

# 장애물이 없는 3차원 네트워크 공간에서 통신 범위 내에 무선 링크가 가능한 블록 기반의 균등 분포 무작위 노드 배치 방법

임동현<sup>†</sup>, 김창화<sup>\*\*</sup>

## A Block-based Uniformly Distributed Random Node Arrangement Method Enabling to Wirelessly Link Neighbor Nodes within the Communication Range in Free 3-Dimensional Network Spaces

DongHyun Lim<sup>†</sup>, Changhwa Kim<sup>\*\*</sup>

### ABSTRACT

The 2-dimensional arrangement method of nodes has been used in most of RF (Radio Frequency) based communication network simulations. However, this method is not useful for the an none-obstacle 3-dimensional space networks in which the propagation delay speed in communication is very slow and, moreover, the values of performance factors such as the communication speed and the error rate change on the depth of node. Such a typical example is an underwater communication network. The 2-dimensional arrangement method is also not useful for the RF based network like some WSNs (Wireless Sensor Networks), IBSs (Intelligent Building Systems), or smart homes, in which the distance between nodes is short or some of nodes can be arranged overlapping with their different heights in similar planar location. In such cases, the 2-dimensional network simulation results are highly inaccurate and unbelievable so that they lead to user's erroneous predictions and judgments. For these reasons, in this paper, we propose a method to place uniformly and randomly communication nodes in 3-dimensional network space, making the wireless link with neighbor node possible. In this method, based on the communication range of the node, blocks are generated to construct the 3-dimensional network and a node per one block is generated and placed within a block area. In this paper, we also introduce an algorithm based on this method and we show the performance results and evaluations on the average time in a node generation and arrangement, and the arrangement time and scatter-plotted visualization time of all nodes according to the number of them. In addition, comparison with previous studies is conducted. As a result of evaluating the performance of the algorithm, it was found that the processing time of the algorithm was proportional to the number of nodes to be created, and the average generation time of one node was between 0.238 and 0.28 us. ultimately, There is no problem even if a simulation network with a large number of nodes is created, so it can be sufficiently introduced at the time of simulation.

**Key words:** 3-Dimensional Network, 3-Dimensional Node Allocation, Block-Base Node Allocation, Node Allocation Area

---

\* Corresponding Author: Changhwa Kim, Address: (26403) 150, Namwon-ro, Heungeop-myeon, Wonju City, Gangwon-do, Republic of Korea, TEL: +82-33-760-8663, FAX: +82-33-760-8661, E-mail: kch@gwnu.ac.kr  
Receipt date: Jul. 18, 2022, Revision date: Sep. 19, 2022  
Approval date: Sep. 20, 2022

---

<sup>†</sup> Dept. of Computer Science&Engineering, Gangneung-Wonju National University  
(E-mail: yeklys@naver.com)

<sup>\*\*</sup> Dept. of Computer Science&Engineering, Gangneung-Wonju National University

## 1. 서론

전파 통신은 네트워크 공간에서 사용자가 원하는 방향과 세기로 전파를 자유롭게 전송할 수 있으며, 전파 속도가 빛의 속도와 같아 매우 빠른 속도로 원하는 메시지를 전달할 수 있다. 이러한 편의성과 속도 덕분에 전파 통신은 통신 기술에서 가장 많은 부분을 차지하게 되었으며 현재 인류가 사용하고 있는 통신 방식의 대부분은 이 전파 통신으로 이루어지고 있다. 전파 통신은 주파수에 따라 수 미터에서 수만 킬로미터에 이르기까지 통신할 수 있어 매우 짧은 거리부터 굉장히 먼 거리까지 매우 빠르게 통신할 수 있다[1-3]. 이러한 전파 특징 때문에 일반적인 전파 통신 시뮬레이션의 경우, 대부분 물리적 네트워크 공간을 2차원 평면으로 고려하고 이 환경에 노드들을 배치하여 실행하는 시뮬레이션으로도 충분한 신뢰도를 얻을 수 있었다. 하지만 통신 거리가 짧거나 통신속도가 느린 환경에서 두 노드 사이의 높이 차이가 무시된 상태로 시뮬레이션을 진행하는 경우 그 결과에 큰 영향을 미칠 수 있다.

WSN (Wireless Sensor Network), IBS (Intelligent Building System) 혹은 스마트홈과 같은 시스템에 구축되는 네트워크에서 저가형 통신 모듈이 주가 되어 노드 간의 통신 거리가 제한적인 경우가 있는데, 이러한 경우 노드들이 설치된 높이차로 인해 2차원 노드 배치 기반의 시뮬레이션을 진행하는 경우 실제 환경과 크게 다른 결과 나타날 수 있다[4-5].

특히, 수중 통신의 경우에 수중 노드는 수표면에 위치하는 해상 부이나 선박과 같은 해상 노드, 수중에서 이동하는 AUV, 앵커를 이용해 수중에서 적당한 깊이에 위치시킨 고정된 노드 등 다양한 유형의 노드들이 존재하는데, 해상 노드 이외의 수중 노드는 같은 유형의 노드라 하더라도 그 깊이가 대부분 서로 다르다[6-7]. 이러한 환경에서 수중 음파 통신은 초당 약 1.5 km 내외의 매우 느린 전파 지연(propagation delay)을 가지며, 이 속도와 에러율은 시시각각으로 변하는 수온과 염도 및 기타 수중 환경에 따라 달라지는 특성이 있다[8]. 따라서, 수중 음파 통신은 수신 수중 노드의 깊이에 따라 송신 노드와의 물리적 거리 차이로 인해 전파 지연이 달라지고 게다가 수중 환경 또한 다를 수 있어 전파 지연 속도와 에러율 또한 다르게 되므로 수중 통신 네트워크 시뮬레이션

은 2차원이 아닌 3차원 노드 배치를 기반으로 하는 시뮬레이션이 필요하다.

이에 본 논문에서는 노드의 3차원 공간 배치를 기반으로 하는 시뮬레이션이 필요한 경우에 사용할 수 있도록 하기 위한 목적으로 장애물이 없는 3차원 네트워크 공간에서 통신 범위 내에 이웃 노드와 무선 링크가 가능하도록 노드를 무작위로 골고루 배치하는 방법을 제안한다.

본 논문은 2절에서 관련 연구를 살펴보고 3절에서 노드 배치 방법에 대한 개요 설명 후 4절에서 알고리즘에 사용되는 개념인 블록의 구조를 소개한다. 다음으로 5절에서는 블록의 생성 방법을 제안한 다음, 6절에서 생성된 블록을 가지고 무작위로 노드를 배치하는 알고리즘을 제안한다. 7절에서는 제안한 알고리즘이 반복적인 시뮬레이션 시에도 사용하기 적합한지 알아보기 위해 직접 구현한 후 처리시간을 측정해보았으며 마지막으로 8절에서 결론을 제시한다.

## 2. 관련 연구

기존의 많은 네트워크 시뮬레이터와 특히, WSN 연구들에서는 ns-2, TOSSIM 등의 범용 네트워크 시뮬레이터 툴을 사용하여 개발하려는 프로토콜 혹은 알고리즘 기술을 평가하거나 혹은 직접 시뮬레이션 환경을 마이크로프로세서를 사용하여 구축하고 평가해왔다. [9]에서는 큐롬 기반 이웃 노드 탐색 프로토콜 스케줄링 기법을 TOSSIM 시뮬레이션 툴을 이용하여 100\*100 m 실험 환경에서 50개의 센서 노드를 임의로 배치한 랜덤 토폴로지를 구성하여 평가하였고, [10]에서는 TOSSIM을 최대 100 × 100 m 크기의 네트워크에서 100개의 노드를 3가지의 서로 다른 유형의 노드 배치 방법으로 배치하여 에너지 효율적인 라우팅 프로토콜을 평가하였다. ns-2를 사용한 연구로는 [11]에서 WSN에서 지역-중앙 클러스터 라우팅 방법을 제안하였으며 이를 100 × 100 m 크기의 네트워크에서 100개의 노드를 사용하여 시뮬레이션하였다. 위에서 언급한 기존 연구들과 대부분의 많은 전파 통신 기반 연구들은 2차원 노드 배치 기반의 통신 및 네트워크 시뮬레이션을 진행해 왔다. 서론에서도 언급한 것처럼 송수신 노드의 높이나 깊이 차이로 인해 통신 거리와 통신 환경이 달라짐에 따라 통신속도와 에러율과 같은 통신 네트워크의 주

요 성능 요인에 영향을 끼치는 통신 네트워크 환경에서는 이러한 2차원 노드 배치 기반의 시뮬레이션으로는 성능측정 결과의 정확도와 신뢰도를 기대하기 어렵다.

3차원 네트워크에서 노드 무작위 배치 방법에 관한 연구는 그 필요성에도 불구하고 극소수의 연구만이 진행되어온 상태이다.

극소수의 연구 중 하나로서 [12]에서는 노드 사이의 거리 측정 방법 및 장치와 함께 저차원 공간에 배치된 네트워크를 고차원 공간으로 재배치하는 방법이 제안되어 있다. 해당 방법을 사용하여 노드와 노드 사이의 거리를 보다 효과적으로 측정할 수 있지만, 해당 방법을 사용하기 위해서는 이미 저차원 공간에 배치된 네트워크가 필요하며 통신 기술을 시뮬레이션하기 위한 무작위 네트워크를 만들기 위한 노드 배치 알고리즘을 제안하는 본 연구와는 목적과 적용 분야가 다르다. [13]에서는 감시 정찰 센서 네트워크 시스템에서 사용할 수 있는 3차원 공간에서 노드 배치 장치와 방법을 제안하고 있다. 해당 기술은 센서의 탐지거리와 통신 거리를 비롯하여 식생, 기상, 노드 배치 환경과 같은 네트워크 환경들을 가중치로 만들어 노드를 배치하는 방법을 제안하여 실제 환경과 유사한 시뮬레이션이 가능한 네트워크를 생성할 수 있다. 하지만 해당 기술은 특정 센서를 이용하는 감시 정찰 센서 네트워크에만 적용되며 지표면 노드와 지상 노드로 구성된 2계층 네트워크에만 적용되어 아주 제한적인 네트워크에서만 사용할 수 있다는 문제점이 있다.

[14]에서는 수중이나 우주통신과 같은 넓은 공간에서의 통신은 3차원적인 해결 시도가 필요함을 언급하며 공간 내의 노드들의 최단연결을 위한 3차원 유클리드 최소신장 트리 제안하고 있다. 해당 연구에서는 3차원 통신 기술 검증에 위해 단순히 생성될 수 있는  $x, y$  축의 범위를 정하고  $x, y$  축의 범위의 일정 비율로 계산된 범위 내에 무작위로 노드를 배치하는 방법을 사용하였다. 해당 방식은 정해진 알고리즘 없이 정해진 범위 내에 노드를 무작위로 배치하기 때문에 통신 거리 밖에 노드가 생성되어 연결이 안되는 노드가 생성되거나 특정 범위 내에 노드가 밀집되어 생성될 수 있다는 문제점이 있다.

[15]에서는 셀룰러 네트워크에서는 점점 셀의 크기를 줄여가며 나노 혹은 피코 셀룰러 네트워크를

사용해야 할 필요성을 언급하고 있다. 이런 작은 셀로 이루어진 네트워크는  $x, y$  축에 더하여 높이,  $z$  축을 포함한 3차원의 관점에서의 문제 해결이 필요함을 제시하며 이를 해결하기 위해 육각형 매쉬 레이어를 기반으로한 3차원 피코 셀룰러 네트워크에 제안하고 있다.

[16]에서는 3차원 네트워크에서 연결성과 커버리지 문제를 해결하는 방법을 제안하고 있다. 해당 연구에서는 노드의 탐지 범위와 통신 범위로 만들어진 셀을 이용하여 최소한의 노드 수로 3차원 공간 내에 최대의 커버리지와 연결성을 확보하는 방법을 제시하고 있으며 이때 노드가 항상 고정된 위치에 배치되어야 한다는 문제점을 무작위로 노드를 배치하는 방법으로 해결하고 있다. 해당 방법은 무작위로  $k$  개의 노드를 배치하였을 때  $k$  개의 노드들의 탐지 범위로 이루어진  $k$ -커버리지를 제안하고  $k$ -coverage를 통해 3차원 네트워크의 연결성과 커버리지 문제를 해결하는 방법을 제시하고 있다. 하지만 해당 연구에서는 시뮬레이션용 네트워크를 만드는 것이 아닌 WSN에서 최대의 커버리지와 연결성을 확보하는 네트워크를 만드는 것이 목적으로  $k$ -coverage에 대한 계산식만 제시되어 있을 뿐 노드의 무작위 배치 방법에 대한 절차 및 방법은 기술되어 있지 않다.

### 3. 노드 배치 방법 개요

시뮬레이션을 위한 네트워크를 생성할 때 노드들은 적절한 거리를 두고 생성되어야 한다. 특히, 노드들이 생성될 때 너무 거리가 멀면 통신 거리보다 먼 거리를 갖는 노드들이 생성되어 통신이 불가능한 네트워크가 생성될 수 있기 때문이다. 반대로 거리가 너무 짧은 경우 또한 문제가 발생할 수 있다. 네트워크 내의 모든 노드가 근접하여 네트워크의 모든 공간이 노드로 가득 차거나 특정 구역에만 노드가 몰려 생성되어 실제 환경과 동떨어지거나 노드 거리가 지나치게 짧게 되어 설치, 운영, 관리 및 유지보수에서 비용상 비효율적인 네트워크가 생성될 수 있기 때문이다.

이 때문에 시뮬레이션을 위한 시뮬레이션 네트워크의 배치는 한 노드의 통신 범위 내에 최소한 한 개 이상의 노드가 존재하여 링크가 가능할 수 있어야 하고 네트워크 범위 내에서 균등하게 분배되어 있어

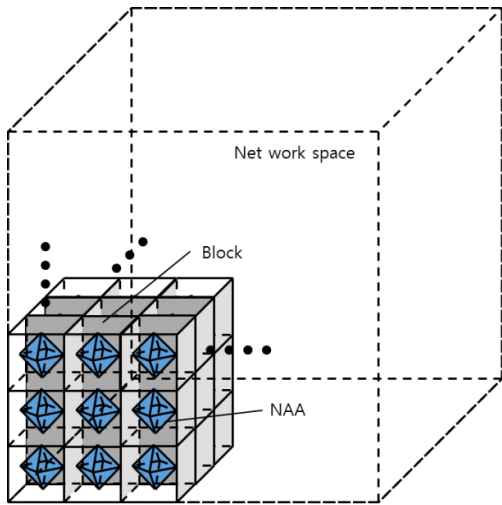


Fig. 1. Node placement using blocks.

야 한다.

이와 같은 문제를 해결하기 위해 본 연구에서 제시하는 노드 배치 알고리즘은 블록이란 개념을 사용해 해결하는 방법을 제안한다. 3차원의 물리적 네트워크 공간은 통신 가능 거리를 기반으로 계산된 블록들로 Fig. 1과 같이 구성한다. 이 그림에서 8면체(NAA라 함)를 포함하는 작은 정육면체들은 각각 블록을 나타내며 Fig. 1은 이 블록들로 3차원 네트워크 공간을 채워가는 모습을 보여준다. 각 블록은 내부에 작은 8면체(NAA)를 포함하는데, 이 8면체(NAA)는 일정한 통신 거리 이상을 유지하면서 통신 범위 내의 노드가 배치되도록 도입한 가상입체이다. 노드 배치 절차를 요약하여 설명하면 다음과 같다:

단계 1. 통신이 가능한 거리를 고려하여 배치하는 블록을 생성한 후,

단계 2. 그 블록의 8면체(NAA)를 이용하여 이웃 블록에 존재할 수 있는 노드와 통신 거리 범위 내에 있도록 노드 한 개를 이 블록 내부에 무작위 위치로 생성하고 배치한 다음,

단계 3. 이 블록을 네트워크가 가득 찰 때까지 채워 넣는다.

블록은 네트워크 내에서 노드 한 개가 생성될 수 있는 사실상, 가상 영역으로 한 블록 내에는 오로지 한 개의 노드만 존재할 수 있다. 이로 인해 노드들은 네트워크의 한 영역에 몰려 생성될 수 없으며 적당한 거리 범위를 유지하면서 생성될 수 있다. 블록 구조와 생성 그리고 배치에 관한 상세한 방법과 절차에 대해서는 4절, 5절, 6절에서 각각 소개된다.

#### 4. 블록 구조

블록의 구조는 Fig. 2와 같이 두 영역으로 구성된다. Fig. 2 (a)의 블록은 제한하려는 노드 배치 알고리즘에서 시뮬레이션을 위한 3차원 네트워크 공간을 구성하기 위한 영역으로 Fig. 2 (b)에서 보이는 노드 할당 영역인 NAA (Node Allocation Area)와 Fig. 2 (c)에서 보이는 빈 영역으로 구성된다. 블록의 크기는 네트워크 내에서 사용되는 통신 매체의 통신 거리와 노드가 생성될 수 있는 영역인 NAA의 범위에 의해 결정된다. Fig. 2 (a), (b), (c)에서 위에 있는 그림들은 블록, NAA, 빈 영역을 모두 3차원으로 표현한 것이고 아래 그림들은 이들을 평면(plane)으로 각각 사상(projection)한 것이다.

NAA는 블록 영역 내에서 노드가 생성될 수 있는 영역이다. 블록이 노드를 생성할 때 노드가 항상 중앙에 생성되면 매번 같은 네트워크를 생성하게 되고,

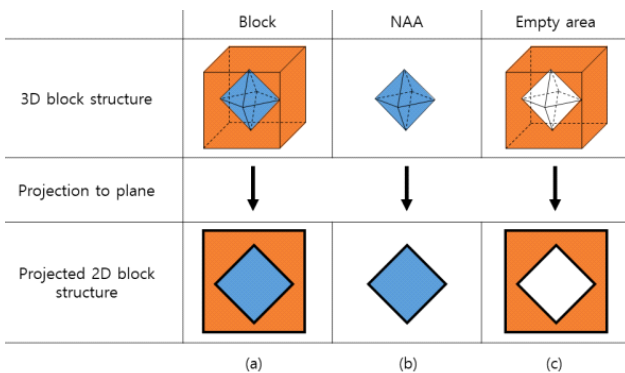


Fig. 2. 3D block structure and its X-Y, Y-Z, and Z-X plane projections. (a) 3D block structure (the top) and its plane projection (the bottom), (b) 3D NAA(Node Allocate Area) (the top) and its plane projection (the bottom), and (c) 3D empty area (the top) and its plane projection (the bottom).

이를 방지하기 위해 블록 내에 무작위로 생성하게 된다면 두 블록이 맞닿는 지점에 노드들이 생성되는 경우가 발생할 수 있다. 이를 해결하기 위해 블록 내에 노드가 생성될 수 있는 영역을 제한하여 생성되는 시뮬레이션 네트워크가 무작위성을 가지게 하며 균등하게 노드가 분포될 수 있도록 한다. NAA의 크기는 네트워크의 시뮬레이션 환경에 따라 사용자가 조절한다.

빈 영역은 블록 영역에서 노드 영역을 제외한 공간으로 노드가 생성될 수 없는 영역이다. 이 빈 영역은 노드가 생성되지 않기 때문에 블록과 블록 사이에 생성된 노드들이 빈 영역만큼의 거리를 유지할 수 있도록 한다. 이로 인해 근접한 모든 블록 영역에 생성된 노드들이 최대통신 거리 내에 배치되고 링크를 만들 수 있게 된다.

5. 블록 생성 절차

블록의 생성 절차는 사용자가 시뮬레이션 네트워크의 특성에 따라 적절한 NAA의 크기를 정하면서 시작된다.

NAA는 Fig. 3의 왼쪽 그림과 같이 정팔면체의 형태로 형성된다. NAA를 블록의 정 중앙에 위치시키고 X-Y, Y-Z, Z-X 평면 중 어떤 평면으로든 사상시켰을 때 정팔면체이기 때문에 Fig. 3의 오른쪽 그림처럼 같은 크기의 정사각형으로 동일하게 사상되는데, NAA의 크기는 이 정사각형의 대각선 길이에 의해 결정된다(사실상, 이 대각선의 길이는 NAA를 높이에 해당하며, 따라서, NAA의 크기는 NAA의 높이이다).

NAA의 크기가 결정되면 블록 영역의 크기를 결정한다. 블록 영역의 크기를 구하기 위해선 블록의 한 변의 길이를 구해야 한다. 우선 Fig. 4와 같은 NAA의 크기가 하나의 노드만 포함하는 네트워크를

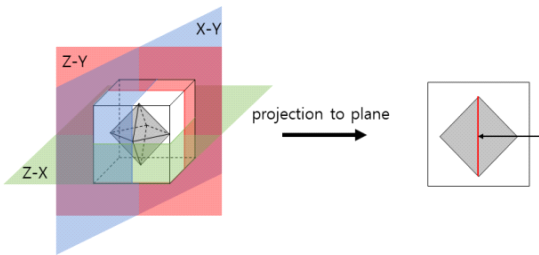


Fig. 3. NAA structure and its plane projection,

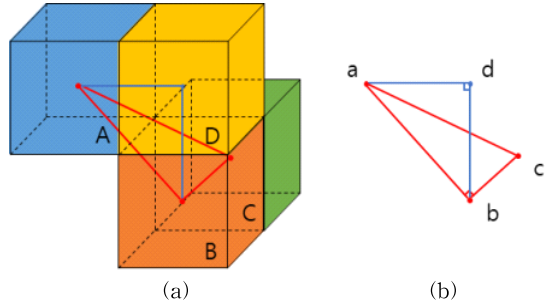


Fig. 4. Node placements of the neighbor blocks of the block A, where NAA includes only one node. (a) neighbor blocks of the block A and (b) the centers, a, b, c, and d, of block A, B, C, and D.

가정해보자. Fig. 4에서 (a)는 네트워크의 3차원 블록과 노드들을 나타낸 그림이고 (b)는 이해를 돕기 위해 해당 네트워크에 생성·배치된 노드들만을 그린 그림이다. 이 네트워크에 블록 A와 A의 오른쪽 아래에 이웃한 블록 B, A와 대각선 위치에 있는 C 그리고 A의 우측에 이웃한 블록 D가 존재하고 각 블록의 중심에 NAA가 하나의 노드만 포함하므로 노드 a, b, c, d가 생성되었다고 하자. Fig. 4의 (b)와 같이 이 a, b, c를 3점으로 하는 평면  $\triangle abc$ 와  $\triangle abd$ 를 그려 보면 서로 한 면을 맞대고 있는 블록 B와 C의 중점을 이은  $\overline{bc}$ 를 볼 수 있다. 이  $\overline{bc}$ 는 블록의 한 변의 길이와 같으므로  $\overline{bc}$ 의 길이를 구한다면 블록 영역의 크기를 구할 수 있다.

블록 A에서 가장 먼 노드는 대각선 위치에 있는 C이다. 따라서 이들의 중점인 노드 a와 c 사이의 거리  $\overline{ac}$ 가 통신 매체의 최대통신 거리가 되어야 한다 ( $\overline{ac}$ 는 6절에서 제안한 3D\_Node\_Allocation 알고리즘의 입력 파라미터로서 시뮬레이션 사용자가 제시한다.).

평면  $\triangle abc$ 는 직각삼각형이므로  $\overline{ab}$ 의 길이를 안다면  $\overline{ac}$ 의 길이를 구할 수 있다.  $\overline{ab}$ 의 길이는 평면  $\triangle abd$ 를 이용하면  $\overline{bc} = \overline{ad} = \overline{bd}$ 이므로 피타고라스의 정리에 따라 다음과 같은 식 3을 구할 수 있다.

$$\overline{ab} = \sqrt{2} \cdot \overline{bc} \quad (\because \overline{ab}^2 = \overline{ad}^2 + \overline{bd}^2 = 2\overline{bc}^2) \quad (1)$$

식 1을 이용하면 다음과 같은 식 2를 통해  $\overline{bc}$ 의 길이를 유도할 수 있다.

$$\overline{ac} = \sqrt{\overline{ab}^2 + \overline{bc}^2} = \sqrt{(\sqrt{2} \cdot \overline{bc})^2 + \overline{bc}^2} = \sqrt{3} \cdot \overline{bc} \quad (2)$$

$$\therefore \frac{\overline{ac}}{\sqrt{3}} = \overline{bc}$$

이때  $\overline{ac}$ 는 통신 매체의 최대통신 거리  $\overline{bc}$ 는 블록의 한 변의 길이이므로 블록의 크기는 다음과 같은 식 3을 통해 구할 수 있다.

$$\left(\frac{\text{통신 매체의 최대통신거리}}{\sqrt{3}}\right)^3 = \text{블록의 한변의 길이}^3 \quad (3)$$

$$= \text{블록의 크기}$$

이번엔 NAA의 크기가 두 개 이상의 노드가 존재 가능한 경우를 가정해보자. Fig. 5는 대각선상에 존재하는 세 블록 A, B에 대한 그림으로 Fig. 5의 (a)는 이웃한 세 블록 A, B, C를 3차원 공간에 나타낸 그림이고 점 a, b, c의 위치에 노드가 생성되었다고 가정한다.  $b'$ ,  $c'$ 는 각각 블록 B와 C의 중점이다. Fig. 5의 (b)는 B와 C 블록을 위에서 바라본 그림이다. Fig. 5의 (c)는  $\overline{b'c'}$ 를 구하는 과정의 이해를 돕기 위한 그림이다. 앞의 예시와 마찬가지로  $b'$ 과  $c'$  사이의 거리는 블록의 한 변의 길이와 같으므로  $\overline{b'c'}$ 의 길이를 구한다면 블록의 크기를 구할 수 있다.

(a)와 같이 A, B 두 블록만 있는 상황에서 두 블록 사이에 가장 멀리 노드가 생성되는 경우는 그림의 a와 b의 위치에 노드가 생성되는 경우라고 할 수 있다. 따라서  $\overline{ac}$ 의 길이가 최대통신 거리가 되어야 하며 식 2에 따라  $\overline{bc}$ 의 길이는  $\overline{bc} = \frac{\overline{ac}}{\sqrt{3}}$ 가 된다.

이때 (b)와 같이  $b'$ 과  $c'$ 에서  $\overline{bc}$ 에 직각이 되는 선을 그었을 때 이들이 만나는 점을 각각  $b''$ 과  $c''$ 이라고 하자.  $\overline{bb''}$ 과  $\overline{cc''}$ 의 길이 만큼을 뺀  $\overline{b''c''}$ 의 길이를 구하면  $\overline{b'c'}$ 의 길이를 구할 수 있다.

Fig. 5의 (c)와 같이 B블록의 NAA에 중점  $b'$ , NAA의 아래 모서리인 h,  $\overline{b'j}$  중점인 j로 만들어진 평면  $\Delta b'hj$ 를 고려하자. 이때 점 b는  $\overline{b'j}$ 의 중점에 놓이게 되며 이 점 b로부터  $\overline{b'j}$ 로 직각이 되도록 직선을 그었을 때 만나는 점 m을 구할 수 있다. 그리고 다시 이 점 m을  $\overline{b'i}$ 에 직각이 되도록 직선을 그었을 때 만나는 점인 n과 선분  $\overline{mn}$ 을 구할 수 있다. 이  $\overline{mn}$ 의 길이는  $\overline{bb''}$ ,  $\overline{cc''}$ 와 같으므로  $\overline{mn}$ 의 길이를 구한다면 블록의 한 변의 길이인  $\overline{b'c'}$ 를 계산할 수 있다.

NAA의 크기를  $\overline{NAA}$ 라 할 때  $\overline{b'i}$ 의 길이는  $\frac{\overline{NAA}}{2}$ 이고 평면  $\Delta b'jl$ 은 직각 이등변 삼각형이므로 피타고라스의 정리에 의해  $\overline{b'j}$ 의 길이는  $\frac{\overline{NAA}}{2\sqrt{2}}$ 가 된다.

이때 m은  $\overline{b'j}$ 의 중점이므로  $\overline{b'm}$ 의 길이는  $\frac{\overline{NAA}}{4\sqrt{2}}$ 가 된다. 이때 평면  $\Delta b'mn$ 은 직각 이등변 삼각형이므로  $\overline{mn}$ 의 길이는  $\frac{\overline{NAA}}{8}$ 가 된다.

따라서 블록의 한 변의 길이인  $\overline{b'c'}$ 는 다음과 같은 식 4를 통해 구할 수 있다.

$$\overline{b'c'} = \overline{bc} - \overline{bb''} - \overline{cc''} = \overline{b'c'} = \overline{bc} - 2\overline{mn}$$

$$= \overline{b'c'} = \frac{\overline{ac}}{\sqrt{3}} - \frac{\overline{NAA}}{4} \quad (4)$$

$$\therefore \overline{b'c'} = \frac{\text{통신 매체의 최대통신거리}}{\sqrt{3}} - \frac{\overline{NAA}}{4}$$

### 6. 노드 배치 알고리즘

노드 배치 알고리즘은 Algorithm. 1과 같은 절차에 따라 생성된다. 우선 알고리즘은 실행하기 위해

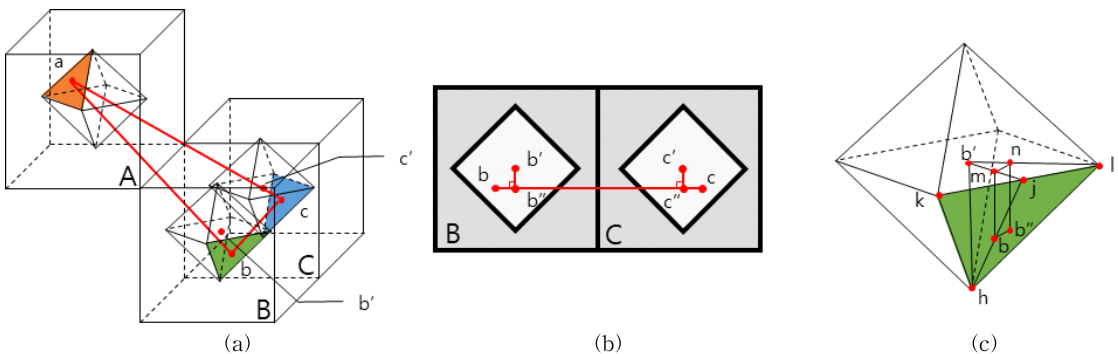


Fig. 5. Node placements of the Neighboring blocks A and B, without NAA. (a) Neighboring blocks A and B, (b) top view of blocks b and c, and (c) points to calculate  $\overline{mn}$ , a, a', b, c, d and e, of block A's NAA.

네트워크의 정보를 매개 변수로 입력받는다. 이때 네트워크의 정보에는 네트워크의 크기(Net\_side), 매체의 최대통신 거리(Media\_range), NAA\_Size가 포함되어야 한다. 그 후 알고리즘의 2번 라인과 같이 식 4를 이용하여 블록의 한 변의 크기를 계산한다. 그 후 x, y, z축으로 현재 블록의 위치를 비교하여 네트워크의 끝에 도달할 때까지 노드를 생성하게 된다. 노드 생성 절차는 알고리즘의 6~22번 라인과 같이 진행된다. 노드는 x, y, z 순서로 좌표를 생성하여 생성하며, 알고리즘의 6번 라인과 같이 NAA 내에서 x 좌표를 생성한다. 그 후 알고리즘의 7~11번 라인과 같이 생성된 x 좌표에서 생성 가능한 y 값의 범위를 계산한다. 다음으로 알고리즘의 12번 라인과 같이 y 좌표를 생성한다. 다음으로 알고리즘의 13~17번 라인과 같이 y와 같은 방식으로 z 좌표를 생성한다.

마지막으로 생성된 노드의 (x, y, z) 좌표는 알고리즘의 19~21번 라인과 같이 현재 블록의 원점 좌표인 (bl\_x, bl\_y, bl\_z)와 더해져 네트워크에서 노드의 위치가 결정된다.

## 7. 알고리즘 구현과 성능평가

이 절에서는 3차원 공간에서 통신 범위 내에 링크가 가능한 균등 분포를 갖는 무작위 노드 배치 방법에 대한 6절의 알고리즘을 구현하고 구현된 코드를 중심으로 3차원 네트워크 크기와 통신 범위에 따라 측정된 성능을 제시한다.

### 7.1 구현과 성능평가 프로세스

6절에서 제시한 3D\_Node\_Allocation 알고리즘의

Algorithm 1. 3D Node Allocation Algorithm.

---

```

1 3D_Node_Allocation(Net_side, Media_range, NAA_Size)
2  Block_side ←  $\frac{1}{\sqrt{3}} * \text{Media\_range} - \frac{\text{NAA}}{4}$ 
3  For bl_x ← 0 to Net_side
4      For bl_x ← 0 to Net_side
5          For bl_x ← 0 to Net_side
6              x ← rand(-(NAA / 2), NAA / 2)
7              If x > 0 then
8                  range_of_y ← NAA - x
9              Else
10                 range_of_y ← NAA + x
11             End If
12             y ← rand(-range_of_y, range_of_y)
13             If y > 0 then
14                 range_of_z ← range_of_y - y
15             Else
16                 range_of_z ← range_of_y + y
17             End If
18             z ← rand(-range_of_z, range_of_z)
19             x ← bl_x + (Block_side / 2) + x
20             y ← bl_y + (Block_side / 2) + y
21             z ← bl_z + (Block_side / 2) + z
22             bl_z ← bl_z + Block_side
23         End for
24         bl_y ← bl_y + Block_side
25     End If
26     bl_x ← bl_x + Block_side
27 End If

```

---

\* Net\_side (length of one side of the network), Media\_range (Node's communication range), NAA (Node allocate area range)

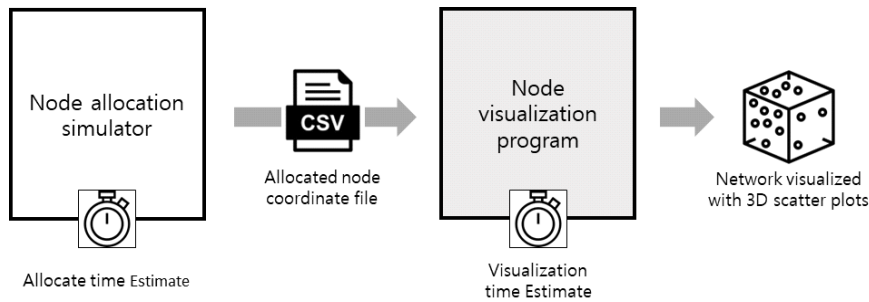


Fig. 6. Process for the implementation and performance evaluation of 3D Node Allocation algorithm.

구현과 성능평가는 Fig. 6과 같은 절차로 진행된다. 우선 C 언어로 구현된 노드 배치 시뮬레이터에서 노드를 생성한 후 생성된 노드들의 좌표를 CSV(Comma Separation Value) 파일로 저장하였다. 이렇게 생성된 CSV 파일은 python으로 구현한 네트워크 시각화 프로그램을 통해 생성된 노드 위치를 3D 산점도로 표현하였다.

노드를 생성하여 3차원으로 배치하고 CSV 파일에 저장하는 과정에서 통신 범위와 네트워크의 크기에 따라 노드 한 개의 위치를 생성·배치하는데 소요(처리)되는 평균 시간과 이 시간과 함께 노드 생성 위치를 CSV 파일에 저장하는 시간을 포함하여 전체 시간을 측정했다. 또한, 노드의 개수에 따라 CSV에 저장된 노드의 위치 데이터를 가시화하는데 걸리는 시간을 측정했다.

### 7.2 노드 생성·배치 평균 시간

네트워크의 크기와 통신 범위가 3차원 노드 배치 알고리즘의 노드 생성·배치 시간에 어떤 영향을 주는지를 알아보기 위해 네트워크의 크기와 통신 범위에 따른 노드 생성·배치에 대한 평균 시간을 측정했다.

Fig. 7의 (a)와 (b)는 각각 통신 범위와 네트워크 범위에 따른 노드 한 개의 생성·배치에 소요된 평균 처리시간으로 통신 범위와 네트워크 크기와는 무관하게 모두 0.238~0.28 us 사이의 시간이 소요되는 것으로 나타났다.

Fig. 8은 노드 개수에 따른 전체 노드의 평균 생성 시간을 그래프로 나타낸 것으로 CASE 1은 노드들의 생성·배치와 CSV 파일에 노드의 좌표를 저장하는 시간을 측정한 평균 시간을, CASE 2는 노드의 생성·배치만 측정한 평균 시간을 각각 나타낸다. CASE

1과 CASE 2에서 두 그래프 간의 간격은 노드의 개수가 주어질 때 파일에 노드의 좌표를 CSV 파일에 저장하는 시간을 의미한다고 볼 수 있다.

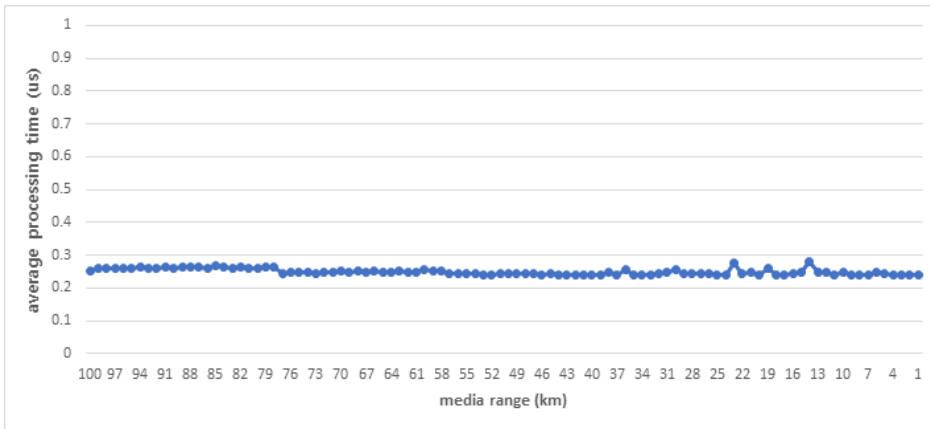
측정 결과로서 Fig. 8은 노드 개수의 증가에 따라 노드 생성·배치 시간이 증가하고 파일 저장에 걸리는 시간 또한 증가하는 모습을 보여준다. 따라서 전체 노드 생성 시간은 통신 범위에 영향받지 않고 생성하려는 노드의 개수에 영향을 받으며, 노드 개수가 증가하기 때문에, 당연히 생성된 노드들의 정보를 담은 CSV 파일의 처리 속도 또한 증가한다는 사실을 알 수 있다. 결국, 네트워크 공간의 물리적 크기가 작더라도 노드의 통신 범위가 작으면 생성되는 노드의 개수가 늘어나므로 노드의 생성·배치 시간이 늘어나는 반면, 네트워크의 물리적 크기가 크더라도 노드의 통신 범위가 크면 생성되는 노드의 개수가 감소하여 노드의 생성·배치 시간이 감소함을 유추할 수 있다.

### 7.3 노드 시각화 처리시간

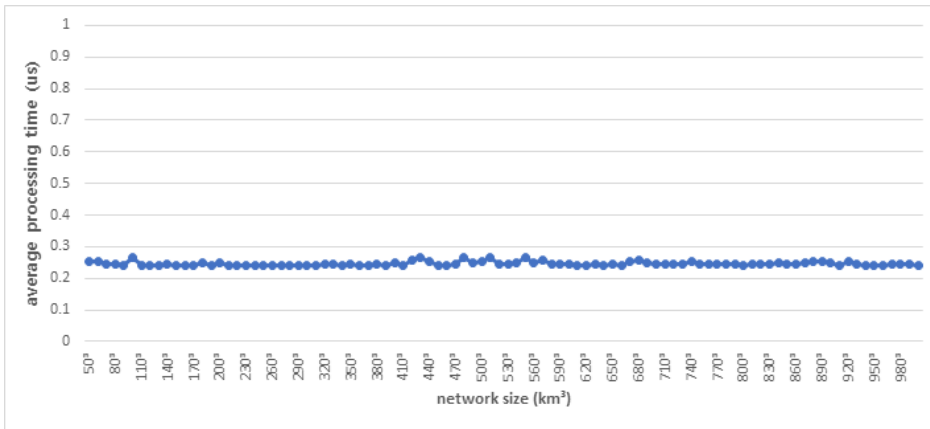
Fig. 9는 노드의 개수에 따라 CSV 파일을 읽고 노드 위치에 따라 산점도 분포로 모든 노드를 가시화하는데 소요된 시간을 나타낸다. 이 그래프는 CSV 파일로 생성된 노드들을 시각화하는데 소요되는 시간 또한 노드의 수에 비례함을 보여준다.

한편, Fig. 10는 노드의 개수에 따라 생성된 노드들을 산점도로 시각화한 그래프이다. Fig. 10 (a)는 생성된 노드 분포가 특정 영역에 집중됨이 없이 무작위로 골고루 분포되어 있음을 보여준다. 단, 노드의 개수가 많은 그래프에서 노드들이 집중되어 보이는 것은 3차원으로 분포된 노드들을 2차원으로 보여주는 과정에서 겹쳐 보이는 것일 뿐 실제로 집중된 것은 아니며 Fig. 10 (b)는 이에 대한 예를 보여준다.





(a)



(b)

Fig. 7. Average processing time for the generation and 3D placement of a node. (a) average processing time according to media range and (b) average processing time according to network size.

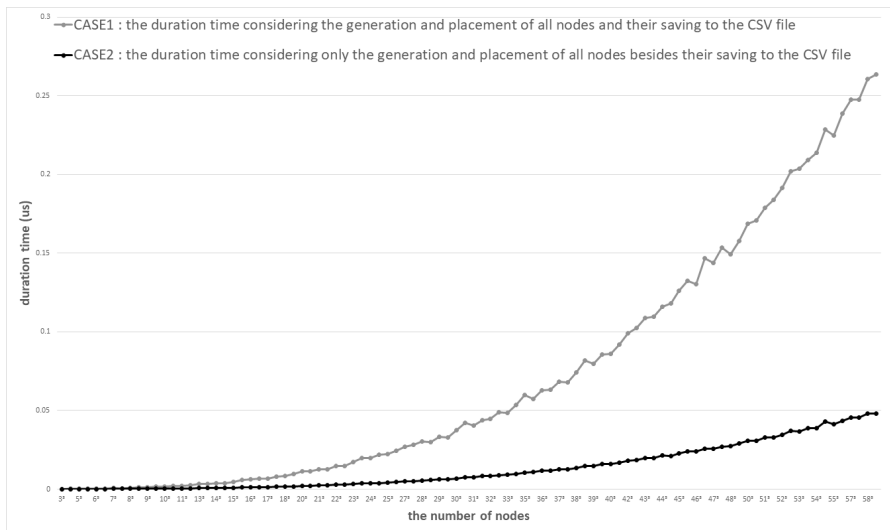


Fig. 8. Duration time for generating and placing all the nodes according to the number of them.

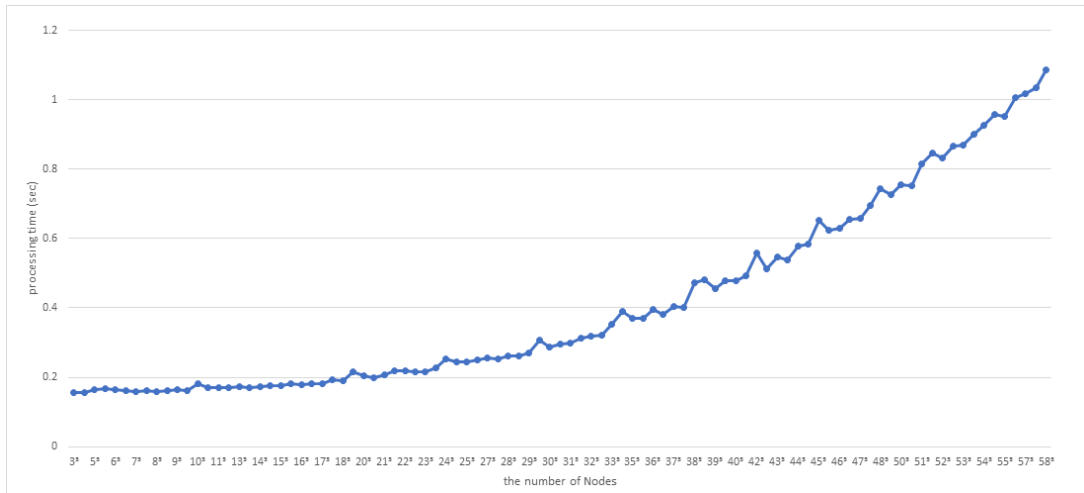


Fig. 9. Visualization time according to the number of nodes.

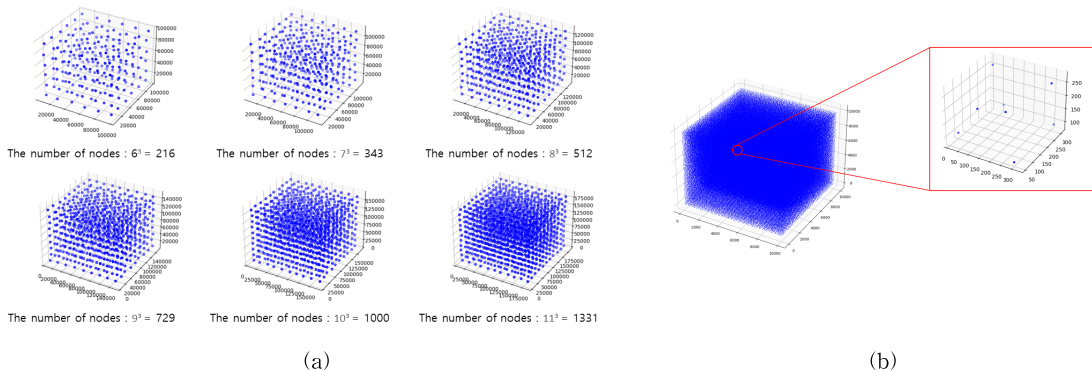


Fig. 10. Scatter plotted visualization's of all generated and placed nodes, (a) Scatter plotted visualizations according to the number of nodes and (b) An enlarged scatter plot of the uniformly randomly generated and placed nodes within a particular area.

### 7.4 분석

성능측정을 해본 결과로 제안된 알고리즘은 많은 수의 노드를 가진 네트워크라도 노드 한 개당 0.238 ~ 0.28 us로 짧은 시간 내에 생성해 낼 수 있으며 생성된 노드들을 시각화할 때에 1.08초에 166,375개의 노드를 처리할 수 있다. 따라서 제안된 알고리즘은 특정 라우팅 알고리즘의 에너지 효율을 측정하거나 네트워크 기술에 대한 라이프타임 측정과 같이 새로 연구되는 통신 기술의 성능측정 시 반복적인 시뮬레이션이 필요한 환경에서도 문제없이 다수의 무작위 네트워크를 짧은 시간 내에 생성하고 시각화할 수 있다.

### 8. 결론

본 논문은 3차원 공간에서 통신 범위 내에 링크가 가능한 균등 분포를 갖는 무작위 노드 배치 방법과 알고리즘을 제안하였다. 논문에서 제안한 방법과 알고리즘은 노드의 3차원 공간 배치를 기반으로 하는 통신과 네트워크의 시뮬레이션이 필요한 경우에 매우 유용하게 적용 가능하며, 2차원의 노드 배치만 고려하는 시뮬레이션의 결과 때문에 발생 가능한 성능 결과와 예측에 대한 부정확성과 불신의 문제점을 해결할 수 있는 장점이 있다. 본 논문은, 또한, 제안된 방법의 알고리즘을 구현하고 노드의 통신 범위와 네트워크의 크기에 따라 노드의 생성·배치에 걸리는

시간과 노드 개수에 따라 전체 노드를 생성·배치하는데 걸리는 시간과 시각화에 걸리는 시간을 각각 직접 측정하여 제시하였다. 그 결과로서 노드 한 개를 생성·배치하는데 소요되는 평균 시간은 노드의 통신 범위와 네트워크 크기에는 무관한 것으로 나타났다. 전체 노드 생성 시간과 시각화 시간은 전체 노드의 개수에 비례하는 것으로 나타났다. 이 결과는 우리가 시물레이션에서 노드의 생성·배치와 시각화에 소요되는 시간을 예측할 수 있게 해준다는 점에서 그 의미가 있다.

본 연구 결과에 따른 추후 연구로서 수중 통신 환경과 같이 기본적으로 통신속도가 느리고 깊이에 따라 주변 환경이 달라져서 통신속도와 에러율에 영향을 크게 미치는 3차원 환경을 고려해야 하는 네트워크에서 이 논문에서 제안한 3차원 노드 배치 알고리즘에 기반한 라우팅을 위한 시뮬레이터를 개발하고자 한다. 또한, 논문에서 제안한 방법의 확장 연구로서 통신 거리, 전송 속도, 에러율 등의 성능에서 서로 다른 성능을 갖는 다수의 통신 매체를 사용하는 임의의 두 노드 사이에서 여러 매체 통신을 이용하여 통신이 연결될 수 하도록 하는 3차원 네트워크 공간에 노드를 배치하는 방법에 관해서도 연구해 보고자 한다. 또한, 제안된 알고리즘에서는 실제 환경에서 통신에 영향을 미치는 다양한 환경요소와 장애물이 배제되었는데 시물레이션 네트워크에 이를 반영하는 방법에 관한 연구를 진행할 예정이다.

## REFERENCE

- [1] D. Shin and C. Kim, "A Method for Applying Multiple Wireless Communication Media to Underwater Communication," *Proceedings of the Korea Software Congress 2017*, pp. 1334-1336, 2017.
- [2] D. You and C. Kim, "Comparison Analysis of Integrated MAC Design Approaches for Multiple Media Communication in Underwater Environments," *Proceedings of Symposium of the Korean Institute of Communications and Information Sciences*, pp. 1219-1220, 2019.
- [3] D. You, D. Shin, S. Lim, S. Jeon, and C. Kim, "Implementation of a Fragmentation Method for Flow Control in Underwater Multi-Media Communication," *Journal of Korea Multimedia Society*, Vol. 23, No. 7, pp. 819-829, 2017.
- [4] B. Lee and H. Kim, "Design and Implementation of a Secure IBS Platform Using RFID and Sensor Network," *IEEE International Symposium on Consumer Electronics*, pp. 1-4, 2006.
- [5] N.-T. Dinh and Y. Kim, "Auto-Configuration in Wireless Sensor Networks: A Review," *Sensors*, Vol. 10, No. 19, pp. 1-24, 2019.
- [6] D. You and C. Kim, "Approaches to the Design and Modularization for Implementing Multimedia-Based Underwater Communication to Use Integrated MAC," *Journal of Korea Multimedia Society*, Vol. 22, No. 11, pp. 1259-1268, 2019.
- [7] D. Shin, S. Park, and C. Kim, "Underwater Acoustic Communication technology and trend," *The Magazine of the IEEE*, Vol. 45, No. 5, pp. 32-48, 2018.
- [8] D. Shin and C. Kim, "Requirements for Underwater Communications Based on Multiple Wireless Underwater Communication Media," *Proceedings of the Korean Institute of Communication Sciences Conference*, pp. 621-622, 2018.
- [9] W. Lee, J.-H. Youn, and T.-S. Song, "Combined Quorum-based NDP in Heterogeneous Wireless Sensor Networks," *Journal of Digital Contents Society*, Vol. 18, No. 4, pp. 753-760, 2017.
- [10] J.-J. You, Y.-K. Kim, and J.-W. Chang, "Design and Implementation of A Location-Based Energy-Efficient Routing Protocol Using Quantity of Energy Consumed," *Journal of Korea Spatial Information System Society*, Vol. 12, No. 1, pp. 1-9, 2010.
- [11] J.-M. He and C.-S. Rhe, "An Energy Efficient Routing Algorithm based on Center of Local Clustering in Wireless Sensor Networks," *Journal of Knowledge Information Technology and Systems*, Vol. 9, No. 4, pp. 502-508,

2014.

[12] Y.-G. Lee and S.-Y. Cho, *Method and Apparatus for Measuring the Distance Between Nodes*, 10-0984479, Korea, 2010.

[13] Agency for Defense Development, *A Node Deployment Strategy Considering Environmental Factors and the Number of Nodes in Surveillance and Reconnaissance Sensor Networks*, 10-1327325, Korea, 2012.

[14] C. Kim and I. Kim, "Three Dimensional Euclidean Minimum Spanning Tree for Connecting Nodes of Space with the Shortest Length," *Journal of The Korean Society of Computer and Information*, Vol. 17, No. 1, pp. 161-169, 2012.

[15] J. Carle, J.-F. Myoupo, and D. Seme, "A Basis for 3-D Cellular Networks," *Proceedings 15th International Conference on Information Networking*, pp. 631-636, 2001.

[16] S.M. Nazrul Alam and Z.J. Haas, "Coverage and Connectivity in Three-Dimensional Networks with Random Node Deployment," *Ad Hoc Networks*, Vol. 34, pp. 157-169, 2015.



임 동 현

2015년~2021년 강릉원주대학교  
컴퓨터공학과(공학사)  
2021년~현재 강릉원주대학교 컴  
퓨터공학과 (석사과정)  
관심분야: Underwater  
Communication, IoT/  
IoUT, Embedded System



김 창 화

1985년 고려대학교 수학교육과  
(이학사)  
1987년 고려대학교 전산학전공  
(이학석사)  
1990년 고려대학교 전산학전공  
(이학박사)  
1994년~1995년 University of Toronto, Enterprise  
Integration Lab. Post-Doc. & Visiting  
Professor  
2002년~2004 미국 Texas A&M 대학 Visiting Scholar  
1989년~현재 강릉원주대학교 컴퓨터공학과 교수  
2005년~현재 강릉원주대학교 해양센서네트워크시스템  
기술연구센터 센터장  
관심분야: Underwater Communication and Sensor  
Network, IoT/IoUT, Distributed System,  
Intelligent System