네트워크 기반 자율이동로봇을 위한 장애물 회피 알고리즘 개발

論 文
54D-5-2

Development of an Obstacle Avoidance Algorithm for a Network-based Autonomous Mobile Robot

孫 粹 耿† · 金 弘 烈* · 金 大 元** · 金 弘 錫***
(SooKyung Sohn · Hongryeol Kim · Dae Won Kim · Hong-Seok Kim)

Abstract - An obstacle avoidance algorithm for a network-based autonomous mobile robot is proposed in this paper. The obstacle avoidance algorithm is based on the VFH(Vector Field Histogram) algorithm and two delay compensation methods with the VFH algorithm are proposed for a network-based robot with distributed environmental sensors, mobile actuators, and the VFH controller. Firstly, the environmental sensor information is compensated by prospection with acquired environmental sensor information, measured network delays, and the kinematic model of the robot. The compensated environmental sensor information is used for building polar histogram with the VFH algorithm. Secondly, a sensor fusion algorithm for localization of the robot is proposed to compensate the delay of odometry sensor information and the delay of environmental sensor information. Through some simulation tests, the performance enhancement of the proposed algorithm in the viewpoint of efficient path generation and accurate goal positioning is shown here.

Key Words : VFH, Network Delay, Sensor Fusion, Kalman Filter.

1. 서 론

네트워크 기반의 시스템은 다양한 이기종 시스템의 결합을 보장할 뿐만 아니라 이에 따르는 비용을 최소화할 수 있기 때문에 최근 많은 관련 연구가 활발히 진행되고 있다. 뿐만 아니라 이러한 네트워크 기반 시스템은 최근에 자율 이동 로봇 같은 복잡한 시스템으로 그 연구 분야가 확대되고 있는 실정이다. 자율 이동 로봇과 같은 복잡한 시스템을 네트워크 기반으로 구성할 경우에는 여러 가지 고려해야 할 사항이 발생한다. 첫째, 어떠한 분산화 전략을 수립할 지를 결정하여야 한다. 이러한 분산화 전략은 초기의 분산화 분해능(resolution)에 대한 전략의 수립에서 시작하여, 소프트웨어적인 구성요소를 어떠한 방식으로 분산화 된 각 네트워크 노드(node)에 최적의 효율을 가지고 배치를 할지를 목적으로 한다[1][2]. 둘째, 배치된 로봇의 소프트웨어적인 구성 요소를 어떠한 방식으로 실시간 보장 성능을 만족하면서 운영할지를 결정하여야 한다. 이러한 소프트웨어의 운영 전략은 각 네트워크 노드 내부에서의 실시간성 만족을 위한 운영 체계 스케줄링(scheduling) 같은 전략뿐만 아니라, 종단 간의 지연 시간에 대한 실시간성 고려와 한정된 자원의 운영에 있어 QoS(Quality of Service)를 최대한 보장하는 것을 목적으로 한다[3][4]. 셋째, 제어 안정도 확보를 위해 네트워크 지연시간으로 인한 제어 성능 저하의 보상 방식을 결정하여야 한다. 이러한 제어 보상 방식은 시변하는 시스템 지연 요소인 네트워크 지연 시간의 발생에 강인성을 갖는 제어기 개발을 목적으로 한다[6].

본 논문은 네트워크 기반 자율 이동 로봇을 위해 네트워크 지연으로 인한 제어 성능 저하의 보상 방식 개발을 목적으로 하며, 구체적으로 네트워크 기반 자율 이동 로봇에 있어 반응 제어(reactive control) 방식을 사용하는 장애물 회피 알고리즘을 대상으로 한 네트워크 지연 보상 방식을 제안한다. 본 논문에서는 대표적인 반응 제어 방식의 장애물 회피 알고리즘인 VFH(Vector Field Histogram) 알고리즘[7]을 네트워크 기반 자율 이동 로봇에 적용하였을 때 나타나는 문제점에 대해 분석 및 보상 방식을 제안하고, 최종적으로 효율적인 이동 경로 생성 능력과 목표 지점 안착율의 관점에서 제안된 보상 방식을 사용했을 때와 그렇지 않은 경우의 성능 비교를 통해 제안된 보상 방식의 효율성을 입증한다.

본 논문의 대상 시스템은 유연한 시스템 구성을 위해 큰 분해능을 갖는 시스템으로서 환경 정보를 인식하는 센서부와 로봇의 이동을 수행하는 구동부, 그리고 알고리즘이 수행되는 제어부가 지역 네트워크 상에 분산화 되어 있는 시스템으로 가정한다. VFH 알고리즘을 네트워크 지연에 대한 보상 없이 이러한 네트워크 기반 자율 이동 로봇에 적용하였을 경우에는 두 가지 문제가 발생하게 된다.

첫째, VFH 알고리즘의 수행 시점과 네트워크를 통해 전송되어지는 주위 환경 센서 정보의 취득 시점이 불일치하기 때문에 결과적으로 VFH 알고리즘에서는 과거의 환경 센서 정보가 이용되게 되고, 그로 인해 장애물 회피의 성능 저하를 발

† 교신저자, 學生會員 : 明知大 工大 情報工學科 工學碩士
E-mail : sukaa@dasatech.co.kr
* 正 會 員 : (株) 캐리어 主任硏究員
** 正 會 員 : 明知大 工大 情報工學科 敎授 · 工博
*** 正 會 員 : 韓國生産技術院 首席硏究員 · 工博
接受日字 : 2004年 12月 24日
最終完了 : 2005年 3月 14日

생시키게 된다. 본 논문에서는 이러한 취득 시점의 불일치에 대한 보상으로 타임스탬프 (time-stamp) 방식을 통한 네트워크 지연 시간 측정과 로봇의 기구학 모델(kinematic model)을 통해 예측된 현재의 환경 센서 정보를 이용하는 VFH 알고리즘을 제안한다.

둘째, VFH 알고리즘의 중요한 요소 중 하나인 정확한 로봇의 현재 위치판단(localization)에 있어 네트워크 지연은 위치판단의 부정확성을 야기할 수 있다. 이러한 위치판단의 부정확성은 로봇의 주행거리계(odometry) 정보에만 의존하는 위치판단에서뿐만 아니라[8], 위치판단의 정확성 향상을 위해 센서 융합(sensory fusion)을 사용하는 경우에도 센서 융합의 시점과 융합 대상의 감지 시점 차이로 인해 발생한다. 본 논문에서는 환경 센서 정보를 이용하여 확률적으로 위치 판단을 수행하는 기존의 알고리즘[9]과 주행거리계 정보의 센서 융합을 대상으로 하여, 네트워크 지연을 포함한 기존 위치 판단 정보와 주행거리계 정보, 그리고 각각의 네트워크 지연을 센서 융합 시점에서 로봇의 기구학적 모델을 이용하여 보상하는 칼만 필터(Kalman filter) 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문의 대상인 네트워크 기반의 자율 이동 로봇 시스템의 구성에 대해 서술하고, 3장에서 네트워크 기반 자율 이동 로봇에서 장애물 회피 알고리즘인 VFH 알고리즘을 구현할 때 발생하는 환경 센서 정보의 지연에 의해 문제점에 대해 서술하며 그 보상 방식을 제안한다. 또한 4장에서 주행거리계에 의존하는 위치 판단의 부정확성 개선을 위한 센서 융합 방식 중 하나인 칼만 필터(Kalman Filter) 알고리즘[10]을 서술하고, 환경 정보 기반의 위치판단 정보와 주행거리계 정보의 네트워크 지연이 고려된 칼만 필터 알고리즘을 제안한다. 5장에서 모의실험을 통해 제안된 보상 방식이 포함된 장애물 회피 알고리즘의 성능을 입증하고, 마지막으로 6장에서 결론을 맺고 향후 과제를 밝힌다.

2. 네트워크 기반 자율 이동 로봇 시스템의 구성

그림 1은 본 논문의 대상인 네트워크 기반 자율 이동 로봇이다[11]. 본 로봇은 브레인 모듈(brain module), 센서 모듈(sensor module), 모바일 모듈(mobile module)로 구성되어 있으며, 모든 모듈은 이더넷(Ethernet) 기반 네트워크로 연결되어 있다. 이러한 분산화 분해능은 낮은 분해능을 제공하는 네트워크 기반 자율 이동 로봇에 비해 높은 유연성을 확보하는 대신에, 네트워크 지연이 로봇의 제어 성능에 직접적인 영향을 준다는 단점을 갖는다.

센서 모듈은 환경 정보 취득을 위한 센서를 제어하는 모듈로서, 전방 190도 범위의 133점 분해능을 가지는 레이저 파인더(laser finder)와 360도로 배치된 16개의 초음파 센서(ultrasonic sensor)가 부착되어 있다. 모바일 모듈은 로봇의 구동을 제어하는 모듈로서, 3개의 바퀴로 구성된 단방향 구동(omni-directional) 방식을 사용하고, 구동부의 엔코더(encoder)를 이용하여 로봇의 현재 위치 정보인 주행거리계 정보를 계산하여 제공하는 역할을 한다. 브레인 모듈은 위치 판단의 확률적 계산 같은 복잡한 계산을 수행하는 역할을 한다.

그림 2는 본 논문의 대상이 되는 네트워크 기반 자율 이동 로봇에서 각 모듈의 소프트웨어 구성도이다. 각 모듈에는 리눅스(Linux) 운영체계가 탑재되어 있고, 이기종 플랫폼간의 소프트웨어 호환성을 위해 가상머신(virtual machine)이 탑재되어 있다. 또한 이기종 통신간의 호환성 있는 운영을 위해 통신 미들웨어(middleware)가 탑재되어 있다. 로봇 제어 프로그램 단위인 하나의 응용 프로그램은 하나의 가상머신에서 수행되며, 다중 태스크 작업을 위해 한 개 이상 다수의 가상머신이 하나의 모듈 내부에서 운영될 수 있다. 이렇게 다중 태스크 작업 수행을 위해 운영체계를 채용할 경우에는 네트워크 지연시간과 비동기적으로 결합하여 수백 밀리초(milli-second) 단위의 지연시간을 발생시키게 된다[5].

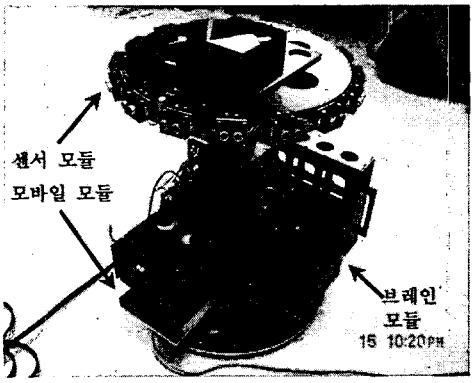


그림 1. 네트워크 기반 자율 이동 로봇
Fig 1. Network-based autonomous mobile robot

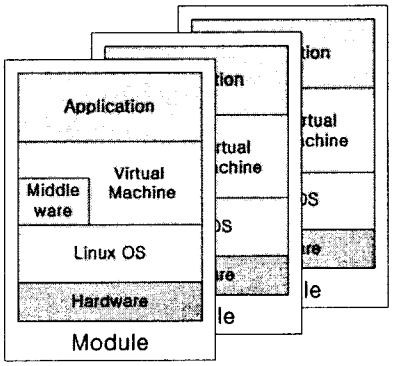


그림 2. 모듈의 소프트웨어 구성도
Fig 2. Software block diagram of module

그림 3은 본 논문의 대상이 되는 네트워크 기반 자율 이동 로봇의 소프트웨어 구성도이다. 그림 2의 가상머신 상에서 운영되는 소프트웨어는 하드웨어 의존적인 소프트웨어와 그렇지 않은 소프트웨어로 구분된다. 여기서 하드웨어 의존적인 소프트웨어는 어떠한 소프트웨어 구성 요소 배치 전략을 사용하더라도 특정 모듈에 존재할 수밖에 없는 소프트웨어 태스크(task)를 의미한다. 다시 말해, 센서 제어 소프트웨어는 실제 초음파 센서와 레이저 센서 정보를 수집하고, 수집된 센서 정보를 다른 모듈에 전송하는 역할을 하므로 센서 모듈에 배치되게 된다. 또한 모바일 모듈에 탑재되어 있는 드라이브 제어 소프트웨어는 실제 구동부를 제어하는 역할을 수행하며, 엔코더 정보를 이용하여 주행거리계 정보를 계산한 후 다른 모듈

에 전송하는 역할을 하므로 모바일 모듈에 배치된다.

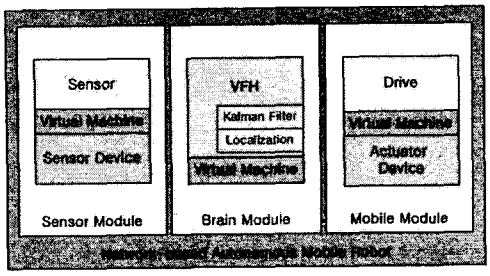


그림 3. 네트워크 기반 자율 이동 로봇의 소프트웨어 구성도
Fig 3. Software block diagram of network-based autonomous mobile robot

본 논문에서 VFH 알고리즘 제어 소프트웨어는 브레인 모듈에서 수행되며, 센서 모듈로부터의 환경 정보 센서 값과 모듈 내부의 위치 판단 정보, 그리고 모바일 모듈로부터의 주행거리계 정보를 칼만 필터 알고리즘을 이용하여 센서 융합한 최종 위치판단 정보에 의해 장애물 회피를 수행한다. 모듈 내부에서의 위치판단은 HRI(Human Robot Interface) 장치를 통해 사용자로부터 맵(map) 정보를 취득하여 취득된 맵 정보와 센서모듈로부터의 환경 정보를 이용한 확률 연산을 통해 수행된다. 따라서 VFH 알고리즘 수행에 있어 주위 장애물을 감지하기 위한 환경 정보 센서 값은 네트워크를 통해 전송되며, VFH 알고리즘 수행에 있어 위치 판단을 위한 환경 정보 센서 값과 주행거리계 정보 역시 네트워크를 통해 전송되게 된다. 이러한 센서 정보의 네트워크 전송은 운영체계의 지연 시간과 비동기적으로 결합하여 모듈 종단 간 네트워크 지연 시간을 발생시키게 된다.

3. VFH 알고리즘 적용을 위한 환경 센서 정보의 네트워크 지연 보상

VFH 알고리즘은 대표적인 반응 제어 방식의 장애물 회피 알고리즘으로서 불확실성을 갖는 환경 정보 센서 값의 보완을 위해 그림 4와 같이 히스토그램 격자 맵(histogram grid map) 방식의 활성 윈도우(active window) 정보를 지속적으로 갱신하여 그림 5와 같은 폴라 히스토그램(polar histogram)을 생성한다. 그림 4에서 Ws는 활성 윈도우의 크기를 의미한다. 그림5에서 Sector k는 로봇을 중심으로 한 절대 각도를 의미하고, H'(k)는 장애물에 대한 존재의 확률을 의미한다. 즉 H'(k) 값이 커질수록 k 방향에 장애물이 존재할 확률이 커짐을 의미하며, 반대로 H'(k)값이 작을수록 장애물이 존재할 확률이 작아짐을 의미한다. 따라서 로봇은 통과할 수 있는 기준인 임계값(threshold) 이하인 H'(k)값을 갖는 영역을 선택하여 장애물을 회피한다. 여기서, 한 개 이상의 임계값 이하 영역이 존재할 때 영역 선택은 목표 벡터(target vector)와 가장 가까운 빈 영역을 선택하게 된다.

본 논문의 대상이 되는 네트워크 기반 자율 이동 로봇에서, 센서 정보는 네트워크 전송 지연 시간을 포함하고 있기 때문에 VFH 알고리즘이 히스토그램 격자 맵을 생성할 시점의 센서 정보는 과거의 정보가 된다. 이로 인해 VFH 알고리즘의 수행 결과가 회피하지 않아도 될 수준이라고 판단되었다 할지라도 과거 시점에서 현재 시점까지의 로봇 이동에 의해 실제로는 장애물 회피가 필요한 시점일 수 있고, 부정확한 폴라 히스토그램의 생성으로 목표 지점으로의 효율적인 이동이 어렵게 된다. 최악의 경우에는 네트워크 지연으로 장애물 회피가 실패하는 결과가 야기될 수도 있다.

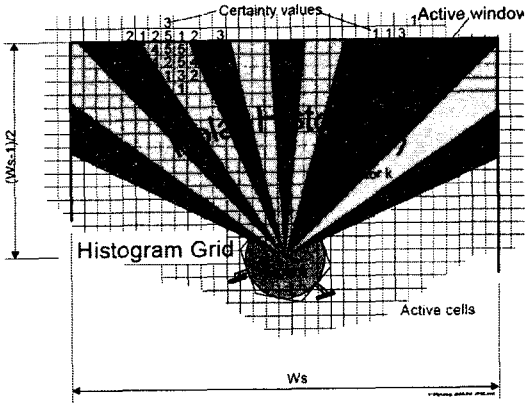


그림 4. VFH 알고리즘의 히스토그램 격자 맵
Fig 4. Histogram grid map of VFH algorithm

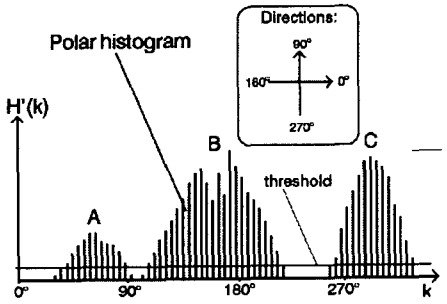


그림 5. VFH 알고리즘의 폴라 히스토그램
Fig 5. Polar histogram of VFH algorithm

따라서 본 논문에서는 네트워크 지연에 의한 과거의 환경 센서 정보를 이용한 히스토그램 격자 맵을 그림 5와 같이 네트워크 지연 시간 동안의 이동 방향에 대해 이동거리 만큼 이동하여 현재의 히스토그램 격자 맵을 예측하는 보상 방식을 제안한다. 제안된 보상 방식은 히스토그램 격자 맵의 중심에서 활성 윈도우 영역 안에 있는 확신값(certainty value)들을 사용하여 폴라 히스토그램을 생성한다. 그림 6에서 활성 윈도우 영역 밖에 있는 확신값들은 네트워크로 전송되어진 과거의 센서 정보이고, 측정된 네트워크 지연 시간을 이용하여 활성 윈도우 영역으로 들어오게끔, 확신값들을 로봇 기구학 모델을 이용하여 로봇 진행 방향의 반대 방향으로 이동시킨다. 그림 7의 절대 좌표계에서 네트워크 지연 시간 동안 로봇의 진행 방향 θ 와 이동 거리 l에 의해 변환되는 절대 좌표 상의 로봇 위치는 식 (1)과 같다. 따라서 일정한 속도 v로 이동하는 로봇에서 환경 센서 정보의 네트워크 지연 시간인 Δt_{loc} 동안 로봇을 중심으로 한 히스토그램 격자 맵의 이동은 식 (2)와 같이 표현된다. 식 (1)에서 H_{ij}는 과거의 센서 정보를 이용해 생성한 히스토그램 격자 맵이고, H'_{ij}는 로봇의 진행 방향 θ 와 반대 방향으로 네트워크 지연 시간동안 이동한 거리만큼

이동시킨 히스토그램 격자 맵이다.

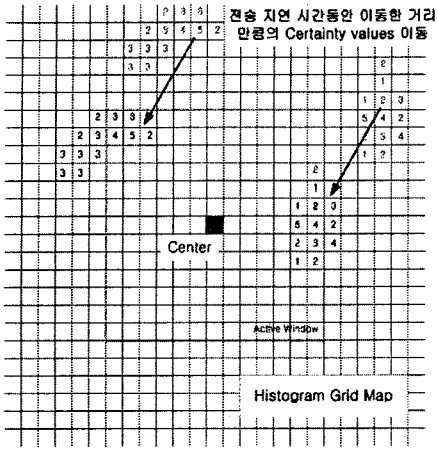


그림 6. 네트워크 전송 지연 시간을 이용하여 센서 정보를 보상한 히스토그램 격자 맵

Fig 6. Histogram grid map with compensated sensor information using network delay

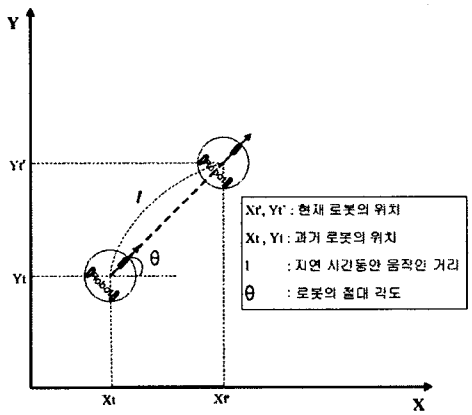


그림 7. 네트워크 지연 시간 동안의 로봇의 이동 거리

Fig 7. Travel distance of robot during network delay

$$Xt' = Xt + l\cos\theta$$
$$Yt' = Yt + l\sin\theta \tag{1}$$

$$H_{ij}' = H_{ij} + \begin{bmatrix} v\Delta t_{loc}\cos(180+\theta) \\ v\Delta t_{loc}\sin(180+\theta) \end{bmatrix} \tag{2}$$

4. 위치판단을 위한 네트워크 지연 보상 칼만 필터 알고리즘

자율 이동 로봇의 정확한 위치 판단은 정확한 목표지점으로의 이동에 필수적인 기능이다. 상기에 서술된 VFH 알고리즘의 경우, 그림 5의 폴라 히스토그램의 임계값 이하 영역 중 목표 벡터와 가장 가까운 빈 영역을 선택하게 된다. 기존의 VFH 알고리즘의 목표 벡터 B_{ij}는 주행거리계 정보를 이용한 현재 위치 정보와 목표 위치 정보를 이용하여 식 (3)과 같이 계산된다. 식 (3)에서 x_i, y_i는 목표 위치를 의미하고, x_0, y_0는 로봇의 현재 위치를 의미한다.

$$B_{ij} = \tan^{-1}\frac{y_i - y_0}{x_i - x_0} \tag{3}$$

네트워크 기반 이동 로봇의 경우에는 네트워크 지연에 의해 지연 시간을 갖고 전달된 주행거리계 정보를 이용하여 식 (3)의 목표 벡터를 계산하게 된다. 이러한 주행거리계 정보는 로봇의 목표 벡터 계산 시점보다 과거의 시점에 획득된 정보로 실제의 현재 로봇 위치와 편차를 갖고 있다. 따라서 잘못된 목표 벡터가 계산되게 되며 이로 인해 목표지점에 도달하는 목표지점 안착율의 성능 저하를 가져오게 된다.

특히 주행거리계 정보만을 이용한 위치판단은 로봇 기구학의 오차나 로봇 운동에서 발생하는 슬립(slip) 등의 현상으로 인해 부정확한 값을 갖기 쉽다. 이렇게 부정확한 주행거리계 정보의 보상을 위해 다양한 센서 융합 기법이 소개되었으며, 이러한 센서 융합 기법을 통해 다양한 센서 정보를 서로 융합함으로써 좀더 정확한 로봇의 현재 위치를 계산할 수 있다. 이러한 센서 융합에 사용되는 대표적인 방식이 칼만 필터 알고리즘이다[10]. 본 논문에서는 기존의 VFH 알고리즘에서 주행거리계 정보만을 이용하여 수행하는 위치판단의 부정확성과 네트워크 지연으로 인한 부정확성을 보상하기 위해 칼만 필터 알고리즘의 적용을 제안한다.

기존의 다른 연구 결과와 유사하게 주행거리계 정보만을 이용하는 경우의 부정확한 위치판단에 대한 보정을 목표로 한 칼만 필터의 상태 벡터(state vector)는 식(4)와 같이 정의된다. 식 (4)에서 X와 Y는 절대 좌표계에서 로봇의 현재 위치 좌표이고, θ는 절대 좌표계에서의 x축을 기준으로 한 로봇의 절대 각도이다.

$$x = \begin{pmatrix} X \\ Y \\ \theta \end{pmatrix} \tag{4}$$

칼만 필터의 예측 모델에 주행거리계 정보를 이용하면 엔코더 정보를 이용한 주행거리계 정보 계산은 모바일 모듈에서 수행되며 로봇의 기구학 모델을 이용하여 식 (5)와 같이 정의된다. 식(5)에서 δs_k^R와 δs_k^L은 k번째 샘플링 동안 이동한 오른쪽 바퀴와 왼쪽 바퀴의 이동 거리이고, a는 로봇 중심으로부터 바퀴까지의 거리이다. $(\delta s_k^R + \delta s_k^L)/2$는 로봇의 중심을 기준으로 해서 샘플링 시간동안 움직인 거리이고, $(\delta s_k^R - \delta s_k^L)/2a$는 로봇 중심을 기준으로 해서 샘플링 시간동안 움직인 로봇의 각도이다. $\hat{x}_{k-1}$는 *k-1*번째의 샘플링 시간의 로봇 상태 벡터이고, $\hat{x}_k^-$는 *k*번째 샘플링 시간의 로봇 상태 벡터로 *k-1*의 상태 벡터와 샘플링 시간동안 측정된 움직

인 거리와 각도를 기구학 모델을 이용하여 계산되어진다.

$$\hat{x}_k^- = \hat{x}_{k-1} + \begin{bmatrix} \cos\theta_k & 0 \\ \sin\theta_k & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{\delta s_k^R + \delta s_k^L}{2} \\ \frac{\delta s_k^R - \delta s_k^L}{2a} \end{bmatrix} \quad (5)$$

예측이 수행되는 예측 시간(time update) 중에 갱신되는 오차 공분산(error covariance) 행렬은 식 (6)과 같다. 식(6)에서 P_{k-1}은 k-1번째 샘플링 시간의 오차 공분산 행렬이고 P_k^-는 k번째 샘플링 시간에 추정된 오차 공분산 행렬이며, Q는 예측 모델에 대한 오차 공분산 행렬이다. 식 (5)에 의해 식 (6)의 행렬 A는 단위행렬 I가 된다.

$$P_k^- = A^T P_{k-1} A + Q \quad (6)$$

칼만 필터의 관찰 모델은 센서 정보와 HRI 장치를 통해 사용자에 의해 주어진 전역 맵 정보를 이용한 위치판단 알고리즘을 사용한다. 센서 정보와 사전에 주어진 맵 정보를 이용하는 위치판단 알고리즘은 MCL(Monte Carlo Localization) 알고리즘을 기반으로 한다. 본 논문에서는 기존의 MCL 알고리즘에서 지형 특성 형태를 고려한 Hough 변환(Hough transform)을 추가하여 개선한 알고리즘[9]으로서 환경 센서 정보와 맵 정보를 이용한 위치판단 알고리즘을 사용한다. 본 알고리즘에서는 전역 맵에서의 지역 트래킹(tracking)을 위해 주행거리계 정보를 사용하기 때문에 예측 모델 값 $\hat{x}_k^-$를 관찰 모델 값 계산에 사용하며, 예측 모델, 환경 센서 정보, 그리고 맵 정보를 이용하는 위치판단 알고리즘을 칼만 필터의 관찰 모델에 적용하면 관찰 값은 식 (7)과 같이 표현된다. 식(7)에서 S는 환경 센서 정보이고, M은 전역 맵 정보이다.

$$\hat{z}_k = f(\hat{x}_k^-, S, M) \quad (7)$$

본 논문에서는 칼만 필터의 보정 시간(measurement update) 중에 갱신되는 오차 공분산을 최소화하기 위한 칼만 이득을 사용한다[10]. 칼만 필터의 오차 공분산을 기존의 칼만 이득 연산 방식을 적용할 경우 식 (5), (6)에서 예측 모델과 관찰 모델은 동일한 도메인을 가지므로 칼만 필터의 이득은 식 (8)과 같이 표현 같다. 식 (8)에서 R은 관찰 모델에 대한 오차 공분산 행렬이다. 여기서 오차란 위치판단 결과의 확률적 오차를 의미하며, 본 논문에서는 관찰 모델의 관찰 결과로서 제공되는 오차 발생 확률을 이용하여 행렬을 식 (9)와 같은 행렬로 정의한다. 식 (9)에서 Err_x, Err_y, 그리고 Err_θ은 각각 관찰 모델인 위치 판단 알고리즘으로부터의 X축 상의 위치, Y축 상의 위치, 그리고 로봇 방향의 오차 발생 확률이다.

$$K_k = P_k^- (P_k^- + R)^{-1} \quad (8)$$

$$R = \begin{bmatrix} Err_x^2 & 0 & 0 \\ 0 & Err_y^2 & 0 \\ 0 & 0 & Err_\theta^2 \end{bmatrix} \quad (9)$$

따라서 최종 칼만 필터의 추정 상태 벡터 계산은 식 (10)과 같고, 보정 시간 중에 갱신되는 최종 오차 공분산은 식 (11)과 같이 계산 된다.

$$\hat{x}_k = \hat{x}_k^- + K(\hat{x}_k^- - \hat{z}_k) \quad (10)$$

$$P_k = (I - K)P_k^- \quad (11)$$

상기 서술한 칼만 필터 알고리즘을 이용한 센서 융합을 네트워크 기반 자율 이동 로봇에 적용할 경우, 네트워크 지연 시간으로 인해 융합 시점과 융합의 대상이 되는 정보의 취득 시점 간에 시간 차이가 발생한다. 따라서 본 논문에서는 측정된 네트워크 지연 시간과 로봇의 기구학 모델을 기반으로 하여 네트워크 기반 자율 이동 로봇에 적합한 칼만 필터 알고리즘을 제안한다. 제안된 칼만 필터 알고리즘은 앞에서 식 (5)를 이용하여 정의된 칼만 필터의 예측 모델에 대한 네트워크 지연 보상과 식 (7)을 이용하여 정의된 칼만 필터의 관찰 모델에 대한 네트워크 지연 보상이 포함된다.

식(5)의 칼만 필터 예측 모델은 모바일 모듈에서 연산되며, 연산된 예측 모델 값이 브레인 모듈에 전송되어 칼만 필터를 통해 센서 융합이 이루어지기까지 네트워크 지연이 발생하게 된다. 그림 7의 로봇 기구학 모델을 이용하여 네트워크 지연 시간동안 로봇이 이동한 거리를 고려하여 보상된 절대 좌표상에서 로봇의 현재 예측 위치는 식 (12)와 같다. 식 (12)에서 x와 y는 네트워크로 전송되기 이전에 식 (5)에 의해 모바일 모듈에서 계산된 로봇의 예측 위치판단 결과이고, x'과 y'은 브레인 모듈에서 전송된 주행거리계 정보를 기반으로 한 예측 위치판단 결과를 측정된 네트워크 지연시간을 고려하여 보상한 예측 위치판단 결과이다. Δt_{odm}은 모바일 모듈로부터 브레인 모듈까지 정보 전송에서 발생한 네트워크 지연 시간이고, v는 로봇의 이동 속도를 의미한다.

$$\begin{aligned} x' &= x + v\Delta t_{odm}\cos\theta \\ y' &= y + v\Delta t_{odm}\sin\theta \end{aligned} \quad (12)$$

식 (12)와 같이 로봇 위치판단의 예측 모델에 로봇의 이동 속도 성분이 포함되어야 하기 때문에 식 (4)의 상태 벡터는 식 (13)과 같이 속도 성분을 포함한 상태 벡터로 확장 정의된다. 식 (13)에서 $\dot{X}$와 $\dot{Y}$는 절대 좌표계에서 x축과 y축에서의

속도 성분이고, 식(14)와 같이 정의된다.

$$x=\begin{pmatrix} X \\ Y \\ \dot{X} \\ \dot{Y} \\ \theta \end{pmatrix} \tag{13}$$

$$\begin{aligned} \dot{X} &= v\cos\theta \\ \dot{Y} &= v\sin\theta \end{aligned} \tag{14}$$

식 (5)와 (12)를 기반으로 하여 정의된 네트워크 전송 지연 시간이 고려된 칼만 필터의 예측 모델은 식 (15)와 같다. 식 (15)에서 A 행렬은 네트워크 지연을 고려하여 예측 모델을 보상하기 위한 행렬로서 식 (16)과 같이 정의된다.

$$\hat{x}_k^- = \hat{x}_{k-1} + A\left(\begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ \frac{\cos\theta}{\Delta t_{odm}} & 0 \\ \frac{\sin\theta}{\Delta t_{odm}} & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} \frac{\delta s_k^R + \delta s_k^L}{2} \\ \frac{\delta s_k^R - \delta s_k^L}{2a} \end{bmatrix}\right) \tag{15}$$

$$A=\begin{bmatrix} 1 & 0 & \Delta t_{odm} & 0 & 0 \\ 0 & 1 & 0 & \Delta t_{odm} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{16}$$

식(7)의 칼만 필터 관찰 모델 값은 브레인 모듈에서 연산이 이루어지지만 모바일 모듈에서의 예측 모델 값과 센서 모듈에서의 환경 센서 정보를 이용하여 연산이 수행되므로 네트워크 지연으로 인한 과거 시점의 정보에 의해 계산이 이루어지게 된다. 식(7)의 칼만 필터 관측 모델에서 예측 모델은 전역 맵에서의 확률적인 지역 트래킹만을 위해 사용되기 때문에 로봇의 이동 속도에 비해 상대적으로 짧은 네트워크 지연에 대한 보상이 큰 의미를 갖지 않는다. 따라서 본 논문에서는 식(7)에서 환경 센서 정보만을 식 (17)과 같이 보상한다. 식 (17)에서 B 행렬은 로봇 기구학 모델을 이용하여 전송 지연 시간에 대한 오차를 개선하기 위한 행렬이며 식 (16)과 유사하게 식 (18)과 같이 정의된다. 식 (18)에서 Δt_{loc}는 센서 모듈로부터 전송되어진 환경 센서 정보의 네트워크 지연 시간이다.

$$z_k = Bg(\hat{x}_k, S, M) \tag{17}$$

$$B=\begin{bmatrix} 1 & 0 & \Delta t_{loc} & 0 & 0 \\ 0 & 1 & 0 & \Delta_{loc} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{18}$$

5. 모의실험 및 결과

그림 8은 본 논문의 모의실험을 위한 각 모듈별 응용 소프트웨어 구성도이다. 모듈의 모든 응용 소프트웨어들은 가상머신 환경에서 수행된다. 리눅스 운영체제 상에서 수행되는 가상머신은 라운드 로빈(round robin) 방식으로 스케줄링 되고, 라운드 로빈의 스케줄링 타임 슬라이스(time slice)는 200ms이다.

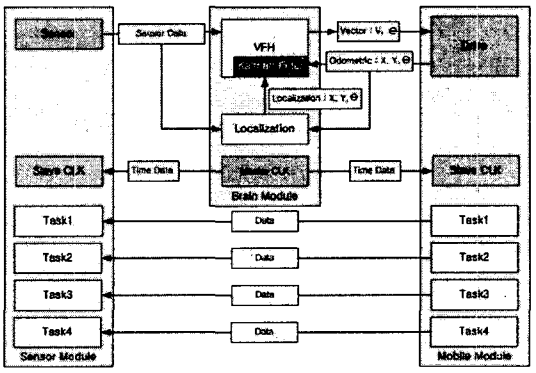


그림 8. 각 모듈별 소프트웨어 구성도
Fig 8. Software block diagram of each module

표 1. 네트워크 전송 메시지의 정의
Table 1. Definition of network transmission message

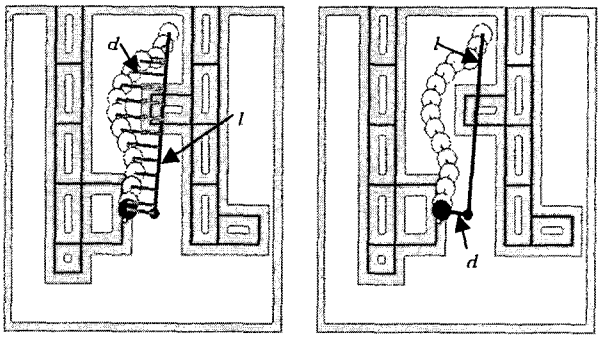
네트워크 메시지	전송 주기	데이터 크기
환경 센서 정보	500	532 Byte
주행거리계 정보	500	22 Byte
글로벌 클록	100	14 Byte
태스크 1	100	16 Byte
태스크 2	120	16 Byte
태스크 3	140	16 Byte
태스크 4	160	16 Byte

본 논문에서는 운영체계의 스케줄링 네트워크 전송 지연 시간이 포함된 종단간 네트워크 지연 시간을 측정하기 위해 타임스탬프 방식을 사용한다. 타임스탬프 방식을 사용하는 경우 전송 메시지의 전송 시점이 각 모듈의 지역 클록에 의해 코딩이 이루어지진다. 따라서 각 모듈의 지역 클록을 동기화하기 위해 브레인 모듈의 마스터 클록(Master CLK)은 브레인 모듈의 시간을 각 모듈의 슬레이브 클록(Slave CLK)으로 전송된다. 본 논문에서는 모든 기능이 구현된 네트워크 기반 로봇을 모사하기 위해 그림 8과 같이 임의의 더미 태스크(dummy task: Task1-Task4)와 더미 메시지(dummy message)를 포함한 시스템을 구성하였다. 모바일 모듈의 더미 태스크들은 센서 모듈의 더미 태스크들에게 더미 네트워크 메시지를 주기적으로 전송한다. 더미 네트워크 메시지가 포함된 전체 네트워크 메시지의 전송 주기와 전송 데이터 크기는 표 1과 같이 정의된다.

본 논문에서는 제안된 장애물 회피 알고리즘의 효율을 정량적으로 평가하기 위해 평가 항목을 제안한다. 평가 항목은 그림 9와 같이 이동 경로의 효율 평가와 목적 지점 안착율로 구분한다. 이동 경로의 효율 평가는 식 (19)와 같이 장애물이

없는 경우의 가장 이상적인 이동 경로인 시작 지점으로부터 목표 지점으로의 직선 경로와 로봇의 실제 이동 경로 사이의 편차 d_p를 직선 경로의 길이 l로 나눈 값의 평균으로 이루어진다. 따라서 이동 경로의 정량적 효율 평가 값은 작을수록 더 효율적인 경로가 생성됐음을 의미한다.

목적 지점 안착율은 도달된 로봇의 최종위치에서 도착 지점까지의 오차 거리 d_a를 구하여, 오차 거리 d_a를 출발 지점에서 목적 지점까지의 직선 경로 거리 l로 나눈 값이며, 식 (20)과 같다. 따라서 목표 지점의 안착율 역시 정량적 효율 평가 값이 작을수록 더 뛰어난 안착율을 갖는 것을 의미한다.



(1) 이동 경로 효율 평가 (2) 목적 지점 안착율 평가

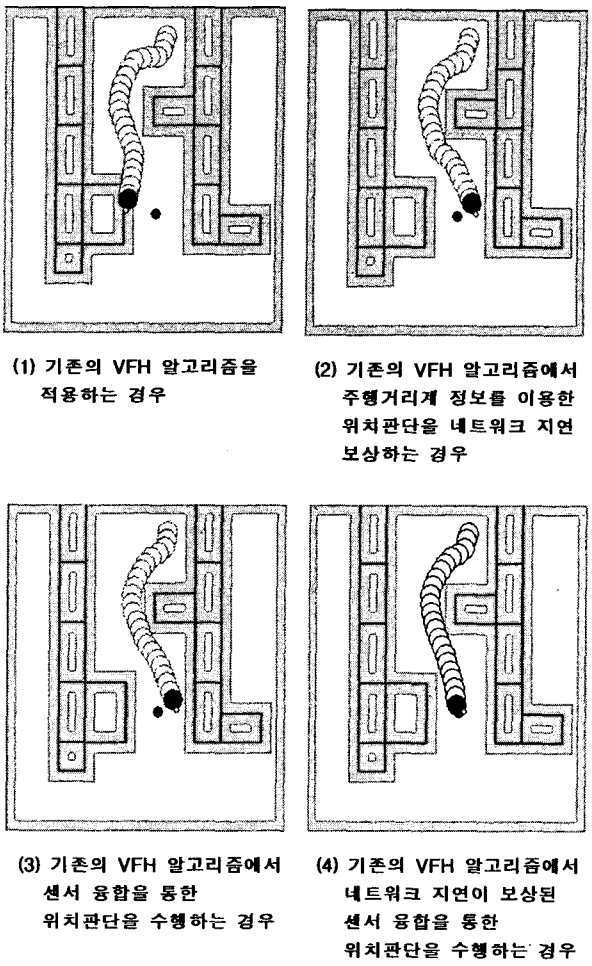
그림 9. 장애물 회피 알고리즘의 정량적 평가
Fig 9. Quantity estimation of obstacle avoidance algorithm

$$s = \sum (d_p / l) / n \tag{19}$$

$$a = d_a / l \tag{20}$$

모의실험을 위해 식 (6)으로 표현되는 제안된 칼만 필터 알고리즘의 예측 시간 중에 갱신되는 오차 공분산 행렬을 구하기 위한 예측 모델에 대한 오차 공분산 행렬 Q는 기존 연구 결과의 주행거리계 오차 측정 결과[9]를 이용하여 설정하였다. 일반적으로 Q는 칼만 필터의 성능 최적을 위한 조정 요소로서 사용되며, 실험적인 방식을 통해 구해지곤 한다. 본 논문에서는 기존 연구 결과에서 단방향 구동 방식의 로봇에 대해 1000mm를 주행하는 반복 실험한 결과인 평균 2%의 주행거리계 오차 결과를 이용하여 Q를 식 (21)과 같이 정의하였다.

$$Q = \begin{bmatrix} 0.0004 & 0 & 0 \\ 0 & 0.0004 & 0 \\ 0 & 0 & 0.0004 \end{bmatrix} \tag{21}$$



(1) 기존의 VFH 알고리즘을 적용하는 경우

(2) 기존의 VFH 알고리즘에서 주행거리계 정보를 이용한 위치판단을 네트워크 지연 보상하는 경우

(3) 기존의 VFH 알고리즘에서 센서 융합을 통한 위치판단을 수행하는 경우

(4) 기존의 VFH 알고리즘에서 네트워크 지연이 보상된 센서 융합을 통한 위치판단을 수행하는 경우

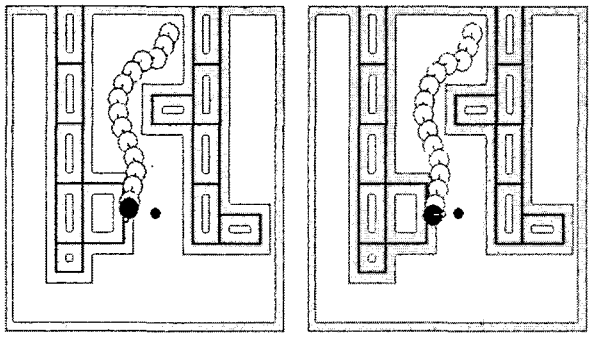
그림 10. 40mm/s 속도의 장애물 회피 경로 비교
Fig 10. Comparison of obstacle avoidance path in 40mm/s velocity

표 2. 40mm/s 속도에서의 장애물 회피 경로 평가
Table 2. Quantity estimation of obstacle avoidance path in 40mm/s velocity

알고리즘 / 성능	(1)의 경우	(2)의 경우	(3)의 경우	(4)의 경우
이동 경로(s)	0.163	0.105	0.99	0.101
목표지점 안착율(a)	0.19	0.11	0.11	0.03

표 2에서와 같이 (1)의 경우 보다는 (2)의 경우가, 그리고 (3)의 경우 보다는 (4)의 경우처럼 네트워크 지연 보상을 하는 경우가 그렇지 않은 경우 보다 목표 지점으로의 안착율이 더 뛰어남을 알 수 있다. 실제로 최악의 안착율을 갖는 (1)의 경우에는 목표 지점으로의 수렴이 실패한 결과를 보인다. 뿐만 아니라 생성된 이동 경로에 있어서도 네트워크 지연 보상을 하는 경우가 그렇지 않은 경우에 비해 더 부드럽고 짧은 경로를 형성함을 알 수 있다. 이러한 경로 생성의 특징은 이동 로봇에 있어 중요한 요소인 에너지 소비에 중대한 영향을 미치게 된다. 일반적으로 칼만 필터 알고리즘을 이용한 센서 융합을 이용하는 경우, 주행거리계 정보만을 이용하는 경우에 비

해 위치판단이 정확한 것으로 알려져 있다. 그러나, (2)와 (3)의 경우에서 보는 바와 같이 칼만 필터 알고리즘을 이용하여 센서 융합을 하는 경우에도 네트워크 지연 보상이 이루어지지 않으면 단순히 주행거리계 정보만을 이용하고 네트워크 지연을 고려한 경우와 성능 상에 큰 차이가 없음을 알 수 있다. 표 2에서 본 논문에서 제안된 방식이 가장 부드럽고 짧은 이동 경로와 높은 목표 지점 안착율을 갖는 것을 알 수 있다. 로봇의 이동 속도가 40mm/s인 경우 장애물 회피를 실패하여 장애물에 충돌하는 경우는 발생하지 않았지만, 최악의 경우인 (1)의 경우 목표 지점으로 수렴하지 못하여 VFH 알고리즘의 신뢰성 자체가 감소함을 확인하였다.



(1) 기존의 VFH 알고리즘을 적용하는 경우

(2) 기존의 VFH 알고리즘에서 주행거리계 정보를 이용한 위치판단을 네트워크 지연 보상하는 경우

(3) 기존의 VFH 알고리즘에서 센서 융합을 통한 위치판단을 수행하는 경우

(4) 기존의 VFH 알고리즘에서 네트워크 지연이 보상된 센서 융합을 통한 위치판단을 수행하는 경우

그림 11. 70mm/s 속도의 장애물 회피 경로 비교

Fig 11. Comparison of obstacle avoidance path in 70mm/s velocity

표 3. 70mm/s 속도에서의 장애물 회피 경로 평가

Table 3. Quantity estimation of obstacle avoidance path in 70mm/s velocity

알고리즘 / 성능	(1)의 경우	(2)의 경우	(3)의 경우	(4)의 경우
이동 경로(a)	0.167	0.154	0.131	0.130
목표지점 안착율(s)	0.17	0.15	0.09	0.05

그림 11은 그림 10과 같은 경우에 대해 속도를 70mm/s로 증가시켜 장애물 회피 실험을 수행한 결과이다. 표 3은 식 (19)와 식 (20)을 이용하여 그림 11의 결과를 정량적으로 평가한 결과이다.

표 3에서도 네트워크 지연 보상을 하는 경우가 그렇지 않은 경우에 비해 생성된 경로와 목표 지점 안착율에서 좋은 성능을 가짐을 알 수 있다. 또한 (2)와 (3)의 경우에서 보는 바와 같이 칼만 필터 알고리즘을 이용하여 센서 융합을 하는 경우에도 네트워크 지연 보상이 이루어지지 않으면 단순히 주행거리계 정보만을 이용하고 네트워크 지연을 고려한 경우와 성능 상에 큰 차이가 없음을 알 수 있다. 표 2와 표 3의 비교에서 이동속도가 70mm/s 인 경우 이동 속도고 40 mm/s인 경우에 비해 성능이 전반적으로 저하됨을 확인할 수 있다. 뿐만 아니라 (1), (2), 그리고 (3)의 경우 장애물 회피가 실패하여 장애물과 충돌을 일으키는 경우가 발생하여 VFH 알고리즘의 신뢰성 자체가 심각하게 감소함을 확인하였다. 따라서 본 논문에서 제안한 (4)의 경우만이 70mm/s의 속도에서 일정한 수준의 효율적인 경로 생성과 목표 지점 안착율을 가지며, VFH 알고리즘 자체의 신뢰성을 유지할 수 있음을 확인하였다.

6. 모의실험 및 결과

본 논문에서는 대표적인 반응 제어 방식의 장애물 회피 알고리즘인 VFH 알고리즘을 네트워크 기반 자율 이동 로봇에 적용하였을 때 나타나는 문제점에 대한 분석 및 보상 방식을 제안하였다. 본 논문에서 대상으로 하는 네트워크 기반 자율 이동 로봇의 경우에 VFH 알고리즘의 수행 시점과 네트워크를 통해 전송 되어지는 주위 환경 센서 정보의 취득 시점에 불일치하기 때문에 결과적으로 VFH 알고리즘에서는 과거의 환경 센서 정보가 이용되게 되고, 그로 인해 장애물 회피의 성능 저하를 발생시키게 된다. 본 논문에서는 이러한 취득 시점의 불일치에 대한 보상으로 타임스탬프 방식을 통한 네트워크 지연 시간 측정과 로봇의 기구학 모델을 통해 예측된 현재의 환경 센서 정보를 이용하는 VFH 알고리즘을 제안하였다. 또한 VFH 알고리즘의 중요한 요소 중 하나인 정확한 로봇의 현재 위치판단에 있어 네트워크 지연은 위치판단의 부정확성을 야기할 수 있다. 따라서 본 논문에서는 환경 센서 정보를 이용하여 확률적으로 위치판단을 수행하는 기존의 알고리즘과 주행거리계 정보의 센서 융합을 대상으로 하여, 네트워크 지연을 포함한 기존 위치판단 정보와 주행거리계 정보, 각각의 네트워크 지연을 센서 융합 시점에서 로봇의 기구학적 모델을 이용하여 보상하는 칼만 필터 알고리즘을 제안하였다.

본 논문에서는 정량적인 성능 비교를 통해 이동 경로의 효율성과 목표 지점 안착율에 대한 정량적인 평가 방법을 제안하였으며, 최종적으로 효율적인 이동 경로 생성 능력과 목표 지점 안착율의 관점에서 제안된 보상 방식을 사용했을 때와 그렇지 않은 경우에 대해 모의실험을 통한 성능 비교에 의해 제안된 보상 방식의 효율성을 입증하였다.

향후 연구과제로는 선형 시스템에 적합한 칼만 필터를 로봇과 같은 비선형 시스템에 적용하기 위한 확장된 칼만 필터를 네트워크 기반 자율 이동 로봇에 적용할 예정이며, 네트워크 지연 시간을 고려한 확장된 칼만 필터에 대한 연구를 수행할 예정이다.

참 고 문 헌

[1] Dar-Tzen Peng, Kang G. Shin and Tarek F. Abdelzaher, "Assignment and Scheduling Communicating Periodic Tasks in Distributed Real-Time Systems," *IEEE Transaction on Software Engineering*, Vol. 23 No. 12, pp.745-758, 1997.

[2] 오재준, 김홍렬, 김대원, "실시간 네트워크 시스템의 이용률 최적화를 위한 태스크 배치 전략 개발", 정보제어학술대회, pp.72-75, 2004.

[3] Chia-How Lin and Kai-Tai Song, "Flexible Real-Time Control of Home Robots Using a Multi-Agent Based Approach," *Proceedings of IEEE/RSJ International Conference on Intelligent Robot Systems*, pp.3092-3097, 2004.

[4] Hongryeol Kim, Joomin Kim, and 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.

[5] Hongryeol Kim, Joomin Kim, and Daewon Kim, "Development of Coordinated Scheduling Strategy with End-to-End Response Time Analysis for the CAN-Based Distributed Control systems," *Proceeding of 2004 IEEE/RSJ International Conference on Intelligent Robot and Systems*, pp.2099-2104, 2004.

[6] 김승용, 김홍렬, 김대원, "네트워크 기반 실시간 제어시스템을 위한 지연보상기 개발", 정보제어학술대회논문집, pp. 82-85, 2004.

[7] J. Borenstein and Y. Koren, "The Vector Field Histogram-Fast Obstacle Avoidance for Mobile Robot," *IEEE Journal of Robotics and Automation*, Vol. 7 No. 3, pp. 278-288, 1991.

[8] Sookyung Sohn, Hongryeol Kim, Daewon Kim, Hongseok Kim, and Hogil Lee, "Obstacle Avoidance Algorithm for a Network-based Autonomous Mobile Robot," *International Conference on Control, Automation, and System*, pp.831-833, 2004.

[9] Dongheui Lee, Woojin Chung, and Munsang Kim, "Probabilistic Localization of the Service Robot by Map Matching Algorithm," *International Conference on Control, Automation, and System*, 2002.

[10] Greg Welch and Gary Bishop, "An Introduction to the Kalman Filter," *http://www.cs.unc.edu/~welch/kalman/.*

[11] 산업자원부, "퍼스널로봇을 위한 시스템 엔지니어링 기술 개발에 관한 연구 (1단계 보고서)", 2004.

저 자 소 개

손 수 경 (孫 粹 耿)

1979년 5월 5일생. 2002년 명지대학교 전자정보통신공학부 졸업. 2005년 동대학원 졸업. 2005년~현재 ㈜다사테크 재직 중. 관심분야는 이동로봇 알고리즘, 임베디드 시스템 등.

Tel : (031)330-6755, Fax : (031)330-6226
E-mail : sukaa@dasatech.co.kr

김 홍 열 (金 弘 烈)

1973년 11월 25일생. 1996년 명지대학교 전기공학과 졸업. 1998년 동대학원 졸업. 2004년 명지대학교 정보공학과 박사과정 수료. 1997년~현재 ㈜캐리어 기술연구소 주임연구원. 관심분야는 실시간 분산시스템, 계층형 로봇 제어구조 등.

Tel : (031)330-6755, Fax : (031)330-6226
E-mail : hr.kim@carrier.co.kr

김 대 원 (金 大 元)

1960년 2월 15일생. 1983년 서울대학교 제어계측공학과 졸업. 1985년 동대학원 석사과정졸업(공학석사). 1990년 동대학원 박사과정졸업(공학박사) 1987년~1992년 ㈜대우중공업 중앙연구소 선임연구원. 1992년~현재 명지대학교 정보공학과 교수. 관심분야는 실시간 분산 시스템, 퍼스널로봇, 웹기반 응용 등.

Tel : (031)330-6755, Fax : (031)330-6226
E-mail : dwkim@mju.ac.kr

김 홍 석 (金 弘 錫)

1957년 6월 9일생. 1980년 서울대학교 전기공학과 졸업. 1983년 동대학원 제어계측공학과 석사과정졸업(공학석사). 1990년 동대학원 박사과정졸업(공학박사) 1983년~1987년 한국과학기술연구원 연구원. 1991년~현재 한국생산기술연구원 로봇기술개발본부 제어·지능연구팀 팀장·수석연구원/로봇종합지원센터 총괄센터장. 관심분야는 로봇용 S/W 프레임워크, 실시간제어시스템, 제어이론 등.

Tel : (031)500-3520, Fax : (031)500-4900
E-mail : hskim@kitech.re.kr