

# 실시간 네트워크 시스템의 이용률 최적화를 위한 태스크 배치 전략 개발

## Development of Task Assignment Strategy for the Optimized Utilization of the Real-time Network System

오재준\*, 김홍렬\*\*, 김대원\*\*\*  
(Jaejoon Oh, Hongryeol Kim, Daewon Kim)

**Abstract** - In this paper, the task assignment strategy considering communication delay and the priority of distributed tasks is proposed for the real-time network system in order to maximize the utilization of the system. For the task assignment strategy, the relationship among priority of tasks in network nodes, the calculation time of each task, and the end-to-end response time including the network delay is formulated firstly. Then, the task assignment strategy using the genetic algorithm is proposed to optimize the utilization of the system considering the LCM(Least Common Multiple) period. The effectiveness of proposed strategy is proven by the simulation for estimating the performance such as the utilization and the response time of the system in case of changing the number of tasks and the number of network nodes.

**Key Words** : real-time network system, task assignment, genetic algorithm, utilization, response time

### 1. 서 론

네트워크 기반의 시스템은 최소한의 비용으로 기능별로 독립적인 모듈(module)의 결합을 통한 시스템 구현을 가능하게 한다. 또한 네트워크 기반의 시스템은 전체 시스템에 영향을 주지 않으면서 모듈 단위 변경을 통해 시스템 전체의 성능 개선을 가능하게 하는 등의 장점을 갖기에 다양한 연구가 진행되고 있으며, 고속 철도, 이동로봇 등의 분야와 같이 복잡한 시스템의 경우에도 네트워크 기반 시스템의 연구가 진행 중이다[1][2].

네트워크 시스템은 네트워크라는 하나의 매체를 공유하기 때문에 필연적으로 지연시간이 발생한다. 이러한 지연시간 발생의 문제점을 극복하기 위해, 시스템을 초기 구성하는 과정에서 지연시간의 관점에서 최적화를 위해 우선순위 제한 등을 고려한 태스크(task) 우선순위 할당 연구와 네트워크 지연시간을 고려한 태스크 배치의 연구가 진행되고 있다[3][4]. 관련된 기존의 연구로 인스트럭션(instruction) 수준의 제한 조건에서 유전자 알고리즘을 사용한 스케줄링(scheduling) 우선순위 최적화에 관한 연구가 수행된 바 있다[3]. 하지만 이동로봇과 같은 시스템의 경우에는 운영 태스크의 조합이 복잡해지기 때문에 인스트럭션 수준의 제한 조건보다 전체 태스크 간의 제한 조건이 시스템의 실시간성 보장 및 성능에 더 중요한 요소가 된다. 또한 네트워크 기반 시스템의 경우

에 태스크는 각 모듈에 분산되어 할당되어지며 규모가 큰 시스템의 경우 각 모듈은 개별적으로 개발되어 하나의 시스템을 구성하게 된다. 이 경우 각 모듈안의 소프트웨어적 구성은 공개되지 않으며 따라서 각 모듈의 인스트럭션을 파악, 그 특성을 분석하는 일은 불가능하게 진다. 따라서 인스트럭션 수준이 아닌 각 태스크 간의 연결관계와 네트워크 지연, 그리고 응답 시간 등을 고려한 태스크 배치 및 스케줄링이 필요하게 된다. 태스크 수준으로 수행된 기존의 연구로 태스크를 통신 태스크와 계산 태스크로 구분하여 네트워크 지연과 계산 시간을 사용하여 태스크 배치를 수행한 연구가 수행된 바 있다[4]. 하지만 기존의 연구에서는 태스크의 배치로 인한 각 모듈 내에서 발생하는 태스크의 지연 시간과 이용률의 변화는 고려되지 않았다. 또한 시스템이 모듈화 됨에 있어 특정 태스크는 기능상의 이유로 특정 모듈에만 배치되어야 하는 제한 조건도 역시 고려되지 않았다.

본 논문에서는 운영체제 스케줄링 방식으로 대표적인 정적 스케줄링 방식인 RMS(Rate Monotonic Scheduling)[6] 방식을 기반으로 한 네트워크 기반 시스템의 태스크 배치 전략을 제안한다. 제안된 태스크 배치 전략은 태스크 배치 위치 변경에 의한 각 프로세스 노드(process node)에서의 우선순위 변화와 네트워크를 통해 전달되는 데이터 크기의 변화, 그리고 그로 인한 네트워크 지연의 변화 등을 고려한 시스템 이용률의 최적화를 목적으로 한다. 이를 위하여 본 논문에서는 먼저 태스크 배치 위치의 변경에 따른 각 네트워크 노드의 우선순위의 변화와 그로 인한 수행 주기 안에서의 태스크 계산시간 변화, 그리고 네트워크 지연의 변화로 인한 응답 시간 변화의 상관관계를 정리한다. 또한 본 논문에서는 상기의 상관관계와 유전자 알고리즘을 사용하여 최소 공배 주기 안에서 이용률을 최적화하는 태스크 배치 전략을 제안한다. 본 논문에서 제안된 태스크 배치 전략의 유용성을 검증하기 위

### 저자 소개

- \* 准 會 員 : 明知大學 情報工學科 碩士課程
- \*\* 正 會 員 : 明知大學 情報工學科 博士課程
- \*\*\* 正 會 員 : 明知大學 情報工學科 教授 · 工博

하여 모의 실험을 통해 프로세스 노드의 수와 태스크 수를 변경시키면서 시스템 이용률에 대한 성능평가를 수행한다. 대상이 되는 네트워크 시스템은 실시간 네트워크 프로토콜인 CAN(Controller Area Network)[5]을 이용하는 것으로 가정하며, CAN의 메시지 스케줄링 또한 RMS 방식을 기반[6]으로 한다.

본 논문의 2장에서는 일반적인 네트워크 기반 시스템의 문제와 본 논문에서 고려한 시스템의 문제를 정의하며, 3장에서는 유전자 알고리즘과 2장에서 고려된 유도된 문제 수식을 사용한 태스크 배치 전략을 소개한다. 4장에서 모의 실험의 구성에 대해 설명한 후, 실험 결과를 사용하여 제안된 태스크 배치 전략의 유용성을 증명하고, 마지막으로 5장에서 본 논문을 요약하고 보완점 등을 논의한다.

## 2. 문제 정의

본 논문에서 대상으로 하는 네트워크 기반 실시간 제어 시스템은 그림1과 같다. 그림1에서 각 네트워크 노드에는 하나 이상의 다중 태스크가 운영되며, 운영되는 태스크들은 실시간 운영체계에 의해 동작의 실시간성을 보장받게 된다. 네트워크 환경에서의 이러한 다중 태스크 운영 환경은 최근의 네트워크 기반 로봇[7]과 같이 복잡한 분산 시스템의 일반적인 형태이다.

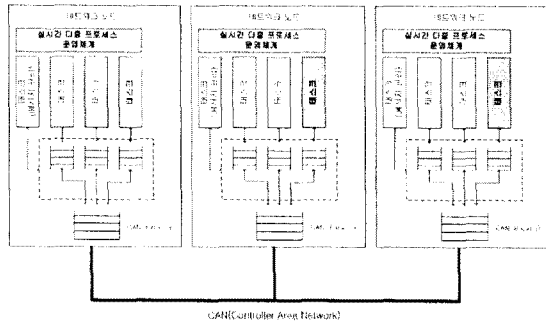


그림 1. 다중 운영 체계를 갖는 네트워크 시스템

그림 1과 같은 네트워크 시스템에서의 시스템 이용률은 전체 시스템을 구성하는 태스크들의 이용률의 총합이다. 즉 태스크의 번호를  $i$ , 주기를  $T_i$ , 계산 시간을  $C_i$ 라고 표현하고, 시스템의 총 이용률  $U$ , 각 태스크의 이용률을  $u_i$ 라 하면, 총 이용률은 다음 식 (1)과 같이 표현된다[8][9].

$$U = \sum_{i=1}^n u_i = \sum_{i=1}^n \frac{C_i}{T_i} \quad (1)$$

일반적으로 네트워크 시스템의 경우, 네트워크 노드는 서로 다른 하드웨어 및 운영 체제 등으로 구성되어 진다. 따라서 이를 고려하면, 각 태스크의 이용률은 식 (2)와 같이 태스크가 배치된 네트워크 노드의 계산 능력  $CP$ , 메모리 크기  $MS$ , 네트워크 노드의 운영 방법  $OST$ , 함께 할당된 다른 태스크의 주기  $T_j$ , 그리고 계산량  $TCS$  등에 영향을 받는다. 본

논문에서는 계산과 시뮬레이션의 편의를 위하여 각 네트워크 노드는 동일한 하드웨어와 동일한 운영 체제를 갖는다고 가정한다.

$$u_i = f(CP, MS, OST, T_j, TCS) \quad (2)$$

실시간 시스템을 대상으로 하는 경우, 시스템을 평가 하는데 있어서 이용률과 함께 지연 시간 또한 중요한 시스템 평가 지표가 되며, 또한 모든 태스크는 각각의 데드라인 (deadline) 안에 수행이 완료되어야 한다. 따라서 이 또한 태스크 배치의 고려 사항이 된다. 일반적으로 오프라인 상에서 태스크 배치를 수행할 경우, 실시간성 보장을 위해 최악의 지연 시간에 대한 선행 분석 결과가 사용된다[9][10]. 운영체제가 포함된 네트워크 시스템의 종단간 지연은 운영체제 지연 시간과 네트워크 지연시간, 그리고 운영체제 태스크의 수행과 네트워크 메시지 전송의 비동기성으로부터 발생하는 대기 지연시간으로 구성된다[9]. 따라서, 이러한 세 가지 지연 요소를 고려한 종단간 최악 지연시간  $D_{etc}$ 는 다음 식 (3)과 같이 표현된다. 여기서,  $r(trans)$ 는 송신부 운영체제 계층에서의 최악 응답시간이며,  $W_{max}(net)$ 은 네트워크 계층에서의 비동기성으로 인한 최악의 대기 지연시간,  $r_m$ 은 네트워크 최악의 대기행렬 지연시간을, 그리고  $C_m$ 은 네트워크 전송 지연시간을 나타낸다. 또한  $W_{max}(recv)$ 는 수신부 운영체제 계층에서의 비동기성으로 인한 최악의 대기 지연시간을 나타내며,  $r(recv)$ 는 수신부 운영체제 계층에서의 최악의 응답시간을 나타낸다.

$$D_{etc} = r(trans) + W_{max}(net) + r_m + C_m + W_{max}(recv) + r(recv) \quad (3)$$

본 논문에서는 실시간성 만족을 위한 데드라인 주어진 경우, 우선적으로 종단간 지연시간을 만족하면서 전체 시스템의 이용률을 최적화하는 태스크 배치 전략 개발을 수행한다. 태스크 배치 전략 개발을 위해 본 논문에서는 식 (1)과 식 (3)을 통해 최적화의 성능 지표를 식 (4)와 같이 제안한다. 식 (4)에서  $\alpha$ 와  $\beta$ 는 시스템을 최적화하는데 있어서 이용률과 종단간 지연시간의 중요도에 따른 가중치이다.

$$F = \frac{1 + \alpha U}{1 + \beta D_{etc}}, \text{ where } \alpha \geq 0, \beta \geq 0, \alpha + \beta = 1 \quad (4)$$

## 3. 유전자 알고리즘을 이용한 태스크 배치 전략 개발

본 논문에서는 유전자 알고리즘을 사용하여 실시간성 만족 조건에서의 식 (4)의 시스템 이용률 최적화 문제를 해결하고자 한다. 본 논문에서 사용되는 유전자 알고리즘은 네트워크 시스템의 제약 조건으로 인해 다음 세대 생성에 있어 제약 조건을 갖는 것을 특징으로 한다. 모들의 조합으로 이루어진 네트워크 시스템의 경우 센서 정보 수집 태스크와 같이 하드웨어에 종속되어 있어서 다른 모듈로 배치되어서는 안 되는 태스크들이 존재한다. 또한 특정 하드웨어 종속적인 태스크는 시스템의 제어 안정성 등의 이유로 수행 주기의 변경에 제한을 받는다. 따라서 본 논문에서는 교배(crossover)와 돌

연변이(mutation) 과정에서 하드웨어 종속적인 태스크의 위치를 변경하지 않고 유전자 알고리즘을 수행한다. 또한 본 논문에서는 하드웨어에 종속되어 하드웨어 제어를 수행하는 태스크의 주기 범위를 식 (5)와 같이 이산 시스템의 제어 안정도 보장 범위로 제한한다[11]. 식 (5)에서  $T_r$ 은 시스템의 상승 시간(rising time) 시간을 의미한다.

$$4 \leq \frac{T_r}{T_i} \leq 10 \quad (5)$$

본 논문에서 제안하는 배치 전략은 그림 2와 같이 두 단계를 거쳐 최적의 태스크 배치를 찾아낸다. 첫 번째 단계는 하드웨어 종속적인 태스크와 하드웨어 독립적인 태스크를 중단 간 지연 시간이 최소화 되도록 네트워크 노드에 배치를 한다. 이 때 제안된 식(4)에서  $a$ 는 0,  $\beta$ 는 1로 설정한다. 두 번째 단계는 각 네트워크 노드에 할당된 하드웨어 독립적인 태스크의 주기를 변경함으로써 각 네트워크 노드의 이용률을 높이며 이를 통해 시스템 전체의 이용률을 최적화한다. 이 때 제안된 식(9)에서  $a$ 는 1이 되며  $\beta$ 는 0으로 설정된다.

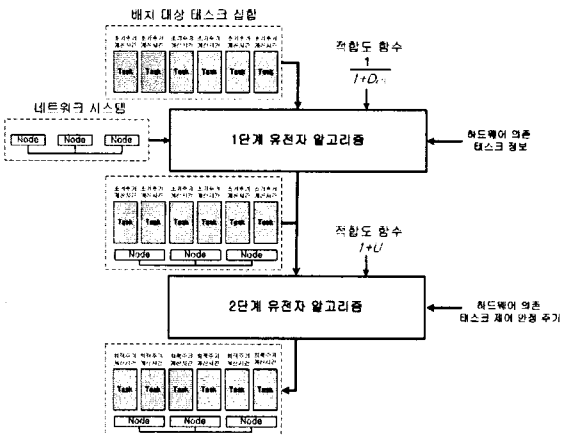


그림 2. 유전자 알고리즘을 이용한 2단계 태스크 배치 전략

#### 4. 모의 실험

본 논문에서는 첫 번째로 그림 3의 (a)와 같이 모바일(mobile), 브레인(brain), 그리고 센서(sensor) 모듈로 구성되어 있는 이동 로봇이 CAN 네트워크로 구성되어 있다고 가정하고 기본 모의실험을 진행한다. 이 때 모바일 부에 반드시 있어야 하는 모터 드라이빙(motor driving) 태스크와 센서 모듈에 있어야 하는 센서 정보 수집 태스크와 같은 하드웨어 종속적인 태스크가 존재하게 된다. 두 번째 모의 실험으로는 그림 3의 (b)와 같이 임의의 시스템이  $m$ 개의 네트워크 노드와  $n$ 개의 태스크로 이루어져 있으며 태스크들 중에는  $k$ 개의 하드웨어 종속 태스크가 있다는 가정하에 진행된다. 첫 번째 실험으로 본 논문에서 제안하는 배치 전략의 적용 가능성을 보이며, 두 번째 실험을 통하여 배치 전략의 확장성을 보임으로서 본 논문에서 제안한 배치 전략의 유용성을 입증한다.

위의 모의 실험에 사용되는 태스크의 설정은 표 1과 표 2와 같다.

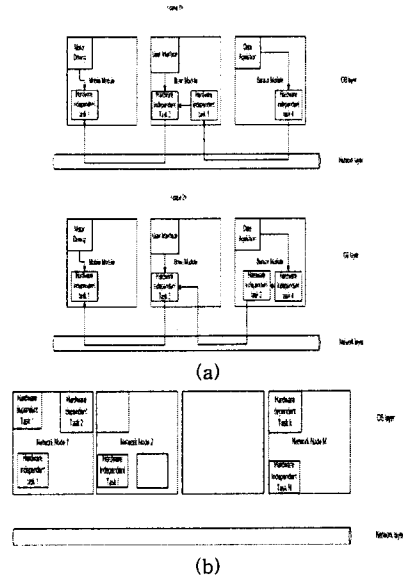


그림 3 (a) 네트워크 기반 로봇의 개념도, (b) 일반적인 네트워크 시스템  
표 1. 태스크 설정

Task No.	$T_i$	$C_i$	dependenc y	Trans. Task	Data Size(byte)	Required Dete
0	100	20	none	none	0	500
1	100	20	0	0	4	500
2	200	30	1	1	2	800
3	200	40	2	2	10	800
4	200	40	none	3	5	800
5	200	20	none	2	4	800
6	200	30	none	2	8	800
7	300	20	none	4	4	1200
8	300	40	none	7	10	1200
9	400	20	none	4	6	1300
10	400	40	none	4	5	1300
11	500	40	none	8	2	1500
12	5600	50	none	11	3	1500
13	900	30	none	12	3	1500
14	900	70	none	6	4	1700

첫 번째 모의 실험에서 시스템의 각 노드의 이용률은 표 1과 같다. 표 2는 표 1에서 12번까지의 태스크만을 3개의 네트워크 노드에 배치할 경우의 각 네트워크 노드의 시스템 이용률이다. "Utilization Bound"는 중단간 지연시간을 고려하지 않고 각 노드에 동등하게 배치할 경우의 이용률이다. 네트워크 지연이 존재하므로 시스템의 이용률은 낮아진다. 본 논문에서 제안한 태스크 배치 전략은 표 2와 같이 두 단계를 거치면서 중단간 지연 시간을 만족하면서 시스템 이용률을 최대화 한다. 그림 4와 그림 5는 각각 네트워크 노드 수에 따른 시스템 이용률과 태스크 들의 중단간 지연 시간의 변화를 보여준다. 이때 태스크 4에서 9번은 네트워크 노드가 증가함에 따라 3에서부터 8까지 네트워크 노드에 하드웨어 종속적인 태스크로 변경하여 실험하였다. 그림 6은 네트워크 노드의 수를 5개로 고정된 상태에서 하드웨어 독립적인 태스크가

증가함에 따른 시스템 이용률의 변화를 보여 주고 있다.

	after step 1.	after step 2	Utilization Bound
Network Node 1	0.400000	0.400000	0.756828
Network Node 2	0.296667	0.646667	0.756828
Network Node 3	0.933333	0.933333	0.756828

표 2. 네트워크 노드의 시스템 이용률

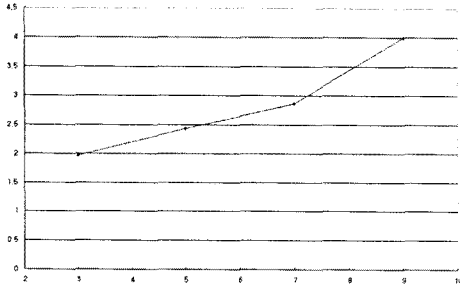


그림 4. 네트워크 노드 수에 따른 시스템 이용률

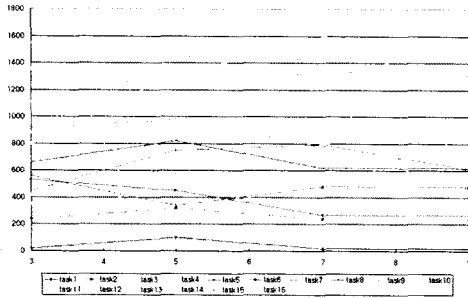


그림 5. 네트워크 노드 수에 따른 중단 간 최대 지연시간

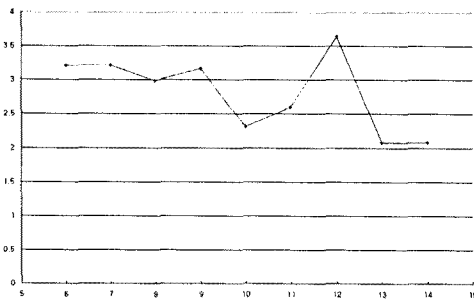


그림 6. 태스크 노드 수의 증가에 따른 시스템 이용률

## 5. 결론

본 논문에서는 네트워크 기반 시스템에서의 이용률을 최적화하는 태스크 배치 전략을 제안하였다. 이를 위하여 태스크 배치에 따른 이용률의 변동을 고려하였고, 또한 태스크 배치에 따른 지연 시간을 함께 고려하여 시스템을 수식화 하였으며, 이를 유전자 알고리즘을 사용하여 최적화하는 배치 전략을 수행하였다.

제안된 태스크 배치 전략의 유용성을 평가하기 위하여 네트워크 기반 이동 로봇에서의 모의 실험과 임의의 시스템에

서 네트워크 노드와 태스크 수를 변경시키면서 이용률의 변화와 지연시간의 변화를 모의 실험을 수행하여 제안된 태스크 배치 전략이 네트워크 기반 시스템에 유용함을 입증하였다.

향후, 네트워크 기반 시스템을 사용한 실험을 통하여 제안된 배치 전략의 유용성을 검증할 것이다. 또한, 네트워크 노드의 하드웨어적인 특성을 동일하다고 가정한 본 논문을 다양한 하드웨어 및 운영체제로 구성된 이기종 네트워크 시스템으로 확장할 것이며, 나아가 이미 구성된 시스템에 태스크가 추가될 경우 최적으로 태스크를 배치 가능한 동적 태스크 할당 전략까지 확장 연구할 예정이다.

## 참 고 문 헌

- [1] 김홍렬, 광권천, 김대원, "CAN 기반 피드백 시스템의 한국형 고속전철 여압시스템 적용", Journal of Control, Automation, and Systems Engineering, Vol. 9, No. 11, pp. 963-968, November, 2003.
- [2] Hongryeol Kim, Joomin Kim, Daewon Kim, "Rate Modulation Strategy for the Behaviors with Hybrid Robot Control Architecture", IFAC Workshop on Adaptation and Learning in Control and Signal Processing, and IFAC Workshop on Periodic Control Systems, pp. 303-308, 2004
- [3] Tai M. Chung, Hank G. Dietz, "Adaptive Genetic Algorithm: Scheduling Hard Real-time Control Programs with Arbitrary Timing Constraints",
- [4] Dar-Tzen Peng, Kang G. Shin, Tarek F. Abdelzaker, "Assignment and Scheduling Communicating Periodic Tasks in Distributed Real-Time Systems", IEEE Trans, vol. 23, no. 12, December 1997
- [5] CAN Specification Version 2.0, Robert Bosch GmbH, 1991.
- [6] K. W. Tindell, A Burns, and A. J. Wellings, "Guaranteeing Hard Real-Time End-to-End Communications Deadlines," RTRG/91/107, Department of Computer Science, University of York, 1991.
- [7] Sookyoung Sohn, Hongryeol Kim, Daewon Kim, Hongseok Kim, Hogil Lee, "Obstacle Avoidance Algorithm for a Network-based Autonomous Mobile Robot", ICCAS2004, august, 2004.
- [8] N.Audsley, A. Burns, "Real-Time System Scheduling", first report from the ESPRIT B RA Project, vol. 2, Chapter 2, Part 2, 1993. 9.
- [9] John Lehoczky, Lui Sha, Ye Ding, "The Rate Monotonic Scheduling Algorithm Exact Characterization And Average Case Behavior", IEEE , vol. , no. , pp. , 1989.
- [10] Hongryeol Kim, Joomin Kim, Daewon Kim, "Development of Coordinated Scheduling Strategy with End-to-End Response Time Analysis for the CAN-Based Distributed Control Systems", Proceeding fo 2004 IEEE/RSJ International Conference on Intelligent Robot and System, pp. 2099-2104 ,2004
- [11] K.J.Astrom, B.Wittenmark, "Computer-Controlled Systems", Prentice Hall, 1997.