

다중 작업환경의 AGV 경로 선정과 충돌 회피에 관한 연구

김 종 선, 유 영 선, 김 세 진, 주 영 훈
 군산대학교 전자정보공학부

A study on the AGV path determination and collision avoidance for multiple environment

Jong Seon Kim, Yeong Seon Yu, Se Jin Kim, Young Hoon Joo
 School of Electronic & information Engineering, Kunsan National University

Abstract - 본 논문은 AGVS(Automated Guide Vehicle System)가 여러 작업환경 및 변경 시 좀더 유연하게 대응할 수 있도록 작업환경 내에서 AGVS에 필요한 작업공간요소로 분류하고 이들을 모델링하는 방법을 제안하였다. 또한, 최단경로 생성 시 A* 알고리즘에서의 평가함수 선정 방법과 최단경로 상에 존재하는 경로의 충돌 모델을 제안하였으며 시뮬레이션을 통해 제안한 방법을 증명하였다.

1. 서 론

물류자동화의 기술에서 빠질 수 없는 것이 바로 AGV이다. AGV는 작업 공간 내에서 이동시켜야 할 물류들을 자동으로 이동 시켜주는 이동로봇으로써, 1953년부터 사용되어지기 시작했다. 기술의 발전에 힘입은 FA(Factory Automation)의 가속화는 FMS(Flexible Manufacturing System) 및 FAS(Flexible Assembly System)를 추구하게 되었으며 이에 따른 다수의 AGV를 작업 공간 내에서 효율적으로 운영하기 위한 연구가 진행되기 시작했다 [1]. AGV의 최단경로(Minimum Cost Path 이하 MCP)를 선정하기 위한 많은 연구가 이루어졌으며 특히, MCP 선정 시에 발생하는 AGV의 충돌을 방지하기 위해 통제구역(Traffic Zone)을 다룬 연구들이 주류를 이루고 있다 [2-5]. 또한 MCP 생성의 연구에서는 이미 그 우수성이 증명된 평가함수를 이용하는 휴리스틱 탐색기법을 사용하고 있다[7]. 최근에는 양방향경로(Bidirectional Path)를 갖는 AGV의 MCP 및 충돌 및 교차 방지와 관련한 연구가 이루어지고 있으나 그 분석이 까다로운 현실이다[6].

본 논문은 양방향 경로에서의 MCP 생성과 충돌 및 교차에 대하여 작업 요소의 측면(경로, 작업장의 위치 등...)에서 접근하여 작업 환경 요소를 일반화 시키는 모델을 제안하고, 모델을 통하여 MCP의 생성과 MCP소비시간(Minimum Cost Path Time) 및 AGV의 이동거리 및 위치를 손쉽게 예측할 수 있음을 증명한다. 그리고, 마지막으로 시뮬레이션을 통해 제안한 방법의 적용 가능성을 보인다.

2. 작업 환경 모델링과 AGVS

FMS(및 FAS의 주요 관심사중 하나는 제품을 생산하기 위한 각 공정단위로 제품을 얼마나 빠르게 이동시키느냐는 것이다. 더욱이 제품은 다수의 공정을 거쳐 완성이 됨은 물론, 공정라인의 종류도 다양하므로 공정 간의 물류이동은 더욱 찾아지게 된다. 따라서 AGVS는 FA로부터 요청된 공정간 물류이동의 복잡함(경로 및 작업소의 위치)을 분석하고 AGV를 충돌 없이 요청된 작업소로 최단시간에 이동시키기 위한 시스템이다.

2.1 작업 환경의 모델링

작업환경을 AGVS가 지나야할 기능으로 살펴봄으로써 작업환경요소를 분류하고 이에 대한 요소의 모델을 제안한다.

2.2.1 AGVS의 기능과 작업환경 요소

작업환경의 요소를 AGVS가 지나야할 기능으로서 살펴 보기 위해 먼저 AGV가 지나야하는 기능을 다음과 같이 3가지로 축약한다..

1. AGVS가 적용된 작업 환경 내에서 작업소의 요청에 따라 AGV의 MCP를 생성한다.
2. 각 AGV를 이동경로 상에서의 충돌이 발생하기 이전에 이를 예측하고 충돌을 회피하도록 유도한다.
3. 모든 AGV와 통신하며 정보를 송수신하여 AGV의 상태를 관리 및 감독한다.

이와 같은 AGVS의 기능을 구현하기 위한 작업환경의 요소로써 경로, 작업소, 통제구역, AGV로 분류할 수 있다. MCP를 생성하기 위해서는 작업을 요청한 작업소의 위치정보와 경로의 정보가 필요하며 운행 중인 AGV 상호간의 충돌을 방지하기 위해서도 경로의 정보가 필요하며 경로가 서로 교차하는 지점 즉, 통제구역의 정보역시 필요하다. 또한, AGV

를 실시간으로 운행하며 관리할 수 있는 무선통신의 프로토콜이 요구되나, AGV의 관리는 작업환경의 요소라고 하기 보다는 작업 환경 내에서 통제되어야 할 대상이기 때문에 환경요소의 분류에서는 제외되어야 한다.

분류된 작업환경의 요소는 AGVS의 데이터베이스로 만들기 위해 각각의 고유 ID, 작업공간상에서의 위치, 거리, 명명전환점(check point) 등의 정보를 추가로 지녀야 하며, 작업환경의 변경 시 이를 수정 가능케 하여 작업환경의 변화에 유연하게 적용할 수 있어야 한다.

2.2.2 제안된 작업환경 모델

분류된 작업환경의 요소는 그림 1과 같이 집합의 요소로써 표현될 수 있으며, 표현된 요소는 AGVS 프로그램의 구현 시 구조체의 구성의 뼈대가 된다.

	ID	Position	Check point	Length	Property
Path	P_{i1}	$P_i = (Type, pos_1, pos_2)$	$P_0 = \{CP_1, CP_2\}$	$P_3 = (length)$	P_5
Station	Working	WS_{j1}	$WS_{j2} = (Type, pos)$	$WS_{j4} = (Type, len_1, len_2)$	WS_{j5}
	Waiting	WA_{k1}	$WA_{k2} = (Type, pos)$	$WA_{k4} = (Type, len_1, len_2)$	WA_{k5}
	Charging	WC_{l1}	$WC_{l2} = (Type, pos)$	$WC_{l4} = (Type, len_1, len_2)$	WC_{l5}
Repair	WR_{m1}	$WR_{m2} = (Type, pos)$	$WR_{m3} = (CP)$	$WR_{m4} = (Type, len_1, len_2)$	WR_{m5}
	Traffic zone	TZ_{n1}	$WR_{n2} = (pos)$	$TZ_{n2} = (zone-length)$	$TZ_{n3} = (Junction-Type)$

그림 1. 작업 환경 요소의 모델

입력된 모든 작업환경요소에 대한 표현은 아래와 같이 각 요소가 지닌 특성들의 집합들로서 표시된다. 하나의 경로입력에 대한 정보는 P_i 로서

$P_i = \sum_{\beta=1}^5 P_{i\beta}$ 로 표현되며 α 개의 경로 입력 P_i 는 식 (1)과 같다.

$$P_i = \sum_{i=1}^{\alpha} \left(\sum_{\beta=1}^5 P_{i\beta} \right) \quad (1)$$

작업소에 대한 표현(1)은 이들이 같은 요소들을 취하기 때문에 하나의 통합된 집합 W_i 로 표현 가능하며, ψ 개(총 작업소)의 입력에 대한 작업소 W_i 는 식 (2), 통제 구역 σ 는 식 (3)과 같다.

$$W_i = \sum_{i=1}^{\psi} W_i = \sum_{i=1}^{\psi} \left(\sum_{\beta=1}^5 W_{i\beta} \right) \quad (2)$$

$$TZ_i = \sum_{i=1}^{\sigma} TZ_i = \sum_{i=1}^{\sigma} \left(\sum_{\beta=1}^5 TZ_{i\beta} \right) \quad (3)$$

2.3 모델을 통해서 본 최단 경로와 충돌 예측

2.3.1 최단 경로(MCP)

MCP생성을 위한 기법으로 휴리스틱 탐색 기법인 A* 알고리즘을 선택하였다[7]. A* 알고리즘은 평가함수(evaluate function)를 사용하기 때문에 생성된 MCP가 항상 과소평가하도록 경로를 생성한다. A* 알고리즘의 최적성은 조건에 따라 다르지만, 보통 *허용성의 가장아래 최적의 해를 가지는 것으로 본다.

[*허용성: $\hat{h}(n)$ 은 실제 최단 거리 $h(n)$ 과 $\hat{h}(n) \leq h(n)$ 의 조건을 만족] A*알고리즘은 임의의 노드 n에 대해 (4)와 같은 평가함수를 갖는다.

$$f(n) = g(n) + \hat{h}(n) \quad f(n): \text{평가함수} \\ g(n): \text{시작노드에서 } n\text{노드까지의 최단거리} \\ \hat{h}(n): n\text{노드에서 목표노드까지의 추정치} \quad (4)$$

본 논문에서는 목표노드 상에서 작업을 요청한 작업소의 위치를 w 라 하고, 휴리스틱 함수 $\hat{h}(n)$ 을 $\hat{h}(n) = \sqrt{(w-n)^2}$ 로서 정의한다. 즉, 탐색 중인 현재 노드 n에서 평가함수 $f(n)$ 이 갖는 추정치이다.

이를 모델링 하면, $\hat{h}(n) = \sqrt{(W_{\beta 2} - TZ_{n2})^2}$ 이며, $W_{\beta 2}$ 는 작업소의 위치(=w)이다. 또한, $W_{\beta 2}$ 는 속해있는 경로의 양 끝이 아닌 지점에 위치하고 A* 알고리즘은 $W_{\beta 2}$ 가 속해있는 통제구역(TZ)을 찾는다. 따라서

