에 관한 제약조건을 만족시켜 네트워크의 효율적인 운영을 가능하게 한다[4].

3. 네트워크 시뮬레이션 모델

IEEE802.4 토큰버스 프로토콜은 SIMAN을 이용하여 이산 사상 시뮬레이션(Discrete Event Simulation)으로 모델링 되었으며, Fig.4

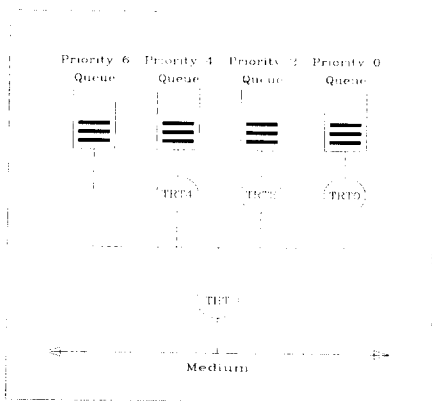


Fig.2 Schematic Structure of Station

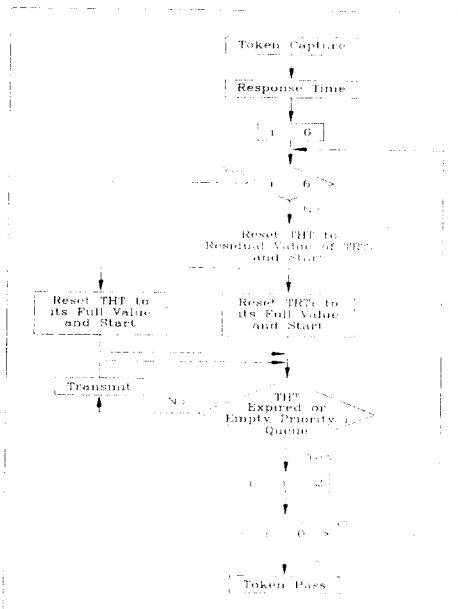


Fig.3 Flow Chart of Priority Mechanism

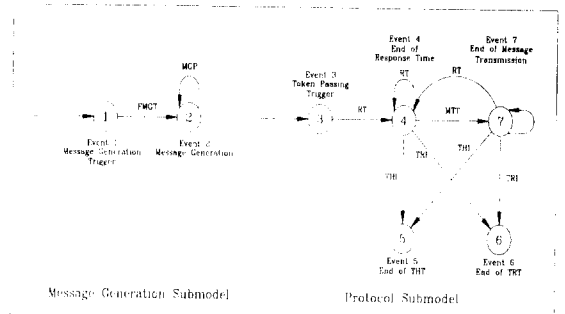


Fig.4 Event Diagram for Message Generation and Protocol Submodel

와 같이 메시지발생 서브모델(Message Generation Submodel)과 프로토콜 서브모델(Protocol Submodel)로 나뉘어진다[4]. 메시지발생 서브모델(MGS)은 사용자가 정의한 확률분포에 따라 네트워크 트래픽(메시지 발생간격, 메시지 길이)을 발생시키고, 프로토콜 서브모델(PS)은 스테이션의 상태에 따라 프로토콜 로직을 실행시킨다.

Table 1은 IEEE802.4 토큰버스 프로토콜 시뮬레이션 모델의 각 우선순위에 대한 파라미터를 보인다. 시뮬레이션 모델의 스테이션 개수는 10개이며, 각 스테이션에는 4개의 우선순위를 위한 큐(Queue)가 있다.

최상위 우선순위(Priority 6)는 센서신호와 제어신호 등을 주로 전송하는 작업장 레벨을 위한 것으로, 이 레벨에서는 비교적 일정한 간격의 메시지 발생 분포와 대체로 균일한 메시지 길이를 가지는 특성이 있다. 그리고, 다음의 상위 우선순위들(Priority 4 과 Priority 2)은 생산주문이나 신제품에 관한 텍스트 파일(text file)과 NC 공작기계의 프로그램 등을 송수신하는 것으로, 우선순위4는 비교적 일정한 메시지 간격과 길이를 가지는 경우로써, 균일분포로 하였으며, 우선순위2는 메시지 간격과 길이에 있어 불규칙한 특성을 반영하기 위하여 지수분포로 설정하였다. 그리고, 우선순위0은 장기계획이나 CAD파일, 공정총괄 등을 전송하기 위한 것이며, 비교적 긴 메시지 길이와 발생간격을 지수분포에 의해 일어나도록 설정하였다.

	THP	TRT4	TRT2	TRT0
Number of Station		10		
Number of Queue per Station		4		
Type of Probability Distribution for Message Generation Interval		uniform	uniform	expon. expon.
Average Message Generation Interval(μsec)	10^4	5×10^4	10^5	5×10^6
Type of Probability Distribution for Message Length		uniform	uniform	expon. expon.
Average Message Length(bit)	2×10^3	10^4	2×10^4	10^6

Table 1. Simulation Parameter Condition

4. 유전자 알고리즘에 의한 프로토콜 변수의 선정

불연속적이거나 멀티모달(multi-modal), 비선형 등과 같은 복잡

한 문제에 있어서 제한된 자원(resource)을 가지고 최적 혹은 준 최적의 해를 구하는 데는 주로 휴리스틱(heuristic) 알고리즘이 사용된다. 하지만, 대부분의 휴리스틱한 알고리즘들은 주어진 문제나 영역(domain)에 의존하여 그 적용범위가 매우 제한적일 뿐 아니라 국부적인 최적(local optima)에 빠져들어 해의 질이 만족할 만한 수준이 되지 못하는 경우가 많다는 결점을 지니고 있다.

유전자 알고리즘도 휴리스틱 알고리즘의 일종이지만, 자연계에서 관찰되는 적자생존의 유전법칙에 근간을 두고 있어 휴리스틱 알고리즘의 한계를 극복할 수 있는 대안으로 제시된 매우 강인한(robust) 알고리즘이다. 특히, 유전자 알고리즘은 문제에 대한 지식을 얻기 힘든 난이도가 높은 문제일수록 더욱 그 효과를 발휘하는 것으로 알려져 있다[5-8].

유전자 알고리즘의 연산자(operator)는 재생산(reproduction), 교배(crossover), 돌연변이(mutation) 등을 들 수 있는데, 재생산은 높은 적합도(fitness)를 가진 염색체(chromosome)가 다음 세대에서도 많은 자손(offspring)을 구성할 수 있도록 하는 것이다. 그리고, 교배는 부모 염색체의 임의 교배점을 기준으로 스트링(string)을 교환(exchange)시키는 것을 의미하며, 돌연변이는 염색체 내의 임의로 선택된 점을 비트 전환하는 것이다.

IEEE802.4 토큰버스 시뮬레이션 모델의 프로토콜 변수인 타이머들의 선정을 위한 유전자 알고리즘의 적용 형태는 Fig.5와 같다. 먼저, 정의된 염색체의 길이(chromosome length)를 고려하여 랜덤 발생기에 의해 초기 모집단을 형성한다. 모집단의 각 염색체는 토큰버스 시뮬레이션 모델을 통하여 벌칙(penalty)을 구하며, 각 염색체에 대한 평가(evaluation)를 수행하여 양질의 염색체들을 재생산(reproduction)한다. 그리고, 선택(selection) 모듈에 의해 임의로 선택된 두개의 부모 염색체에서 확률에 따라 교배와 돌연변이를 수행하여 새로운 개체군을 형성함으로써 진화해 간다.

Fig.6은 이상의 네트워크 프로토콜 변수를 선정하기 위한 유전자 알고리즘(Genetic Algorithm Parameter Selection, GNPS)에서의 교배와 돌연변이 연산을 보이는 것으로, 교배는 각 타이머의 영역을 하나의 소 염색체(sub-chromosome)로 하여 각각의 소 염색체에서 교배점이 임의로 발생하여 수행되도록 하였다.

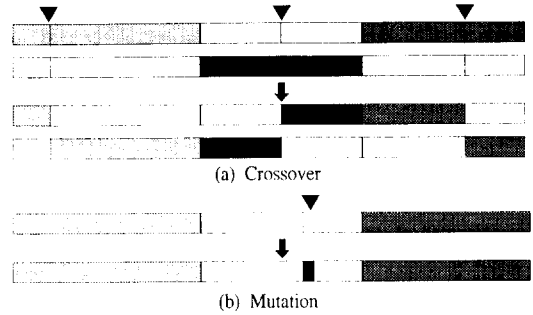


Fig.6 Crossover and mutation of chromosome for GNPS

	THT	TRT4	TRT2	TRT0	
population size		20			
crossover probability		0.9			
mutation probability		0.01			
chromosome length (bit)	fixed (1000 μ sec)	13	13	13	
fitness function (μ sec) (penalty P)	θ 6E3	2E4	1E5	1.6E6	
	b 2000	2000	1000	500	

Table 2. Simulation Parameter Condition

이와 같은 유전자 알고리즘을 수행하기 위한 각 파라미터들은 Table 2와 같으며, 모집단에 속하는 염색체의 수는 20개이다. 또한, 교배 확률과 돌연변이 확률은 각각 0.9와 0.01로서, 모집단의 크기가 30개 이하인 경우에 일반적으로 추천되는 값이다[5].

적합도 함수는 IEEE802.4 토큰버스 네트워크에 있어서 전송된 각 메시지의 지연에 근거하여 계산하였다. 여기서, $i(6,4,2,0)$ 는 우선순위, δ_j^i 는 우선순위 i 의 j 번째 메시지의 전송지연, $P_i(\delta_j^i)$ 는 우선순위 i 의 j 번째 메시지에 해당하는 벌칙으로서 다음 식과 같이 정의하였다.

$$P_i(\delta_j^i) = \begin{cases} 0 & \text{if } \delta_j^i \leq \theta_i \\ (\delta_j^i - \theta_i)^2 & \text{if } \theta_i < \delta_j^i \leq \theta_i + b_i \\ b_i^2 & \text{if } \delta_j^i > \theta_i + b_i \end{cases}$$

여기서, θ_i , b_i 는 각각 우선순위가 i 인 메시지의 penalty threshold와 penalty band이다. 이와 같이 각각의 메시지에 대한 벌칙을 평균하여 적합도 함수를 구성하였다.

그리고, 염색체 길이는 전체 39bit이며, 이것은 타이머 값은 이진수로 코딩한 것으로 각 TRT들에 대해 13bit으로 할당하여 타이머가 최고 8192 μ sec까지 조정될 수 있도록 하였다. 또, 토큰버스 시뮬레이션으로부터 THT의 변화가 전송지연에 크게 영향을 미치지 않는다고 밝혀졌기 때문에 1000 μ sec로 고정하여 수행하였다[11].

5. 실험결과와 고찰

본 실험은 IEEE 802.4 토큰버스 네트워크의 최소 벌칙(전송지연)을 탐색하기 위한 것으로, 유전자 알고리즘이 환경에 적응하

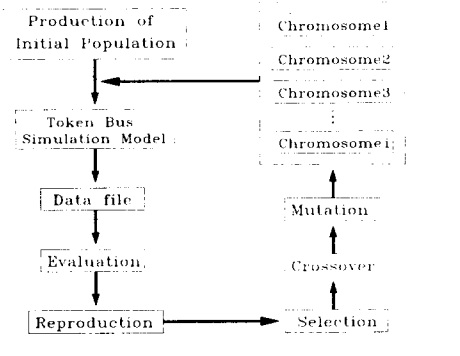


Fig.5 Configuration of GNPS operation

여 적절한 프로토콜 변수를 선정하는가에 대한 실험이다.

GNPS에서 새로운 개체군이 유전자 알고리즘의 연산자에 의해 부모 세대보다 더 나쁜 결과를 초래하는 것을 방지하기 위하여, 부모 세대의 가장 우성의 염색체를 교배나 돌연변이 없이 다음 세대에 재생산하는 엘리트 모델(Elite Model)을 적용하여 실험을 수행하였다.

실험은 두 가지의 경우로 수행하였는데, 80%의 트래픽에서 단

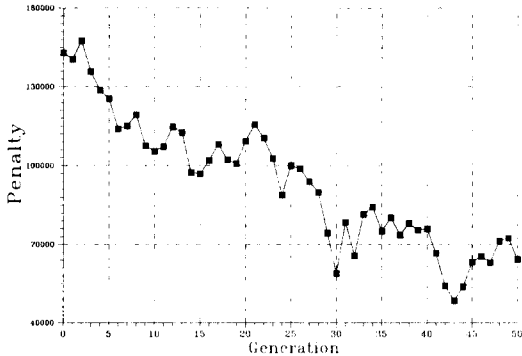


Fig.7 Average Penalty from Single Seed Experiment

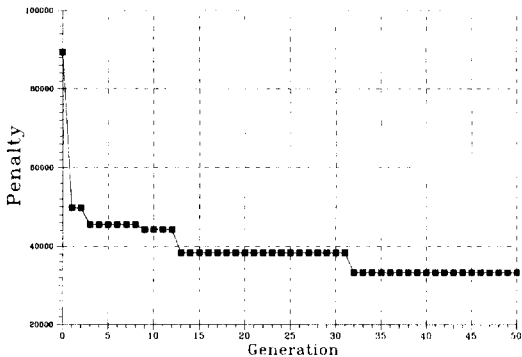


Fig.8 Minimum Penalty from Single Seed Experiment

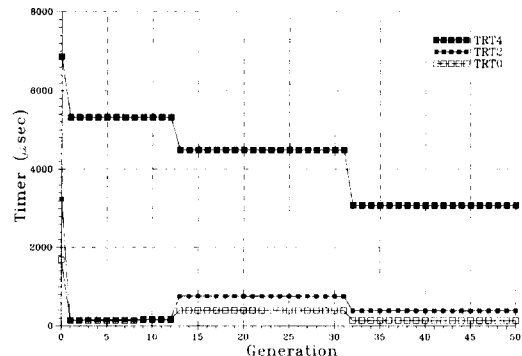


Fig.9 Timer Settings for the Minimum Penalty with Multiple Seed Experiment

일 씨드(single seed)를 통한 GNPS의 유효성을 검증하고, 이를 바탕으로 동일 트래픽을 기초로 한 51가지의 랜덤 씨드(random seeds)를 이용하여 네트워크의 시변성을 반영한 실험을 수행하였다. 단일 씨드로 실험을 수행하는 경우에는 매 세대마다 벌칙을 계산하는 시뮬레이션이 완전히 동일하기 때문에 유전 알고리즘의 효과를 쉽게 알아볼 수 있다. 한편, 랜덤 씨드를 이용한 실험에서는 시뮬레이션의 입력인 통계적 특성이 세대간에 서로 동일하지만, 사건들이 일어나는 순서와 시간들이 다르기 때문에 보다 현실적인 상황을 상정한 것이다.

Fig.7은 단일 씨드 하에서의 각 세대의 평균 벌칙을 보인다. 평균 벌칙이 초기에 $1.4E5 \mu\text{sec}^2$ 이상의 값을 가지나, 세대를 거쳐 감에 따라 급격히 감소하여 50 세대 지난 후에는 $6E4 \mu\text{sec}^2$ 정도의 값을 가져, 55% 이상 감소하였다. Fig.8은 각 세대에서 최소의 벌칙을 보이고 있으며 Fig.9는 이에 해당하는 타이머의 값들을 보이고 있다. 우선 타이머 값과 최소 벌칙의 값의 변화가 별로 나타나지 않고 있는데 이는 엘리트 모델의 특성을 그대로 나타내고 있다. 두 번째 세대에서 최소 벌칙과 대응하는 타이머 값들을 살펴보면 벌칙에서 큰 비중을 차지하는 우선순위 4의 전송지연을 감소시키기 위하여 TRT2와 TRT0이 급격히 감소하여 토큰만을 순환하는데 걸리는 시간인 토큰 순환 시간(Token Circulation Time, TCT)에 근접하는 것을 관찰할 수 있다. 50 세대가 지난 후에 TRT4와 TRT2, TRT0이 각각 3083, 387, $138 \mu\text{sec}$ 에 도달하였다. 이러한 결과를 토대로, GNPS가 IEEE 802.4 토큰버스 프로토콜의 우선순위 도구의 의도를 만족시키는 타이머 값을 선정할 수 있다는 가능성을 확인하였다.

다음은 51가지의 랜덤 씨드에서 수행된 것으로, 이 경우는 씨드의 영향으로 변동 폭이 상당히 크므로 누적 평균(cumulative average)을 적용하였다.

Fig.10은 각 세대의 평균 벌칙을 보이고 있다. 2세대에서 씨드의 영향으로 벌칙이 큰 폭으로 증가하는 것을 했다. 그러나, 전반적인 추세는, 누적평균이 보여주는 것 같이 감소하는 경향을 보이므로 GNPS가 의도한 바대로 작동함을 알 수 있다. Fig.11은 각 세대 별 최소 벌칙을 보이고 있으며 Fig.12는 이에 대응하는 타이머의 값들을 보여준다. Fig.11에 따르면 비록 최소 벌칙의 변동이 있으나 누적평균에서 보이는 것과 같이 전반적으로 감소하는 추세를 나타내고 있어서 GNPS의 유효성을 입증하고 있다. 50세대를 거치면서 약 60%정도의 감소를 보이고 있다. 최종 타이머 값들을 살펴보면 TRT4, TRT2, TRT0이 각각 3402, 322, $287 \mu\text{sec}$ 에 도달하여 단일 씨드와 별다른 차이를 보이지 않고 있다. 40세대 이후의 TRT4의 값이 크게 변동하는 것을 관찰할 수 있는데 이는 랜덤 씨드의 불확실성과 한정된 시뮬레이션 길이에 의한 영향이라고 생각된다.

6. 결론

본 연구에서는 공장자동화용 표준통신 프로토콜인 IEEE 802.4 토큰버스 프로토콜의 네트워크 성능변수의 조정을 위해 엘리트 모델의 유전자 알고리즘을 기초로 하여 작성된 GNPS (Genetic

Network Parameter Selection)를 이용하여, 그 성능을 SIMAN에 의해 개발된 토큰버스 통신망 시뮬레이션 모델을 통해서 검증하여 다음과 같은 결론을 얻었다.

1) 단일 씨드를 사용한 실험을 통하여 50세대 이후 벌칙이 55% 감소하는 것을 관찰하였으며, 네트워크의 타이머 값들을 선정할 수 있었다.

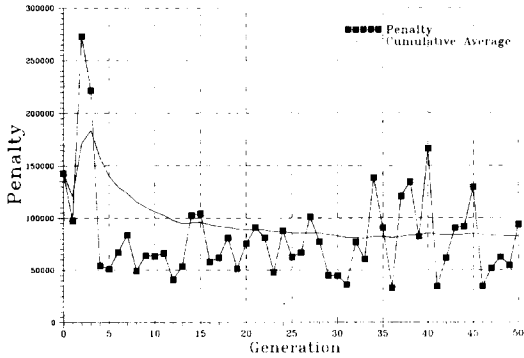


Fig.10 Average Penalty Settings from Multiple Seeds Experiment

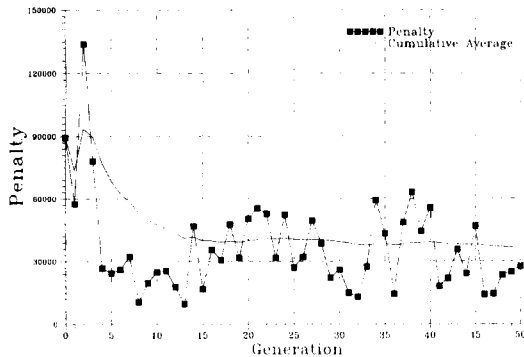


Fig.11 Minimum Penalty from Multiple Seed Experiment

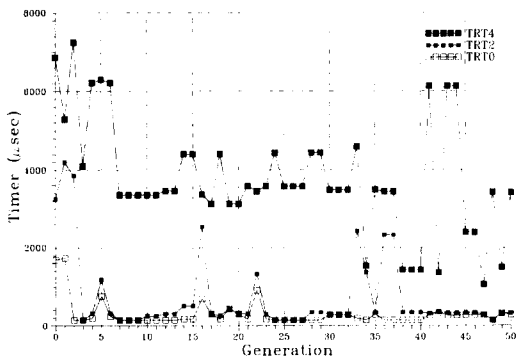


Fig.12 Timer Settings from the Minimum Penalty with Multiple Seed Experiment

2) 랜덤 씨드를 사용한 실험을 통하여 50세대 이후 벌칙이 60% 감소하는 것을 관찰하였으며, 환경이 변화하는 네트워크에서도 GNPS는 네트워크의 타이머 값들을 선정하는데 효과적으로 사용될 수 있음을 알 수 있었다.

3) 본 연구의 두 실험 결과 엘리트 모델을 이용한 유전자 알고리즘은 토큰버스의 타이머 선정에 효과적인 성능을 가지고 있음을 알 수 있었다.

향후의 연구 과제로서는 GNPS에 THT를 포함한 네 가지 타이머 선정에 대한 연구와, 유전자 알고리즘의 연산자에 대한 보강, 다양한 트래픽에 대한 검증, 컴퓨터 연산시간의 감소 등을 들 수 있다.

참고문헌

1. J.T. O'Rourke, "A Case for Computer Integrated Manufacturing," Keynote Address, NSF Workshop on Computer Networking for Manufacturing System, 1987.
2. W.L. Genter, K.S. Vastola, "Performance of the Token Bus for Time Critical Messages in a Manufacturing Environment," American Control Conference, 1989.
3. IEEE Computer Society, "Information processing systems-local area networks-Part4", IEEE Inc., 1990.
4. Suk Lee and Asok Ray, "Performance Management of Multiple Access Communication Networks", IEEE Jour. on SELECTED AREAS IN COMMUNICATION, Vol.11, No.9, pp.1426-1437, DEC. 1993.
5. 이상호, 윤정아, 이석, "큐용량과 시간할당에 의한 토큰버스 네트워크의 퍼지 성능관리기," 한국정밀공학회집, '95년도 추계 학술대회 논문집, pp.664-669, Nov., 1995
6. D. E. Goldberg, "Genetic Algorithms; In Search, Optimization & Machine Learning", Addison-Wesley Publishing Company Inc. 1989
7. K. A. DeJong, "An Analysis of the Behavior of a Class of Genetic Adaptive Systems", Dissertation Abstracts International 36, pp102, 1975
8. M. Srinivas, Lalit M. Patnaik, "Genetic Algorithms; A Survey," IEEE COMPUTER, pp.17-26, June 1994
9. J. J. Grefenstette "Optimization of Control Parameters for Genetic Algorithms", IEEE Trans. Systems, Man, and Cybernetics, Vol SMC-16, No 1, pp122-128, Jan/Feb 1986
10. M. Srinivas, Lalit M. Patnaik, "Adaptive Probabilities of Crossover and Mutation in Genetic Algorithms", IEEE Trans. Systems, Man, and Cybernetics, April, 1994
11. K. A. DeJong and W. M. Spears, "An Analysis of the Interfacing Roles of Population Size and Crossover in Genetic Algorithms", Proc. First Workshop Parallel problem Solving from Nature, pp38-47, Springer-Verlag, Berlin, 1990