Journal of Korea Multimedia Society Vol. 16, No. 3, March 2013(pp. 354-365)
http://dx.doi.org/10.9717/kmms.2013.16.3.354

센서 필드 설계를 위한 배치 시뮬레이터

권오흠†, 송하주††

요 약

무선 센서 네트워크의 QoS (Quality of Service)에 큰 영향을 미치는 요소들 중 하나는 센서 노드의 배치이다. 본 연구는 감시정찰 센서 네트워크에서 센서 필드 설계시에 사용자의 의사결정을 효과적으로 지원하기 위하여 다양한 변수들을 고려한 배치 결과를 제공하고, 정량적인 분석을 실시하며, 또한 세밀한 센서 배치 시뮬레이션을 제공하는 시스템을 개발하는 것이다. 배치 형태를 영역 채우기, 경로 감시, 그리고 장벽 감시의 3가지 형태로 분류하여 격자 기반의 초기 배치 알고리즘을 제공하고, 배치 영역의 비정형성과 센서들의 감지 범위의 상이성을 고려하여 초기 배치를 수정하는 알고리즘을 제공한다. 배치된 노드들에 대해서 경로 커버리지와 장벽 커버리지를 분석하는 기능과 네트워크 시뮬레이션을 제공한다. 제안된 시뮬레이터는 감시정찰용 센서네트워크 시스템의 개발 환경의 일부로 활용될 수 있다.

Sensor Deployment Simulator for Designing Sensor Fields

Oh-Heum Kwon†, Ha-Joo Song††

ABSTRACT

Node deployment is one of the important problems in achieving good quality of service in wireless sensor network. The purpose of this paper is to develop an interactive system that supports user's decision makings in designing sensor fields. The system provides grid-based initial deployment algorithm supporting three types of node deployment pattern, area-fill, path-cover, and barrier-cover deployment pattern. After initial deployment, an iterative refinement algorithm can be applied, which takes care of the irregularity of the deployment area and the heterogeneity of sensors. The proposed system helps users to effectively deploy nodes in the sensor field, analyse the detection performance of the deployment, and perform network simulations. The developed system can be utilized as a part of the development environment of the surveillance sensor network system.

Key words: Surveillance(감시정찰), Sensor network(센서 네트워크), Simulator(시뮬레이터), Sensor deployment(센서 배치)

1. 서 론

네트워크의 정상적인 작동과 탐지 성능의 확보를 위해 노드들을 최적으로 배치하는 문제는 센서 네트워크 분야의 주요 연구 주제의 하나로서 지속적으로 연구되고 있다. 하지만 이 분야에서의 기존의 연구는 현실적인 고려사항을 충분히 고려하기 보다는 다소 이론적인 수준에서 진행되어 왔다고 할 수 있다 [1-7].

감시 정찰을 위한 센서 필드의 설계는 수없이 다

※ 교신저자(Corresponding Author) : 권오흠, 주소 : 부산광역시 남구 용소로 45 부경대학교 컴퓨터멀티미디어공학부(608-737), 전화 : 051) 620-6886, FAX : 051) 620- 6880, E-mail : ohkwn@pknu.ac.kr
접수일 : 2013년 1월 21일, 수정일 : 2013년 2월 21일
완료일 : 2013년 3월 3일

† 정회원, 부경대학교 컴퓨터멀티미디어공학부
†† 정회원, 부경대학교 IT융합응용공학과
E-mail : hajoosong@pknu.ac.kr
※ 이 논문은 2007학년도 부경대학교 연구년 교수 지원사업에 의하여 연구되었음(C-D-2007-0630).

양한 변수들을 고려해야하는 매우 복잡하고 어려운 과제이다. 탐지대상의 특성, 센서 및 노드의 기기적 특성, 센싱 프로세스와 통신에 대한 환경적 영향, 그리고 네트워크의 운영 정책 등 매우 다양한 요소들을 고려해야 한다. 또한 배치의 적합성을 평가하는 방법에 있어서 한두 가지의 정량적인 기준을 최적화하는 것이 항상 최선은 아니다. 때로는 정량화하거나 명확히 기술하기 어려운 요소들이 더 중요할 수도 있다. 즉, 배치 적합성의 평가 기준에 있어 본질적인 모호성이 존재한다고 할 수 있다. 이러한 고려사항의 다양함과 평가 기준의 본질적인 모호성으로 인하여 센서 배치 문제를 하나의 알고리즘적 문제로 접근하는 것은 비현실적이라고 할 수 있다. 센서 필드의 설계는 사용자의 적극적인 개입과 고차원적인 의사결정에 의존하는 매우 복합적인 과정을 통해 이루어질 수밖에 없으며, 그 과정에서 사용자의 의사결정을 지원하는 대화형 시스템의 필요성이 제기된다.

본 연구는 이러한 관점에서 필드의 형태, 센서 디바이스 및 센싱 프로세스의 특성, 감시 정찰의 대상 및 임무 등에 관한 세부적인 모델을 구축하고, 사용자가 요구하는 필드에 대한 배치 솔루션을 제공하며, 그것에 대한 다양한 정량적인 분석 결과 및 네트워크 시뮬레이션을 제공하고, 유연하고 풍부한 그래픽 사용자 인터페이스를 통해서 각종 매개변수를 조정하고 그에 대한 풍부한 피드백을 제공함으로써, 궁극적으로 센서 필드 설계에 관한 사용자의 의사결정을 효과적으로 지원하는 시스템을 개발하는 것을 목적으로 한다.

이에 본 연구에서는 다음과 같은 특징을 갖는 센서 배치 및 필드 시뮬레이터를 제안한다.

• 센서 네트워크를 구성하는 기기의 특성 및 센서 네트워크의 구성과 운영에 관련된 다양한 요소들을 설정하고 변경할 수 있다.

• 다양한 센서 배치 방법을 제공하고 배치된 센서들에 관하여 감지율 특성을 정량적으로 분석하는 기능을 제공한다.

• 정교한 네트워크 시뮬레이션을 제공하고 GUI 통해 이를 시각화하고 제어할 수 있는 기능을 제공한다.

논문의 전체 구성은 다음과 같다. 제2장에서는 관련 기술들에 대한 기존의 연구 결과를 기술한다. 제3장에서는 센서 배치 및 탐지율 분석 방법을 기술한다. 제4장에서는 본 시스템의 설계와 기능에 대해 기술한다. 마지막으로 제5장에서 결론을 맺는다.

2. 관련 연구

비용을 줄이고 신뢰할만한 통신 QoS를 제공하면서 센서필드를 효율적으로 커버하도록 노드를 배치하는 것은 센서네트워크에 있어서 매우 중요한 문제의 하나이며, 따라서 다양한 형태의 배치 알고리즘들이 연구되어져 왔다. 노드 배치 전략들은 센서 네트워크의 커버리지(coverage)를 최적화하고 연결성을 보장하는데 초점을 둔다[8]. 가장 기본적이면서 현실적인 노드 배치 전략은 노드들을 격자(grid)의 형태로 배치하는 것이다. 대표적인 형태는 그림 1에서 보여주듯이 삼각 격자, 사각 격자, 6각 격자 등이 있다. 격자에 따른 노드의 규칙적 배열은 노드의 소모 수량을 예측하기 쉽고, 배치의 특성을 수학적으로 분석하기 용이하며, 또한 센서 필드의 관리 측면에서 장점이 있을 수 있다. 반면 필드의 형태가 정방형 등의 정형(regular)이 아닌 경우나 감지 범위가 서로 다른 이종의(heterogeneous) 센서들이 섞여 있는 경우 최적의 감시 성능을 달성하지는 못한다.

Chakrabarty 등은 필드에 대한 커버리지를 최대화하는 배치 최적화 문제를 정수 계획법(integer programming) 문제로 정형화하여 다루었으며[9], 또한 다수의 연구들에서 Voronoi 다이어그램과 Delauny 삼각분할과 같은 계산기하학(computational geometry)적 도구를 이용한 알고리즘을 제시하였다[9]. 이러한 연구들은 필드 영역 내에서 센서에 의해 감지되는 영역을 최대화하는 것에 초점을 두며, 그런 의미에서 보통 “영역 커버리지(area coverage)”라고 불린다.

감시 정찰 센서 네트워크의 경우 일반적으로 탐지의 대상은 일정한 궤적을 가지는 침입자이다. 이는 필드 내의 임의의 지점에 임의의 시점에 발생할 수 있는 가령 화재 등과 같은 이벤트와는 차별된다. 즉

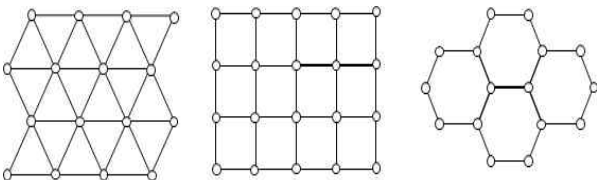


그림 1. 격자형 배치의 예

탐지 대상은 시공간적으로 연속성을 가지며 그런 의미에서 영역 커버리지와는 구별되는 “트랙 커버리지(track coverage)”의 개념을 사용한다[10]. 트랙 커버리지는 일정한 궤적을 지닌 탐지 대상이 필드를 지나갈 때 이를 탐지하는 능력이라고 포괄적으로 정의할 수 있으며, 이에 대한 다양한 연구 결과들이 제시되어 있다. 최소 노출 경로(minimal exposure path)는 필드 내의 일정 지점에서 목표 지점까지 도달하는 경로 중 센서들에 의해 탐지될 가능성이 가장 낮은 경로를 말하며 [10-13], 최대 위반 경로(maximal breach path)는 가장 가까운 센서와의 거리가 최대가 되는 경로를 말한다[12].

일반적으로 다항시간 최적화 알고리즘을 기대하기 어려운 복잡한 최적화 문제에 대한 전형적인 접근법의 하나는 반복적(iterative) 근사(approximation) 알고리즘을 사용하는 것이다. 이러한 접근법에서는 우선 적절한 초기배치에서 출발하여 반복적으로 배치를 수정함으로써 최적해에 근접한 해를 찾는 것을 목적으로 한다. 힘 기반 접근법(force-based method)라고 지칭되는 일련의 배치 알고리즘들은 이러한 접근의 대표적인 예이다[9,14]. 초기 배치 상태에 대해서 서로 근접한 노드들 간에는 서로 밀어내는 힘이 작용한다고 가정하고, 멀리 떨어진 노드들 간에는 서로 당기는 힘이, 그리고 필드 내의 장애물이나 주변 지형지물 등에 대해서도 적절한 가정하에 밀어내거나 당기는 힘이 존재한다고 가정한다. 하나의 노드에 대해 작용하는 이러한 힘들을 종합하면 노드가 움직여야하는 방향이 정해지고, 노드를 그 방향으로 일정 정도 이동함으로써 새로운 배치를 얻게 된다. 이 과정을 반복적으로 수행함으로써 최적해에 근접하는 해를 기대하는 것이다.

3. 센서 배치 및 배치 분석 알고리즘

3.1 배치 알고리즘

센서필드에 노드들을 배치하는 것은 매우 복잡한 고려사항들을 반영해야하는 어려운 문제이다. 제안 시스템에서는 노드 배치에 대해서 기본적으로 반복적인 접근을 택한다. 먼저 노드 배치를 위해 필요한 요구사항들과 제약조건들을 설정한다. 여기에는 필드의 형태와 배치될 노드의 유형, 성능 혹은 수량 등이 포함된다. 둘째, 노드들의 초기 배치를 계산한다. 초기배치는 배치 목적과 필드의 형태에 따라 몇 가지 기본적인 배치 패턴으로 나누어진다. 셋째, 배치된 노드들에 대해서 네트워크의 연결성 및 커버리지 분석을 제공하고, 필요한 경우 네트워크 시뮬레이션을 실시한다. 넷째, 이러한 분석의 결과를 반영하여 노드들의 배치를 수정한다.

본 절에서는 초기 배치의 패턴들과 배치 알고리즘들을 기술한다. 본 연구에서는 기본적으로 다음과 같은 3가지 유형의 초기 배치 형태를 지원한다.

- 영역 채우기 배치
- 경로 감시 배치
- 장벽 감시 배치

영역 채우기(area-fill) 배치

영역 채우기 배치는 일정한 영역으로 정의된 필드 내부에 가능한 균일하게 노드들을 배치하는 것을 말한다. 필드 영역은 영역의 외부 경계를 표현하는 하나의 다각형 P와 영역 내부에서 노드들이 배치될 수 없는 지역을 표현하는 여러 개의 홀(hole)들의 집합 $H=\{H_i : i=1,2,...,k\}$로 표현된다. 각각의 홀 H_i는 하나의 다각형이다.

영역 채우기 배치의 목적은 소요될 노드의 개수 N 혹은 노드간의 거리 Δ를 입력받아 배치 영역 전체가 가능한 한 균일하게 커버되도록 노드들을 규칙적으로 배치하는 것이다. 배치의 패턴은 삼각 격자 혹은 사각 격자의 형태를 지원하며 이하에서는 삼각 격자 형태의 배치 방법에 대해서 기술한다. 그림 2는 삼각 격자의 형태로 배치한 한 예를 보여준다. 그림에서 빗금 영역은 노드가 배치될 수 없는 홀을 표시한다.

필드의 형태가 내부에 홀을 가진 임의의 다각형의 형태이므로 배치 패턴이 격자형이라고 하더라고 노드들을 필드 내부에 배치하는 문제는 자명하지 않다.

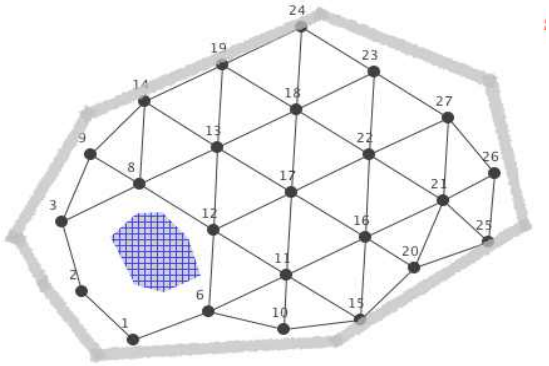


그림 2. 삼각격자 배치의 예

본 연구에서는 휴리스틱 알고리즘을 제안한다. 격자 배치에 있어서 첫 번째 고려사항은 격자의 축(axis)의 방향을 결정하는 것이다. 격자의 축은 사람들의 동선이나 주변의 지형, 등고선 등의 외적인 요소를 종합적으로 고려하여 결정하는 것이 바람직한데 이는 시스템이 자동으로 결정하기 어려운 문제이다. 이런 측면을 고려하여 본 시스템에서는 사용자가 격자의 수평축을 지정할 수 있게 구현하였으며 사용자가 축을 지정하지 않은 경우에는 다음과 같이 자동으로 격자의 축을 결정한다. 필드의 외부경계를 표현하는 다각형 P를 둘러싼 가장 작은 직사각형(MER: minimum enclosing rectangle)을 구한 후 그것의 변을 축의 방향으로 정한다. 평면상의 점들을 둘러싼 가장 작은 직각 사각형을 찾는 문제는 다항시간 알고리즘이 존재한다.

이제 노드간의 간격이 Δ가 되는 무한 격자를 평행이동하여, 격자의 각 정점에 노드가 놓인다고 가정하였을 때, 가장 많은 수의 노드가 다각형 P의 내부에 포함되는 위치를 찾는다. 이때 영역의 외부에 놓인 노드들 중에서 영역의 경계와의 거리가 적절한 상수 c에 대해서 Δ/c이하인 노드들은 영역의 내부로 끌어온다. 이는 영역 밖으로 약간 나간 노드로 인해 발생할 수 있는 그 주변의 커버리지의 공백을 매우기 위함이다. 그림 3은 이 과정을 설명한다.

노드간의 간격 Δ 대신 소요될 노드의 수 N이 입력으로 주어지는 경우에는 먼저 다음과 같은 조건을 만족하는 노드간의 간격 Δ_0를 계산한다.

$$\left(\frac{W}{\Delta_0}+1\right)\cdot\left(\frac{H}{\sqrt{3}\frac{\Delta_0}{2}}+1\right)=N \qquad (1)$$

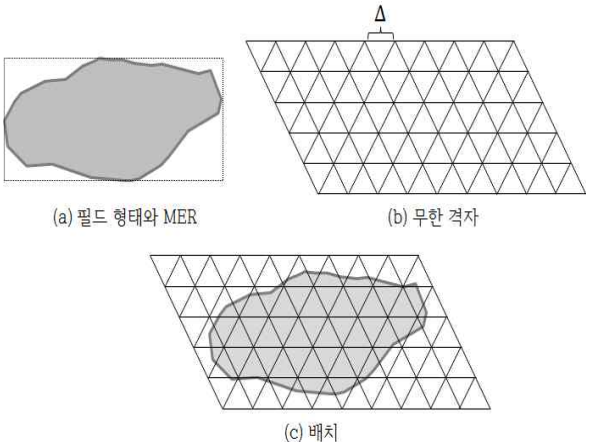


그림 3. 영역 채우기 배치 - 삼각 격자

여기서 W와 H는 각각 둘러싼 사각형 MER의 넓이와 높이이며, (수식 1)을 만족하는 N은 MER 전체를 노드 간격 Δ_0인 삼각 격자로 채울 때 소요되는 대략적인 노드 수이다. 이제 노드 간격을 Δ_0로 지정한 후 위에 설명한 방법으로 노드를 배치한다. 이 경우 필드의 형태에 따라 소요된 노드의 수는 N을 초과할 수도 있고 반대로 미달할 수도 있다. 이제 일종의 이진 탐색(binary search) 알고리즘을 이용하여 $\Delta_1=2\Delta_0$ 혹은 $\Delta_1=\Delta_0/2$로 하고 다시 배치한다. 이 과정을 소요된 노드의 수가 목적한 N에 가깝게 될 때 까지 반복한다. 이 방법은 정확하게 노드의 수가 N이 되는 것을 보장하지는 않는다.

경로 감시 배치

경로감시 배치는 주어진 일정한 폭의 경로를 따라서 노드들을 배치하는 것이다. 도로나 교량 등을 따라 이동하는 대상을 탐지하기 위하 목적의 배치이다. 이 경우 경로의 한쪽 편을 따라 배치하거나 혹은 양쪽을 지그재그 형태로 오가며 배치한다.

장벽감시 배치

장벽 감시 배치는 지정된 개수의 열로 노드들을 배치하여 장벽의 역할을 하도록 하는 것이다. 열의 개수와 열 간의 간격, 그리고 노드들 간의 간격이 입력으로 주어진다. 그림 4는 경로감시 배치와 장벽 감시 배치의 예이다.

노드 배치 향상 알고리즘

초기배치 알고리즘에서는 영역의 경계나 홀 부근의 노드들은 강제적으로 영역 내부로 이동 배치되므로 엄밀한 의미에서 등간격 배치라고 할 수 없다. 보다 근본적인 문제로는 노드들이 서로 다른 유형의 센서들을 탑재하고 있고 센서들의 감지 범위가 서로 다를 경우 등간격 배치가 반드시 균일한 커버리지를 제공하는 것은 아니라는 점이다. 재배치 알고리즘은 이러한 측면들을 고려하여 배치된 노드들의 위치를 부분적으로 수정하는 알고리즘이다. 기본적으로 반복적인(iterative) 방식으로 배치를 수정한다. 목적은 필드내의 영역에 대한 커버리지의 균일성을 최대화하는 것이다. 알고리즘의 개요는 다음과 같다.

먼저 배치된 노드들에 대한 Voronoi 다이어그램을 구한다. Voronoi 다이어그램은 평면상의 모든 점

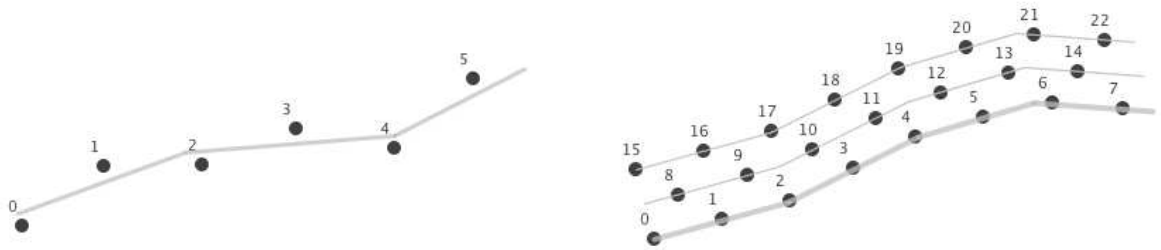


그림 4. 경로감시 배치와 장벽감시 배치의 예

들을 자신과 가장 가까운 입력점들에 따라 분할(partition)하는 다이어그램이다. 구체적으로는 다음과 같이 정의된다. n개의 노드의 위치 $p_1, p_2, \ldots, p_n$이 주어질 때 각각의 점 p_i에 대한 Voronoi 영역(region) R_i는 다음과 같이 정의된다.

$$R_i = \{x \in R^2 : d(x, p_i) \le d(x, p_j) \text{ for all } j \ne i\}$$

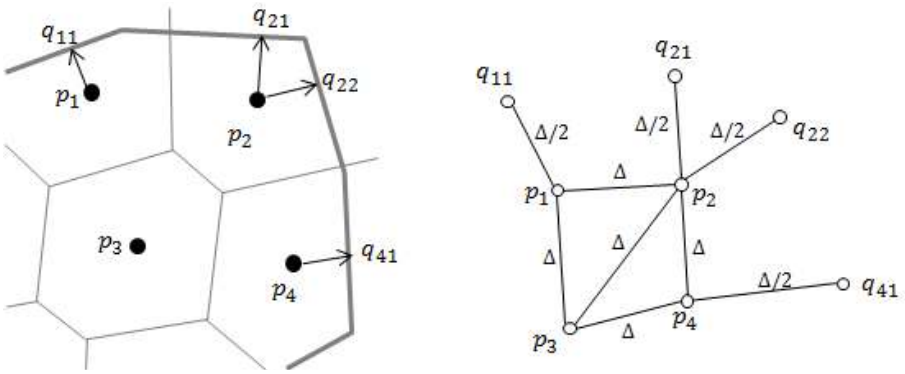
Voronoi 다이어그램의 듀얼(dual) 그래프 $D = (V, E)$를 Delaunay 삼각분할(triangulation)이라고 부른다. 즉 각각의 노드들을 정점으로 하고 두 노드의 Voronoi 영역이 하나의 변을 공유하면 두 정점 간을 에지로 연결한다.

$$V = \{p_1, p_2, \ldots, p_n\}$$
$$E = \{(p_i, p_j) | R_i \text{와} R_j \text{가 변을 공유}\}$$

Delaunay 그래프 D를 정점과 에지를 추가하여 다음과 같은 확장된 그래프 $D' = (V', E')$를 정의한다. 먼저 다음과 같이 정점을 추가한다. 만약 어떤 노드 i의 Voronoi 영역 R_i가 필드의 영역 외부 경계 P 혹은 필드 내부의 홀 H_k와 교차할 경우 P 혹은 H_k의 각각의 변과 노드 i로부터의 수선의 교차점을 찾아 그것을 정점 V에 추가한다. 그림 5에서 보여주듯이 이 교차점은 하나 이상일 수 있으므로 각각을 $q_{i,j}, j = 1, 2, \cdots,$ 로 표기하자. 정점 p_i와 $q_{i,j}$간을 에지로 연결한다.

이제 이렇게 정의된 그래프 $D' = (V', E')$의 각 에지에 가중값을 할당한다. 이 가중값은 두 노드 간의 바람직한(desired) 거리를 표현한다. 기본적인 방법은 새로 추가된 에지들의 가중값은 $\Delta/2$으로 하고, 원래 D에 속한 에지들의 가중값은 Δ로 하는 것이다. 이것은 노드들간의 거리는 Δ가 되고, 노드와 영역의 경계면까지의 거리는 $\Delta/2$가 되기를 원한다는 의미이다. 만약 노드들이 서로 다른 유형의 센서를 탑재하고 있고, 센서 유형마다 최대 감지 범위에 차이가 있는 경우 그러한 특성을 반영하여 거리를 설정할 수 있다. 가령 노드 i와 j에 탑재된 센서의 감지 범위가 각각 Δ_i와 Δ_j라면 두 노드를 연결하는 에지의 가중값은 적절한 상수 $0 < \alpha < 1$에 대해서 $\alpha \cdot (\Delta_i + \Delta_j)$로 설정할 수 있다.

이제 그래프 $D' = (V', E')$의 각 정점들의 현재 위치를 다음과 같이 표기하자.



(a) Voronoi 다이어그램 (b) 확장된 Delaunay 그래프

그림 5. Voronoi 다이어그램과 확장된 Delaunay 그래프의 예

$X=\{x_i,\ i=1,2,...,n+m\}$

여기서 $X_1=\{x_1,x_2,...,x_n\}$은 각각 노드들의 현재 위치이며, $X_2=\{x_{n+1},...,x_{n+m}\}$은 새로 추가된 영역 혹은 홀의 경계상의 점들의 위치들이다. 여기서 $x_{n+1},...,x_{n+m}$는 고정된 값이며, 우리의 목표는 노드들 간 거리가 대응하는 에지의 가중값과 가능한 한 일치되도록 $x_1,x_2,...,x_n$의 값을 결정하는 것이다. 즉 다음과 같은 일종의 그래프 실현(graph realization) 문제로 기술될 수 있다.

$$X_{opt}=\mathrm{argmin}_{X_1}=\sum_{(i,j)\in E'}(\delta_{ij}-d_{ij})^2$$
$$d_{ij}=\|x_i-x_j\|$$

여기서 δ_{ij}는 에지의 가중값, 즉 우리가 원하는 거리이며 d_{ij}는 두 점 간의 실제 거리이다.

이 수식에 대한 닫힌 해나 최적해를 구하는 알고리즘은 알려져 있지 않으나 반복적 최적화 기법을 적용하여 근사해를 구할 수 있다. SMACOF 반복적 최적화기법은 그래프 실현 문제에 적합한 전통적인 반복적 최적화 기법의 하나이며, 반복이 진행되는 과정에서 적어도 오차가 증가하지는 않음이 증명되어 있다[15]. SMACOF 알고리즘의 정확도는 초기조건에 의존적이다. 본 논문에서는 초기 배치 위치를 알고리즘의 초기값으로 사용하여 SMACOF 알고리즘을 적용하여 노드의 위치를 개선한다.

그림 6은 삼각격자 배치에서 초기 배치와 배치 향상 알고리즘을 적용한 후의 예이다. 그림에서 초록색 원은 각 센서의 감지 범위를 표시한다. 이 예에서는 감지 범위가 서로 다른 2 종류의 센서가 사용되었다.

3.2 커버리지 분석

3.2.1 경로 커버리지 분석

센서 필드에 대한 경로 커버리지 분석은 탐지 대상의 유형과 이동 경로 및 속도가 지정될 때 센서네트워크가 그 대상을 탐지할 확률을 분석하는 기능이다. 탐지 대상의 궤적은 시간에 따른 대상의 위치를 정의하는 함수 $l:Z\rightarrow R^2$으로 표현된다. $l(t),\ t=0,1,...$는 시간 t에서 탐지 대상의 위치를 나타낸다. N개의 노드가 배치되어 있고, 노드 i에 장착된 센서를 s_i로 표기한다. 각각의 시간 t에 탐지 대상이 각각의 센서 s_i에 의해서 탐지되는 사건은 서로 독립적인 Bernoulli 시도(trial)로 간주되며 $Bernoulli(p(i,t))$ 분포를 따른다고 가정한다. 본 논문에서 $p(i,t)$는 시간 t에서의 감지 대상과 센서 s_i의 위치의 함수, 즉 $p(i,t)=f(x_i,l(t))$라고 가정한다. 여기서 x_i는 센서 i의 위치이다.

함수 f에 대해서는 기존의 연구들에서 다양한 방법으로 다루어졌다. 가장 간단한 경우는 다음과 같이 정의하는 것이다.

$$f(p,q)=\begin{cases}\alpha, & d(p,q)<\delta\\ \beta, & \text{otherwise.}\end{cases}$$

여기서 $d(p,q)$는 두 점 p와 q간의 거리이다. 즉 센서와 타겟 간의 거리가 임계값 δ이하이면 고정된 확률 α로 탐지에 성공하고, 그렇지 않으면 탐지 확률을 β으로 정의하는 것이다. 일반적으로 $\alpha=1$이고 $\beta=0$인 경우를 "단위 디스크(unit disk) 모델"이라고 부른다. $\beta>0$인 경우 이것은 오감지(false detection)를 표현하는 것으로 해석할 수 있다. 즉 탐지 대상이 충분히 멀리 떨어져 있거나 혹은 아예 존재하지 않은 경우에도 센서는 단위 시간당 β의 확률로 오감지 할 수 있다는 사실을 표현한다.

PIR 센서와 같이 지향성을 가지는 센서의 경우 센서의 설치 방향 $\theta, 0\le\theta<360$, 와 감지각의 범위 ϕ가 추가되어야 한다. 즉 센서로부터 타겟의 상대적 방향이 $\theta-\phi/2$와 $\theta+\phi/2$ 범위에 속할 경우 α의 확률로 탐

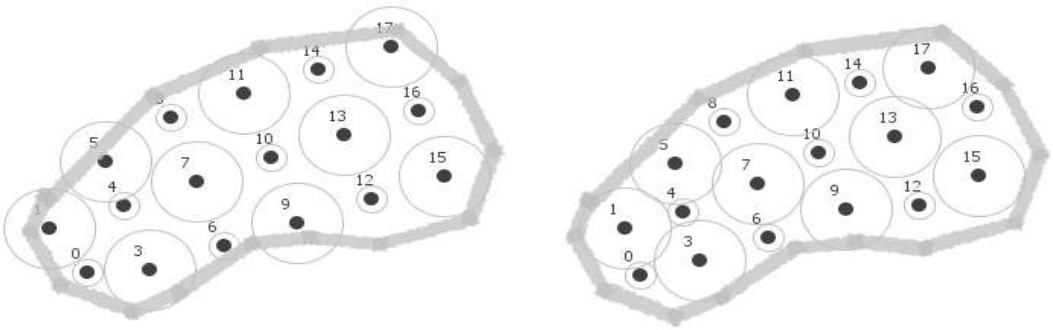


그림 6. 배치 향상 이전과 이후의 예

지에 성공한다. 즉 지향성 센서의 경우 다음과 같이 정의된다.

$$f(p,q)=\begin{cases}\alpha, & d(p,q)<\delta \text{and} \theta-\phi/2<dir(\overrightarrow{pq})<\theta+\phi/2 \\ \beta, & \text{otherwise.}\end{cases}$$

이 모델에서 각각의 센서는 전방향(omnidirectional) 센서의 경우 세 가지 매개변수 (α,β,δ)에 의해서, 그리고 지향성 센서의 경우 $(\alpha,\beta,\delta,\theta,\phi)$로 기술된다.

일반적으로 단일 센서의 단일 감지값에 기반하여 침입여부에 대해 판정(decision)을 내리는 것보다 센서들의 감지값을 시간적 혹은 공간적으로 융합(fusion)하여 판정을 내리는 것이 바람직하며, 특히 감시정찰 목적의 센서 네트워크의 경우 오감지(false alarm)을 개선하는데 있어서 매우 중요한 역할을 할 수 있다. 본 논문에서는 서로 다른 센서간의 융합은 고려하지 않으며, 단일 센서 차원에서 시간적인 융합, 즉 단일 센서 상에서의 복수의 연속된 감지값을 융합하여 침입 여부를 판단하는 정책만을 고려한다. 이 경우 대표적인 방법으로는 지정된 횟수만큼 연속적으로 탐지에 성공할 경우 침입으로 판단하는 "K-consecutive" 정책과 혹은 연속된 R번의 감지값 중에서 K번 이상이 성공할 경우 침입으로 판단하는 "K-out-of-consecutive-R" 정책 등이 있다. 이하에서는 최종적으로 침입으로 판단하는 것을 "경보(alarm)"를 발행한다고 부른다.

K-consecutive 정책 하에서 시간 t에 센서 s_i가 경보를 발행할 확률 $p_{alarm}(i,t)$은 다음과 같이 표현된다.

$$p_{alarm}(i,t)=1-\prod_{t'=t-K+1}^{t}(1-p(i,t'))$$

이제 시간 t 혹은 그 이전에 센서 s_i가 대상을 경보할 확률 $CDF(i,t)$를 계산해보자. 탐지의 경우와는 달리 경보를 발행하는 사건들은 시간에 따라 서로 독립적이지 않으므로 위와 같이 단순한 방법으로는 누적확률을 계산할 수는 없다. (가령 시간 $t-1$에 경보를 발행하는 사건과 시간 t에 경보를 발행하는 사건은 서로 독립적이지 않다.) $CDF(i,t)$를 구하기 위하여 먼저 다음과 같이 값 $CDF_i(k,t)$를 정의한다.

정의: $H_i(k,t), k\le K, t\ge 0,$ 를 센서 s_i가 시간 구간 $[0,t]$동안 K번 연속해서 대상을 탐지했거나 혹은 시간 $[t-k+1,...,t]$ 동안 연속하여 k번 대상을 탐지하였을 사건이라고 하고, $CDF_i(k,t)$를 사건 $H_i(k,t)$가 일어날 확률이라고 정의한다.

이제 사건 $H_i(k,t)$가 일어날 확률 $CDF_i(k,t)$을 계산하기 위하여 다음과 같이 두 가지 경우로 나누어 생각할 수 있다. 첫째, 시간 t에 탐지에 성공한 경우이다. 이 경우 사건 $H_i(k,t)$가 성립하기 위해서는 시간 $t-1$에 사건 $H_i(k-1,t-1)$이 성립해야 한다. 즉 시간 $t-1$ 이내에 K번 연속해서 대상을 탐지했거나 혹은 마지막 $t-1$ 시간 구간동안 연속하여 $k-1$번 대상을 탐지하여야 한다. 둘째, 시간 t에 대상을 탐지하지 못한 경우이다. 이 경우 사건 $H_i(K,t-1)$이 성립해야 한다. 이것은 시간 $t-1$ 이내에 K번 연속해서 대상을 탐지했어야 한다는 것을 의미한다. 이는 시간 t에 이미 탐지를 실패했으므로 마지막 k시간 구간 동안 연속해서 대상을 탐지할 수는 없기 때문이다. 따라서 K-consecutive 정책 하에서 센서 s_i가 대상을 경보할 시간에 따른 누적 확률은 다음과 같은 순환식으로 표현가능하다.

$$CDF_i(k,t)=\begin{cases}0, & t<k, \\ 1, & k\le 0, \\ (1-p(i,t))\cdot CDF_i(K,t-1)+p(i,t)\cdot CDF_i(k-1,t-1), & \text{otherwise}\end{cases}$$

이제 센서 s_i에 의해 시간 t 혹은 그 이전에 K번 연속하여 임계값을 넘는 값이 탐지되어 침입으로 판단할 확률은 $CDF(i,t)$는 다음과 같이 계산된다.

$$CDF(i,t)=CDF_i(K,t)$$

이것은 위의 순환식으로부터 간단한 동적계획법(dynamic programming) 알고리즘을 적용하여 $O(Kt)$ 시간에 계산할 수 있다.

각 센서 별로 경보를 발행할 누적확률함수가 계산되면 각 센서별 경보 발행 사건들은 서로 독립적이므로 전체 네트워크에 의해서 대상이 탐지될 누적확률 $CDF(t)$는 다음과 같이 계산된다.

$$CDF(t)=1-\prod_{i=1}^{N}(1-CDF(i,t))$$

K-out-of-consecutive-R 정책 하에서 누적 확률을 계산하는 문제에 대해서는 몇몇 상한(upper bound)과 하한(lower bound)를 구하는 알고리즘이 알려져 있을 뿐 정확한 값을 계산하는 다항시간 알고리즘이 알려져 있지 않다. 현재 알려져 있는 알고리

즘들은 지수함수(exponential) 시간 복잡도를 가지며 본 연구에 적용할 수 없다. 본 연구에서는 통계적인 방법으로 K-out-of-consecutive-R 정책 하에서의 탐지율을 계산한다. 즉, 1000회 이상의 랜덤 데이터를 생성하여 시뮬레이션을 통해 각 시간 t에 대해서 경보가 발행될 확률을 통계적으로 계산하여 사용자에게 제공한다.

3.2.2 장벽 탐지율과 최소탐지 경로

장벽 탐지율은 배치된 노드들이 침입에 대한 장벽으로서의 기능을 수행하는 성능에 대한 척도가 될 수 있다. 본 연구에서 장벽 탐지율은 탐지 대상이 센서 필드를 통과할 때 네트워크에 의해서 탐지될 "최소" 확률로 정의한다. 경로 탐지율과 장벽 탐지율의 차이점은 경로 탐지율의 경우 지정된 특정한 경로를 따라 대상이 이동하는 경우의 탐지율이며, 장벽 탐지율의 경우 그러한 지정된 경로가 존재하지 않는다는 것이다.

장벽 탐지율을 정의하기 위해서는 우선 탐지 대상이 센서필드를 통과한다는 것의 의미에 대한 정의가 필요하다. 본 연구에서는 모든 노드들을 포함하는 최소 사각형(minimum enclosing rectangle)의 한쪽 변에서 반대편 변으로 이동하는 경우에만 센서 필드를 통과했다고 간주한다.

일반적으로 최소 탐지 경로는 복잡한 곡선을 그리게 되며 최소 탐지경로를 다항시간에 찾아내는 알고리즘은 알려져 있지 않다[17]. 하지만 주어진 필드를 세밀한 격자로 분할하여 격자 상으로만 움직인다고 가정하면 최소 탐지 경로를 찾는 문제는 이산적인(discrete) 문제로 변환되며 일종의 최단경로 문제로 변환가능하다. 필드를 가로 세로 일정한 크기의 격자로 분할하고 격자를 따라서만 움직인다고 가정하면 그리드를 하나의 그래프 $G=(V,E)$ 로 볼 수 있다. 여기서 V는 격자의 모든 정점에 해당하고 E는 격자를 구성하는 에지들의 집합이다. 이제 이 그래프의 모든 에지 $(u,v)\in E$에 대해서 대상이 그 에지를 따라 움직일 때 네트워크에 의해 탐지될 확률을 $p(u,v)$라고 하자. 이 확률은 경로 탐지률 모델에 따라서 계산가능하다. 이제 격자상의 임의의 경로 $p=v_0v_1\ldots v_k$의 탐지 확률 $p_{\mathrm{detect}}(p)$는 다음과 같이 정의된다.

$$p_{\mathrm{detect}}(p)=1-\prod_{i=1}^{k-1}(1-p(v_i,v_{i+1}))$$

탐지 확률이 최소가 되는 경로를 찾는 것은 $\prod_{i=1}^{k-1}(1-p(v_i,v_{i+1}))$이 최대가 되는 경로를 찾는 문제와 동일하다.

$$\log\prod_{i=0}^{k-1}(1-p(v_i,v_{i+1}))=-\sum_{i=0}^{k-1}(-\log p'(v_i,v_{i+1}))$$

이다. 여기서 $p'(v_i,v_{i+1})=1-p(v_i,v_{i+1})$이다. 따라서 탐지 확률이 최소가 되는 경로는 $\sum_{i=0}^{k-1}(-\log p'(v_i,v_{i+1}))$가 최소가 되는 경로를 찾는 것과 동일한 문제이다. 즉, 그래프의 각 에지 (u,v)에 가중값 $-\log p'(u,v)>0$을 할당한 후 최단경로를 찾으면 된다. 최단경로를 찾는 문제는 Dijkstra의 알고리즘에 의해 $O(|V|\log|V|)$ 시간에 해결가능하다. 이는 이 그래프에서 $|E|=O(|V|)$이기 때문이다.

격자의 가장자리에 위치한 모든 정점에 대해서 Dijstra의 알고리즘을 실행한 후, 한 가장자리 정점에서 맞은편 가장자리 정점으로의 최단 경로의 길이가 가장 짧은 두 정점을 찾으면 된다. 따라서 전체 시간 복잡도는 $O(|V|^2\log|V|)$을 넘지 않는다.

4. 배치 시뮬레이터 시스템의 설계와 기능

4.1 시스템 구성

그림 7은 제안하는 시스템을 구성하는 주요 구성요소를 보인 것으로 크게 뷰어(viewer), 배치엔진(DE: Deployment Engine), 네트워크 시뮬레이터(NS: Network Simulator), 그리고 기기정보 모델(Device Model)의 네 부분으로 이루어진다. 뷰어는

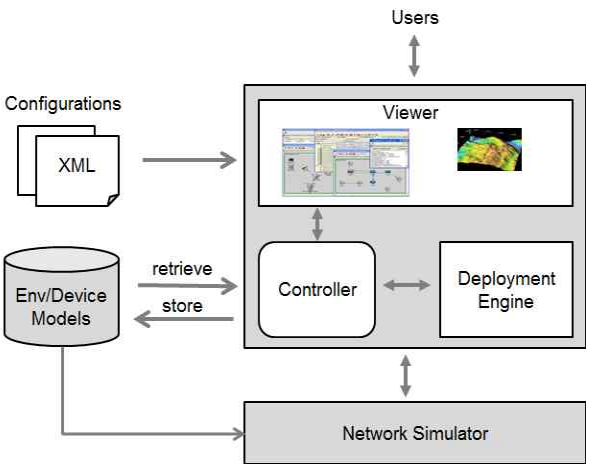


그림 7. 센서필드 시뮬레이터의 구조

사용자와의 상호작용을 위한 GUI 컨트롤들을 제공하고, 배치, 분석 및 시뮬레이션 결과를 시각화한다. 배치엔진은 주어진 매개변수들로부터 최적의 노드 배치를 계산한다. 네트워크 시뮬레이터는 노드들이 배치되면 네트워크 시뮬레이션을 제공한다. 기기정보 모델은 노드, 센서 등의 특성에 관한 모델링 정보를 저장한다.

본 연구에서 사용된 네트워크 시뮬레이터는 NICTA에 의해 개발된 네트워크 시뮬레이터인 Castalia를 기반으로 하였다. Castalia는 대표적인 네트워크 시뮬레이션 플랫폼이라고 할 수 있는 OMNeT++을 기반으로 하고 있으며 센서 네트워크에 적합하도록 개발된 시뮬레이터이다[16].

Castalia가 기본적으로 제공하는 Customizable Physical Channel 모듈은 단순한 기본 기능만을 지원한다. 다양한 센서의 유형과 다양한 탐지 대상의 유형 및 행동을 기술하기 위해서 기본 모듈을 바탕으로 제안 시스템의 용도에 적합하도록 수정하였다. 타겟의 이동 경로와 타겟이 발생시키는 소스 시그널 타입별로 시간적 변이를 기술하기 위해서 시나리오(scenario) 기반의 소스 시그널 타겟 기술 기법을 추가 확장하였으며 또한 지향성 센서를 표현할 수 있도록 확장하였다.

Castalia를 실행하기 위해서는 시뮬레이션 스크립트를 작성해야 한다. 시뮬레이션 스크립트는 시뮬레이션을 위한 각종 설정들을 정의하는 문서이다. 시뮬레이션 스크립트 내에는 노드의 개수와 위치, 이동성, 각 노드에 장착된 라디오의 유형과 설정값, 사용될 프로토콜 및 그 세부 설정 등 센서 네트워크를 정의하는 모든 정보들이 포함된다. 시뮬레이터는 기본적으로 시뮬레이션 스크립트를 읽고, 실행하여, 시뮬레이션 결과를 파일 형태로 출력하는 배치(batch) 방식으로 작동한다. 따라서 본 시스템과 Castalia 시뮬레이터와의 상호작용은 스크립트 파일 및 트레이스 파일을 기반으로 이루어진다.

4.2 시스템의 기능

본 절에서는 개발된 시스템이 사용자에게 제공하는 기능들을 센서 설정에서부터 필드 배치 시뮬레이션까지 차례로 설명한다.

4.2.1 센서 및 기기 정보의 등록

제안 시스템의 주요 기능은 노드들을 배치하고, 정량적 특성을 분석하며, 네트워크 시뮬레이션을 제공하는 것이다. 이러한 일들에 필요한 모든 모델링 정보들을 하나의 저장소(repository)에 저장하여 일목요연하게 사용자에게 제공한다. 또한 새로운 기기 및 모형을 추가하거나 수정할 수 있는 기능을 제공한다. 표 1은 저장소에 저장되는 기본 항목들의 구성을 보여주며, 그림 8은 사용자에게 제공되는 GUI의 예이다.

표 1. 저장소 구성 항목

모 듈	정 보
센서	종류, 감지 특성, 지향성 여부 및 감지 가능 각도 등
노드	장착한 센서의 유형과 개수
Radio 및 RF 채널	Radio의 특성 및 전파 모델 파라매터 등

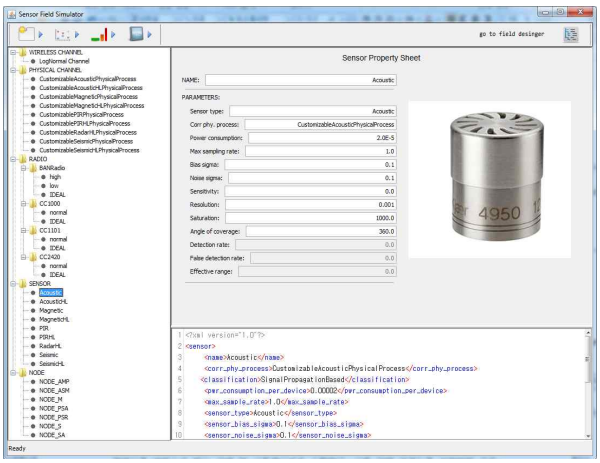


그림 8. 저장소 GUI의 예

4.2.2 필드 설정 기능

네트워크 시뮬레이션 및 필드설계의 사실성을 높이기 위해 실제 지도를 사용한다. 대부분의 기존 시스템이 센서 필드를 가상적인 정방형 공간으로 가정하는 반면, 제안 시스템은 실제 지형정보를 바탕으로 필드 설계 및 시뮬레이션을 수행한다. 위경도를 지정하면 구글 맵으로부터 필요한 지도 및 고도 정보를 읽어오는 방식을 지원하며 필드의 형태를 임의의 다각형으로 정의할 수 있도록 하였다.

4.2.3 노드의 배치 및 분석

3.1절에서 기술한 알고리즘에 따라 3각 격자, 4각

격자, 혹은 지정된 경로를 따르는 패턴 등 다양한 방법으로 노드를 배치하고, 감지율과 네트워크의 연결성을 분석하고, 필요에 따라 노드를 자동 혹은 수작업으로 재배치하는 기능을 제공한다. 노드가 배치된 상태에서 각 노드의 유형이나 각 노드에 장착된 센서를 변경하거나 편집하는 기능을 제공한다. 그림 9~11는 이러한 기능을 제공하는 GUI화면의 예들이다. 커버리지 분석 결과는 다양한 차트를 이용해서 가시화된다. 그림 12는 분석 결과를 차트를 이용해 가시화한 예이다.

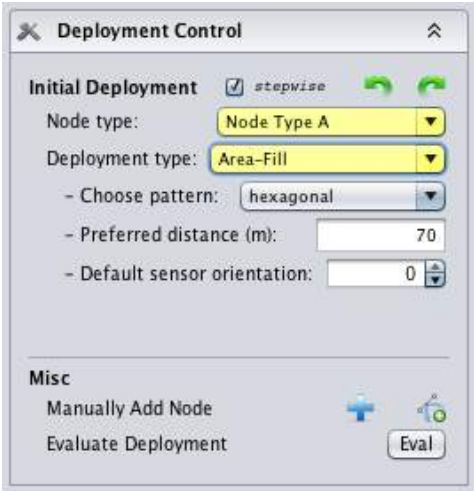


그림 9. 배치 설정 화면

그림 10. 경로 커버리지 설정 화면

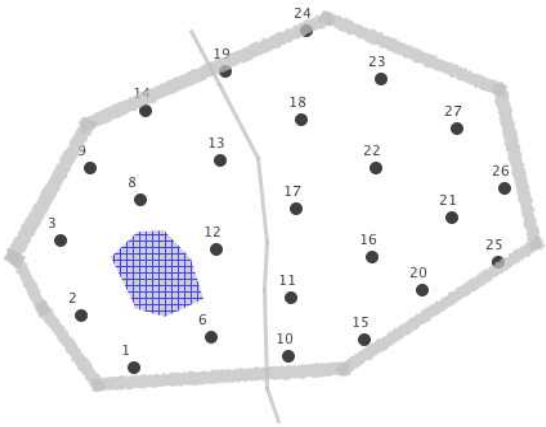


그림 11. 분석할 경로의 설정

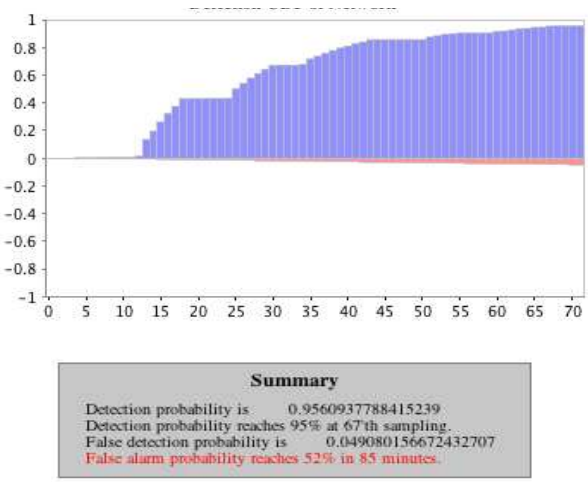


그림 12. 경로 커버리지 분석 결과의 시각화 예

4.2.4 네트워크 시뮬레이션

배치된 노드들에 대해서 네트워크 시뮬레이션을 실시하고 그 결과는 애니메이션의 형태로 시각화하는 기능을 제공한다. 차트, 이벤트 리스트, 그리고 지도 화면상의 애니메이션을 통해 시뮬레이션의 실행 과정을 시각화하여 제공한다.

5. 결 론

본 논문에서는 감시정찰 센서 네트워크에서의 노드 배치를 위한 시뮬레이터를 설계하고 개발하였다. 노드 배치와 관련된 연구동향을 살펴보았고, 노드 배치 문제를 과학적으로 표현하고 해결하기 위해 필요한 센싱 프로세스 모델링과 배치 전략들, 그리고 배치의 적절성을 평가하기 위한 다양한 방법들에 대해 기술하였다. 또한 본 연구에서 개발된 배치 엔진 및

시뮬레이터에 대해 기술하였다. 본 연구에서 개발된 시스템의 주요 특징은 다음과 같다. 첫째, 추후 필요한 기능이 추가될 수 있도록 확장가능한 구조를 추구하였다. 적용된 센서의 특성과 센싱 프로세스와 관련된 설정 값들은 모두 임의적인 것이며 객관적인 실험을 통해 검증되지 않았다. 이는 추후 지속적인 연구를 통해서 분석되어야할 과제이며, 그 결과들은 본 시스템의 수정 없이 적용 가능한 구조를 지향하였다. 둘째, 배치된 노드들에 대해 경로 커버리지 분석, 장벽 커버리지 분석 등의 다양한 분석과 평가가 가능하도록 하였다. 셋째, 유연하고 편리한 시각적 사용자 인터페이스를 제공함으로써 사용자의 편의성을 추구하였다.

본 연구의 한계점은 앞서 기술된 바와 같이 센싱 프로세스의 모델링이 실제 실험 데이터에 근거하여 기술되지 못한 점이다. 이것은 본 연구를 통해 개발된 시스템의 분석 결과가 신뢰성을 획득하는데 있어서의 부족함이 있음을 의미하며 추후 이에 대한 보다 깊은 연구가 진행될 예정이다.

참 고 문 헌

[1] P. Cai, L. Hou, Y. Shi, and Y. Shen, "Optimal Base- Station Locations in Two-Tiered Wireless Sensor Networks," *IEEE Transactions on Mobile Computing*, Vol. 4, No. 5, pp. 458-473, 2005.

[2] X. Cheng, D. Du, L. Wang, and L. Xu, "Relay Sensor Placement in Wireless Sensor Networks," *IEEE Transactions on Computers,* Vol. 56, No. 1, pp. 134-138, 2007.

[3] S.S. Dhillon and K. Chakrabarty, "Sensor Placement for Effective Coverage and Surveillance in Distributed Sensor Networks," *Proc. IEEE Wireless Communications and Networking Conference,* pp. 1609-1614, 2003.

[4] A. Efrat, S. Har-Peled, and J.S.B. Mitchell, "Approximation Algorithm for Two Optimal Location Problems in Sensor Networks," *Proc. the 3rd International Conference on Broadband Communications, Networks and Systems*, pp. 714-723, 2005.

[5] S.R. Grandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy Efficient Schemes for Wireless Sensor Networks with Multiple Mobile Base Stations," *Proc. the IEEE Globecom,* pp. 377-381, 2003.

[6] M. Younis and Kemal Akkaya, "Strategies and Techniques for Node Placement in Wireless Sensor Networks: A Survey," *Ad Hoc Networks*, Vol. 6, Issue 4, pp. 621-655, 2008.

[7] 이기욱, 성창규, "에너지 효율 제약조건을 가진 센서 네트워크 모니터링 시스템 구현," 멀티미디어학회논문지, 제13권, 제1호, pp. 10-16, 2010.

[8] Y. Zou and K. Chakrabarty, "Sensor Deployment and Target Localization based on Virtual Forces," *Proc. IEEE INFOCOM,* pp. 1293-1303, 2003.

[9] Y. Zou, K. Chakrabarty, "Sensor Deployment and Target Localization in Distributed Sensor Networks," *Transactions on IEEE Embedded Computing Systems,* Vol. 3, No. 1, pp. 61-91, 2004.

[10] S. Ferrari and Guoxian Zhang, "Probabilistic Track Coverage in Cooperative Sensor Networks," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 4, No. 6, pp. 1492-1504, 2010.

[11] S. Meguerdichian, F. Koushanfar, G. Qu, and M. Potkonjak, "Exposure in wireless ad-hoc sensor networks," *Proc. the 7th Annual International Conference on Mobile Computing and Networking, Mobicom'01,* pp. 139-150, 2001.

[12] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M.B Srivastava, Coverage Problems in Wireless Ad-Hoc Sensor Networks, *INFOCOM*, pp. 1380-1387, 2001.

[13] G. Veltri, Q. Huang, G. Qu, and M. Potkonjak, "Minimal and Maximal Exposure Path Algorithms for Wireless Embedded Sensor Networks," *Proc. the 1st International*

Conference on Embedded Networked Sensor Systems, pp. 40-50. 2003.

[14] A. Howard and M. Mataric, "G. Sukhatme, Mobile Sensor Network Deployment using Potential Fields: A Distributed Scalable Solution to the Area Coverage Problem," *Proc. the 6th International Symposium on Distributed Autonomous Robotic Systems,* pp. 299-308, 2002.

[15] T. Cox and M. Cox, *Multidimensional Scaling*, 2nd Edition, Chapman & Hall, Boca Raton, 2001.

[16] A. Boulis, *Castalia: A Simulator for Wireless Sensor Networks and Body Area Networks*, NICTA, v.2.3, 2011.

권 오 흠

1988년 서울대학교 컴퓨터공학과 졸업
1991년 KAIST 전산학과 졸업 (공학석사)
1996년 KAIST 전산학과 졸업 (공학박사)
1997년~현재 부경대학교 교수
관심분야 : 알고리즘 설계 및 분석, 무선 센서 네트워크

송 하 주

1993년 서울대학교 컴퓨터공학과 졸업
1995년 서울대학교 대학원 컴퓨터공학과 졸업 (공학석사)
2001년 서울대학교 대학원 전기컴퓨터공학부 졸업 (공학박사)
2003년 8월 ㈜아이티포웹 부장
2003년 9월~현재 부경대학교 교수
관심분야 : 데이터베이스 시스템, RFID/USN