

다중 로봇 시스템의 결합, 모델링 및 시뮬레이션

이 기동 홍 지민 이 범희 고 명삼

로보틱스 및 지능 시스템 연구실
서울대학교 대학원 제어 계측 공학과

An Interconnection, Modelling and Simulation for a Multi-Robot Systems(MRS)

Ki-Dong Lee Jee-Min Hong Bum-Hee Lee Myoung-Sam Ko

Automation and Systems Research Institute,
Department of Control & Instrumentation Engineering,
Seoul National University,

ABSTRACT

For a robotic workcell, which consists of multiple robots, several interconnection methods are presented in terms of the processor based architecture. Since few attempts have been made to formulate and analyze multiple robot system(MRS), we turn the knowledge of multiple processor system(MPS) or multiple computer system(MCS) to good account. The performance evaluation is achieved through queueing analysis, the aim being to compare their response time, utilization, probability of service failure under different workload. To verify the validity of the proposed analysis methods, a computer simulation is performed. The results together with comments presented here give some useful guidelines for the selection of an appropriate interconnection method.

I. 서론

생산시스템 분야는 상품제조의 경비절감과 소비자의 다양한 요구를 만족시키는 다품종 소량생산 체제로의 전환을 위하여 생산설비의 자동화, 고속화, 유연화를 추구하고 있다. 따라서 전 세계적으로 FA(factory automation)를 확산시켜 나가려는 움직임이 활발히 진행되고 있는데, 이것은 최근 급격히 발전한 전자기술로 인하여 더욱 가속화되었으며, 특히 FA기술의 기본 개념인 FMS(flexible manufacturing system)에 대한 기술개발은 컴퓨터기술 및 로봇기술, 인공지능기술의 개발과 더불어 더욱 발전하고 있다. FMS에서 가장 핵심적인 역할을 수행하고 있는 로봇트는 조립작업, 반송작업, 기구작업 등 다양한 동작기능을 발휘함으로써 새로운 시대의 중핵적 역할을 감당하고 있다. 한편 최근에는 여러대의 로봇에 의한 공동작업의 실현에 많은 관심이 집중되고 있는데, 이것은 유연성 및 신뢰성의 향상으로 작업시간의 단축 및 생산성의 증가를 가져올 수 있기 때문이다.

이와같이 여러대의 로봇으로 구성된 조립설에서 로봇을 중심으로 생각하면 이것은 MRS(multiple robot system)로 볼수 있는데, MRS의 연결 방법 및 성능 평가에 대한 연구는 많지 않으므로 MPS(multiple processor system) 또는 MCS(multiple computer system)의 연결 방법 및 성능 평가에 대한 분석을 원용

한다.

MPS나 MCS의 연구 결과를 MRS의 분석에 사용한 논문은 거의 없고[1] SRS(single robot system)의 분석에 적용한 논문은 [2], [3] 등이 있다. [2]은 실시간 비선형 로봇 제어 시스템을 MPS를 이용하여 분석하였고, [3]은 OSU 6지 로봇(OSU hexapod vehicle) 시스템을 MPS로 구성하여 해석하였다. 한편 MRS의 성능 분석을 위해서는 MRS의 통신 방법에 대한 연구[4], [5] 및 MRS의 구성 방법에 대한 연구[6], [7]도 필요하다. 또한 MRS의 운영 체제에 대한 조사[8], [9]도 병행되어야 한다.

오류 회복(error recovery)에 대한 연구로는 [3], [10], [11], [12]등이 있는데 [10]에서는 자기 발견적인(heuristic) 방법을 사용하였고, [11]에서는 센서를 이용한 모니터링 모듈(sensor-based monitoring module)로 오류를 감지하고 규칙을 이용한 오류 분석 모듈(rule-based failure analysis module)로 오류를 처리한다. [12]은 지식을 이용한 로봇 제어의 오류 회복에 대한 결과를 보여 주고 있다.

본 논문의 구성은 다음과 같다. 2장에서는 다중 로봇 시스템의 내부 연결 방법과 그것들의 모델을 제시하고, 제시된 시스템의 수식적 분석을 3장에서 기술한다. 4장에서는 수식적 분석의 구체적인 결과를 보여주고 5장에서는 컴퓨터 시뮬레이션 결과와 4장에서의 결과와 비교한다. 마지막으로 6장에서 결론을 맺는다.

II. 문제 설정

여기서 제시하는 MRS는 4가지 인데 그것들은 각각 SPS(single processor system)-I, SPS-II, HSS(hierarchical structured system), MSS(master slave system)등이다. MRS와 비교하기 위하여 SRS를 도입하며 HSS의 일반화된 형태인 EHSS(extended hierarchical structured system)도 제시한다.

2.1 다중 로봇 시스템(MRS)의 구성

SPS-I와 SPS-II는 하나의 프로세서가 두대의 로봇을 구동하는 시스템인데 프로세서 한개로 로봇 두대를 순차적으로 구동하는 것이 SPS-I이고 프로세서 한개로 로봇 두대를 동시에 구동하는 것이 SPS-II이다.

HSS는 각각의 로봇이 자신의 프로세서를 가지고 있어서 로

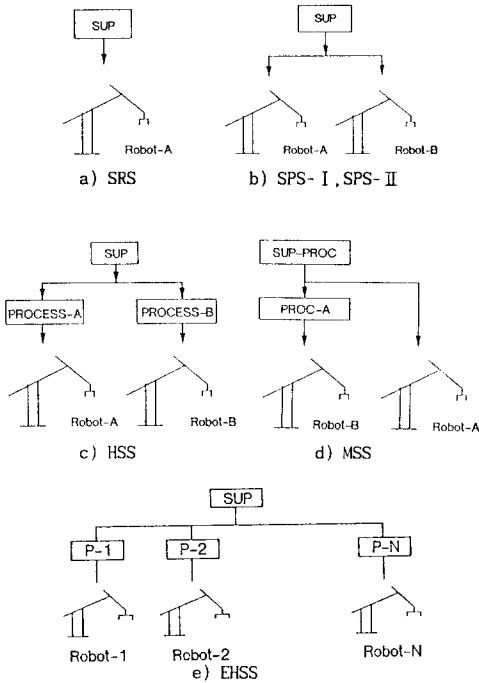


그림 2.1 MRS의 종류
Fig 2.1 Types of MRS

봇을 독립적으로 동작시키고 각각의 프로세서를 통합 제어 하는 또다른 프로세서가 있는 계층적 구조이며 MRS에서 가장 일반적으로 사용되는 구조이다[6],[7]. 프로세서간의 통신은 주 프로세서가 총괄하며 주프로세서에서 부프로세서로 명령을 이송할 경우에는 전송시간을 고려하고, 부프로세서에서 주프로세서로 명령을 이송할 경우에는 역전송 시간을 고려한다. 전송 시간 및 역전송 시간은 전송할 명령의 개수에 비례하고 기본적으로 데이터 전송에 필요한 시간(startup time)을 첨가한 형태로 가정하여 일차함수로 다음 식(1)에 나타내었다[13].

$$\begin{aligned}
 f(C) &= k_1 \cdot C + k_2 \quad (k_1, k_2 \text{는 전송 상수}) \\
 g(C) &= k_3 \cdot C + k_4 \quad (k_3, k_4 \text{는 역전송 상수}) \\
 C &: \text{전송할 명령수}
 \end{aligned} \tag{1}$$

MSS는 HSS와 유사하나 주프로세서가 직접 로봇을 구동시키면서 전체를 총괄하기 때문에 주프로세서의 부담이 크지만 전체 전송 및 역전송에 의한 지연 시간이 HSS 보다 작다.

비교를 위한 SRS를 포함하여 여기서 제시된 시스템을 그림 2.1에서 보여주고 있다.

2.2 다중 로봇 시스템의 모델링

MRS의 성능 평가를 위해서 MRS를 느슨히 결합된 MPS(loosely coupled MPS)로 간주하고 큐잉 모델을 사용한다. 그러면 로봇이 수행해야 할 작업 명령들이 큐의 방문자(customer)가 되며 처리자(server)는 로봇 또는 프로세서가 된다. 여기서 고려하는 성능 평가 지표는 전체 실행 완료 시간과

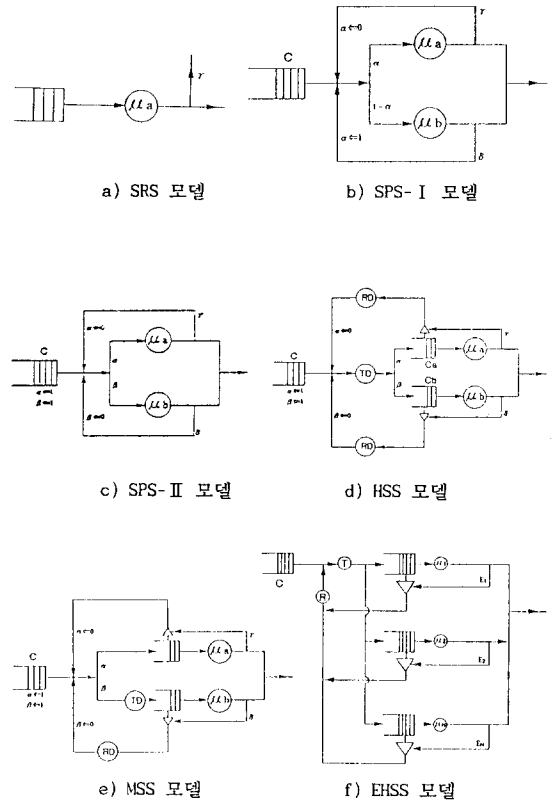


그림 2.2 MRS의 큐잉 모델
Fig 2.2 Queueing Models of MRS

실행 실패 확률, 실행 실패하여 남은 명령 개수의 기대값, 프로세서와 로봇의 사용률 등이다. 로봇 작업의 종류[4],[5]는 편의상 독립 작업을 대상으로 하며 시스템 해석을 위해 다음 가정을 한다. 1) 로봇의 명령 실행 시간의 분포는 지수 분포(exponential distribution)이다. 2) 명령을 전송하는 도중에는 오류(error)가 발생하지 않는다. 3) 오류가 생긴 로봇은 사용하지 못한다. 4) 오류 발생 명령도 실행 완료된 명령과 같은 기대값의 실행 시간을 가진다. 여기서 명령(customer)이란 각 로봇에 할당되는 작업(task)을 말하며, 오류라는 것은 명령 한 개에 대해서 로봇이 실행 완료하지 못하는 경우를 가리킨다. 이 오류는 로봇자체 뿐 아니라 기구부, 부품, 부품 피더, 그리퍼 등의 불량을 의미한다.

앞절에서 제시한 시스템들의 큐잉 모델들은 다음 그림 2.2와 같다.

III. 수학적 분석

모든 모델에 대해서 전체 실행 완료 시간과 실행 실패 확률, 실행 실패하여 남은 명령 개수의 기대값, 프로세서와 로봇의 사용률 등을 분석하였으나 지면 관계상 프로세서와 로봇의 사용률에 관한 수식은 생략한다. 또한 [14]에 SRS, SPS-II, HSS에 대한 수학적 분석 결과가 있으므로 여기서는 SPS-I, MSS, EHSS에 대한 수학적 결과만 다룬다. 이후로 사용될 용어

및 표식을 나타내면 아래와 같다.

실행 실패(service failure) : 주어진 명령들을 전부 실행 완료하지 못하고 오류가 발생하는 경우.

평균 실행완료 시간(average completion time) : 실행 실패가 없는 경우, 주어진 명령들을 전부 수행하는데 걸리는 시간의 기대치.

- C : 주어진 명령수
- C_a : 로봇 a의 프로세서에 최초 할당된 명령수
- μ_a : 로봇 a의 수행 비율(service rate)
- ε_a : 로봇 a의 오류 확률
- P_{a11} : 각 로봇들이 오류 발생 없이 명령들을 각각 실행 완료할 확률
- P_a : 로봇 a에서 먼저 오류가 발생하여 나머지를 다른 로봇에서 실행 완료할 확률
- T : 평균 실행 완료 시간
- T_{sch} : 작업 스케줄링 시간의 기대치
- T_{a11} : P_{a11}이 생길 경우의 전체 실행 완료시간의 기대치
- TP_a : P_a가 생길 경우의 전체 실행 완료 시간의 기대치
- CP_a : P_a가 생길 경우의 주프로세서로 전송해야 할 명령수의 기대값
- N_{ab}(i) : 로봇 a에서 i개의 명령이 수행될때까지 로봇 b에서 수행된 명령수의 기대값
- TE_a : 로봇 a에서 먼저 오류가 생길때까지 경과된 시간의 기대값
- PE_{ab} : 로봇 a에서 먼저 오류가 발생하여 나머지 명령을 로봇b로 옮겨 실행할때 다시 로봇 b에서 오류가 발생할 확률(실행 실패)
- P_{sf} : 실행 실패가 일어날 확률
- RC_{ab} : PE_{ab}로 인하여 실행하지 못할 명령수의 기대값
- RC : 실행 실패로 인하여 실행완료하지 못할 명령수의 기대값

분배율(Load Distribution Ratio) : C_a와 C_b의 비율.

여기에서 C_b, μ_b, ε_b, P_b, TP_b, CT_b, N_{ba}(i), TE_b, PE_{ba}, RC_{ba}은 C_a, μ_a, ε_a, P_a, TP_a, CT_a, N_{ab}(i), TE_a, PE_{ab}, RC_{ab}와 유사한 형태로 정의된다. 즉 a, b가 바뀐 경우이다.

3.1 단일 프로세서 시스템 I (SPS-I)의 분석

SPS-I에서만 사용되는 표식은

- α : 명령 한개가 로봇 a에서 수행될 확률
- P(i) : 오류 없이 명령 i개를 실행 완료할 확률
- N_a : P_a가 발생할 경우, 오류가 생길때까지 수행된 명령수의 기대값(단 오류가 생긴 명령 포함)
- N_b : P_b가 발생할 경우, 오류가 생길때까지 수행된 명령수의 기대값(단 오류가 생긴 명령 포함)

인대 위의 표식을 이용하여 오류 발생 없이 주어진 명령들을 실행 완료할 확률을 구해보면

$$P_{a11} = P(C) = \{\alpha(1-\epsilon_a) + (1-\alpha)(1-\epsilon_b)\}^C \quad (2)$$

이다. 이 경우, 이것은 2-단계 병행 서버 (2-stage parallel

server) [15]이므로 명령 한개의 평균 수행 시간을 구해 전체 실행 시간의 기대치를 계산하면

$$T_{a11} = C \left(\alpha \frac{1}{\mu_a} + (1-\alpha) \frac{1}{\mu_b} \right) \quad (3)$$

와 같이 표현되고, 로봇 a에서 먼저 오류가 발생했으나 나머지를 로봇 b에서 실행 완료할 확률 및 그때의 전체 실행 시간의 기대치는 다음 식(4), (5)와 같이 구한다.

$$P_a = \alpha \cdot \epsilon_a \sum_{i=0}^{C-1} [(1-\epsilon_b)^{C-i}] \cdot P(i) \quad (4)$$

$$TP_a = TE_a + \frac{C-N_a+1}{\mu_b} \quad (5)$$

여기에서

$$N_a = \alpha \epsilon_a \left[\sum_{i=0}^{C-1} (1-\epsilon_b)^{C-i} \cdot P(i) \cdot (i+1) \right] / P_a \quad (6)$$

$$TE_a = N_a \left(\alpha \frac{1}{\mu_a} + (1-\alpha) \frac{1}{\mu_b} \right) \quad (7)$$

P_b, TP_b 도 비슷한 방법으로 계산할 수 있다. 위의 T_{a11}, TP_a, TP_b 값을 이용하여 평균 실행 완료 시간을 구하면:

$$T = \frac{P_{a11} \cdot T_{a11}}{P_{a11} + P_a + P_b} + \frac{P_a \cdot TP_a}{P_{a11} + P_a + P_b} + \frac{P_b \cdot TP_b}{P_{a11} + P_a + P_b} + T_{sch} \quad (8)$$

그다음 실행 실패가 일어날 확률 및 실행 실패가 발생하여 남은 명령수를 계산하는 방법이 식(9)과 식(10)에 있다.

$$P_{sf} = PE_{ab} + PE_{ba} \quad (9)$$

$$RC = RC_{ab} + RC_{ba} \quad (10)$$

여기에서

$$PE_{ab} = \alpha \cdot \epsilon_a \cdot \epsilon_b \cdot \sum_{i=0}^{C-1} [P(i) \sum_{j=0}^{C-i-1} [(1-\epsilon_b)^j]] \quad (11)$$

$$RC_{ab} = \alpha \cdot \epsilon_a \cdot \epsilon_b \cdot \sum_{i=0}^{C-1} [P(i) \sum_{j=0}^{C-i-1} [(1-\epsilon_b)^j (C-i-j)]] \quad (12)$$

이며 PE_{ba}, RC_{ba} 는 PE_{ab} 및 RC_{ab}의 계산법과 유사하다.

3.2 이중 시스템(MSS)의 분석

HSS와 유사한 형태이므로 N_{ab}(i), N_{ba}(i), CP_a, CP_b, TE_a, TE_b, P_{a11}, T_{a11}, P_a, P_b 값은 HSS와 같다. 단지 전송 지연 시간에 차이가 있다. 그래서 확률 P_a일 경우, 전체 실행 시간의 기대치는 TE_a 가 C_b/μ_b 보다 크면,

$$TP_a = TE_a + f(CP_a) + CP_a / \mu_b$$

가 되고, 그렇지 않을 경우에는

$$TP_a = \frac{C_b + CP_a}{\mu_b} + f(CP_a) \quad (13)$$

이 된다. 확률 P_b일 경우에도 마찬가지로 계산할 수 있다. 그래서 평균 실행 완료 시간은

$$T = \frac{P_{a11} \cdot T_{a11}}{P_{a11} + P_a + P_b} + \frac{P_a \cdot TP_a}{P_{a11} + P_a + P_b} + \frac{P_b \cdot TP_b}{P_{a11} + P_a + P_b} + T_{sch} + f(C_b) \quad (14)$$

인데 이것과 HSS의 그것과의 차이는 최초 전송 지연 시간에 대한 항 뿐이다. P_{sf}와 RC를 구하는 식도 HSS와 같다.

3.3 확장 계층 구조 시스템(EHSS) 분석

EHSS는 HSS를 일반화 시킨 것으로, HSS를 N개의 로봇이 있는 경우로 확장한 경우이다. EHSS에서만 사용되는 표식은 이송 전략에 관련된

CT_{ij}(k) : i번째 로봇에서 k개(오류 발생 명령 제외)의 명령을 실행 완료한뒤 그다음 명령에서 오류가 발생했을 경우 j번째 로봇으로 이송되는 명령 갯수. 이송 전략(transmission strategy)에 따라 결정된다.

이며 나머지는 앞에서 정의한 표식과 유사하나 아래 첨자 a, b 대신 1에서 N까지의 값을 가지는 변수를 사용한다. 또한 두 로봇 경우와는 달리 반복형(recursive)으로 표현하기 위해 함수 형태로 나타낸다 즉,

P_k(X, Y, Z), CP_k(X, Y, Z), P_{sf}(X, Y, Z), P_{sf}(X_k, Y_k, Z_k(i)), T(X, Y, Z), T(X_k, Y_k, Z_k(i)), T_{a11}(X, Y, Z), TP_k(X, Y, Z), TE_k(X, Y, Z), RC(X, Y, Z), RC(X_k, Y_k, Z_k(i))
여기에서

$$\begin{aligned} X &= \{ \mu_j \mid j=1, \dots, N \} \\ Y &= \{ \varepsilon_j \mid j=1, \dots, N \} \\ Z &= \{ C_j \mid j=1, \dots, N \} \\ X_k &= \{ \mu_j \mid j=1, \dots, N, j \neq k \} \\ Y_k &= \{ \varepsilon_j \mid j=1, \dots, N, j \neq k \} \\ Z_k(i) &= \{ C_{kj}(i) \mid j=1, \dots, N, j \neq k \} \\ & \quad C_{kj}(i) = C_j - N_{kj}(i) + CT_{kj}(i-1) \\ & \quad (i=1, 2, \dots, \text{할당된 명령 갯수}) \\ & \quad (k=1, 2, \dots, N) \end{aligned}$$

k번째 로봇에서 제일 먼저 오류가 발생하였으나 실행실패가 일어나지 않을 확률은

$$P_k(X, Y, Z) = \frac{\varepsilon_k}{1 - \varepsilon_k} \sum_{i=1}^{C_k} \left[\prod_{j=1}^N (1 - \varepsilon_j)^{N_{kj}(i)} \right] \{1 - P_{sf}(X_k, Y_k, Z_k(i))\} \quad (k=1, 2, \dots, N) \quad (15)$$

이며 이때 주 프로세서로 이송해야할 명령 갯수의 기대값은

$$CP_k(X, Y, Z) = \frac{A_k(X, Y, Z)}{P_k(X, Y, Z)} \quad (k=1, 2, \dots, N) \quad (16)$$

이 되는데 여기에서

$$A_k(X, Y, Z) = \frac{\varepsilon_k}{1 - \varepsilon_k} \sum_{i=1}^{C_k} \left[\prod_{j=1}^N (1 - \varepsilon_j)^{N_{kj}(i)} \right] \{1 - P_{sf}(X_k, Y_k, Z_k(i))\} (C_k - i + 1) \quad (17)$$

이다. 또한 k번째 로봇에서 제일 먼저 오류가 발생하여 나머지를 다른 로봇에서 실행 완료하는 경우(P_k)에 오류가 발생할 때까지 걸리는 시간의 기대치는

$$TE_k(X, Y, Z) = \frac{C_k - CP_k + 1}{\mu_k} \quad (k=1, 2, \dots, N) \quad (18)$$

가 되며. 이 경우에 전체 실행 시간의 기대치는

$$TP_k(X, Y, Z) = g(CP_k) + f(CP_k) + TE_k + T(X_k, Y_k, Z_k(C_k - CP_k + 1)) \quad (k=1, 2, \dots, N) \quad (19)$$

이 된다. 한편 주어진 로봇 N개 모두 오류가 생기지 않을 확률은

$$P_{a11}(X, Y, Z) = \prod_{k=1}^N (1 - \varepsilon_k)^{C_k} \quad (20)$$

으로 나타나며, 이 경우 전체 실행 시간의 기대치는

$$T_{a11}(X, Y, Z) = \max\{C_1/\mu_1, C_2/\mu_2, \dots, C_N/\mu_N\} \quad (21)$$

로 표현되므로 평균 실행 완료 시간은

$$T(X, Y, Z) = \frac{P_{a11}}{P_{a11} + \sum_{i=1}^N P_i} T_{a11} + \sum_{k=1}^N \left[\frac{P_k}{P_{a11} + \sum_{i=1}^N P_i} TP_k \right] + f(C) + T_{sch} \quad (22)$$

와 같이 반복 형태(recursive form)로 나타낸다. 그밖에 실행 실패가 일어날 확률과 실행 실패가 발생하여 실행 완료하지 못할 명령 갯수의 기대값은 식(23)과 식(24)에 있다.

$$P_{sf}(X, Y, Z) = \sum_{k=1}^N \frac{\varepsilon_k}{1 - \varepsilon_k} \sum_{i=1}^{C_k} \left[\prod_{j=1}^N (1 - \varepsilon_j)^{N_{kj}(i)} \right] P_{sf}(X_k, Y_k, Z_k(i)) \quad (23)$$

$$RC(X, Y, Z) = \sum_{k=1}^N \frac{\varepsilon_k}{1 - \varepsilon_k} \sum_{i=1}^{C_k} \left[\prod_{j=1}^N (1 - \varepsilon_j)^{N_{kj}(i)} \right] B_k(i) \quad (24)$$

여기에서 B_k(i) = P_{sf}(X_k, Y_k, Z_k(i)) · RC(X_k, Y_k, Z_k(i)) 이다.

IV. 결과 및 비교

실행 실패가 발생할 확률은 로봇의 오류 확률에 따라 결정된다. 그래서 두대의 로봇에 대한 오류 비율이 주어지면 SPS-I, SPS-II, HSS, MSS의 실행 실패 확률은 같고, 실행 실패가 발생하여 남겨진 명령들의 갯수도 동일하다. 그림 4.1은 몇가지 오류 확률에 대하여 SPS-I, SPS-II, HSS, MSS의 실행 실패 발생 확률을 구해본 것이다. 그림 4.2는 SRS, 즉 로봇이 한대 밖에 없는 경우의 그래프이다. 예를들어 명령(customer)수가 5000이고, 두 로봇의 오류 비율이 전부 0.0001일때, SPS-I, SPS-II, HSS, MSS의 실행 실패 발생 확률은 약 0.09 이나 SRS의 그것은 0.39 로 앞의 것에 비해 4배 이상 크다. 개략적으로 말하면 이경우 같은 오류 비율을 가지는 로봇을 두대 연결하여 사용하면, 한대를 단독으로 사용하는 것 보다 4배 이상 신뢰성이 있다고 할 수 있다.

그림 4.3과 그림 4.4는 배분율(load distribution ratio)를 변화 시키면서 명령 수가 1000, 5000, 10000 일때의 HSS와 MSS의 평균 실행 완료 시간을 도식한 것이다. HSS는 방문자의

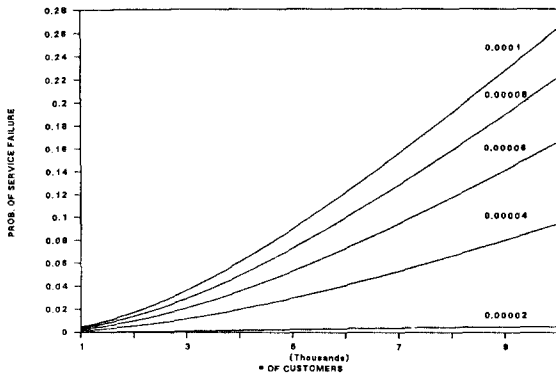


그림4.1 실행 실패 확률의 비교 I (SPS- I, SPS- II, HSS, MSS)
Fig4.1 Comparison of the Probability of Service Failure

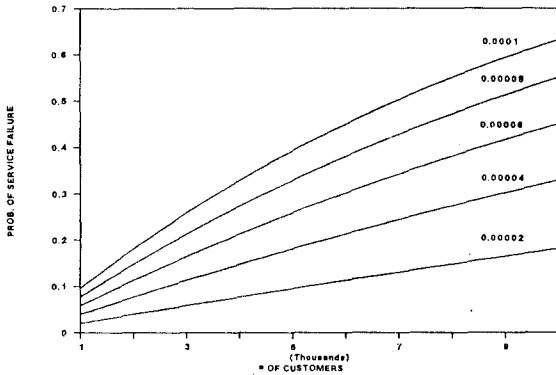
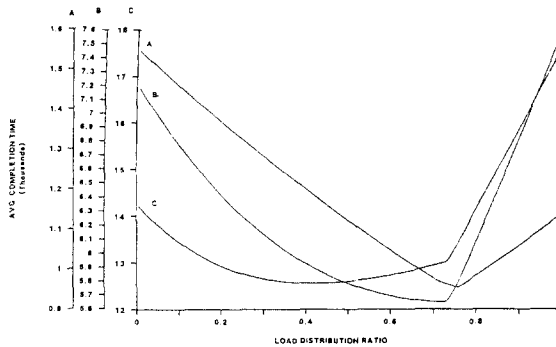


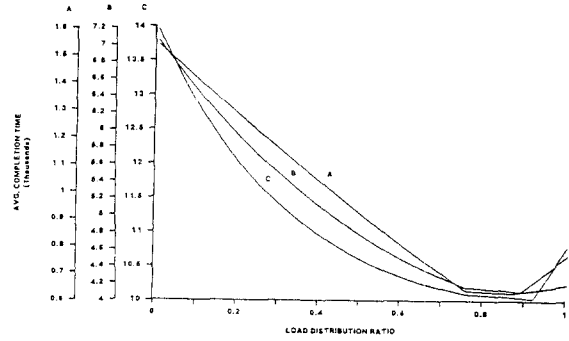
그림4.2 실행 실패 확률의 비교 II (SRS)
Fig4.2 Comparison of the Probability of Service Failure



($T_{sch}=100, \mu_a=3, \mu_b=1, \epsilon_a=0.0004, \epsilon_b=0.0001, k_1=k_3=0.4, k_2=k_4=40$, 명령수: A=1000, B=5000, C=10000)

그림4.3 명령수와 분배율과의 관계(HSS)
Fig 4.3 The Relation of the number of Customers to Load Distribution Ratio(HSS)

수에 따른 최소 평균 실행 완료 시간을 가지는 배분율이 각각 0.75, 0.705, 0.415 이고, MSS는 각각 0.86, 0.885, 0.91 이다. 즉 HSS에서는 방문지의 수가 많을수록 오류 발생 확률이 커지고 또한 로봇 a의 수행 비율은 로봇 b 보다 3배 빠르나 오류 비율이 크기 때문에 배분율은 0.75 보다 작을것으로 예상된다. 그러나 MSS에서는 HSS와는 반대로 오류가 생길 확률이 커질수록 배분율이 0.75 보다 커짐을 알수 있다. 이것은 로봇 a



($T_{sch}=100, \mu_a=3, \mu_b=1, \epsilon_a=0.0004, \epsilon_b=0.0001, k_1=k_3=0.4, k_2=k_4=40$, 명령수: A=1000, B=5000, C=10000)

그림4.4 명령수와 분배율과의 관계(MSS)
Fig 4.4 The Relation of the number of Customers to Load Distribution Ratio(MSS)

표4.1 배분율과 최소 평균 실행 완료 시간의 관계 I

Table 4.1 The relation of load distribution ratio to minimum average completion time I.

($C=5000, T_{sch}=100, \mu_a=3, \mu_b=1, \epsilon_b=0.0001, k_1=k_3=0.4, k_2=k_4=40$)

오류 비율 (ϵ_a)	최적 배분율	
	HSS	MSS
0.0001	0.75	0.085
0.0003	0.72	0.875
0.0005	0.595	0.89
0.0008	0.355	0.915

표4.2 배분율과 최소 평균 실행 완료 시간의 관계 II

Table 4.2 The relation of load distribution ratio to minimum average completion time II.

($C=5000, T_{sch}=100, \mu_a=3, \mu_b=1, \epsilon_a=0.0004, \epsilon_b=0.0001$)

전송, 역전송 지연 상수 (k_1, k_2, k_3, k_4)	최적 배분율	
	HSS	MSS
0.1, 10, 0.1, 10	0.75	0.75
0.3, 30, 0.3, 30	0.75	0.88
0.5, 50, 0.5, 50	0.675	0.885
0.8, 80, 0.8, 80	0.535	0.885

가 주 프로세서에 곧바로 연결되어 있어서 전송 지연이 생기지 않기 때문이다.

표4.1과 표4.2에는 HSS와 MSS에 대해서, 로봇 a의 오류 비율이 달라지는 경우 및 전송 지연 상수가 달라졌을 경우, 최소 평균 실행 완료 시간을 가지는 배분율을 표시하고 있다.

V. 시뮬레이션

앞 절에서 분석한 결과의 타당성을 입증하기 위해 컴퓨터 시뮬레이션을 수행한다. 주어진 조건하에서 제시한 모델의 수

표5.1 모델 분석 결과

Table 5.1 Mathematical analysis results.

모델	명령수	평균 실행 완료시간	실행실패 확률	실행 실패 명령 갯수의 기대값
SRS	1000	1000.00	0.095	48.42
	5500	5500.00	0.423	1269.76
	10000	10000.00	0.632	3679.24
SPS- I	1000	1000.09	0.005	1.59
	5500	5500.36	0.106	212.19
	10000	10000.50	0.264	1036.38
SPS- II	1000	761.43	0.005	1.59
	5500	4369.22	0.106	212.34
	10000	8125.38	0.264	1036.73
HSS	1000	522.77	0.005	1.59
	5500	3238.09	0.106	212.20
	10000	6250.25	0.264	1036.38

표5.2 컴퓨터 시뮬레이션 결과

Table 5.2 Computer simulation results.

모델	명령수	평균 실행 완료시간	실행실패 확률	실행실패 명령 갯수의 기대값	반복 횟수
SRS	1000	1000.00	0.09	44.87	2000
	5500	5500.00	0.423	1295.99	1000
	10000	9999.81	0.617	3650.85	1000
SPS- I	1000	999.91	0.005	1.54	4000
	5500	5514.69	0.109	212.50	3000
	10000	9980.92	0.256	1016.18	3000
SPS- II	1000	779.00	0.005	1.54	3000
	5500	4330.66	0.106	212.07	2000
	10000	7950.74	0.260	1060.22	2000
HSS	1000	541.53	0.005	1.45	4000
	5500	3498.95	0.112	214.57	2000
	10000	6696.54	0.256	1029.74	3000

학적 분석 결과와 그것의 컴퓨터 시뮬레이션 결과를 표5.1과 표5.2에서 보여주고 있다.

$$\begin{aligned}
 \text{조건} \quad & T_{sch} = 0 \\
 & \mu_a = \mu_b = 1 \\
 & \alpha = 0.5 \\
 & \epsilon_a = \epsilon_b = 0.0001 \\
 & \text{배분율} = 0.5 \\
 & \text{전송, 역전송 상수} : k_1=k_2=k_3=k_4=0
 \end{aligned}$$

주어진 조건에서 전송, 역전송 지연 상수를 0으로 했기 때문에 HSS와 MSS가 같은 결과를 나타내므로 표5.1, 표5.2 모두 MSS에 대한 결과를 제외 시켰다. 또한 결과에서 보면 SPS- I, SPS- II, HSS의 실행 실패 확률과 실행 실패하여 남은 명령어 갯수의 기대값이 같음을 알수 있고 대신 평균 실행 완료 시간은 HSS가 가장 작으나 전송 지연을 고려하면 MSS가 가장 작음을 알수 있다.

모델의 수학적 분석 결과와 컴퓨터 시뮬레이션 결과와의 오차는 지수 분포를 표현하기 위해 사용한 난수 발생기 오차의 영향이 크며, 컴퓨터 시뮬레이션의 반복 시행 횟수를 많이하지 못한 원인도 있다.

VI. 결론

본 논문에서는 네 가지의 다중 로봇 시스템의 구성 방법과 모델링 방법 및 수학적 분석 방법을 제시하였다. 특히 오류가 생길 경우 회복하는 기능을 가진, 간단한 독립 작업을 하는 두 로봇 경우에 대해서 살펴보았다. 제시된 시스템 모델의 수식적 분석 결과와 시뮬레이션 결과와의 비교에서 보는바와 같이 두

결과가 상당히 근사한 것을 알수 있다. 따라서 시간이 많이 소요되는 컴퓨터 시뮬레이션 없이 수식을 이용하여 여러가지 최적 상태를 쉽게 빨리 예측할 수 있다. 확장된 형태인 EHSS 경우는 수식적 결과만 제시하였으므로 실제 계산 및 시뮬레이션이 필요하다. 여기에서는 일단 로봇에 오류가 생기면 복구해서 사용하지 못하는 것으로 가정했으나 앞으로는 오류 복구 시간이 주어지는 경우에 대해서도 분석이 가능하며 편의상 수행 비율의 분포가 지수 분포임을 가정했으나 다른 분포에 대해서도 분석이 필요하다. 본연구는 실제적인 MRS를 분석하기 위한 기본적인 도구가 될 수 있을 것이며 나아가서는 FMS 성능 평가의 기초작업이라 할수 있다.

참고 문헌

- [1] V.Dupourgue, "Towards Multiple-Processor and Multi-Robot Controllers," IEEE Int'l Conf. on Robotics & Automation 1986.
- [2] J.Y.Han and C.Y.Wang "Modeling and Performance Evaluation of Multiprocessor Systems for Real-Time Nonlinear Robot Control," IEEE Conf. on R&A 1989.
- [3] C.A.Klein, W.Wahawisan, "Use of a Multiprocessor for Control of a Robotic System," Int'l Journal of Robotics Research, Vol.1, No.2, pp.45-59, 1982.
- [4] K.G.Shin, "Communication Primitive for a Distributed Multi-Robot System," IEEE Int'l Conf. on Robotics & Automation 1985.
- [5] K.G.Shin, "Intertask Communication in an Integrated Multirobot System," IEEE Journal of Robotics & Automation, Vol. RA-3, No.2, pp.90-100 Apr. 1987.
- [6] J.Y.S.Luh and Y.F.Zheng "An Interactively Hierarchical Control Scheme for Two Coordinating Industrial Robots", Proceedings of 25th CDC Dec.1986
- [7] S.S.Hussaini, "Multiple Manipulators and Robotic Workcell Coordination," IEEE Int'l Conf. on Robotics & Automation 1983.
- [8] Y.F.Zheng, J.Y.S.Luh, P.F.Jia, "A Real-Time Distributed Computer System for Coordinated-Motion," IEEE Int'l Conf. on Robotics & Automation, pp.1236-1241, 1987.
- [9] Insup Lee, R.King, X.Yun, "A Real-Time Kernel for Distributed Multi-Robot Systems," American Control Conference, pp.1083-1088, 1988.
- [10] S.J.Chang, "The Generation of Diagnostic Heuristics for Automated Error Recovery in Manufacturing Workstations," IEEE Journal of Robotics & Automation, pp.522-527, 1989.
- [11] E.L.Mellado, R.Alami, "A Failure Recovery Scheme for Assembly Workcells," IEEE Int'l Conf. on Robotics & Automation, pp.702-707, 1990.
- [12] R.E.Smith, M.Gini, "Robot Tracking and Control Issues in an Intelligent Error Recovery System," IEEE Int'l Conf. on Robotics & Automation, pp.1070-1075, 1986.
- [13] D.A.Reed, D.C.Grunwald, "The Performance of Multicomputer Interconnection Network," IEEE Computer, Vol.20, No.6, pp.63-73, Jun. 1987.
- [14] 이기동, 조혜경, 이범희, 고명삼, "다중 로봇트 시스템의 작업 스케줄링 및 성능 평가," 한국 자동 제어 학술 회의 (KACC) 논문집, 제 1 권, pp.423-428, 10월 1990.
- [15] L.Kleinrock, Queueing Systems, John Wiley & Sons, New York, pp.141-143, 1975.