

그리드 단위의 제어에 기반한 자동화 터미널의 AGV 시스템 운영 방안

이용환* · 박은경** · 박태진* · 류광렬* · 김갑환***

*부산대학교 컴퓨터공학과, **LG CNS, ***부산대학교 산업공학과

AGV System Operating Scheme based on Grid Level Control in Automated Terminal

Yong-Hwan Lee* · Eun-Kyung Park** · Tae-Jin Park* · Kwang-Ryel Ryu* · Kap-Hwan Kim***

*Dept. of Computer Engineering, Pusan National University, Pusan 609-735, Korea

**LG CNS Co. Ltd.

***Dept. of Industrial Engineering, Pusan National University, Pusan 609-735, Korea

요 약 : 본 논문에서는 자동화 컨테이너 터미널에서의 AGV 시스템 운영을 위한 방안을 제시하고 그 검증을 위한 시뮬레이터를 개발한다. 우선적으로 AGV의 생산성을 고려한 자동화 터미널의 구조를 제시하고 그러한 구조상에서의 AGV의 교통통제 방안을 제시한다. 기존에 연구된 AGV 시스템들은 주로 존 제어 방식을 사용하고 있다. 그러나 본 연구에서는 AGV 운행의 유연성을 위해 그리드 기반의 제어 방식을 제안하고 사용한다. 최종적으로 실제 현장과 유사한 작업을 할 수 있고 운영규칙의 수정 개선이 용이한 시뮬레이션 환경을 개발한다.

핵심용어 : AGV, 자동화 컨테이너 터미널, 교통 통제, 시뮬레이션

Abstract : In this paper, we present a scheme for developing a simulator which verify the AGV system operation in automated container terminal. First of all, we propose a guide path in automated terminal considering AGV productivity and present a traffic control scheme in a such guide path. Generally, the AGV system can be controlled by zone control scheme, but there are no flexibility and use simple operation. In this paper, we propose and use grid based control scheme to improve the flexibility of the AGV movement. Lastly, we construct the simulation environment which can do similar to that of real tasks and ease the modification and improvement of the control logics.

Key words : AGV, Automated Container Terminal, Traffic Control, Simulation

1. 서 론

산업이 발전함에 따라 수출입 화물량이 급격하게 증가하고 있으나 그에 비해 물류 처리 시설 능력은 한정되어 있기에 빠른 시간에 수출입 화물을 처리할 수 없으며, 이로 인해 물류비는 상승하고 국가 경쟁력은 약화된다. 그러므로 각 항만에서는 수출입 화물의 취급을 자동화하여 시간과 비용을 절감하고 물류비가 차지하는 비율을 낮추려는 노력을 하고 있다. 네덜란드, 싱가포르 및 일본 등 항만 선진국에서는 이미 항만의 자동화가 많이 이루어졌으며 현재도 이에 관한 연구를 지속적으로 추진하고 있다. 현재 세계적으로 항만 자동화 시스템을 갖춘 대표적인 곳은 네덜란드 로테르담의 ECT 항만이다. 하지만 ECT 항만은 AGV 운영을 극도로 단순화시켜 생산성이 낮고 소요 비용이 많이 드는 형태로 운영되고 있다. 독일 CTA는

ECT의 문제점을 보완하여 보다 효율적인 AGV 운영 방안을 제시, 적용하여 현재 시험 운행 중에 있다. 이밖에 영국, 일본 등 항만 자동화를 실험하고 있는 국가들은 여러 기술적인 문제로 인해 AGV의 완전자동화 운영을 하지 못하고 있다.

항만 자동화란 물류의 공정 전반에 있어서의 자동화를 뜻하는 것이지만 본 논문에서는 그 중 선석과 장치장 사이에서 컨테이너를 운반하는 AGV(Automated Guided Vehicle) 시스템 운영방안에 대해서 기술한다. AGV의 운영은 AGV시스템 컨트롤러에 의해 이루어지며 AGV 시스템 컨트롤러는 다음의 기능을 가진다. 우선, 작업현장(shop floor) 컨트롤러로부터 컨테이너 운반요구를 받아서 그 운반을 수행할 AGV를 선정하고 AGV의 이동을 위한 경로를 설정해 주며(routing), 교착상태(dead-lock) 발생 시 그 상황을 해소 시켜주는 역할(dead-lock resolution)과 경로상의 traffic을 통제해 주는 역할(traffic

* yhlee1@pusan.ac.kr, 051-510-3531

** eunkpark@lgcns.com, 019-285-1971

* p.rktj@pusan.ac.kr, 051-510-3531

* krryu@pusan.ac.kr, 051-510-2453

*** 중신회원, kapkim@pusan.ac.kr, 051-510-1483

control), 유휴 AGV의 대기위치 등을 지정해서 이동을 명령하는 역할(positioning) 등을 수행한다.

AGV 시스템 운영에 관한 연구들은 주로 존(zone) 제어 방식(임동순, 2000) (J. J. M, 1996) (Yeh, M. S, 1998)을 기반으로 하고 있다. 존 제어 방식은 AGV의 전체 주행경로들을 존(zone)과 링크(link)로 구분한다. 주행 경로에서 교차로, 작업장 그리고 이들을 연결하는 트랙을 존으로 보고 존과 존들을 링크로 연결해서 표현한다. 존 제어 방식은 주로 존에서의 교통통제를 통하여 AGV 시스템을 운영한다. 존 제어 방식은 비교적 교통 통제가 용이하지만 각 존에 진입할 수 있는 AGV의 수(일반적으로 1대)가 제한되기 때문에 AGV 운행의 효율성이 떨어진다. 또한 존 제어 방식은 대상 시스템의 guide path가 존의 구분이 힘들도록 설정되어 있는 경우에 적용이 힘들다. 따라서 본 연구에서는 guide path의 형태에 관계없이 적용가능 하면서 AGV 운행의 효율성을 높일 수 있는 방안을 제시한다.

본 논문에서 제시하는 AGV 운영 제어 방식은 그리드(grid) 기반의 제어 방식이다. 우선, 대상 AGV 시스템의 전체 guide path 영역을 격자모양으로 그리드라는 작은 단위로 구분하고 이 그리드를 제어를 위한 기본 단위로 사용한다. AGV가 주행 시 이들 그리드들을 점유하도록 하고, 하나의 그리드들은 한 시점에 하나의 AGV만이 점유 가능하도록 한다. 이상의 기본 골격 하에서 교착상태가 발생 가능한 경우를 알아보고 일부 그리드들의 예약을 통하여 교착상태를 해결한다.

그리드 단위의 제어 방식은 guide path의 형태에 관계없이 사용가능하고 기본 제어 단위가 그리드로서 기존의 존 제어 방식에 비해 작은 단위의 영역이기 때문에 원활한 교통 통제가 가능하다면 보다 효율적인 AGV 운영 제어 방식이 될 것이다.

AGV 시스템 컨트롤 소프트웨어를 개발하기 위해서 포함되는 각 모듈에서 구현될 여러 운영 논리들은 작업장의 특성을 고려하여 면밀히 평가해 볼 필요가 있다. 그렇게 함으로써 그런 운영 논리들의 조합으로 구현되는 컨트롤 소프트웨어가 대상시스템에 적용되었을 때 최상의 성능을 발휘할 수가 있게 된다. 따라서 이러한 운영논리들을 실제 현장의 AGV 시스템에 설치하기 전에 충분한 테스트를 거쳐 검증하는 것이 중요하고 현장 테스트와 유사하게 작업을 재현할 수 있는 소프트웨어 환경이 필요하다.

시뮬레이션은 그 기법의 특성상 현장의 상세한 형태까지 묘사가 가능하므로 AGV 시스템 설계안의 분석 뿐 아니라 컨트롤 소프트웨어의 기능에 따른 운영규칙의 대안을 평가해 보는 데 좋은 도구라 할 수 있다. 본 연구에서는 사용자로 하여금 컨트롤 소프트웨어의 운영규칙을 수정 개선하는 것을 용이하게 하고 시스템의 상황과 환경을 수시로 변경하여 테스트해 볼 수 있도록 시뮬레이션 환경을 개발하고자 한다. 그리하여 향후 이 시뮬레이션 환경을 이용하여 다양한 운영규칙의 평가를 통한 효율적인 알고리즘 개발의 기초를 마련하고자 한다.

2. 국내외 연구동향

AGV 시스템 컨트롤러에 포함되어야 할 알고리즘에는 시스

템의 종류에 따라 다소 차이가 나겠으나 guide path 네트워크를 동적(dynamic)으로 바꾸어 줄 네트워크 configuration 모듈, AGV의 운행경로를 결정해줄 routing 모듈, AGV에 작업을 부과해 줄 dispatching 모듈, 유휴 AGV의 대기위치를 정해 줄 positional 모듈 등으로 구성된다.

이상과 같은 문제들이 다른 학자들에 의해서 어떻게 연구되어 왔는지를 살펴보면 guide path 네트워크 설계 문제(Gaskin, R. J, 1987) (Siniech, D. 1992)는 몇몇 학자들에 의해 무궤도 AGV를 위한 guide path 네트워크 설계 알고리즘이 개발되었으나 수리적으로 수식화되었기 때문에 계산소요시간이 길어서 운영논리의 일부로서 사용하기에는 부적절하다고 할 수 있다. 그 이후 계산시간을 줄이기 위한 노력이 있었으나 그 결과가 운영논리의 일부로 사용될 수 있을 정도인지는 확인되지 않았다. 더구나 이상의 연구들은 대부분 congestion을 고려하지 않은 정적인(static) 모형이라는 제약이 있다.

Routing의 문제(Alami, R. 1998)는 몇몇 학자들에 의해서 좋은 성능을 낼 수 있는 해법이 개발되어 왔는데 특히 blocking을 고려한 운행경로의 결정해법은 무궤도 AGV의 제어에 유용하리라 기대되나 해법의 복잡성으로 인해 실용적인 적용은 좀 더 검토해 볼 문제라고 생각되며 이보다는 분산화된 차량의 지능을 이용하여 학습기능을 갖춘 최적화 지능을 부여하는 것이 더욱 미래지향적인 방법이라고 생각된다.

Dispatching 문제도 여러 학자들(Evers, J. J. 1996) (Ihsan, S. 1992) (Kim, S. H. 1999) (Klein, C. M. 1996) (Lim, J. K. 2003)에 의해서 연구되었으나 실용적인 측면에서 간단한 결정규칙에 의해서 순서를 결정하는 방식이 보편적으로 많이 사용되고 있다. 그러나 항만과 같이 여러 장비들이 협동 작업을 하는 경우에는 동기화가 매우 중요한 바, 이를 위한 배차 논리의 개발이 필요하다고 할 수 있다.

유휴 AGV의 positioning 문제는 그 동안 제조시스템에서의 간단한 시스템을 대상으로 연구가 이루어졌으므로(Kim, K. H. 1995; Kim, K. H. 1997) 항만과 같은 문제를 해결하기 위해서는 추가의 연구가 필요하다.

본 논문에서는 위의 각 모듈 개발의 전 단계가 되는 기본적인 AGV 시스템의 통제방안으로서 기존의 존 제어 방식이 아닌 그리드 기반의 새로운 방안으로서 AGV 시스템의 통제방안을 제시한다. 그리고, AGV 시스템 시뮬레이션 환경을 개발하고 향후 이 시뮬레이션 시스템을 통하여 각 모듈의 효율적 알고리즘을 개발하고 검증하는 것이 가능하도록 할 것이다.

3. 자동화 터미널의 개요

현재 우리나라가 동북아 물류거점으로 발돋움하고 있는 시점에서 컨테이너 터미널의 작업 생산성 향상은 중요한 이슈로 부각되고 있다. 이러한 생산성 향상을 위해 우리나라에서도 광양항에 무인자동화터미널의 건설을 추진중이다. 본 논문은 광양항의 자동화 터미널 건설을 위한 계획안을 기본으로 한다. 자동화 터미널의 일반적인 구조를 살펴보면 Fig. 1과 같

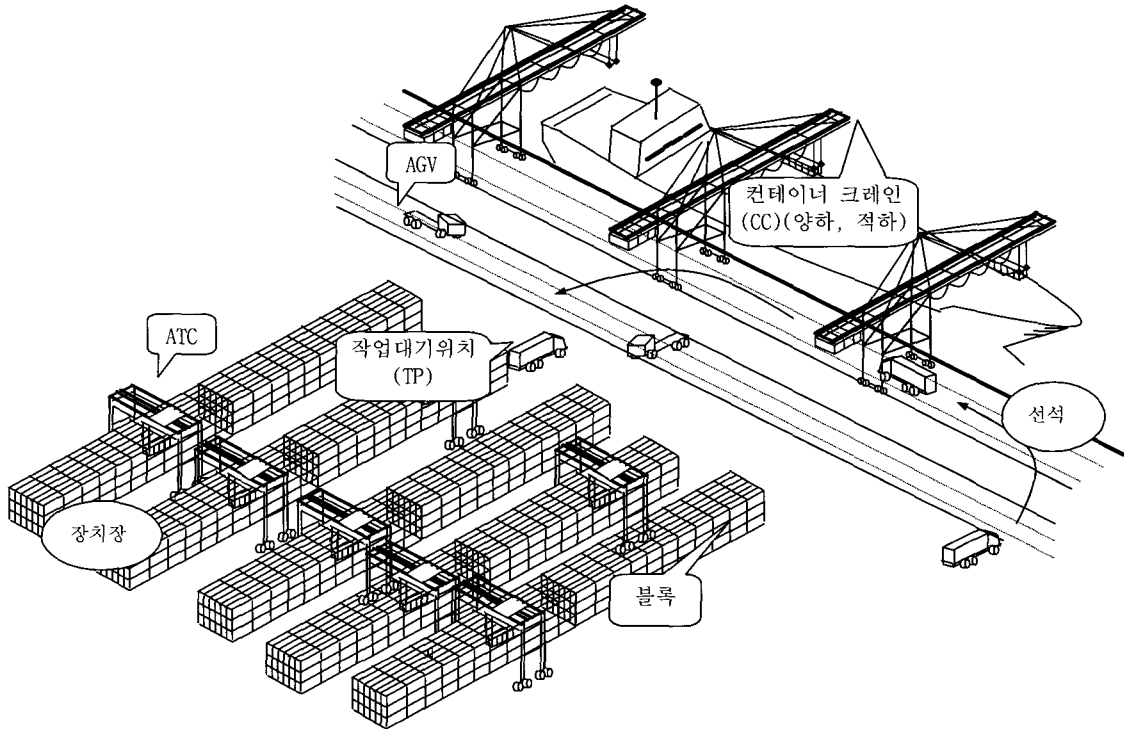


Fig. 1 Automated container terminal

다. Fig. 1에서 보는 것처럼 자동화 터미널은 해측의 선박과 육측의 장치장으로 구성된다. 선박에는 선박으로부터 수입컨테이너를 양하하거나 선박으로 수출컨테이너를 적재하는 CC(Container Crane)들이 있다. 장치장은 수출입 컨테이너를 적재해두는 곳으로 ATC(Automated Transfer Crane)가 컨테이너를 들어내거나 적재하는 일을 한다. 장치장과 CC사이의 컨테이너 운반은 AGV가 담당하게 된다. 장치장은 여러개의 장치장 블록으로 구성되어 되며 각 블록마다 TP(Transfer Point)들이 있고 AGV가 이들 TP에 도착하면 ATC가 AGV로 수출컨테이너를 적재하거나 AGV로부터 수입컨테이너를 집어 장치장에 적재하게 된다. 따라서 AGV들은 장치장 블록의 TP들과 선박의 CC사이를 오가면서 컨테이너를 운반하게 된다. 일반적으로 컨테이너 터미널의 생산성은 CC의 작업이 얼마나 원만하게 이루어지는가에 있다. 즉, CC가 쉬지 않고 지속적으로 작업을 할 때 생산성이 가장 높아지게 된다. 따라서 CC에서의 작업의 지연이 발생하지 않도록 AGV들이 CC가 요구하는 시간 이전에 CC에 도달해있어야 한다. 결국 AGV의 효율적 운영은 자동화 컨테이너 터미널의 생산성에 있어 중요한 요소라고 할 수 있다.

4. AGV 경로 설계(Path Design)

앞장에서 설명하였듯이 AGV는 TP와 CC사이를 오가면서 컨테이너를 운반한다. 따라서 장치장과 선박사이에서 AGV가 주행할 수 있는 가상의 경로를 설계하여야 한다. 선진 자동화 컨테이너 터미널에서 사용중인 AGV의 경로는 Closed

Loop방식과 Cross Lane방식이 있다.

Closed Loop방식은 선박들과 장치장을 연결하는 하나 또는 그 이상의 폐 루트로 경로를 설계하는 방식이다. Fig. 2에서 보는 것처럼 선박에서 장치장으로 컨테이너를 옮기기 위해서 AGV는 CC에서 컨테이너를 받아 폐 루트를 따라 TP까지 운반한다. Closed Loop 방식은 교통 통제 논리는 비교적 간단한 반면 AGV가 폐 루트를 따라 한 방향으로 진행을 하게 되므로 AGV의 이동거리가 길어지게 되고 AGV의 생산성이 떨어지게 된다. 이러한 이유로 Closed Loop방식을 채택하여 운영했던 네덜란드의 ECT도 현재는 Cross Lane방식으로 설계를 변경 적용하고 있다.

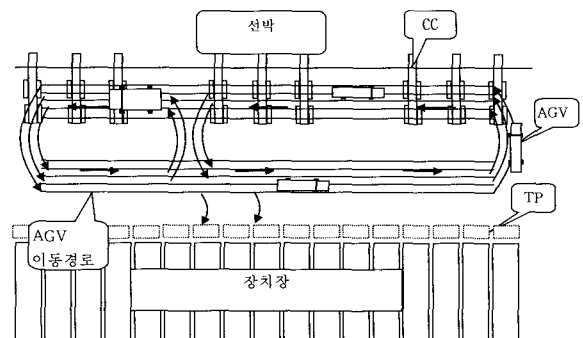


Fig. 2 Closed Loop path design

Cross Lane방식은 독일의 CTA가 채택한 방법으로 AGV가 특정 폐 루트로서 운행하도록 경로를 정하지 않고 어떠한 형태로든지 운행이 가능하도록 하는 방식이다. Fig. 3에서 보

듯이 선석쪽과 장치장쪽에 가로방향의 주행이 가능(일반적으로 여러개의 가상 레인(lane)을 둔다)하고 중간영역에는 세로 방향의 주행이 가능(일반적으로 여러개의 가상 레인을 둔다)하도록 하여 장치장과 선석 사이를 오가는 AGV가 어디서든지 회전이 가능하도록 한다. 따라서 Closed Loop방식에 비해 AGV의 주행 거리가 짧아지게 되고 그에 따라 AGV의 생산성도 향상시킬 수 있다. 하지만 Closed Loop에 비해 교통 통제는 다소 복잡해진다.

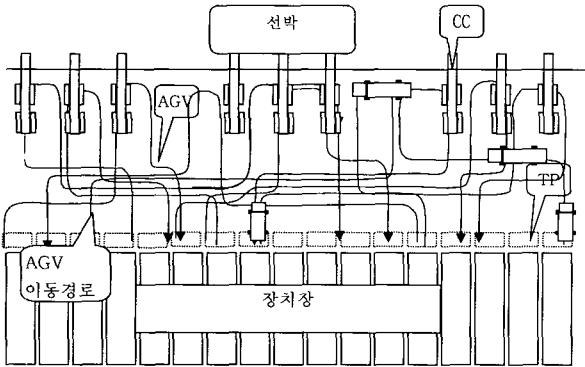


Fig. 3 Cross Lane path design

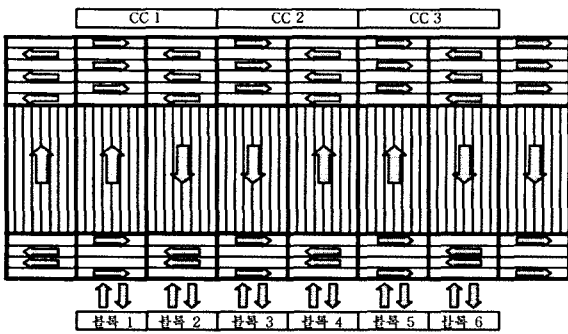


Fig. 4 Path design in this paper

본 연구에서는 대상 자동화 터미널의 경로 설계로서 Cross Lane방식을 채택하였다. 하지만 교통 통제를 용이하게 하기 위하여 Fig. 4처럼 각 경로마다 진행방향을 지정한다. 본 연구에서는 실제 대상 자동화컨테이너 터미널의 규모를 축소하여 3개의 CC와 6개의 장치장 블록(block)이 있는 형태로 설계한다. 선석쪽에는 6개의 레인을 두되 1, 3, 5 레인(레인 번호는 위쪽부터 차례대로 부여)은 오른쪽 방향으로 AGV가 주행하도록 하고 2, 4, 6 레인은 왼쪽 방향으로 주행하도록 한다. 장치장쪽은 4개의 레인을 두고, 2, 3 레인은 오른쪽 방향, 1, 4 레인은 왼쪽 방향으로 주행하도록 한다. 그리고 장치장과 선석 사이의 수직 레인들은 장치장 블록의 경계에 따라 구역을 나누고 그림과 같이 위, 아래의 진행 방향을 지정하고 각 CC마다 전용 레인도 지정한다(선석 1번부터 차례대로 1, 6, 3 레인). 이와 같이 경로를 설정함으로써 AGV의 이동거리도 짧으면서 교통 통제도 용이하도록 하여 AGV의 생산성을 좋도록

한다. 각 레인의 주행방향을 지정치 않는 때 AGV의 이동거리는 더 짧아질 가능성도 있지만 이 경우 교통 통제 논리가 복잡해져서 효율적 교통 통제를 하지 못하게 됨으로써 생산성이 오히려 더 떨어질 수도 있다.

5. AGV 운영 시스템

본 논문에서는 AGV 운영 시스템을 위하여 그리드 기반의 제어 방식을 제안하고 사용한다. 또한 그리드 기반의 통제 시스템의 검증을 위한 운영 시뮬레이션 환경을 개발한다.

Closed Loop방식 경로 설계 또는 생산공장 시스템에서 흔히 볼 수 있는 그물망 형태의 경로 설계 방식에서는 주행 구간과 교차로 등의 구분이 명확하기 때문에 쉽게 존을 구분 지을 수 있고 존 제어 방식을 통한 제어가 가능하다. 그러나 본 논문에서 대상으로 하고 있는 Cross Lane 방식에서는 거의 모든 곳이 교차로가 될 수 있어 존의 구분이 힘들고 존 제어 방식의 적용이 힘들다. 따라서 본 논문에서는 그리드의 개념을 도입하여 사용한다.

5.1 그리드 단위의 제어 방안

본 논문에서는 AGV의 제어를 위하여 앞 장에서 설명한 경로 설계에 의해 나타나는 guide path를 길이 5-7m내외의 짧은 그리드들로 나눈다. 폭은 선석 쪽 레인은 4m, 장치장 쪽은 7m이고 중간 영역은 5-7m이다. 그리고 이러한 그리드들을 모듈이라는 단위로 묶는다.

모듈은 가로 방향으로는 선석모듈과 주차장 모듈, 장치장 모듈(각 블록의 TP영역까지 장치장 모듈에 포함시킴)로 구분하고 세로방향으로는 장치장 블록의 기본선을 중심으로 모듈을 구분한다. 따라서 본 연구의 대상 터미널에서는 선석 쪽에 8개의 모듈이 있고 주차장과 장치장에도 각각 8개의 모듈이 있다. Fig. 4에서 선석쪽과 장치장쪽은 굵은 실선 및 점선으로 모듈이 구분되어 있고 주차장 쪽은 굵은 실선으로 모듈이 구분되어 있다. 각 모듈은 선석 모듈의 좌측부터 시작해서 1번부터 총 24번까지 번호를 부여한다. 각 모듈내의 그리드들도 같은 방식으로 번호를 부여한다.

그리드 기반의 AGV 제어 방식의 기본은 AGV들이 그리드들을 점유하면서 주행을 할 때 자신이 진행 할 앞 일부 그리드들을 미리 예약을 함으로써 교착상태 발생 가능성이 있는 다른 차량의 진입을 사전에 방지하도록 하는 것이다. 하나의 그리드는 동시에 하나의 AGV만이 점유 또는 예약이 가능하다. AGV의 크기(15.8m)에 비해 그리드의 크기가 작기 때문에 AGV는 직진 주행 시 동시에 3, 4개의 그리드를 점유하게 되고 회전 시에는 더 많은 그리드들을 점유하게 된다. 본 논문에서 AGV가 그리드 영역에 조금만 들어가게 되더라도 그 그리드를 완전히 점유하는 것으로 본다. 따라서 회전 시에는 AGV의 축이 비스듬하게 되면서 인접 그리드들도 점유하게 된다.

그리드 기반의 제어 방안에서 AGV들은 자신의 routing이 결정되고 나면 그 routing따라 그리드들을 점유하면서 운영을

한다. 그러나 단순히 그리드들을 점유하면서 운영을 하게 되면 교착상태의 발생이 가능하고 이를 방지하기 위해서 자신의 routing상의 일부 그리드들을 예약하고 예약된 그리드에는 타 AGV가 진입하지 못하도록 한다.

5.2 AGV 운영 시스템

본 연구에서는 간단한 교통제어 논리 및 시물레이션 환경을 개발한다. 교통제어 논리의 개발은 단순히 AGV들이 충돌 없이 운행할 수 있는 논리를 개발하는 것으로 앞장에서 설명한 형태로의 경로 설정을 통하여 어느 정도의 교통제어가 용이해진다. 현재는 여기에 간단한 형태의 교착상태 처리 알고리즘을 적용하고 보다 효율적인 알고리즘의 개발은 향후에 연구할 것이다. 현재 개발중인 시스템에 적용하는 교착상태 처리 알고리즘은 교착상태를 사전에 방지하기 위하여 자신이 진행해야 할 일부 영역을 미리 예약하여 다른 차량이 진입하지 못하도록 하는 방식을 사용하고 있다.

본 논문에서 시물레이션을 위하여 eM-Plant라는 시물레이션 소프트웨어를 사용한다. eM-Plant는 이벤트 중심의 시물레이션 소프트웨어로 이벤트들을 정의하고 이벤트들 사이의 관계를 정의해 주면 이벤트 컨트롤러가 전체 이벤트를 시간순서대로 관리 수행해 주면서 시물레이션을 수행한다. 즉, 현재 등록되어 있는 이벤트 중에서 시간상 가장 빠른 이벤트를 수행시켜주고 그 이벤트로 인해 발생하는 또 다른 이벤트를 이벤트 리스트에 추가시키는 방식으로 시물레이션을 한다.

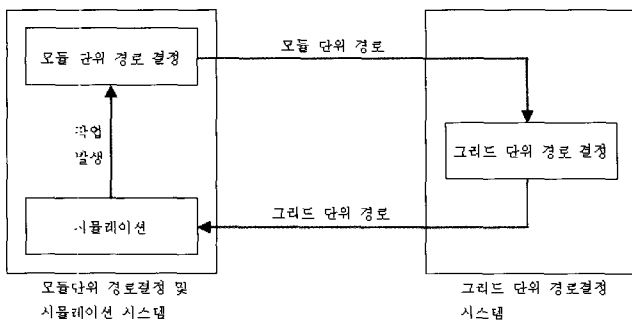


Fig. 5 Outline of operation system

AGV 운영 시스템을 위하여 두 개의 하위 시스템으로 구분하여 개발한다. 첫 번째 하위시스템은 모듈단위의 경로 결정 및 시물레이션 시스템이고 두 번째 하위시스템은 그리드 단위의 경로 결정 시스템이다. Fig. 5처럼 모듈단위 경로결정 및 시물레이션 시스템에서 AGV에 새로운 작업이 발생하면 먼저 모듈 단위의 경로를 결정하고 결정된 모듈 순서를 그리드 단위 경로 결정 시스템에 전달한다. 그리드 단위 결정 시스템은 이 정보를 받아서 그리드 단위의 경로를 결정하고 시물레이션 시스템에 그 결과를 넘겨준다. 시물레이션 시스템은 받은 결과에 따라 시물레이션을 수행한다. 두 시스템의 연동에 대한 보다 자세한 내용은 (4)절에 다시 설명한다.

(1) 모듈단위의 경로 결정 및 시물레이션 시스템

이 절에서 설명할 시스템은 eM-Plant를 사용하여 실제 시물레이션을 수행하는 시스템이다. 현재는 특별한 dispatching 알고리즘을 적용하는 대신 AGV의 작업을 무작위적으로 생성하도록 하였다. AGV가 CC나 TP에 도착하면 일정시간 뒤에 임의의 다른 CC나 TP로 이동만 하도록 작업을 할당하였다. 이는 AGV가 이동해야 하는 모든 경우와 상황에 대해서 AGV가 실제로 효율적으로 움직이는지를 테스트하기 위한 방안이다.

AGV가 작업을 할당받고 나면 목적지까지 이동하기 위해 모듈단위의 경로를 결정한다. 모듈단위의 경로 결정은 AGV의 현재 위치와 다음에 수행될 작업이 완료될 위치에 따라 크게 3가지 방식으로 이루어진다. 첫째, AGV의 현재 위치와 작업 완료 위치가 선석/장치장일 경우 AGV는 현재위치로부터 주차장 모듈직전까지 수평주행을 하고 주차장 모듈에서 수직주행, 주차장 모듈직후부터 작업 완료 위치까지 수평 주행을 한다. 둘째, 현재 위치가 선석/장치장이고 작업 완료 위치가 주차장일 경우에는 AGV는 현재위치부터 주차장 모듈직전까지 수평주행을 하고 주차장 모듈에서 수직 주행을 한다. 셋째, 현재 위치가 주차장이고 작업 완료 위치가 선석/장치장일 경우에 AGV는 주차장 모듈에서 수직 주행을 하고 주차장 모듈직 후부터 작업 완료 위치까지 수평 주행을 한다.

첫째 경우의 경로 결정은 먼저 적절한 주차장 모듈을 선택한 다음 시작위치, 주차장 바로 아래/위 모듈, 주차장 모듈, 주차장 바로 위/아래 모듈, 작업 완료 위치까지의 순서로 통과할 모듈들을 결정한다. 예를 들어 Fig. 6에서 AGV의 현재 위치가 2이고 작업완료 위치가 22일 경우 AGV가 지나갈 주차장 쪽 모듈 선택을 12로 했을 때, 모듈 단위의 경로는 2, 3, 4, 12, 20, 21, 22의 순서가 된다. AGV가 지나갈 주차장 모듈의 선택은 작업 완료 위치와 모듈내의 주차 상태를 감안하여 이루어진다. 둘째 경우의 경로 결정은 시작위치, 주차장 바로 아래/위 모듈, 주차장 모듈(작업 완료 위치) 순서로 결정한다. 예를 들어 Fig. 7에서 현재 위치가 2이고 작업 완료 위치가 12일 경우 모듈 단위의 경로는 2, 3, 4, 12의 순서가 된다. 셋째 경우의 경로결정은 주차장 모듈, 주차장 바로 아래/위 모듈의 순서로 결정한다. 예를 들어, Fig. 8에서 현재 위치가 12이고 작업 완료 위치가 22일 경우의 모듈 단위의 경로는 12, 20, 21, 22의 순서가 된다.

1	2(S)	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22(D)	23	24

S는 AGV 시작 위치, D는 AGV 작업 완료 위치

Fig. 6 When both S and D are in berth/yard

모듈 단위의 경로를 결정하고 나면 그리드 단위의 경로결정 시 장치장에서 어떠한 레인을 선택하는 것이 좋을지 추천 레인을 결정한다. 추천 레인은 각 레인의 혼잡한 정도와 AGV

의 이동 거리를 고려하여 결정한다. 장치장 쪽의 각 모듈들은 각 레인마다의 혼잡 정도(추천 빈도 수)를 기록하는 테이블을 가지고 있다. 추천트랙은 AGV의 주행방향과 일치하는 레인 중 혼잡 정도가 낮은 레인으로 결정한다. 시뮬레이션 방법과 시뮬레이션 시 교착상태의 방지 방안에 대해서는 (3)절, (4)절에서 설명한다.

1	2(S)	3	4	5	6	7	8
9	10	11	12(D)	13	14	15	16
17	18	19	20	21	22	23	24

Fig. 7 When S is in berth/yard and D is in parking area

1	2	3	4	5	6	7	8
9	10	11	12(S)	13	14	15	16
17	18	19	20	21	22(D)	23	24

Fig. 8 When S is in parking area and D is in berth/yard

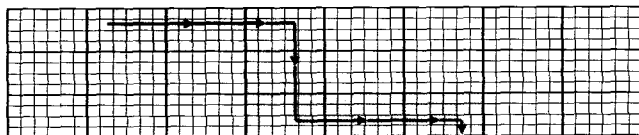


Fig. 9 Grid level routing

(2) 그리드 단위의 경로 결정 시스템

그리드 단위의 경로 결정은 Fig. 9와 같은 그리드 상에서 AGV의 그리드 점유 순서를 결정하는 것이다. 선석 모듈의 주행 레인과 장치장 모듈의 주행 레인은 모듈별 경로 결정 시 미리 결정된다. 따라서 주차 모듈에서의 진행 레인만 결정하면 대략적인 이동 경로는 정해진다. 주차 모듈에서의 레인결정은 간단한 방법으로 AGV가 회전을 할 수 있는 충분한 거리가 되는 레인들을 선택하되 경로 결정 요청이 들어올 때마다 다른 레인을 번갈아 가며 사용하도록 함으로써 서로 다른 AGV들이 서로 다른 레인으로 주행하게 되도록 한다. 이는 물론 교통의 분산을 위해서이다. 주차 모듈에서의 레인이 결정되면 회전지점은 계산을 통하여 찾을 수 있고 그리드들의 순서 또한 계산으로 구할 수 있다. 그러나 AGV의 크기가 그리드의 크기보다 크므로 동시에 여러 개의 그리드를 AGV가 점유하게 된다. 따라서 계산을 통하여 AGV의 앞 차축을 중심 위치로 하여 그 위치를 기준으로 AGV가 점유하는 그리드들을 찾아낸다. eM-Plant는 이벤트 중심의 시뮬레이션 소프트웨어이므로 eM-Plant를 통한 시뮬레이션을 위해서 본 연구에서는 AGV의 앞 차축의 위치가 다른 그리드로 이동하는 것을 AGV의 이동 이벤트로 본다.

따라서 그리드 단위의 경로 결정 시 일정 시간 간격으로 AGV의 앞 차축의 위치를 계산함으로써 각 그리드에서의 앞 차축의 점유시간을 계산할 수 있다. 직진 주행에 비해 회전 주

행 시 그리드의 위치변화가 빈번히 발생하기 때문에 직진 주행 시와 회전 주행 시 같은 시간 간격으로 계산하는 것은 효율적이지 못하다. 또한 직진 주행 시는 그리드의 점유시간이 쉽게 계산될 수 있다. 따라서 직진 주행 시에는 일정거리마다 이동을 하면서 위치 계산을 하였고 회전 주행 시는 0.5도씩 회전 진행을 하면서 위치를 계산하였다. AGV의 앞 차축의 위치가 결정되고 나면 AGV의 네 모서리의 위치의 계산이 가능하고 네 모서리의 위치와 각 그리드의 위치에 대한 계산을 통해 AGV가 점유하는 그리드들을 계산할 수 있다. 결국 모든 위치에 대해서 계산을 하고 나면 주행 경로상의 각 위치마다 앞차축이 속한 그리드와 그때 AGV가 점유하는 그리드들의 리스트들을 확보하게 된다. 앞 차축이 점유한 그리드가 동일한 것들(Fig. 10의 오른쪽 창에서 흰수 1, 2의 경우처럼, 앞 차축이 48로 동일)에 대해서 서로 점유 그리드들을 합치고 나면 최종적으로 주행 경로상의 각 그리드에 대해서 점유 시간과 그때 같이 점유되는 그리드들의 리스트가 결정된다. 이것이 결국 그리드 단위의 경로가 된다.

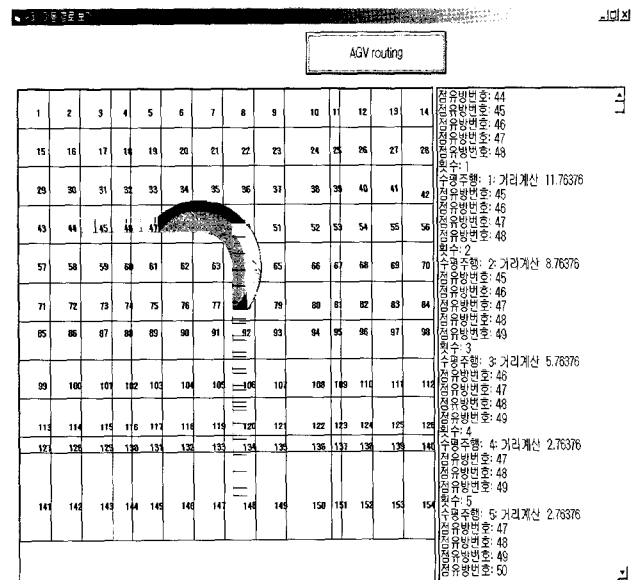


Fig. 10 Grid level routing system

(3) 교착 상태의 방지

본 논문에서는 대상 터미널의 모든 레인에 대해서 AGV 주행 방향이 단 방향으로 지정되어 있으므로 직진 주행을 하는 AGV들간에 교착 상태는 발생하지 않는다. 하지만 AGV들이 회전을 할 때 차 축이 비스듬해지면서 인접레인의 그리드를 점유하게 되고 인접레인에 타 AGV가 주행 중이면 교착상태의 발생이 가능하다. Fig. 11에서 AGV 2는 TP 1로 진입하려고 하고 AGV 1은 TP 2로 진입하려고 할 경우 교착상태가 발생한다. AGV 2가 TP 1에 진입하기 위해서는 앞 칸으로 이동하면서 회전이 되므로 꼬리부분이 위로 올라가게 되고 AGV 1과 충돌을 하게된다. 따라서 AGV 2는 이동을 할 수가 없다.

마찬가지로 AGV 1이 TP 2로 진입하려고 하면 역시 회전을 하면서 AGV 2의 앞 부분과 충돌을 하게 된다. 그러나 만약 AGV들이 3개의 그리드 이상 떨어져 있다면 충돌 없이 회전 주행이 가능하다. 따라서 모든 AGV들은 자신의 앞 3개의 그리드가 비어 있는지 확인하고 그 그리드들을 예약하면서 주행한다. 그러나 AGV가 회전 중에 예약 영역이 확보되지 않아서 AGV가 비스듬하게 정지하게 되면 여러 그리드들을 점유하게 되고 타 AGV의 주행을 방해하거나 교착상태를 야기할 수 있다. 따라서 회전 주행을 할 때에는 회전 시작 전(실제 AGV가 회전 시작하는 지점의 세 칸 이전 그리드)에 회전이 끝나는 지점까지의 모든 그리드를 예약을 하여 반드시 AGV는 평행(수직 또는 수평)하게 정차를 하도록 한다.

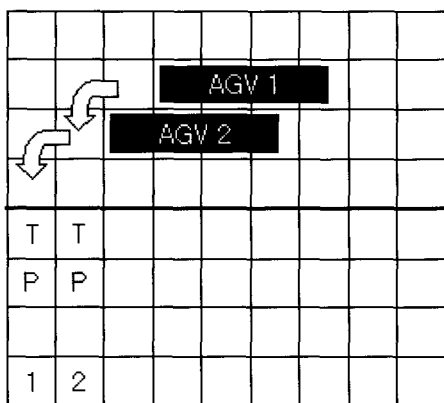


Fig. 11 Deadlock situation 1

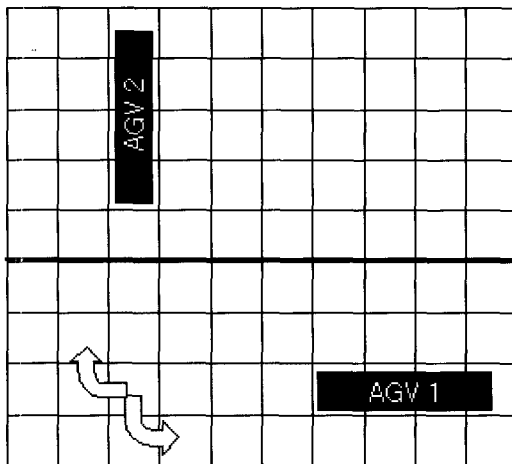


Fig. 12 Deadlock situation 2

하지만, Fig. 12의 경우처럼 예약영역에 의한 교착상태가 발생 가능하다. 그림에서 음영으로 나타나 부분은 각 AGV의 예약영역이다. 두 대의 AGV가 Fig. 12와 같은 상황에 놓여있을 때 AGV 1이 하나의 그리드를 전진하기 위해서는 회전이 끝나는 지점까지의 모든 그리드들을 예약을 하여야 한다. 그러나 일부 예약하여야 할 그리드들이 AGV 2에 의해 이미 예약되어 있어 AGV 1은 예약을 하지 못하고 결국 정지하게 된

다. AGV 2도 같은 이유로 정지하게 되고, 두 AGV는 교착상태가 된다. 그러나 이러한 교착상태는 물리적인 교착상태가 아니라 예약 영역으로 인해 발생하는 논리적인 교착상태이다. 따라서 이러한 경우 어느 하나의 AGV가 자신의 예약 영역을 해제하게 되면 교착상태에서 벗어나게 된다.

이와 같이 매우 단순한 방법에 의해 교착상태를 방지하고 해결할 수 있다. 물리적인 교착상태는 AGV가 주행 시 앞 일부 그리드들 예약을 함으로써 방지할 수 있고, 이러한 예약영역에 의한 논리적 교착상태 발생은 예약 영역을 해제함으로써 해결할 수 있다. 일반적으로 교착상태가 발생하면 이를 해결하기 위하여 AGV가 후진을 하여야 하고 결국 효율성이 떨어지게 된다. 하지만 본 논문에서의 방식은 사전에 물리적 교착상태의 발생을 방지하고, 논리적 교착상태가 발생하더라도 AGV가 후진할 필요 없이 예약 영역을 해제만 하면 해결 가능하다. 따라서 유연성 있는 AGV의 운행이 가능하다.

(4) 두 하위 시스템의 연동 및 시뮬레이션

5.2.1절 및 5.2.2절의 두 시스템간의 데이터 전달을 위하여 Microsoft Access 데이터베이스를 이용하였다. eM-Plant에서 AGV가 작업을 할당받고 모듈 단위의 경로가 결정되면 데이터베이스의 Request 테이블에 Request정보를 기록한 다음 그리드 단위의 경로 결정 시스템을 호출한다. Request정보는 Request 번호(AGV번호)와 함께 모듈의 순서로서 구성된다. 그리드 단위의 경로 결정 시스템은 호출되면 데이터베이스의 Request정보로부터 모듈 순서를 읽어 이를 바탕으로 그리드 단위의 경로를 결정한다. 경로가 결정되고 나면 다시 데이터베이스에 그리드 단위의 경로를 Routing테이블에 기록한다. Routing테이블에는 각 그리드의 방문 순서(AGV의 앞 차축)와 점유시간을 기록하고 그때 같이 점유되는 그리드들의 리스트를 기록한다. 그리드 단위의 경로 결정 시스템의 수행이 종료되면 eM-Plant는 Routing 테이블의 정보를 읽고 이 정보에 따라 시뮬레이션을 수행한다. 시뮬레이션 시 AGV의 교통 제어는 앞 절에 설명한 규칙에 따른다.

6. 실험

본 논문에서 제시한 운영방안의 검증을 위해서 시뮬레이션 실험을 수행하였다. 대상 터미널은 앞장에서 제시한 구조를 사용하였다. AGV의 수를 2대부터 2대씩 늘려가면서 30분 동안 처리하는 작업의 양과 AGV의 utility(전체작업시간중에서 실제 AGV가 주행을 한 시간)를 측정해보았다. AGV가 TP에서 CC로 이동하거나 그 반대의 경우를 하나의 작업으로 보았다.

Fig. 13에서 보듯이 작업량은 AGV의 수가 증가함에 따라 같이 증가하다가 12대가 될 때는 오히려 작업량이 감소하고 있다. 실험 대상으로 하고 있는 터미널의 규모가 매우 작아 AGV가 12대 이상이 되면 교통이 혼잡해지고 AGV의 정지가 빈번하게 발생하게 된다. 그러나 교착상태에 빠지지 않고 AGV들이 원활히 운행되었다. Fig. 14를 보면 AGV의 utility

는 AGV의 수가 많아짐에 따라 서서히 감소함을 알 수 있다. 이는 대상 터미널의 규모가 비교적 좁기 때문에 전체적으로 하나의 작업을 위한 AGV의 운행거리는 짧고, 또한 CC의 수가 적기 때문에 동시에 CC쪽으로 AGV가 집중하는 경우가 빈번하게 되어 결국 AGV들이 정지하는 시간이 길어짐에 기인한 것으로 보인다. 대상 터미널의 규모가 커지게 되면 AGV의 수가 늘더라도 운행 영역의 면적이 넓어지기 때문에 교통 혼잡도는 오히려 낮아질 수도 있을 것으로 예상된다.

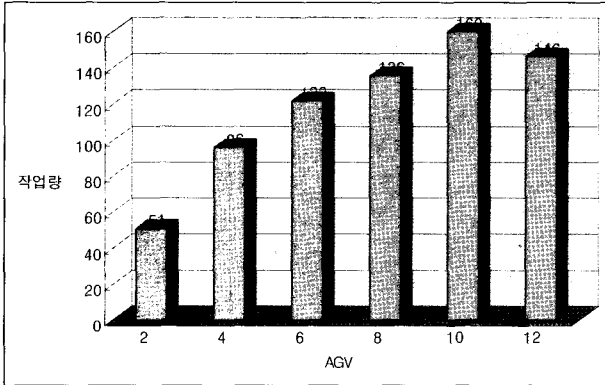


Fig. 13 Comparison of the number of delivery for different number of AGVs

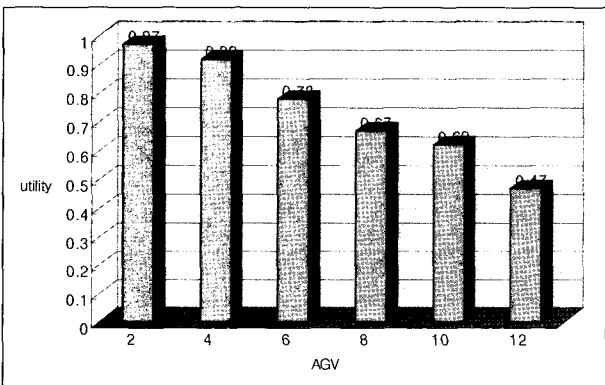


Fig. 14 Comparison of utility for different number of AGVs

7. 결론 및 향후연구

본 논문은 자동화컨테이너 터미널에서의 AGV 시스템의 운영을 위한 기본 논리로서 그리드 단위의 제어 방안을 제시하고 이를 이용한 운영 시스템을 개발하였다. 그리고 이러한 운영논리의 타당성 검증에 위한 시뮬레이터의 개발에 대해서 기술하고 있다. 우선 간단한 형태의 교통 통제 방안을 제시하고 개발된 시스템의 검증을 위한 시뮬레이션 시스템을 개발하였다. 또한 시뮬레이션 시스템은 운영규칙의 변화 시 시스템의 변화를 최소화하기 위하여 두 개의 하위시스템으로 구현을 하였다. 시뮬레이션을 통하여 본 논문에서 제시한 그리드 단위의 운영 방안을 이용하여 교착상태의 발생 없이 유연성 있는

AGV의 운영이 가능함을 확인하였다. 그리고 개발된 시뮬레이션 시스템을 이용한 다양한 운영 규칙에 대한 시뮬레이션을 통하여 자동화터미널을 위한 효율적 운용 시스템의 개발이 가능할 것이다.

현재 시스템의 운영 논리는 아직 기본적인 규칙을 적용하고 있다. 그리고 시뮬레이션 시스템의 일부 보완도 필요하다. 향후 과제로서 시뮬레이션 시스템의 보완과 더불어 개발된 시뮬레이션 시스템을 이용하여 보다 효율적인 교통 통제 논리를 개발, 검증하고 그리드 단위의 제어에 기반한 AGV 운영 시스템 하에서 적용 가능한 AGV 시스템의 하위 모듈들 (Dispatching, Routing 알고리즘 등)의 개발에 대해서 연구를 할 것이다.

후 기

본 연구는 한국과학재단지정 지능형통합항만관리연구센터의 지원에 의한 것입니다.

참 고 문 헌

- [1] 임동순, 존 조정하에서의 AGV 고착 방지, 한국산업공학 회논문지, 26(4), pp. 392-401 (2000)
- [2] Alami, R., Ingrand, F. and Qutub, S., A Scheme for Coordinating Multi-robot Planning Activities and Plans Execution, 13th European Conference on Artificial Intelligence (1998)
- [3] Co, C. G. and Tanchoco, J. M. A., A review of research on AGVS vehicle management, Engineering Costs and Production Economics, Vol. 21, pp. 35-42 (1991)
- [4] Evers, J. J. M. and Koppers, S. A. J., Automated guided vehicle traffic control at a container terminal, Transportation Research-A 30, 2134 (1996)
- [5] Gaskin, R. J. and Tanchoco, J. M. A., Flow path design for automated guided vehicles systems, International Journal of Production Research, 25(5), pp. 667-676 (1987)
- [6] Ihsan, S. and Hommertzheim, D. L., dispatching algorithm for scheduling machines and automated guided vehicles in a flexible manufacturing system, International Journal of Production Research 30, pp. 1059-1079 (1992)
- [7] Kim, K. H., Positioning of automated guided vehicles in a loop layout to minimize the mean vehicle response time, International Journal of Production Economics, 39, pp. 201-214 (1995)
- [8] Kim, K. H. and Kim, J. Y., Estimating Mean Response Time and Positioning Idle Vehicles of Automated Guided Vehicle Systems in Loop Layout, Computers & Industrial Engineering, 33(3,4), pp. 669-672 (1997)

- [9] Kim, S. H. and Hwang, H., An adaptive dispatching algorithm for automated guided vehicles based on an evolutionary process, International Journal of Production Economics, 60-61, pp. 465-472 (1999)
- [10] Kleia, C. M. and Kim, J., AGV dispatching, International Journal of Production Research, 34, pp. 95-110 (1996)
- [11] Lim, J. K., Kim, K. H., Yoshimoto, K., Lee, J. H. and Takahashi, T., A dispatching method for automated guided vehicles by using a bidding concept, OR Spectrum, 25, pp. 25-44 (2003)
- [12] Sinriech, D. and Tanchoco, J. M. A., Intersection graph method for AGV guide-paths design, International Journal of Production Research, 29(9), pp. 1725-1732 (1992)
- [13] Yeh, M. S. and Yeh, W. C., Deadlock Prediction and Avoidance for Zone-control AGVS, International Journal of Production Research, 36(10), pp. 2879-2889 (1998)

원고접수일 : 2002년 12월 3일

원고채택일 : 2003년 6월 11일