

다중 사이클 시스템을 위한 실시간 위치 기반 컨테이너 야드 트랙터 최단거리 계산

김한수[†], 박만곤^{††}

요 약

생산성은 컨테이너를 취급하는 작업소요시간의 최소화에 의해 극대화된다. 따라서 기존에는 전산시스템의 개선을 통해 생산성 향상을 추구해왔으나 이송장비의 운영 및 이송방식의 문제점으로 인해 한계가 있었다. 즉, 대표적인 이송장비인 Y/T(Yard Tractor)는 한 번의 주행에 양하 또는 적하 작업 중 하나만 가능하기 때문에 한 번의 주행에 양하와 적하를 순차적으로 동시에 수행할 수 있다면 작업소요시간이 단축될 수 있다. 따라서 본 논문에서는 RTLS(Real Time Location System) 기반의 Multi-Cycle System을 통하여 문제점을 해결하고자 하였다. 이를 위해 RTLS기술을 활용하여 Y/T의 실시간 위치 파악, GIS기술을 통한 Y/T실시간 위치표시 및 최단거리 알고리즘(다익스트라)을 터미널업무에 활용하여 생산성을 향상시킬 수 있는 시스템을 개발하였다. 시스템에 대한 시뮬레이션 결과, 평균치대비 절감 운전거리율은 9%가 되었고 최대치대비 절감 운전거리율은 19%가 되었음을 확인할 수 있어 결과적으로 컨테이너터미널 생산성이 극대화됨을 알 수 있었다.

Computation of the Shortest Distance of Container Yard Tractor for Multi-Cycle Systems

Han-Soo Kim[†], Man-Gon Park^{††}

ABSTRACT

A container terminal productivity is maximized by a minimized time for processing containers. So, we have been elevated the container terminal productivity through an improvement of computing system, but there are a limitation because of problems for transportation management and method. A Y/T(Yard Tractor), which is a representative transportation, is able to do only one process, loading or unloading, at one time. So if the Y/T can do loading and unloading step by step at a same time, the processing time would be shortened. In this paper, we proposed an effective operating process of Y/T(Yard Tractor) Multi-Cycle System by applying RTLS(Real Time Location System) to Y/T(Yard Tractor) in order to improve the process of loading and unloading at the container terminal. For this, we described Multi-Cycle System. This system consists of a real time location of Y/T based on RTLS, an indicating of Y/T location in real time with GIS technology, and an algorithm(Dijkstra's algorithm) of the shortest distance. And we used the system in container terminal process and could improve the container terminal productivity. As the result of simulation for the proposed system in this paper, we could verify that 9% of driving distance was reduced compared with the existing rate and 19% of driving distance was reduced compared with the maximum rate. Consequently, we could find out the container performance is maximized.

Key words: container terminal productivity(컨테이너터미널 생산성), RYMS(실시간 야드 트랙터 다중 사이클 시스템), computation of shortest distance(최단거리 계산), seeking of optimum path(최적경로 탐색), algorithm of critical path(주경로 알고리즘), Dijkstra's algorithm(Dijkstra 알고리즘)

※ 교신저자(Corresponding Author): 박만곤, 주소: 부산시 남구 대연3동 599-1(608-737), 전화: 051)629-6240, FAX: 051)628-6155, E-mail: mpark@pknu.ac.kr
접수일: 2009년 6월 23일, 수정일: 2009년 7월 29일

완료일: 2009년 12월 21일

[†] 부경대학교 정보공학과 박사과정
(E-mail: hskim3363@hanmail.net)

^{††} 종신회원, 부경대학교 전자컴퓨터정보통신공학부 교수

1. 서 론

동북아시아 교역규모 확대에 의한 컨테이너 물동량의 급증은 초대형 컨테이너선을 등장시켰으며, 이에 대응하는 선진항만의 발걸음도 빨라지고 있다. 독일 CTA(Container Terminal Altenwerder), 네덜란드 ECT(Europe Combined Terminal), 홍콩 HIT(Hongkong International Terminal) 등의 해외선진항만들은 이러한 급격한 변화에 대응하기 위하여 각 부문별 첨단 기술 개발과 도입을 통해 초대형선의 기항지인 메가 허브포트를 구축하려는 노력을 가속화하고 있다. 현재 초대형선이 기항하기 위해서는 기존의 항만하역시스템보다 한층 더 향상된 생산성이 요구되고 있으며 이를 위해 현재보다 발전된 저비용, 고효율 컨테이너터미널 운영시스템 및 관련기술 개발이 필수적인 것으로 인식되고 있다. 따라서 첨단 유비쿼터스 정보기술(u-IT; ubiquitous-Information Technology)등과 같은 기술을 적용시키기 위한 노력이 진행중이며, 특히 컨테이너터미널 하역작업의 생산성을 향상시키기 위해 RTLS(Real Time Location System)[1]를 기반으로 하는 기술이 개발되어 적용중이다.

컨테이너터미널에 RTLS기술을 적용시키면 양·적하 작업장비의 현재위치를 실시간으로 파악할 수 있어 작업을 위한 이동시간 등을 최소화할 수 있기 때문에 생산성 향상에 큰 역할을 하게 된다. 컨테이너터미널의 생산성은 취급하는 작업시간과 관련되어 있다는 점에서 작업시간을 최소화할 수 있는 기술과 운영방법이 중요하며, 작업소요 시간은 곧 비용이라는 점에서 운영 및 작업 방식의 개선과 자동화 등에 대한 연구가 필요하다[2,3].

따라서 본 논문에서는 컨테이너터미널의 양·적하 작업 개선에 요구되는 최적 경로의 탐색, 최단 이동거리에 대한 계산과 적용에 대해 연구하였으며, 이를 가능하게 하는 알고리즘으로 양·적하 다중 사이클 작업을 위한 알고리즘, Y/T(Yard Tractor) 작업 최단거리 및 풀링(Pooling) 알고리즘, 그리고 예상 이동거리 계산에 사용되는 알고리즘을 구현하고 시뮬레이션으로 검증하였다.

본 논문에서 주요하게 고려한 점은 이송장비인 Y/T의 실시간 위치 정확도이다. RTLS장비에서 계산되는 Y/T 위치값이 실제로는 차량의 움직임처럼

선형적으로 움직이지 않기 때문에, Y/T의 움직임이 빠르게 표현되지 않음으로써 표시되는 화면에도 직선화된 라인으로 표시되지 않는다는 것이다. 결과적으로는 화면에 매끄럽지 못하게 나타날 것이며, 사용자로 하여금 혼란이 생길 수도 있다는 점이다.

본 논문과 관련된 연구는 거의 없는 실정으로 유사연구로서 자동화 항만의 야드 운영시스템의 설계[2-4] 및 RTLS 기술적용과 설계[1-2,5-8], RTLS에서 지능형 조정을 사용한 위치 예측 알고리즘[9-10] 등이 있다. 또 다른 유사작업장에서의 관련연구를 살펴보면 조선소에서의 블록운반을 위한 복합 트랜스포터 일정 계획에 관한 연구[10-11]와 트랜스포터 블록 운반 시 시간을 최소화 하는 최단경로 탐색 연구 등이 있다[10].

본 연구의 결과물인 컨테이너터미널에서의 효율적인 Y/T작업 및 모니터링을 통해서 생산성 향상과 비용절감 효과를 기대할 수 있을 것이며, 향후 컨테이너터미널의 급격한 물동량 증가를 고려했을 때, 본 논문에서 제안하는 방법은 컨테이너터미널의 효율적 운영과 개선에 큰 역할을 할 것이다.

본 논문의 구성은 다음과 같다. 2장은 컨테이너터미널 업무프로세스와 최단경로 탐색을 위한 알고리즘에 대해 선행연구를 실시하였고 3장에서는 시스템 설계와 구현으로 야드 맵 정의, 이동경로 설정 및 위치보정을 포함하는 최단경로 탐색과 계산 알고리즘 등을 제시하였다. 그리고 4장에서는 Y/T 흐름을 제어할 수 있는 모니터링 및 그 결과를 평가할 수 있도록 시뮬레이션 결과를 나타내고자 한다. 끝으로 5장 결론에서는 제안된 본 논문이 가지는 기여점과 한계점을 제시하였다.

2. 컨테이너터미널 업무프로세스와 최단경로 탐색을 위한 알고리즘

2.1 컨테이너터미널 업무프로세스

컨테이너터미널 내 하역시스템은 선박, 본선하역장비(C/C; Container Crane), 컨테이너 야드 트랙터라고 불리는 내부차량(Y/T; Yard Tractor), 야드 하역장비(T/C; Transfer Crane, R/S; Reach Stacker), 야드, 게이트, 외부차량으로 구성 되어 있으며, 하역시스템별 구성 요소 및 인터페이스 특성은 그림 1과 같다. 화살표는 구성 요소별 인터페이스에 대한 요구

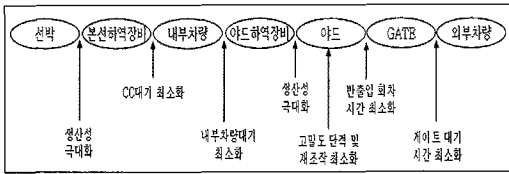


그림 1. 하역시스템 구성요소 및 인터페이스 특성

사항을 나타내고 있으며, 요구사항에 대한 최적의 계획, 실행, 모니터링, 자원배치 등을 제공 했을 때에는 터미널의 운영 효율성이 극대화 된다.

터미널 양·적하 작업 시 컨테이너의 이동절차를 살펴보면, 양하작업의 경우는 선박, C/C(본선하역장비), Y/T(내부차량), T/P(블록-배이), T/C(야드 하역장비), 야드 내 블록의 순으로 이루어지고 적하의 경우는 그 반대절차에 따라 수행된다.

Y/T는 양·적하 컨테이너 화물을 본선과 야드간에 운반하는데 이용되는 운송 장비로써 배정 및 운영 절차는 다음과 같다. 선박이 입항하게 되면 선박별 C/C 투입현황을 확인하고 C/C별 Y/T를 배정(C/C 당 Y/T 4대)하며, 배정된 Y/T는 해당 C/C로 이동한다. 특정 C/C별로 배정된 Y/T에 대하여 Under Clerk(U/C)은 배정된 Y/T를 확인하고 입력관리를 수행한다.

그림 2와 같이 공차구간을 없앤 개선된 양·적하 업무프로세스에서 Y/T 운영형태는 Y/T 풀링 시스템에 의해서 그 목적을 달성할 수 있으며, Y/T 풀링 시스템은 Y/T 배정 및 배차, 교대, 그리고 선박 및 작업지시 등을 Y/T 풀링 엔진에 의해서 수행되도록 한다. Y/T 풀링 시스템에서의 작업흐름은 첫째, 풀(POOL)등록 시 각 C/C별 Y/T를 유동적으로 배분, 각각의 C/C당 Y/T를 최소 1에서 최대 4까지 두고 Y/T의 합이 10이 되도록 한다. 둘째, 작업 중 C/C의

작업 상황을 모니터링 한 후 작업 부하에 따라 Y/T를 유동적으로 배분, 구현한다. 셋째, 풀 등록 후 풀가동, 중지, 일시 정지의 기능을 구현하고, 해당 풀 선택 후에는 풀에 배당할 모션을 선택한다. 그런 다음에는 작업계획 정보에서 배정되어진 C/C에 따라 적절한 Y/T를 배분한다. 넷째, 풀 등록 시, 풀 ID는 일정한 규칙에 의해서 기본 정보를 자동으로 생성하고 현재 풀 상태와 운영시간 및 중단시간을 표시한다. 다섯째, 풀 등록 시 작업 여건에 맞게 속성값들을 지정, 풀이 가동 중이더라도 작업 상황에 따라 C/C별, Y/T를 유동적으로 배분할 수 있게끔 하여 Y/T와 C/C의 작업균형을 맞출 수 있도록 한다.

이때 풀링을 위해서는 작업할당을 위한 시스템이 필요하며 RYMS(RTLS 기반 Y/T Multi Cycle System; 실시간 야드 트랙터 다중 사이클 시스템) 엔진이 그 기능을 수행하게 된다. RYMS는 컨테이너 터미널의 작업 능률을 향상 시킬 수 있는 중요 요인 중의 하나로서 Y/T장비의 작업 능률을 향상시킬 수 있다. 또한, Y/T 작업현황을 관리하고 있으면서 현재 작업진행 중에 있는 C/C, T/C에 가장 근접한 Y/T를 찾아내어 신속하게 Y/T차량을 C/C 및 T/C 작업장에 투입할 수 있도록 하는 시스템이다.

2.2 최단경로 탐색을 위한 알고리즘

최단경로 탐색에 있어서 사용될 수 있는 몇 가지 알고리즘으로는 다익스트라(Dijkstra) 알고리즘[12], 무어(Moore) 알고리즘[13], 쉬어(Shier) 알고리즘[14], Floyd 알고리즘[15-19] 그리고 벨만 포드(Bellman Ford) 알고리즘[20-24] 등이 있다. 다익스트라 알고리즘은 시작점과 연결된 정점 중 최소값을 가진 정점에 표시를 붙여 확정하며, 확정된 정점과 연결된 모든 정점의 거리를 구해서 저장하게 된다. 이를 통해 모든 정점에 표시가 붙어 확정될 때까지 경로 찾기를 반복하게 되며, 이 알고리즘의 대표적인 예로는 네트워크 라우팅의 최단경로 알고리즘이 있다. 벨만 포드 알고리즘은 다익스트라 알고리즘에서 가중치가 음수인 경로가 있을 때 최단경로를 올바르게 구할 수 없던 오류를 수정한 알고리즘으로 시작점과 연결된 정점 중 최소값을 가진 정점에 표시를 붙여 확정하는 알고리즘이다. Floyd 알고리즘은 모든 정점에서 출발해서 출발한 정점을 제외한 모든 정점을 도착점으로 하여 최단거리를 구하게 되

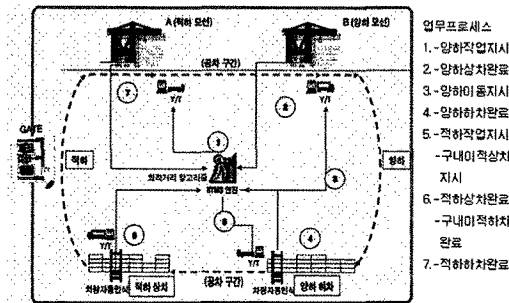
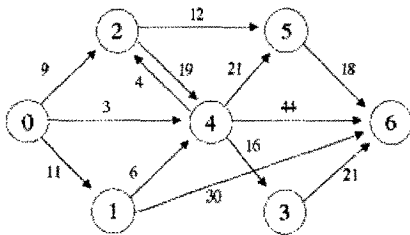


그림 2. 하역시스템에서의 업무프로세스

며 복잡도가 높은 경우 속도가 실행 속도가 빠른 장점이 있다.

그 중에서 다익스트라 알고리즘은 동적으로 계산되며, 속도가 빠르고, 하나의 정점에서 목적지까지의 계산방식이므로 현장중심의 업무협회가 이루어져 요구 반응속도가 빠르게 요구되며, 터미널의 크기 및 모양, 이송차량의 크기 및 회전반경, 외부차량과의 교차점 회피가 요구되는 터미널 운영환경 변수에 적합[11-12,15,25] 하여 본 연구에서 채택하였으며 이에 대해 살펴보면 다음과 같다.

그림 3(a)와 같은 방향그래프의 간선이 양수의 가중치를 가질 때 임의의 출발 정점에서 도착 정점까지의 경로 중 경로의 길이가 최소인 경로를 최단경로라고 정의한다. 이와 같은 그래프에는 정점(Vertex)과 간선(Edge)들로 구성되어 있으며, 간선들은 방향성과 무방향성으로 구분된다. 방향그래프는 최소 신장트리(MST; Minimum Spanning Tree), 최단경로(Shortest path)와 임계경로(CR; Critical Route)를 구하는데 활용된다[26-29]. 이러한 최단경로는 통신망, 도로망, 항공로 지도, 작업공정계산 등에 널리 응용된다[11,14,15,20,21,25]. 최단경로 탐색을 위한 단계별 알고리즘은 다음과 같으며, 그림 3(b)는 간선의 인접행렬을 나타낸다.



(a) 방향그래프

	0	1	2	3	4	5	6
0	0	11	9	∞	3	∞	∞
1	∞	0	∞	∞	6	∞	30
2	∞	∞	0	∞	19	12	∞
3	∞	∞	∞	0	∞	∞	21
4	∞	∞	4	16	0	21	44
5	∞	∞	∞	∞	∞	0	18
6	∞	∞	∞	∞	∞	∞	0

(b) 인접행렬과 가중치

그림 3. 최단경로 설정과 인접행렬

이러한 다익스트라 알고리즘의 수행절차는 다음과 같다.

[Step 1] 인접행렬 상태로 각 간선의 가중치가 존재하도록 한다. 정점 i에서 i까지의 가중치는 0이고 정점 i에서 j까지의 간선 E(i, j)가 존재치 않으면 가중치는 무한대(∞)가 된다.

[Step 2] 집합 S와 T를 정하는데 S는 출발 정점을 초기 값으로 하고 집합 T는 출발 정점을 제외한 모든 정점을 포함하도록 초기화한다.

[Step 3] 출발 정점을 제외한 모든 정점으로부터의 거리의 초기값(dist[i], 2<=i<=n)을 [Step 1]의 인접행렬에서 취한다.

[Step 4] 집합 T의 원소 중 출발점으로부터의 거리가 최소인 정점 v를 택하여 T에서 제거하고 집합 S에 추가한다.

[Step 5] T집합 내의 모든 정점 w에 대해 출발점으로부터의 거리(dist[w])와 간선 E(v, w)의 길이에 v정점의 거리(dist[v])를 합한 값 중 적은 것을 선택하여 정점 w의 거리(dist[w])로 한다.

[Step 6] Step 4와 Step 5의 과정을 T가 공집합이 될 때까지 반복한다.

이와 같이 컨테이너터미널 환경에 적합한 다익스트라 알고리즘을 사용하여 시스템 설계와 구현을 하고자 한다.

3. 시스템 설계와 구현

본 논문에서는 데이터베이스에 구축된 야드 맵(Yard Map) 상에서 Y/T 작업 최단거리 자동 탐색 및 예상 이동거리를 계산한다. 이를 위해서는 터미널 환경에 맞도록 알고리즘이 적용되어야 하며, Y/T 효율적 이동선 재현 그리고 최단거리 재설정 검증을 위한 시뮬레이션 기능을 포함한다. 여기서 야드 맵에 입력된 야드 맵에서의 단위 접점을 나타내는 JP(Junction Point)의 좌표위치값 자동 조회와 터미널 환경에 따른 최단거리 설정을 위해서 다익스트라 알고리즘을 적용하였다.

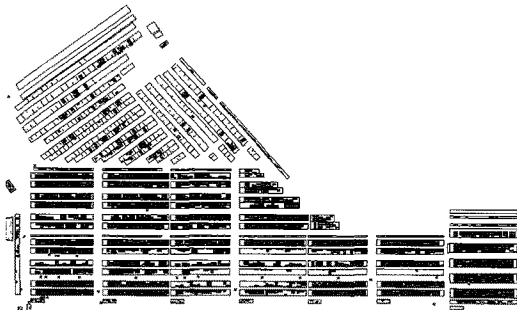
시스템 운영환경은 터미널 내 Y/T의 실시간 모니터링, 양·적하 작업의 모니터링, Y/T의 최단거리 적용과 양·적하 다중 사이클 작업을 통한 항만생산성의 극대화를 위해서 최적의 운영환경이 되도록 구

현하였으며, RYMS 및 하역장치장 모니터링 시스템 개발을 위해 사용된 PC사양은 펜티엄의 윈도우즈 계열과 Framework 2.0이며, VB.NET과 Windows XP가 개발환경으로 사용되었다.

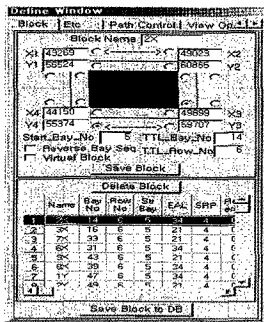
3.1 아드 맵 정의

데이터베이스에 구축된 야드는 먼저, 벡터(Vector)화가 이루어져야 하는데, 이를 위해 야드에 있는 블록, 건물 등에 대한 좌표값을 터미널운영사로부터 획득하였다. 그런 다음 그림 4(b)와 같이 블록을 각각 정의한다. 각각의 블록에 대하여 정의하게 되면 야드 맵상에 블록의 모양이 하나씩 나타나고 전체 블록을 정의하면 전체 야드가 그림 4(a)처럼 나타난다.

이와 같은 정의 절차를 살펴보면 다음과 같다.



(a) 야드 내 블록좌표에 의해서 나타난모양

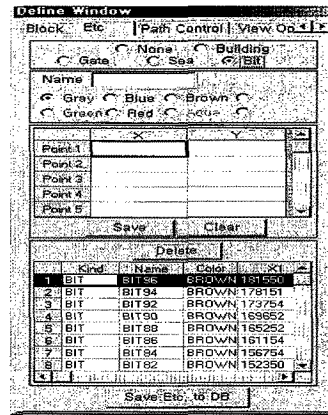


(b) 야드 맵 표현을 위한 블록정의

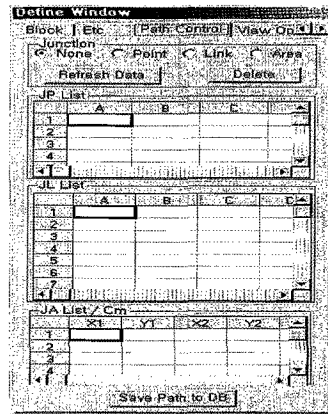
그림 4. 아드 맵 정의

먼저, 하나의 블록을 정의하기 위해서는 4개의 꼭지점 값(X1,Y1; X2,Y2; X3,Y3; X4,Y4)을 그림 4(b)와 같이 입력한 후, 표 1과 같은 나머지 항목들을 기입한다.

둘째, 그림 5(a)와 같이 건물과 기타 모양을 각각 정의하되 Building/Gate/Sea/Bit로 정의하고 각 옵션을 선택한다. 다음은 선택한 옵션에 대해서 이름을 입력하고 각각의 구조물에 대한 색상을 정의한 후, 저장을 하면 야드 맵에 조형물의 모양이 나타난다.



(a) 야드 맵에서의 조형물 출력을 위한 요소



(b) 야드의 경로 제어를 위한 JP/JL/JA

그림 5. 아드 맵에서의 조형물과 경로 제어

표 1. 블록의 기본정의와 추가 항목

항 목	내 용	항 목	내 용
Start_Bay_No	시작되는 Bay의 번호 입력	Reverse_Bay_Seq	Bay번호의 순서를 정의
TTL_Bay_No	총 Bay의 수량을 입력	Virtual Block	가상블록 인지에 대한 정의

셋째, 그림 5(b)와 같이 야드 맵에서의 단위 접점을 나타내는 JP(Junction Point), 점선인 두 지점 간의 거리를 계산해서 나타내는 JL(Junction Line), 현재 차량의 위치에 가장 가까이 있는 JP할당에 필요한 JA(Junction Area)에 대해서 정의하되 Point(JP), Link(JL), Area(JA)로 구분하고 선택한다. JP인 경우 야드 맵의 예상 위치에 마우스를 이용하여 이동한 후, 클릭하면 JP가 입력된다. JL의 경우 화면에 입력된 두개의 JP를 순차적으로 선택하면 JL이 입력되고 JA의 경우 야드 맵 화면에 나타나 있는 JP들을 마우스로 드래그하면 박스가 나타나게 되는데, 그 박스 안에 있는 JP들은 하나의 JA에 묶이게 된다.

야드 맵 전체는 백터로 구성되었으며, 야드맵 상에서 마우스를 활용(확대/축소 가능)하여 경로를 지정하게 되며, 경로 지정 시 포인트와 포인트 간의 속성 즉 방향성과 거리에 대한 정보도 함께 입력하게 된다. 야드 맵에 실시간으로 표현되는 정보는 정확히 표현하면 2초 이내이며, 터미널 내 출발지에서 목적지까지의 최단경로에 대한 정보가 운영 전에 미리 계산되어 저장되어있는 정보를 활용한다. 또한 터미널에서 Y/T차량의 운영상 차로의 변경은 거의 발생하지 않는다. 또한 야드 맵 각 블록 내의 컨테이너 적재정보는 1분 30초에서 30분까지 사용자가 지정할 수 있도록 되어 있으며, 최소 1분 30초 내의 시간은 하나의 컨테이너에 대한 작업은 최소 4분 이상 소요된다는 운영통계를 근간으로 하기 때문에 전체 야드를 통해 볼 때 실시간으로 봐도 무관하다.

3.2 야드 맵에서의 이동경로 설정과 위치정의

야드 맵에서의 이동경로 설정과 위치 정의를 위해서는 야드 맵에서의 단위 접점을 나타내는 JP와 현재 차량의 위치에 가장 가까이 있는 JP할당에 필요한 JA 그리고 JA에 소속된 JP임을 나타내는 JC(Junction Connect), 작업지점을 나타내는 WP(Work Point), 점선인 두 지점 간의 거리를 계산해서 나타내는 JL, Y/T가 임의의 출발점 JP에서 도착점 JP까지의 복수 경로 중 JL의 누적값이 가장 작은 값으로 표현될 수 있는 HOOP 등, 합 연산을 수행한다. 또한, 야드 맵상에서의 이동경로 설정과 위치 정의하는데 요구되는 좌표값과 이동거리를 위해서 JP, JC, WP, JL, JA 관리가 필요하며, 블록 좌표정보 및 야드 블록정보 그리고 최단거리상세 정보에 대한 데이터

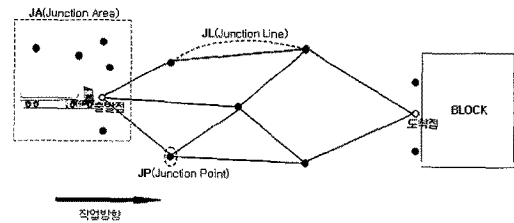


그림 6. 점점경로의 구성요소

베이스 설계가 이루어진다.

3.2.1 점점 경로의 구성

JP할당에 필요한 점점영역은 그림 6과 같이 나타낼 수 있으며, 터미널 환경에 따른 최단거리 알고리즘이 적용되도록 하였다.

특히, 그림 6에서처럼 점선, 두 지점간의 거리를 정의하기 위한 JL은 JP와 다음 JP의 위치를 알 수 있도록 이동거리를 계산하였으며, 이는 JP에서 다음 JP까지의 속성을 의미하는 것으로 점점의 차를 나타낸다. 여기서 작업지점 WP는 JP라는 개념의 연장선에서 표현되며 인접 블록과 연결되어 있는 것으로 하며, 터미널내의 구조는 변하지 않는 구조이므로 실시간 상황에서 점점경로를 구성하는 것을 보장할 수 있다.

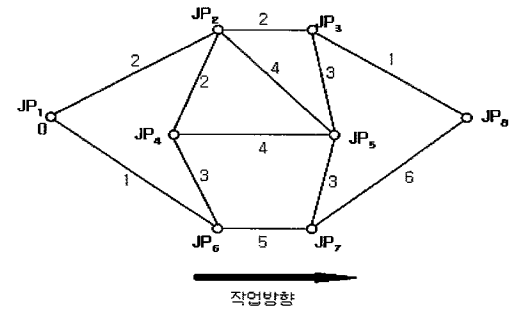
3.2.2 다익스트라 알고리즘에 기초한 HOOP 계산 과정

시작 JP 정점에서 가장 인접한 JP 정점까지의 거리가 최단거리가 되며, 여기서 최단거리가 알려진 JP들의 집합을 식 (1)과 같이 S_p 라고 가정한다. 그런 다음 집합 S_p 에 포함되지 않은 JP들 중에서 시작 JP로 부터 가장 가까운 JP를 찾게 되는데, 이와 같이 탐색된 새로운 Jp_i^s 은 집합 S_p 에 바로 이웃한 JP들 중 하나가 된다. 또한, 목적 JP에서 새로운 Jp_i^s 이 발견되지 않을 때까지, 즉 모든 JP가 집합 S_p 에 포함될 때까지 동일한 과정을 반복한다(식 (2)).

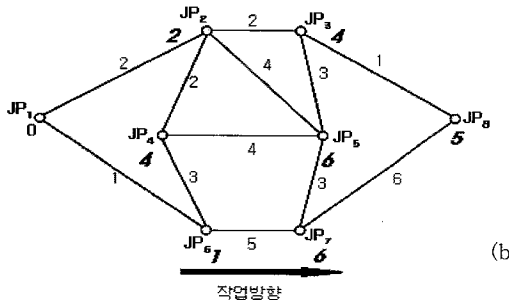
$$If, S_p = \{JP_1, JP_2, \dots, JP_n\}; \text{ where, } JP \rightarrow \text{shortest distance} \quad (1)$$

$$\begin{aligned} &do \\ & \text{if, } \{JP_1, JP_2, \dots, JP_n\} \notin S_p \\ & \quad JP_i^s \leftarrow \text{Result of seek to nearest JP from start JP;} \\ & \quad \text{where, } i = 1, \dots, n \\ & \quad \text{new_}JP_i^s = JP_i^s; JP^s = \text{new_}JP_i^s \\ & \text{while } (JP \in S_p) \end{aligned} \quad (2)$$

여기서, Jp_i^s 는 집합 S_p 에 포함되지 않은 JP들 중



(a) 최단거리 계산이전의 초기화



(b)

최단거리 계산과정

그림 7. HOOP 산출을 위한 수행과정

에서 또 다른 가장 가까운 JP의 발견과정에서 얻어진 결과이다.

그림 7의 작업방향에 따라 최단거리를 계산할 수 있도록 식 (3)과 같이 적용하되, 각 단계별 최단거리 탐색을 위해서 수행되는 과정은 다음과 같다.

$$JL(JP_i) = \min \{ JI_i^w, JL(JP_i^s) + JP_i^d \}; i = 1 : n \quad (3)$$

여기서,

JL_i^w : 탐색된 점정 JP_i 셋에서 인접 JP까지 계산되어 최단거리가 알려진 S_p 중 하나인 JP에 대해서 가장치 w 의 누적값 중 작은값을 가지는 결과 값이다.

$JL(JP_i^s)$: 하나의 점정 JP_i 에서 인접 JP까지의 거리가 계산되고 탐색된 JP_i 셋(s)을 나타낸다.

JP_i^d : 하나의 점정 JP_i 셋에서 인접 JP까지의 거리 d 를 나타내는 값이다.

[Step 1] 먼저, 알고리즘 수행 전 각 변수 값들을 초기화 한다. 각 JP까지의 거리(가중치)를 최대값으로 지정한 후, 출발 JP_1 의 거리를 0으로 지정하고 JP 셋을 검색된 셋과 검색되지 않은 셋으로 구분한

다. 점점 JP_1 에 인접한 JP_2, JP_6 에 대한 거리를 계산한다.

$$JL(JP_2) = \min \{ \infty, 0+2 \} = 2$$

$$JL(JP_6) = \min \{ \infty, 0+1 \} = 1$$

[Step 2] JP_1 의 인접JP까지의 거리가 계산되었으므로 탐색된 JP_2 셋으로 이동하고 탐색되지 않은 JP는 셋에서 제거하는 것으로 한다. 계속해서 거리가 계산된 JP_2 인접 JP 중에서 JP_3, JP_4, JP_5 까지의 거리를 탐색한다.

$$JL(JP_3) = \min \{ \infty, JL(JP_2) + 2 \} = 4$$

$$JL(JP_4) = \min \{ \infty, JL(JP_2) + 2 \} = 4$$

$$JL(JP_5) = \min \{ \infty, JL(JP_2) + 4 \} = 6$$

[Step 3] 거리가 계산된 JP 셋 중, 점정 JP_6 의 인접 점정에 대해서 JP_7 까지의 거리를 계산한다. 계속해서 JP_3, JP_4, JP_5, JP_7 의 인접 JP에 대해 거리(JL)를 탐색한 후, 거리가 탐색된 JP 셋 중 JP_3 의 인접 점정에 대해 JP_5, JP_8 까지의 거리를 계산한다.

$$JL(JP_7) = \min \{ \infty, JL(JP_6) + 5 \} = 6$$

$$JL(JP_5) = \min \{ 6, JL(JP_3) + 2 \} = 6$$

$$JL(JP_8) = \min \{ \infty, JL(JP_3) + 1 \} = 5$$

계속해서 거리가 탐색된 JP 셋 중 JP_7 의 인접 점정에 대해 JP_5, JP_8 까지의 거리를 계산한다.

$$JL(JP_5) = \min \{ 6, JL(JP_7) + 3 \} = 6$$

$$JL(JP_8) = \min \{ 5, JL(JP_7) + 6 \} = 5$$

결과적으로는 JP_1 에서 JP_8 까지의 최단거리는 “5”가 되고 최단경로는 JP_1 - JP_2 - JP_3 - JP_8 순으로 수행되었음을 알 수 있다.

Y/T 경로배정을 위해서는 그림 8(a)와 그림 9에서처럼 알고리즘의 수행과 동시에 유희 Y/T를 찾아내어 목록을 작성하고 Y/T별 실제좌표값을 탐색한다. 이와 같이 탐색되고 계산된 최단경로는 야드 맵에서 적용이 가능하여 최종 목적지로 이동할 수 있는 Y/T를 선택하고 확정한다. Y/T 풀링 시스템은 그림 8(b)에서 보인 것처럼 Y/T 배정 및 배차, 교대, 그리고 선택, 작업지시 등을 Y/T 풀링 엔진에 의해서 수행될 수 있도록 구현된다. 최단 경로의 탐색과정은 그림 9와 같이 이루어지게 되며 A는 출발점으로 거리값은 0이며, 목적지 H까지의 최단경로를 찾기 위한


```

If Trim(rs("r_rtls_X").Value.ToString) <> " " And Trim(rs("r_rtls_Y").Value.ToString)
<> "" Then
    For x = 1 To UBound(g_Ary_CP)
        dist_Btn_rtls =
            f_Distance_between_Two_Points(Double.Parse(Trim(rs("r_rtls_X").Value.ToString)),
            Double.Parse(Trim(rs("r_rtls_Y").Value.ToString)),
            g_Ary_CP(x).g_CP_X, g_Ary_CP(x).g_CP_Y)
        If dist_rtls > dist_Btn_rtls Then
            dist_rtls = dist_Btn_rtls ' //계산된 두점의 거리에 대해서 가장 근접된 것인지 비교함
        End If
    Next
End If

' // 두 점간의 거리를 구하는 함수 //
sub Function f_Distance_between_Two_Points
    Temp_X = x2 - x1 ' // x1 - x2 를 구함
    Temp_Y = y2 - y1 ' // y1 - y2 를 구함
    Temp_X = Temp_X * Temp_X ' // (x1 - x2)^2
    Temp_Y = Temp_Y * Temp_Y ' // (y1 - y2)^2
    Result = Sqrt(Temp_X + Temp_Y) ' // sqrt((x1 - x2)^2 + (y1 - y2)^2)
    Result = Round(Result)
end sub
    
```

그림 11. CP계산 알고리즘과 두 점간의 거리를 구하는 함수

을 “r_rtls_X”, “r_rtls_Y”로 나타낸다. 또한, g_Ary_CP에 있는 점들의 좌표를 현재 비교대상인 YT의 좌표와 거리를 비교해서 가장 가까운 CP의 값을 YT의 위치로 인정하게 된다.

4. 시뮬레이션 및 실험결과

RTLS 기반의 위치추적 기술을 이용해서 컨테이너의 위치를 파악함과 동시에 생성된 정점(JP; Junction Point)정보를 바탕으로 최단거리알고리즘 연산을 수행하고 야드 맵 상의 블록표시와 등록된 JP표시를 통해서 시뮬레이션 기능을 제공할 수 있다. 여기서 Y/T에 대한 최단, 최장 운행거리 조회가 가능하다. 실시간 위치 추적기술은 기존 RTLS기반의 거리측정방식으로써 TDOA(Time Difference of Arrival) 기법을 적용한다. 이와 같은 기법은 삼각측량법에 의존하는 것으로써 동일 직선상에 있지 않은 3지점으로부터의 거리를 계산하여 위치를 측정하는 방법이다. 결과적으로는 Y/T실시간 위치정보와 야드 컨테이너 정보 등을 활용해서 컨테이너터미널 내 하역장치의 상태를 모니터링 할 수 있게 된다. 그림 12에서와 같이 이미 데이터베이스에 저장되

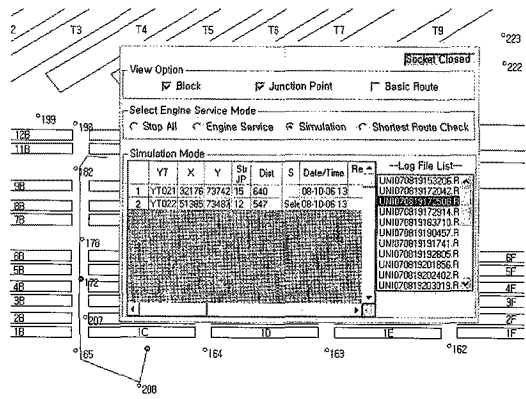


그림 12. 최단거리 탐색 및 시뮬레이션

고 구축된 야드 맵을 출력하고, 출력된 좌표 위치를 탐색할 수 있어서, 탐색된 야드 맵에서는 JP간 최단거리 알고리즘을 적용할 수 있도록 한다.

그림 12는 생성된 정점(Junction point) 정보를 바탕으로 최단거리 알고리즘 연산을 수행한 후, 시뮬레이션을 수행한 결과이다. 시뮬레이션 수행을 위하여 적용된 요소로 먼저 몇 대의 Y/T를 비교해 볼 것인가로 정의되는 비교대상 희망위치수, 둘째로 각각의 희망 위치수별 X, Y좌표, 셋째, 목적지에 대한 X, Y

좌표가 있다. 시뮬레이션에 적용된 시간은 2009년 2월 11일 오후 7시 20분부터 오후 9시 15분까지 약 2시간 동안이며, A터미널 Y/T 17대, C/C 2대가 투입되었다. 시뮬레이션에서는 Y/T 위치 이력조화가 가능하고 Y/T 최단운행과 최장 운행거리, 최적 Y/T 투입수를 확인할 수 있게 된다. 이를 통해서 최단거리가 탐색되고 경로가 확정됨으로써 하역장치장에서의 작업 흐름을 원활히 진행될 수 있음을 알 수 있다. 본 논문에서 제안된 최단경로 탐색과 알고리즘의 효율성을 확인하기 위해서 시뮬레이션을 수행하였다. 표 2에서 선택된 Y/T는 실제 운행한 차량에 대해 컨테이너터미널 내부에서 사용하는 번호를 기록하였으며, 수행된 시뮬레이션에 따라서 Y/T 작업한 결과, RYMS 적용 시의 최단총거리는 51,941m,

최장총거리는 61,604m로 나타났으며, 절감운전거리 비율은 최단거리와 평균치대비 9.3%가 되었고 최대치대비 18.6%가 되었음을 그림 13과 표 2을 통해서 확인할 수 있다.

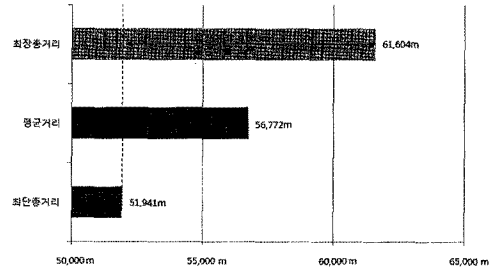


그림 13. 최장/평균/최단 총거리에 대한 절감운전 거리 및 비율 분석

표 2. 최단·최장 총거리에 대한 절감운전비율 분석

순번	선택된 Y/T	최단		YT1		YT2		YT3		YT4		최장	
		No	Distance (cm)	No	Distance (cm)	No	Distance (cm)	No	Distance (cm)	No	Distance (cm)	No	Distance (cm)
1	340	340	48569	340	48569							340	48569
2	302	302	26548	302	26548	340	48569					340	48569
3	314	314	12136	314	12136							314	12136
4	340	340	18317	340	18317	314	82769					314	82769
.....
21	340	340	18317	340	18317	331	98866				331	98866
22	314	314	17219	314	17219	340	18317	331	98866			331	98866
23	331	331	46782	331	46782							331	46782
24	311	311	81169	311	81169							311	81169
.....
65	317	317	82151	317	82151							317	82151
66	314	314	4306	314	4306	308	5634	317	82151			317	82151
67	308	308	5634	308	5634	317	82151					317	82151
68	340	340	11785	340	11785	317	82151					317	82151
.....
80	382	382	47623	382	47623	321	67233	317	82151			317	82151
81	321	321	92688	321	92688	317	98866	305	31509	317	82151	317	82151
82	320	320	14237	320	14237	305	45515	321	78279	317	98866	317	98866
83	325	325	5051	325	5051	321	78279	317	98866			317	98866
84	305	305	28321	305	28321	317	82151	321	48211			321	48211
85	321	317	98866	321	64926	317	98866					317	98866
최단총거리(A)				51,941 (m)				최장총거리(B)				61,604 (m)	
C = B - A								9,663 (m)					
평균거리: D=(B-A)/2+A								56,772 (m)					
절감운전거리율(평균치대비) (D-A) / A				9.3 (%)				절감운전거리율(최대치대비) (B-A) / A				118.6 (%)	

표 3. Y/T 차량 투입 절감에 따른 탄소 발생 분석

구분	탄소발생 절감거리		탄소발생절감율	내 용
Y/T	최단거리	41,570 (m)	46.9 (%)	Y/T 투입이 7에서 4.1 대로 2.9대 절감의 효과
	최장거리	53,237 (m)	50.6 (%)	

또한, 간접적인 효과로서 최단거리 운행시 발생될 수 있는 CO₂ 발생절감율을 표3과 같이 계산하였으며, 이에 에너지 소비량, 발열량, 탄소배출계수 등이 사용되며, 컨테이너 하역장치장에서의 Y/T 투입 증가는 탄소 발생의 증가로 나타나는 만큼 Y/T 투입량 감소 및 탄소 절감 정도를 고려해서 정성적 분석을 수행하고 최단거리 및 최장거리 그리고 평균거리 대비 절감거리와 절감율을 살펴보는 데 정량적 분석을 수행한다. Y/T 차량 투입 절감에 따른 탄소 발생을 분석하면 최단거리에서 41,570(m)와 최장거리에서 53,237(m)로 46.9(%)와 50.6(%)의 탄소 발생 절감 효과가 있음을 표 3를 통해서 알 수 있고 Y/T 투입(7~4.1대)에 따른 탄소 발생 절감거리 및 절감효과도 2.9대임을 알 수 있다.

5. 결 론

본 논문에서는 컨테이너터미널의 생산성을 극대화할 수 있도록 기존의 전산시스템만 개선하는 방법이 아닌 이송장비의 운영 및 이송방식을 개선하는데 초점을 두었다. 즉, 대표적인 이송장비인 Y/T(Yard Tractor)는 한 번의 주행에 양하 또는 적하 작업 중 하나만 가능하기 때문에 한 번의 주행에 양하와 적하를 순차적으로 동시에 수행할 수 있다면 작업소요시간이 단축될 수 있으므로 이를 통해 생산성을 향상시키고자 RTLS(Real Time Location System) 기반의 Multi-Cycle System을 통하여 문제점을 해결하고자 하였다. 이를 위해 RTLS기술을 활용하여 Y/T의 실시간 위치 파악, GIS기술을 통한 Y/T실시간 위치 표시 및 최단거리 알고리즘(다익스트라)을 터미널업무에 활용하여 생산성을 향상시킬 수 있는 시스템을 개발하였다.

알고리즘으로는 CP 계산 알고리즘, 최단거리 및 예상거리 계산을 수행할 수 있는 알고리즘이 포함되었으며, RYMS 엔진이 사용되었다. Y/T 작업 시, 최단경로의 탐색과 이동거리를 계산함으로써 장치장에서 양·적하 작업을 원활히 할 수 있음과 동시에

RTLS 기술 적용을 통해서 컨테이너 하역장비의 위치를 실시간으로 식별한다. 또한, 장비 간 유기적인 연계와 작업지시는 컨테이너터미널에서의 효율적인 Y/T작업 및 모니터링으로 생산성 향상과 비용절감을 할 수 있게 된다.

본 논문에서 제안된 방식의 실험결과를 살펴보면, 평균치대비 절감 운전거리율은 9%가 되었고 최대치 대비 절감 운전거리율은 19%가 되었음을 확인할 수 있었다. 결과적으로는 Y/T 작업에 대한 최단경로 탐색과 이동거리의 계산으로 컨테이너터미널에서의 생산성을 극대화 할 수 있음을 알 수 있다. 향후, 제안된 방식이 전국 컨테이너 항만에 도입될 경우 선박접안시간을 단축하는 등 수출입 물류비용을 크게 절감시킬 것으로 예상할 수 있다.

본 연구에서는 최단거리계산과 예측을 위한 알고리즘을 적용하였지만, 비교분석하는데 있어서 다른 시스템과의 비교가 없어서 한계점이 있다. 따라서 향후 연구에서는 최단거리 적용 시, 혼잡도 계산과 예외 상황을 인식하는 부분에 대한 연구가 포함되어야 할 것이다.

참 고 문 헌

- [1] 권순량, 정광주, 박상훈, 김정훈, “유비쿼터스 항만 운영 효율화를 위한 RTLS 기술 적용,” 한국정보과학회 논문지, Vol.13, No.06, pp. 371-0377, 2007.
- [2] 홍동의, 정태충, “자동화항만의 야드 운영시스템 레이아웃 설계,” 한국정보처리학회 논문지, Vol.10-D, No.01, pp. 0101-0108, 2003.
- [3] 박준표, 이주표, 황대훈, “컨테이너 영상 전처리 및 식별자 인식 시스템의 설계,” 한국 멀티미디어학회 추계학술발표논문집, Vol.5, No.2, pp. 786-791, 2002.
- [4] Joeng Hoon Kim, Hyun Woo Lee, and Soon Ryang Kwon, “A Study on 3D RTLS at Port Container Yards Using the Extended Kalman

- Filter," *International Journal of Fuzzy Logic and Intelligent Systems*, Vol.7, No.4, pp. 228-235, 2007.
- [5] 김수희, 정인환, "Wireless LAN 환경 하에서 Access Point의 RSSI 삼각측량 방식을 이용한 RTLS 설계," *한국정보과학회 가을 학술발표논문집*, Vol.33, No.2(D), pp. 330-333, 2006.
- [6] 정동호, 김정효, 지동환, 백윤주, "능동형 RFID를 이용한 RTLS의 설계 및 구현," *한국통신학회 논문지*, Vol.31, No.12A, pp. 1238-1245, 2006.
- [7] Seok-Yong Lee and Chul-Ho Lee, "A Study on the RTLS based Dynamic Planning of Yard Tractors in Container Terminals," *International Journal of Navigation and Port Research*, Vol.31, No.2, pp. 141-149, 2007.
- [8] Doo-Jin Park and Ki-Chan Nam, "RFID-Based RTLS for Improvement of Loading Productivity in Container Terminals," *International Journal of Navigation and Port Research*, Vol.30, No.4, pp. 285-290, 2006.
- [9] 정승희, 이현재, 오창현, "RTLS에서 지능형 조정을 사용한 위치 예측 알고리즘의 성능 분석," *보안공학연구논문지*, Vol.3, No.2, pp. 15-26, 2006.
- [10] 송우석, 이재빈, 장안진, 유기윤, 김용일, "RFID를 이용한 이동체의 속도계측 기술에 관한 연구," *한국측량학회지*, 제25권 제3호, pp. 215-221, 2007.
- [11] C. M. Joo, W. S. Lee and K. B. Lee, "Transporter Scheduling for Block Transportation in the Shipyard," *Annual Spring Meeting of Korean Operations Research and Management Science Society/Korean Institute of Industrial Engineers*, pp. 348-352, 2005.
- [12] E. W. Dijkstra, "A Note on Two Problems in Connection with Graphs," *Numerisch Mathematik*, Vol.1, pp. 269-271, 1959.
- [13] U. Pape, "Implementation and Efficiency of Moore Algorithms for the Shortest Root Problem," *Mathematical Programming*, Vol.7, pp. 212-222, 1974.
- [14] D. R. Shier and Christoph W., "Arc Tolerances in Shortest Path and Network Flow Problems," *Networks*, Vol.18, pp. 159-171, 1989.
- [15] 이상운, "방향그래프의 접대점 최단경로 탐색 알고리즘," *한국퍼지 및 지능시스템학회논문지*, Vol.17, No.7, pp. 893-900, 2007.
- [16] 이창용, 김현성, 정연기, "에너지 효율성을 고려한 데이터 기반 라우팅 프로토콜," *한국정보과학회 영남지부 학술발표회 논문지*, pp. 7-11, 2006.
- [17] 박노성, 김대영, "에너지 효율적인 데이터 기반 센서 네트워크 라우팅 프로토콜," *정보과학회 논문지*, 정보통신 제32권 제2호, pp. 261-277, 2005.
- [18] 오진석, 곽준호, "독립형 태양광 발전 시스템을 위한 저가의 다중 무선 모니터링 기법에 관한 연구," *한국마린엔지니어링학회지*, 제29권 제5호, pp. 579-586, 2005.
- [19] 박성우, 김영진, "컴퓨터 네트워크에서 루핑 효과 감소에 관한 연구," *전북대학교 논문집*, 제37집 자연과학편, pp. 169-176, 1994.
- [20] 김성수, 차영민, "첨단교통정보시스템의 최적 경로 알고리즘 개발," *산업기술연구(강원대학교 산업기술연구소 논문집)*, 제21권 A호, 2001.
- [21] 장현봉, 박성철, "경로탐색 과정을 고려한 일방통행체계 효과 측정," *건축.도시연구센터 제5집*, pp. 137-150, 1997.
- [22] 김성호, "CCA 투영기법을 사용한 모션 데이터의 대화식 얼굴 표정 애니메이션," *한국인터넷 정보학회 제6권 제1호*, pp. 85-93, 2006.
- [23] 김성호, "모션 데이터에 Isomap을 사용한 3차원 아바타의 실시간 표정 제어," *한국컨텐츠학회 논문지*, Vol.7, No.3, pp. 8-16, 2007.
- [24] 김학신, 오진석, "반도체 공정에 이용되는 지능형 천장 반송 시스템의 최적 중앙제어," *한국박용기관학회지*, 제28권, 제7호, pp. 1159-1164, 2004.
- [25] 임선빈, 노명일, 차주환, 이규열, "트랜스포터의 공주행(空走行) 최소화를 고려한 블록 운반 계획 최적화," *대한조선학회논문집*, Vol.45, No.6, pp. 646-655, 2008.
- [26] 박만곤, 박순형, "동적 시스템 중속 그래프를 사

용한 동적 슬라이싱,” 한국멀티미디어학회 논문지, Vol.5, No.3, pp. 331-341, 2002.

- [27] 박순형; 박만곤, “개선된 동적 객체지향 종속 그래프,” 한국멀티미디어학회 춘계학술발표대회 논문집, pp. 806-809, 2004.
- [28] 정승희, 강철규, 오창헌, 임춘식, “RTLS에서 TDOA 기법을 이용한 위치추정 알고리즘,” 한국해양정보통신학회 추계종합학술대회 제9권 제2호, pp. 459-462, 2005.
- [29] 정주영, 전철민, 최윤수, 조성길, “최적경로 알고리즘을 활용한 경제적인 통신망 설계 적용 사례,” 한국공간정보시스템학회 논문지, 제8권 제1호, pp. 107-118, 2006.



김 한 수

방송통신대 전자계산학과 학사
 부산수산대 전산정보학과 석사
 2002년~현재 부경대학교 정보공학과 박사과정
 1977년~1984 총무처 정부전자계산소 근무
 1984년~2006 해양수산부 국립수산물학원 근무

2006년~현재 국토해양부 부산지방해양항만청 근무
 관심분야 : 컨테이너 터미널의 RTLS/USN 적용성 분석 및 U-IT기술을 활용한 항만인적 보안시스템 개발, RFID기반의 항만물류정보시스템 개발 등



박 만 곤

경북대학교 전산통계학
 (이학사, 교육학석사, 이학박사)
 Philippine Women's University
 (국제행정학석사)
 University of Rizal System,
 Philippines (명예 기술학박사)

Dept. of Electrical & Computer Engineering, University of Kansas (Post Doc.)
 1981년~현재 부경대학교 전자컴퓨터정보통신공학부 교수
 1997년~현재 한국멀티미디어학회(KMMS) 부회장, 회장 및 명예회장
 2002년~2007년 정부간 국제기구 CPSC (콜롬보폴랜기술교육대학) 총재 (Director General and CEO)
 2004년~2007년 Asia Pacific Accreditation and Certification Commission 아태지역 인증 및 검증위원회 위원장
 1990년~1991년 영국 University of Liverpool 전자계산학과 객원교수
 1996년~1997년 호주 University of South Australia 컴퓨터정보과학부 객원교수
 2005년~현재 유네스코 (UNESCO-UNEVOC) 자문위원, 아시아개발은행(ADB) 자문관
 관심분야 : 소프트웨어신뢰성공학, 비즈니스 프로세스 재공학 (BPR), 소프트웨어 공학 및 재공학, 멀티미디어정보처리기술, 정보시스템성능평가, ICT기반 HRD System