



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2018-0037431
(43) 공개일자 2018년04월12일

(51) 국제특허분류(Int. Cl.)
H04L 29/06 (2006.01)

(52) CPC특허분류
H04L 69/161 (2013.01)
H04L 69/26 (2013.01)

(21) 출원번호 10-2016-0127562

(22) 출원일자 2016년10월04일

심사청구일자 2016년10월04일

(71) 출원인

한림대학교 산학협력단

강원도 춘천시 한림대학길 1, 한림대학교(옥천동)

(72) 발명자

김의직

강원도 춘천시 지석로 29, 514동 1103호 (석사동, 휴먼타운아파트)

권정혁

강원도 춘천시 스무숲1길 32-9, 202호 (석사동)

오하라

서울특별시 중랑구 상봉로 131, B동 603호(상봉동, 상봉 듀오트리스)

(74) 대리인

유민규

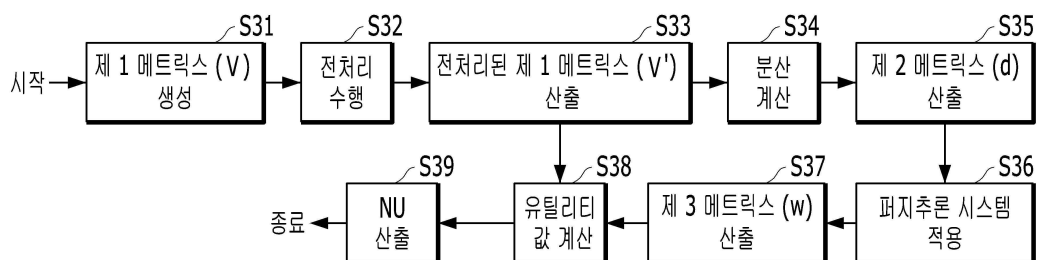
전체 청구항 수 : 총 12 항

(54) 발명의 명칭 클러스터 헤드 선정 방법 및 그를 이용하는 센서 노드

(57) 요약

클러스터 헤드 선정 방법에 관한 것이며, 클러스터 헤드 선정 방법은, (a) 복수의 센서 노드 각각이, 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하는 단계 및 (b) 상기 복수의 센서 노드 각각이, 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정하는 단계를 포함할 수 있다

대표도 - 도3



이 발명을 지원한 국가연구개발사업

과제고유번호 2016H1D5A1910427

부처명 미래창조과학부

연구관리전문기관 한국연구재단

연구사업명 지역신산업선도인력양성사업

연구과제명 실시간 상황정보 빅데이터 기반의 개인 맞춤형 헬스케어 시스템 개발

기 여 율 1/1

주관기관 한림대학교 산학협력단

연구기간 2016.06.01 ~ 2017.05.31

명세서

청구범위

청구항 1

사물 인터넷 환경에서의 클러스터 헤드 선정 방법에 있어서,

- (a) 복수의 센서 노드 각각이, 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하는 단계; 및
- (b) 상기 복수의 센서 노드 각각이, 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정하는 단계,

를 포함하는, 클러스터 헤드 선정 방법.

청구항 2

제1항에 있어서,

상기 (a) 단계는,

- (a1)상기 복수의 센서 노드 각각이, 자신과 이웃 센서 노드의 클러스터 변수 정보 값들(cluster variables)을 이용하여 제1 매트릭스를 생성하는 단계;
- (a2) 상기 복수의 센서 노드 각각이, 상기 제1 매트릭스에 전처리를 수행함으로써 전처리된 제1 매트릭스를 산출하는 단계;
- (a3) 상기 복수의 센서 노드 각각이, 상기 전처리된 제1 매트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 매트릭스를 산출하는 단계;
- (a4) 상기 복수의 센서 노드 각각이, 상기 제2 매트릭스를 퍼지 추론 시스템(fuzzy inference system)의 퍼지 입력 변수로 사용함으로써 퍼지 출력 변수로서 상기 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 나타내는 제3 매트릭스를 산출하는 단계; 및
- (a5) 상기 복수의 센서 노드 각각이, 상기 전처리된 제1 매트릭스와 상기 제3 매트릭스를 이용하여 상기 유틸리티 값을 계산하는 단계,

를 포함하는 것인, 클러스터 헤드 선정 방법.

청구항 3

제2항에 있어서,

상기 (a1) 단계는,

상기 복수의 센서 노드 각각이, 상기 클러스터 변수 정보 값으로서 클러스터 변수의 종류의 수를 고려한 클러스터 변수 각각의 순차 정보 및 이웃 센서 노드의 수를 고려한 이웃 센서 노드 각각의 순차 정보를 이용하여 상기 제1 매트릭스를 생성하는 것인, 클러스터 헤드 선정 방법.

청구항 4

제3항에 있어서,

상기 (a2) 단계는,

상기 복수의 센서 노드 각각이, 상기 제1 매트릭스에 포함된 클러스터 변수의 종류별로 기정의된 카테고리 분류하고 기정의된 카테고리별로 대응하는 분류 함수를 적용함으로써 상기 전처리된 제1 매트릭스를 산출하는 것인, 클러스터 헤드 선정 방법.

청구항 5

제1항에 있어서,

상기 (b) 단계는,

상기 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 제1 센서 노드가 클러스터 헤드로 선정되는 것인, 클러스터 헤드 선정 방법.

청구항 6

제5항에 있어서,

상기 (b) 단계는,

상기 복수의 센서 노드 중 상기 클러스터 헤드로 선정된 제1 센서 노드가,

자신이 클러스터 헤드임을 알리는 클러스터 헤드 알림 메시지를 주변 이웃 센서 노드로 브로드캐스팅하고, 상기 주변 이웃 센서 노드로부터 가입 요청 메시지를 수신하기 위한 가입 요청 시간을 설정하며, 상기 설정된 가입 요청 시간이 만료된 것으로 판단되는 경우 상기 가입 요청 메시지를 송신한 주변 이웃 센서 노드로 가입 응답 메시지를 전송하는 것은, 클러스터 헤드 선정 방법.

청구항 7

제6항에 있어서,

상기 (b) 단계는,

상기 복수의 센서 노드 중 상기 클러스터 헤드로 선정된 제1 센서 노드의 유틸리티 값 보다 큰 유틸리티 값을 가지는 제2 센서 노드가 존재하는 경우, 상기 제1 센서 노드는 자신을 클러스터 멤버로 설정하고, 기설정된 클러스터 헤드 알림 수신 시간 동안에 상기 제2 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 가입 요청 메시지를 상기 제2 센서 노드로 전송하는 것인, 클러스터 헤드 선정 방법.

청구항 8

제7항에 있어서,

상기 (b) 단계는,

자신을 클러스터 멤버로 설정한 상기 제1 센서 노드가 상기 기설정된 클러스터 헤드 알림 수신 시간 동안에 복수의 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 상기 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 센서 노드를 클러스터 헤드로서 선택하는 것인, 클러스터 헤드 선정 방법.

청구항 9

제1항에 있어서,

상기 (a) 단계 이전에,

상기 복수의 센서 노드 각각이, 자신의 유틸리티 변수 값과 클러스터 헤드인지 여부를 나타내는 클러스터 헤드 변수 값을 초기화시키는 단계,

를 더 포함하는 것인, 클러스터 헤드 선정 방법.

청구항 10

제1항의 사물 인터넷 환경에서의 클러스터 헤드 선정 방법을 이용하는 센서 노드에 있어서,

자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하는 유틸리티 값 계산부; 및

상기 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정하는 클러스터 헤드 선정부,

를 포함하는 센서 노드.

청구항 11

제10항에 있어서,

상기 유틸리티 값 계산부는,

상기 자신과 상기 이웃 센서 노드의 클러스터 변수 정보 값들(cluster variables)을 이용하여 제1 매트릭스를 생성하고,

상기 제1 매트릭스에 전처리를 수행함으로써 전처리된 제1 매트릭스를 산출하고,

상기 전처리된 제1 매트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 매트릭스를 산출하고,

상기 제2 매트릭스를 퍼지 추론 시스템(fuzzy inference system)의 퍼지 입력 변수로 사용함으로써 퍼지 출력 변수로서 상기 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 나타내는 제3 매트릭스를 산출하고,

상기 전처리된 제1 매트릭스와 상기 제3 매트릭스를 이용하여 상기 유틸리티 값을 계산하는 것인, 센서 노드.

청구항 12

제1항 내지 제9항 중 어느 한 항의 방법을 컴퓨터에서 실행시키기 위한 프로그램을 기록한 컴퓨터로 읽을 수 있는 기록매체.

발명의 설명

기술 분야

[0001] 본원은 사물 인터넷 환경에서의 클러스터 헤드 선정 방법 및 그를 이용하는 센서 노드에 관한 것이다.

배경 기술

[0002] 사물인터넷(Internet of Things, IoT)은 센서와 통신 기능을 내장한 수많은 사물들이 주변의 물리적 환경 현상을 모니터링하고 제어할 수 있으며, 사물 상호 간에 또는 인터넷에 연결되어 원격의 사물과 통신할 수 있는 기술을 의미한다. 최근 이러한 IoT의 기술적 개념은 홈네트워크 서비스, 지능형 빌딩, 자동화, 헬스케어, 물류, 제조, 엔터테인먼트 등 다양한 응용분야에 활용되고 있다.

[0003] 수많은 센서 노드로 구성된 IoT 네트워크는 '사물의 연결(Connectivity of things)'과 '물리적 센싱 데이터의 전달(Delivery of physical sensing data)'을 위해 핵심적인 역할을 한다. IoT 네트워크에서는, 대부분의 센서 노드가 자원이 제한적인(resource-constrained) 디바이스이며, 그들의 수집된 센싱 데이터를 무선링크의 손실을 통해 주기적으로 기지국(base station, BS) 또는 싱크 노드(sink node)로 전달하는 구조를 가지며, 이는 무선 센서 네트워크(wireless sensor network, WSN)의 네트워크 구조와 매우 유사하다. 따라서, 지금까지 기존의 연구에서는 센서 노드들의 데이터 전달에 있어서 에너지 효율성(Energy efficiency)을 주 목적으로 하는 라우팅 프로토콜(Routing protocol)에 대한 연구들이 진행되어 왔다. 특히, 클러스터 기반의 라우팅 프로토콜은 센서 노드의 부족한 자원을 효율적으로 관리하고, 부하 균등(load balancing)을 통해 IoT 네트워크의 시스템 수명을 증가시킬 수 있는 효율적인 접근방식 중 하나이다.

[0004] 클러스터 기반 라우팅 프로토콜에서는 네트워크 내의 센서 노드들이 그룹화되어 몇 개의 클러스터를 구성하고, 각 클러스터는 기지국 또는 싱크 노드와 통신할 수 있는 하나의 클러스터 헤드(cluster head, CH)를 유지하고, 나머지 센서 노드들은 클러스터 멤버(cluster member, CM)가 되어 계층적인 네트워크 구조를 형성한다. 여기서, 클러스터 헤드는 자신에게 연결된 다수의 클러스터 멤버들로부터 센싱 데이터를 수집하여 싱크 노드로 전송하는 역할을 하므로 에너지가 빨리 소진되며, 이러한 이유로 클러스터 헤드의 선정(CH election) 방법은 클러스터 기반 라우팅 프로토콜에서 매우 중요한 도전 과제이다.

[0005] 앞서 언급한 바와 같이, IoT 기술은 다양한 어플리케이션에 적용될 수 있으며, 각 IoT 어플리케이션에서 사용되는 클러스터 기반 라우팅 프로토콜은 해당 어플리케이션의 특징에 따라 서로 다른 한 개 이상의 클러스터링 변수(clustering variable)들을 사용한다. 최근 연구에서는, 여러 가지 클러스터링 변수들을 종합적으로 고려할 수 있는 퍼지 논리(fuzzy logic)를 이용한 클러스터 헤드 선정 알고리즘들이 제안된바 있다.

[0006] 구체적으로, 퍼지 논리는 이진법의 논리와 양자화된 옴, 톨립의 논리 등을 사용하는 기존 기술의 한계를 극복하기 위해 제안된 기술로서, 근사치나 주관적 값을 사용하는 규칙들을 생성함으로써 부정확함을 표현할 수 있는

규칙 기반 기술을 의미한다. 즉, 퍼지 논리는 정확한 계산보다는 근사값을 사람의 생각과 유사하게 추론하는 개념으로서, 퍼지 추론 시스템(fuzzy inference system, FIS)을 사용한다. 퍼지 추론 시스템은 미리 정해진 퍼지 규칙을 사용해 여러 개의 퍼지 입력 변수들을 하나의 퍼지 출력 변수로 맵핑(mapping)하는 과정을 처리하는 제어 시스템 모델이다. 이러한 퍼지 추론 시스템을 활용하면 클러스터 헤드 선정 과정의 복잡성을 증가시키지 않으면서 필요한 여러 개의 클러스터링 변수들을 종합하여 적절한 클러스터 헤드를 선정할 수 있기 때문에, 최근 퍼지 추론 시스템이 클러스터 기반 라우팅 프로토콜과 관련된 연구에 활발히 적용되고 있다.

[0007] 종래의 한 논문에서는, 확률을 이용해 무작위로 클러스터 헤드를 선택하는 방법을 통해, 종래 클러스터 기반 라우팅 프로토콜에서 클러스터 헤드의 에너지가 다른 노드에 비해 많이 소비된다는 문제점을 개선한바 있다. 그러나 이러한 방법은 적합하지 않은 클러스터 헤드가 연속적으로 선택될 수 있는 가능성의 문제가 여전히 존재하는 단점이 있다.

[0008] 본원의 배경이 되는 기술은 한국등록특허공보 제10-1033665호(등록일: 2011.04.29)에 개시되어 있다.

발명의 내용

해결하려는 과제

[0009] 본원은 전술한 종래 기술의 문제점을 해결하기 위한 것으로서, 클러스터 헤드 선정 시 낮은 복잡도와 클러스터 변수에 대한 고확장성을 유지할 수 있는 클러스터 헤드 선정 방법 및 그를 이용하는 센서 노드를 제공하려는 것을 목적으로 한다.

[0010] 본원은 전술한 종래 기술의 문제점을 해결하기 위한 것으로서, 종래의 클러스터 헤드 선정 기술 대비 처리율, 종단간 지연시간, 네트워크 수명 및 에너지 소비 측면에서 더 나은 성능을 나타내는 클러스터 헤드 선정 방법 및 그를 이용하는 센서 노드를 제공하려는 것을 목적으로 한다.

[0011] 본원은 전술한 종래 기술의 문제점을 해결하기 위한 것으로서, 보편적인 사물인터넷 어플리케이션 환경에서 클러스터 네트워크 구축을 위해 퍼지 추론 시스템을 사용하여 적절한 클러스터 헤드를 선정할 수 있는 클러스터 헤드 선정 방법 및 그를 이용하는 센서 노드를 제공하려는 것을 목적으로 한다.

[0012] 다만, 본원의 실시예가 이루고자 하는 기술적 과제는 상기된 바와 같은 기술적 과제들로 한정되지 않으며, 또 다른 기술적 과제들이 존재할 수 있다.

과제의 해결 수단

[0013] 상기한 기술적 과제를 달성하기 위한 기술적 수단으로서, 본원의 일 실시예에 따른 사물 인터넷 환경에서의 클러스터 헤드 선정 방법은, (a) 복수의 센서 노드 각각이, 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하는 단계 및 (b) 상기 복수의 센서 노드 각각이, 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정하는 단계를 포함할 수 있다.

[0014] 또한, 상기 (a) 단계는, (a1)상기 복수의 센서 노드 각각이, 자신과 이웃 센서 노드의 클러스터 변수 정보 값들(cluster variables)을 이용하여 제1 매트릭스를 생성하는 단계, (a2) 상기 복수의 센서 노드 각각이, 상기 제1 매트릭스에 전처리를 수행함으로써 전처리된 제1 매트릭스를 산출하는 단계, (a3) 상기 복수의 센서 노드 각각이, 상기 전처리된 제1 매트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 매트릭스를 산출하는 단계, (a4) 상기 복수의 센서 노드 각각이, 상기 제2 매트릭스를 퍼지 추론 시스템(fuzzy inference system)의 퍼지 입력 변수로 사용함으로써 퍼지 출력 변수로서 상기 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 나타내는 제3 매트릭스를 산출하는 단계 및 (a5) 상기 복수의 센서 노드 각각이, 상기 전처리된 제1 매트릭스와 상기 제3 매트릭스를 이용하여 상기 유틸리티 값을 계산하는 단계를 포함할 수 있다.

[0015] 또한, 상기 (a1) 단계는, 상기 복수의 센서 노드 각각이, 상기 클러스터 변수 정보 값으로서 클러스터 변수의 종류의 수를 고려한 클러스터 변수 각각의 순차 정보 및 이웃 센서 노드의 수를 고려한 이웃 센서 노드 각각의 순차 정보를 이용하여 상기 제1 매트릭스를 생성할 수 있다.

[0016] 또한, 상기 (a2) 단계는, 상기 복수의 센서 노드 각각이, 상기 제1 매트릭스에 포함된 클러스터 변수의 종류별로 기정의된 카테고리 분류하고 기정의된 카테고리별로 대응하는 분류 함수를 적용함으로써 상기 전처리된 제1 매트릭스를 산출할 수 있다.

[0017] 또한, 상기 (b) 단계는, 상기 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 제1 센서 노드가 클러스터 헤

드로 선정될 수 있다.

- [0018] 또한, 상기 (b) 단계는, 상기 복수의 센서 노드 중 상기 클러스터 헤드로 선정된 제1 센서 노드가, 자신이 클러스터 헤드임을 알리는 클러스터 헤드 알림 메시지를 주변 이웃 센서 노드로 브로드캐스팅하고, 상기 주변 이웃 센서 노드로부터 가입 요청 메시지를 수신하기 위한 가입 요청 시간을 설정하며, 상기 설정된 가입 요청 시간이 만료된 것으로 판단되는 경우 상기 가입 요청 메시지를 송신한 주변 이웃 센서 노드로 가입 응답 메시지를 전송할 수 있다.
- [0019] 또한, 상기 (b) 단계는, 상기 복수의 센서 노드 중 상기 클러스터 헤드로 선정된 제1 센서 노드의 유틸리티 값보다 큰 유틸리티 값을 가지는 제2 센서 노드가 존재하는 경우, 상기 제1 센서 노드는 자신을 클러스터 멤버로 설정하고, 기설정된 클러스터 헤드 알림 수신 시간 동안에 상기 제2 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 가입 요청 메시지를 상기 제2 센서 노드로 전송할 수 있다.
- [0020] 또한, 상기 (b) 단계는, 자신을 클러스터 멤버로 설정한 상기 제1 센서 노드가 상기 기설정된 클러스터 헤드 알림 수신 시간 동안에 복수의 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 상기 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 센서 노드를 클러스터 헤드로서 선택할 수 있다.
- [0021] 또한, 본원의 일 실시예에 따른 사물 인터넷 환경에서의 클러스터 헤드 선정 방법은, 상기 (a) 단계 이전에, 상기 복수의 센서 노드 각각이, 자신의 유틸리티 변수 값과 클러스터 헤드인지 여부를 나타내는 클러스터 헤드 변수 값을 초기화시키는 단계를 더 포함할 수 있다.
- [0022] 한편, 본원의 일 실시예에 따른 사물 인터넷 환경에서의 클러스터 헤드 선정 방법을 이용하는 센서 노드는, 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하는 유틸리티 값 계산부 및 상기 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정하는 클러스터 헤드 선정부를 포함할 수 있다.
- [0023] 또한, 상기 유틸리티 값 계산부는, 상기 자신과 상기 이웃 센서 노드의 클러스터 변수 정보 값들(cluster variables)을 이용하여 제1 매트릭스를 생성하고, 상기 제1 매트릭스에 전처리를 수행함으로써 전처리된 제1 매트릭스를 산출하고, 상기 전처리된 제1 매트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 매트릭스를 산출하고, 상기 제2 매트릭스를 퍼지 추론 시스템(fuzzy inference system)의 퍼지 입력 변수로 사용함으로써 퍼지 출력 변수로서 상기 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 나타내는 제3 매트릭스를 산출하고, 상기 전처리된 제1 매트릭스와 상기 제3 매트릭스를 이용하여 상기 유틸리티 값을 계산할 수 있다.
- [0024] 상술한 과제 해결 수단은 단지 예시적인 것으로서, 본원을 제한하려는 의도로 해석되지 않아야 한다. 상술한 예시적인 실시예 외에도, 도면 및 발명의 상세한 설명에 추가적인 실시예가 존재할 수 있다.

발명의 효과

- [0025] 전술한 본원의 과제 해결 수단에 의하면, 복수의 센서 노드 각각이 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산하고, 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교하여 클러스터 헤드를 선정함으로써, 클러스터 헤드 선정 시 낮은 복잡도와 클러스터 변수에 대한 고확장성을 유지할 수 있는 효과가 있다.
- [0026] 전술한 본원의 과제 해결 수단에 의하면, 클러스터 헤드 선정 방법을 제공함으로써, 종래의 클러스터 헤드 선정 기술 대비 처리율, 종단간 지연시간, 네트워크 수명 및 에너지 소비 측면에서 더 나은 성능을 제공할 수 있는 효과가 있다.
- [0027] 전술한 본원의 과제 해결 수단에 의하면, 보편적인 사물인터넷 어플리케이션 환경에서 클러스터 네트워크 구축을 위해 퍼지 추론 시스템을 사용함으로써 적절한 클러스터 헤드를 선정할 수 있는 효과가 있다.

도면의 간단한 설명

- [0028] 도 1은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에 기초한 클러스터 네트워크의 구축 예를 나타낸 도면이다.
- 도 2는 본원의 일 실시예에 따른 클러스터 헤드 선출 방법을 이용하는 센서 노드의 구성을 개략적으로 나타낸 블록도이다.

도 3은 본원의 일 실시예에 따른 센서 노드가 유틸리티 값을 계산하는 과정을 나타낸 도면이다.

도 4는 본원의 일 실시예에 따른 센서 노드가 전처리 과정을 도식화한 도면이다.

도 5는 본원의 일 실시예에 따른 센서 노드가 전처리를 수행할 때 클러스터 변수의 종류에 따른 분류 함수의 그래프를 나타낸 도면이다.

도 6은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법을 이용하는 복수의 센서 노드 각각에 의하여 측정된 클러스터 변수 값의 예를 나타낸 도면이다.

도 7은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 분산 값의 예를 나타낸 도면이다.

도 8은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 이용되는 퍼지 추론 시스템을 개략적으로 나타낸 도면이다.

도 9는 본원의 일 실시예에 따른 클러스터 헤드 선정 방법을 통해 제안되는 변수 분류 클러스터링 알고리즘(VCCA)의 클러스터링 동작을 나타낸 도면이다.

발명을 실시하기 위한 구체적인 내용

- [0029] 아래에서는 첨부한 도면을 참조하여 본원이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본원의 실시예를 상세히 설명한다. 그러나 본원은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본원을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0030] 본원 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결" 또는 "간접적으로 연결"되어 있는 경우도 포함한다.
- [0031] 본원 명세서 전체에서, 어떤 부재가 다른 부재 "상에", "상부에", "상단에", "하에", "하부에", "하단에" 위치하고 있다고 할 때, 이는 어떤 부재가 다른 부재에 접해 있는 경우뿐 아니라 두 부재 사이에 또 다른 부재가 존재하는 경우도 포함한다.
- [0032] 본원 명세서 전체에서, 어떤 부분이 어떤 구성 요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있는 것을 의미한다.
- [0033] 도 1은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에 기초한 클러스터 네트워크의 구축 예를 나타낸 도면이다.
- [0034] 도 1을 참조하면, 본원은 사물 인터넷(Internet of Things, IoT) 환경에서 클러스터 헤드를 선정하는 방법 및 그를 이용하는 센서 노드에 관한 것으로서, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는 복수의 센서 노드(1, 2, 3, 4, 5)가 네트워크(10)를 통해 상호 연결될 수 있다.
- [0035] 네트워크(10)는 일예로 무선 센서 네트워크(wireless sensor networks, WSNs), 블루투스(Bluetooth) 네트워크, NFC(Near Field Communication) 네트워크, 인터넷(Internet), LAN(Local Area Network), Wireless LAN(Wireless Local Area Network) 등일 수 있으며, 이에 한정된 것은 아니다.
- [0036] 복수의 센서 노드(1, 2, 3, 4, 5)는 네트워크(10)를 통해 다른 센서 노드들과 데이터 송수신 등의 통신을 수행하는 센서 디바이스로서, 서버 또는 장치를 의미할 수 있다.
- [0037] 복수의 센서 노드(1, 2, 3, 4, 5)는 일예로, 고정되거나 이동성을 가지는 단말로서, MS(Mobile Station), UT(User Terminal), SS(Subscriber Station), 무선기기(Wireless Device), 무선 모뎀(Wireless Modem), 휴대기기(Handheld Device), AT(Access Terminal) 등과 같은 용어로 불릴 수도 있다.
- [0038] 또한, 본원의 다양한 실시예들에 따르면 복수의 센서 노드(1, 2, 3, 4, 5)는 다양한 형태의 단말로서, 예를 들어, 단말은 네트워크를 통해 원격지의 서버에 접속할 수 있는 TV 장치, 컴퓨터 또는 휴대용 단말일 수 있다. 여기서, TV 장치의 일 예에는 스마트 TV, IPTV 셋톱박스 등이 포함되고, 컴퓨터의 일 예에는 웹 브라우저(WEB Browser)가 탑재된 노트북, 데스크톱(desktop), 랩톱(laptop) 등이 포함되고, 휴대용 단말의 일 예에는 휴대성과 이동성이 보장되는 무선 통신 장치로서, PCS(Personal Communication System), GSM(Global System for Mobile communications), PDC(Personal Digital Cellular), PHS(Personal Handyphone System), PDA(Personal

Digital Assistant), IMT(International Mobile Telecommunication)-2000, CDMA(Code Division Multiple Access)-2000, W-CDMA(W Code Division Multiple Access), Wibro(Wireless Broadband Internet) 단말, 스마트폰(smartphone), 태블릿 PC 등과 같은 모든 종류의 핸드헬드(Handheld) 기반의 무선 통신 장치가 포함될 수 있으며, 이에 한정된 것은 아니다.

- [0039] 또한, 도 1에 도시하지는 않았으나, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는 싱크 노드(sink node, 미도시)를 포함할 수 있으며, 싱크 노드는 복수의 센서 노드(1, 2, 3, 4, 5) 각각으로부터 전송되는 데이터를 취합하고, 이러한 데이터를 가공하여 서버(미도시)로 전송할 수 있다.
- [0040] 복수의 센서 노드(1, 2, 3, 4, 5)는 네트워크(10)를 구성하여 데이터 송수신을 수행하는 한편, 싱크 노드는 복수의 센서 노드로부터 전송되는 데이터의 수집 및 전송뿐만 아니라 복수의 센서 노드들의 네트워크(20)에 대한 구성 및 관리를 수행할 수 있으며, 이에 한정되는 것은 아니다.
- [0041] 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는 복수의 센서 노드(1, 2, 3, 4, 5) 각각이 자신의 네트워크 역량(network capability)을 나타내는 유틸리티 값을 계산한 후 자신의 유틸리티 값과 이웃한 센서 노드의 유틸리티 값을 비교함으로써, 복수의 센서 노드(1, 2, 3, 4, 5) 각각이 분산적인 클러스터 헤드 선정 과정을 수행할 수 있다. 이때 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는 가장 큰 유틸리티 값을 갖는 센서 노드가 클러스터 헤드로 선정될 수 있으며, 도 1에서의 경우에는 일예로 제2 센서 노드(2)가 가장 큰 유틸리티 값을 가져 클러스터 헤드로 선정될 수 있다.
- [0042] 본원에서 이용되는 노드 각각의 유틸리티 값인 노드 유틸리티(node utility, NU) 값은 해당 노드의 네트워크 역량 레벨에 대한 정략적인 수치를 표현하기 위해 사용되는 지표를 의미할 수 있다. 이에 기초하여 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는 가장 큰 유틸리티 값을 갖는 노드가 클러스터 헤드로 선정될 수 있다.
- [0043] 또한, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법은, IoT 어플리케이션에 따라 다른 클러스터 네트워크를 구성하기 위해, 불특정한 복수의 클러스터 변수들(즉, 클러스터 변수 정보 값, cluster variables)을 특성에 따라 분류하는 과정을 거쳐 복수의 센서 노드 중 높은 네트워크 용량을 갖는 클러스터 헤드를 선정하는 기법으로서, 변수 분류 클러스터링 알고리즘(variable-categorized clustering algorithm, VCCA)을 제안한다. 본원에서 제안하는 VCCA 기법은 클러스터 헤드 선정 과정의 낮은 복잡도와 클러스터 변수에 대한 고확장성을 위해 규칙 기반의 변수 매핑을 통하여 결과값을 산출하는 퍼지 추론 시스템(fuzzy inference system, FIS)을 이용함으로써 클러스터 헤드를 선정할 수 있다.
- [0044] 이하에서는 복수의 센서 노드(1, 2, 3, 4, 5) 각각이 자신의 유틸리티 값을 계산하고, 이에 기초하여 클러스터 헤드를 분산적으로 선정하는 방법에 대해 보다 자세히 설명하기로 한다.
- [0045] 도 2는 본원의 일 실시예에 따른 클러스터 헤드 선정 방법을 이용하는 센서 노드의 구성을 개략적으로 나타낸 블록도이다.
- [0046] 도 2를 참조하면 본원의 일 실시예에 따른 사물 인터넷 환경에서의 클러스터 헤드 선정 방법을 이용하는 센서 노드(20)는 유틸리티 값 계산부(21) 및 클러스터 헤드 선정부(22)를 포함할 수 있다. 이때, 센서 노드(20)에 대하여 설명하는 내용은 도 1에서의 복수의 센서 노드(1, 2, 3, 4, 5) 각각에도 동일하게 적용 가능하다. 즉, 복수의 센서 노드(1, 2, 3, 4, 5) 각각은 센서 노드(20)와 동일한 구조로 이루어질 수 있다.
- [0047] 유틸리티 값 계산부(21)는 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산할 수 있다. 유틸리티 값을 계산하는 과정은 도 3을 참조하여 보다 쉽게 이해될 수 있다.
- [0048] 도 3은 본원의 일 실시예에 따른 센서 노드가 유틸리티 값을 계산하는 과정을 나타낸 도면이다.
- [0049] 도 3을 참조하면, 센서 노드(20)에서 유틸리티 값 계산부(21)는, 단계S31에서, 자신과 이웃 센서 노드의 클러스터 변수 정보 값들(cluster variables)을 이용하여 제1 메트릭스(V)를 생성할 수 있다. 이때, 센서 노드(20)는 이웃 센서 노드들과의 헬로 메시지 교환을 통해 이웃 센서 노드들 각각으로부터 각각의 ID와 클러스터의 변수 정보 값들(cluster variables)을 수신할 수 있으며, 수신된 정보에 기초하여 제1 메트릭스를 생성하여 저장할 수 있다.
- [0050] 제1 메트릭스는 $V=[v_{ij}]$ 로 표현될 수 있으며, 여기서, v_{ij} 의 i 는 각 클러스터 변수의 순차 정보를 의미하고, j 는 각 노드의 순차 정보를 의미한다. 달리 말해, 단계S31에서 센서 노드(20)는 클러스터 변수 정보 값으로서 클

러스터 변수의 종류의 수를 고려한 클러스터 변수 각각의 순차 정보(i) 및 이웃 센서 노드의 수를 고려한 이웃 센서 노드 각각의 순차 정보(j)를 이용하여 제1 메트릭스를 생성할 수 있다. 일례로, 클러스터 변수의 종류의 수가 n 개이고, 이웃 센서 노드의 수가 m 개인 경우, 유틸리티 값 계산부(21)는 제1 메트릭스로서 $n \times m$ 크기의 행렬 V를 생성할 수 있다.

[0051] 다음으로, 유틸리티 값 계산부(21)는 생성된 제1 메트릭스에 전처리를 수행(S32)함으로써 전처리된 제1 메트릭스(V')를 산출(S33)할 수 있다.

[0052] 단계S32의 전처리 과정에서 유틸리티 값 계산부(21)는 클러스터 변수의 특성을 고려하기 위해, 제1 메트릭스의 각 행마다 분류 함수(category function)를 적용함으로써 전처리된 제1 메트릭스를 산출할 수 있다. 달리 표현하여, 유틸리티 값 계산부(21)는 제1 메트릭스에 포함된 클러스터 변수의 종류별로(즉, 제1 메트릭스의 각 행마다) 기정의된 카테고리(예를 들어, 리니어(Linear), 인버스(Inverse) 및 트라이앵글(Triangular))로 분류하고 기정의된 카테고리별로 대응하는 분류 함수를 적용함으로써 전처리된 제1 메트릭스를 산출할 수 있다.

[0053] 보다 구체적으로, 복수의 센서 노드(1, 2, 3, 4, 5) 각각은 주기적으로 여러 종류의 클러스터 변수들을 측정할 수 있는데, 이때, 클러스터 변수의 종류 마다 측정값의 물리적 특성은 서로 다를 수 있다. 따라서, 유틸리티 값 계산부(21)는 클러스터 변수들을 특성에 따라 특정 카테고리(예를 들어, 리니어, 인버스, 트라이앵글)로 분류하여 그에 맞는 분류함수를 적용하는 전처리 과정(S32)을 수행함으로써 전처리된 제1 메트릭스를 산출할 수 있다. 유틸리티 값 계산부(21)는 전처리 과정을 통해 클러스터 변수들의 분류(categorization) 및 정규화(normalization)를 수행할 수 있다.

[0054] 일례로, 본원에서 이용되는 클러스터 변수의 분류를 위한 기정의된 카테고리는 리니어(Linear), 인버스(Inverse) 및 트라이앵글(Triangular)이 포함될 수 있다.

[0055] 리니어 카테고리의 경우, 측정된 값이 클수록 보다 클러스터 헤드로 적합하다고 판단되는 클러스터 변수가 카테고리 A(리니어 카테고리)로 분류될 수 있다. 예를 들어, 측정 값으로서 잔류 에너지(residual energy) 측정 값이 이용될 수 있으며, 이때 복수의 센서 노드 중 잔류 에너지가 가장 많이 남아 있는 센서 노드가 클러스터 헤드로서 적합성이 높다고 판단될 수 있다. 따라서, 유틸리티 값 계산부(21)는 제1 메트릭스에 포함된 클러스터 변수들 중 측정된 값이 클수록 클러스터 헤드로 적합하다고 판단되는 잔류 에너지 등과 같은 변수를 카테고리 A로 분류할 수 있다.

[0056] 인버스 카테고리의 경우, 리니어 카테고리와는 반대로, 감소하는 특성을 가져 측정 값이 적을수록 클러스터 헤드에 더 적합하다고 판단되는 클러스터 변수가 카테고리 B(인버스 카테고리)로 분류될 수 있다. 예를 들어, 모바일 헬스케어 시스템에서, 인체에 부착되는 센서는 온도가 높을수록 인체에 해로운 영향을 끼칠 수 있다. 이러한 경우에는 센서의 온도가 낮을수록 클러스터 헤드로서 적합성이 높다고 평가될 수 있다. 따라서, 유틸리티 값 계산부(21)는 제1 메트릭스에 포함된 클러스터 변수들 중 측정된 값이 작을수록 클러스터 헤드로 적합하다고 판단되는 온도 등과 같은 변수를 카테고리 B로 분류할 수 있다.

[0057] 트라이앵글 카테고리의 경우, 센서 노드(20)와 싱크 노드(미도시) 사이의 거리와 같은 특정한 구간에서 가장 좋은 값을 갖는 클러스터 변수가 카테고리 C(트라이앵글 카테고리)로 분류될 수 있다. 예를 들어, 특정 센서 노드가 이웃 센서 노드 및 싱크 노드 모두와 적절한 신호 세기로 통신을 수행하기 위해서는 특정 노드와 싱크 노드 사이의 거리가 특정 값에 가까울수록 클러스터 헤드로서 적합성이 높다고 평가될 수 있다. 따라서, 유틸리티 값 계산부(21)는 제1 메트릭스에 포함된 클러스터 변수들 중 특정한 구간에서 가장 좋은 값을 가져 클러스터 헤드로 적합하다고 판단되는 싱크 노드와의 거리 등과 같은 변수를 카테고리 C로 분류할 수 있다. 이처럼 유틸리티 값 계산부(21)는 제1 메트릭스에 포함된 클러스터 변수들을 특성에 맞게 기정의된 카테고리(예를 들어, 리니어, 인버스, 트라이앵글)로 분류한 후 각 카테고리의 특성에 맞는 분류 함수를 적용함으로써 전처리된 제1 메트릭스를 산출할 수 있다. 한편, 전처리 과정(S32)의 경우, 도 4 내지 도 6을 참조하여 보다 쉽게 이해될 수 있다.

[0058] 도 4는 본원의 일 실시예에 따른 센서 노드가 전처리 과정을 도식화한 도면이고, 도 5는 본원의 일 실시예에 따른 센서 노드가 전처리를 수행할 때 클러스터 변수의 종류에 따른 분류 함수의 그래프를 나타낸 도면이고, 도 6은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법을 이용하는 복수의 센서 노드 각각에 의하여 측정된 클러스터 변수 값의 예를 나타낸 도면이다.

[0059] 도 4 내지 도 6을 참조하면, 먼저 도 4에서 P_a , P_b , P_c 는 각 카테고리(즉, 리니어, 인버스 및 트라이앵글 카테고리)에 해당하는 클러스터 변수들로 구성된 메트릭스를 의미하며, 각 카테고리에 적용되는 분류 함수는 f_a , f_b , f_c 로 정의될 수 있으며, 이는 도 5와 같은 그래프로 표현될 수 있다.

- [0060] 도 5를 참조하면, 유틸리티 값 계산부(21)는 일예로, 클러스터 변수들의 분류 과정을 거친 변수 값을 정규화함으로써 도 5와 같이 모두 같은 범위 안에서의 값으로 표현할 수 있다. 일예로, f_A 함수는 기울기가 1인 증가 함수로서 x 값이 증가함에 따라 y 값도 증가할 수 있다. f_B 함수는 기울기가 -1인 감수 함수로서 x 값이 작을수록 y 값이 증가할 수 있다. f_C 함수는 특정 값에서 y 가 최대값을 갖고, x 가 특정 값 보다 작으면 x 값이 증가함에 따라 y 값이 증가하고, x 가 특정 값 보다 크면 x 값이 증가할수록 y 값이 감소할 수 있다.
- [0061] 도 6을 참조하면, 표 1(Table 1)은 각 센서 노드가 측정한 클러스터 변수 값에 대한 예를 나타내고, 표 2(Tables 2)는 표 1에 도시된 클러스터 변수에 대하여 전처리를 수행한 후에 각 센서 노드별 클러스터 변수 값의 변화를 나타내며, 표 3(Tables 3)은 표 1에 도시된 클러스터 변수에 대하여 정규화만 수행한 후에 각 센서 노드별 클러스터 변수 값의 변화를 나타낸다.
- [0062] 표 1에서는 3 개의 센서 노드와 4개의 클러스터 변수들이 고려되었으며, 각 클러스터 변수는 일예로 순서대로 카테고리 A, B, A, C로 분류되는 특성을 가질 수 있다. 표1에서의 클러스터 변수 값은 일예로 제1 메트릭스에 포함된 변수 값(v_{ij})을 의미할 수 있다.
- [0063] 표 2에서, 카테고리 A로 분류되는 클러스터 변수 1(cluster variable 1)과 클러스터 변수 3(cluster variable 3)은 높은 클러스터 변수 값을 가질수록 높은 전처리된 제1 메트릭스 값(v'_{ij})을 유지하는 반면, 클러스터 변수 2(cluster variable 2)와 클러스터 변수 4(cluster variable 4)는 낮은 값 또는 특정 값에 가까운 클러스터 변수 값을 가질수록 높은 전처리된 제1 메트릭스 값(v'_{ij})을 유지함을 확인할 수 있다. 이를 통해 따라서, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법은, 제1 메트릭스에 포함된 클러스터 변수의 종류별로 기정의된 카테고리 고리로 분류하고 그에 대응하는 분류 함수를 적용하는 전처리를 수행한 후 클러스터 헤드를 선정함으로써 클러스터 변수의 특성을 반영한 클러스터 네트워크를 구성할 수 있다. 반면, 표 3에서는 클러스터 변수의 특성에 상관없이 클러스터 변수에 대한 자릿수(order of magnitude)만을 맞추었으며, 표 3에서는 카테고리 B 또는 C에 해당되는 클러스터 변수임에도 불구하고 높은 전처리된 제1 메트릭스 값(v'_{ij})을 유지함을 확인할 수 있다.
- [0064] 다음으로, 유틸리티 값 계산부(21)는 전처리된 제1 메트릭스를 산출(S33)한 이후에, 전처리된 제1 메트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 계산(S34)할 수 있으며, 이를 통해 제2 메트릭스(d)를 산출(S35)할 수 있다.
- [0065] 단계S34에서는, 전처리된 제1 메트릭스에 포함된 각 변수(즉, v'_{ij})들이 서로 떨어져 있는 정도를 결정하기 위해, 유틸리티 값 계산부(21)가 클러스터 변수의 종류별로 각 변수(v'_{ij})들의 분산 값을 구함으로써 제2 메트릭스를 산출할 수 있다. 즉, 제2 메트릭스(d)는 클러스터 변수의 종류별 분산 값들의 행렬을 의미할 수 있다.
- [0066] 보다 자세하게는, 유틸리티 값 계산부(21)는 전처리된 제1 메트릭스를 산출한 이후, 전처리된 제1 메트릭스를 이용하여 각 클러스터 변수의 분산 값(d_i)을 계산할 수 있다. 여기서 분산 값은 각 클러스터 변수들의 값이 서로 떨어져 있는 정도를 의미하는 것으로서, 이는 수학식 1을 통해 계산될 수 있다.

수학식 1

$$d_i = \frac{(|v'_{i1} - \bar{v}_i| + |v'_{i2} - \bar{v}_i| + \dots + |v'_{im} - \bar{v}_i|) / m}{\bar{v}_i}$$

$$\bar{v}_i = \frac{v'_{i1} + v'_{i2} + v'_{i3} + \dots + v'_{im}}{m}$$

- [0067]
- [0068] 여기서 i 는 클러스터 변수의 순차 정보를 의미하고, m 은 센서 노드의 수를 의미하며, $\bar{v}_i$ 는 i 번째 클러스터 변수의 평균을 의미한다. 계산된 분산 값은 제2 메트릭스인 $\mathbf{d} = [d_1, d_2, d_3, \dots, d_n]$ 로 정의될 수 있다.
- [0069] 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는, 불특정 다수의 클러스터 변수들을 고려한 클러스터

헤드 선정에 의해, 각 클러스터 변수들의 가중치를 결정하는 기준으로서 분산 값을 이용한다. 달리 표현하여, 이는 후술할 퍼지 추론 시스템을 적용하는 과정(S36)에서, 제2 매트릭스 d가 퍼지 추론 시스템에서의 퍼지 입력 변수로 사용됨을 의미할 수 있다. 이때, 각 클러스터 변수들의 가중치를 결정하는 기준으로서 분산 값이 이용되는 이유는, 각 노드들이 특정 클러스터 변수에 대해 유사한 값을 가지는 경우(즉, 분산 값이 작은 경우)에는 해당 클러스터 변수는 클러스터 헤드 선정에 큰 영향을 끼칠 수 없고, 만약, 반대로 각 노드들이 특정 클러스터 변수에 대해 유사한 값을 가지지 않는 경우(즉, 분산 값이 큰 경우)에는 해당 클러스터 변수가 클러스터 헤드 선정에 큰 영향을 끼치기 때문이다.

[0070] 한편, 도 7은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 분산 값의 예를 나타낸 도면이다.

[0071] 도 7을 참조하면, 표 4(Table 4)는 표2에 리스트된 전처리된 제1 매트릭스 값을 기반으로 한 분산 값의 계산 결과와 표 3에 리스트된 전처리된 제1 매트릭스 값을 기반으로 한 분산 값의 계산 결과를 보여준다. 여기서, 표2에 기초한 분산 값은 클러스터 변수에 전처리를 수행(Cluster variable preprocessing)한 후에 분산 값을 계산한 결과를 나타내고, 표3에 기초한 분산 값은 클러스터 변수에 정규화(Nomalization)만 수행한 후에 분산 값을 계산한 결과를 나타낸다.

[0072] 전처리를 수행한 경우와 표준화를 사용한 경우 간에 분산 값을 비교한 결과, 카테고리 A로 분류된 클러스터 변수 1과 3(cluster variables 1, 3)은 동일한 분산 값을 보였으나, 카테고리 B와 카테고리 C로 분류된 클러스터 변수 2와 4(cluster variables 2, 4)는 각각 0.21, 0.25 정도로 비교적 큰 차이가 나타났다. 이러한 결과에 의하면, 두 경우(즉 전처리를 수행한 경우와 표준화를 사용한 경우) 모두 클러스터 변수 2와 클러스터 변수 3에 대한 분산 값이 다르기 때문에, 서로 다른 클러스터 헤드가 선정될 가능성이 높다고 할 수 있으나, 두 경우 중에서도 전처리를 수행하는 경우에는 클러스터 변수들의 특성을 고려하기 때문에 이웃 센서 노드 대비 높은 네트워크 역량을 갖는 센서 노드를 클러스터 헤드로 선정할 수 있다.

[0073] 한편, 유틸리티 값 계산부(21)는, 제2 매트릭스를 산출(S35)한 이후에, 제2 매트릭스를 퍼지 추론 시스템(fuzzy inference system, FIS)의 퍼지 입력 변수로 사용(S36)함으로써 퍼지 출력 변수로서 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 행렬로 나타낸 제3 매트릭스(w)를 산출(S37)할 수 있다.

[0074] 즉, 유틸리티 값 계산부(21)는, 제2 매트릭스를 퍼지 추론 시스템의 퍼지 입력 변수로 사용(즉, FIS(d))함으로써, 클러스터 변수의 종류별 가중치 값들의 행렬로 표현되는 제3 매트릭스(w)를 산출할 수 있다. 이는 도 8을 참조하여 보다 쉽게 이해될 수 있다.

[0075] 도 8은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 이용되는 퍼지 추론 시스템을 개략적으로 나타낸 도면이다.

[0076] 도 8을 참조하면, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 센서 노드(20)가 이용하는 퍼지 추론 시스템은, 퍼지화기(fuzzifier), 추론 엔진(inference engine) 및 비퍼지화기(defuzzifier)를 포함할 수 있다. 여기서 추론 엔진은 퍼지 규칙(fuzzy rule)을 포함할 수 있다.

[0077] 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 이용되는 퍼지 추론 시스템은, 산출된 제2 매트릭스(d)를 퍼지 입력 변수로 사용하여 제2 매트릭스(d)에 포함된 각 클러스터 변수의 종류별 가중치 값 값을 나타내는 제3 매트릭스(w)를 퍼지 출력 변수로서 산출할 수 있다.

[0078] 퍼지화기는 퍼지 입력 변수에 대한 소속함수(membership function)를 통해 각 분산 값에 대한 소속도(Degree of membership)를 산출할 수 있다.

[0079] 일례로, 퍼지 추론 시스템의 퍼지 입력 변수에 대한 소속함수는 7개의 언어 변수(linguistic variable)로서 매우 작은(very small, vsmall), 보통 작은(medium small, msmall), 작은(small), 보통(medium), 큰(large), 보통 큰(medium large, mlarge), 매우 큰(very large, vlarge)의 변수를 정의하는 소속함수를 포함할 수 있다.

[0080] 추론 엔진은 퍼지 규칙을 사용하여 퍼지 입력 변수를 퍼지 출력 변수로 맵핑(mapping)할 수 있다.

[0081] 퍼지 추론 시스템의 퍼지 규칙은 일례로 if-then 형태로 정의될 수 있다. 일례로, 퍼지 규칙 중 R₁은 분산 값이 매우 작으면 가중치가 매우 작은 것으로 정의되고, R₂는 분산 값이 보통 작으면 가중치가 보통 작은 것으로 정의되고, R₃은 분산 값이 작으면 가중치가 작은 것으로 정의되고, R₄는 분산 값이 보통이면 가중치가 보통인 것으로 정의되고, R₅는 분산 값이 크면 가중치가 큰 것으로 정의되고, R₆는 분산 값이 보통 크면 가중치가 보통 큰 것으로

로 정의되고, R_7 은 분산 값이 매우 크면 가중치가 매우 큰 것으로 정의될 수 있다.

- [0082] 앞서 말한 바와 같이, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 이용하는 퍼지 추론 시스템은 분산 값을 퍼지 입력 변수로 사용할 수 있으며, 또한, 분산 값이 큰 클러스터 변수가 분산 값이 상대적으로 작은 클러스터 변수에 비해 클러스터 헤드 선정에 대한 기여도가 높도록 퍼지 규칙이 정의될 수 있다.
- [0083] 비퍼지화기는 추론 엔진이 맵핑한 결과를 사용하여 제3 메트릭스인 w 를 산출할 수 있다. 일례로, 퍼지 출력 변수에 대한 소속함수(membership function)는 퍼지 입력 변수에서 정의한 언어 변수의 소속함수와 동일하게 정의될 수 있으며, 이하 중복되는 설명은 생략하기로 한다.
- [0084] 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는, 분산 값을 기준(criterion)으로 사용하여 변수의 가중치 값을 산출하므로, 하나의 퍼지 입력 변수를 위한 하나의 소속함수만을 정의할 수 있다. 따라서, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법은, 클러스터 변수의 수가 늘어나더라도 퍼지 규칙의 수는 늘어나지 않아 시스템의 부담을 줄일 수 있다.
- [0085] 한편, 유틸리티 값 계산부(21)는, 제3 메트릭스를 산출(S37)한 이후에, 전처리된 제1 메트릭스와 제3 메트릭스를 노드 유틸리티 함수에 적용함으로써 유틸리티 값(노드 유틸리티 값, NU 값)을 계산(S38)하고 산출(S39)할 수 있다.
- [0086] 즉, 센서 노드(20)는 전처리된 제1 메트릭스(V')와 제3 메트릭스(w)를 노드 유틸리티 계산식의 입력 변수로 사용(즉, $NU(v'_{ij}, w_i)$)함으로써 유틸리티 값으로서 NU_j 를 산출할 수 있다.
- [0087] 구체적으로, 유틸리티 값 계산부(21)는, 자신과 이웃 센서 노드의 클러스터 변수 정보 값을 종합적으로 고려하여 유틸리티 값을 산출할 수 있으며, 이때, 유틸리티(NU) 값은 전처리 이후의 변수 값(즉, 전처리된 제1 메트릭스(V')의 각 변수(v'_{ij})들의 값)에 가중치 값을 곱한 결과를 모두 더함으로써 산출될 수 있으며, 큰 유틸리티 값을 갖는 센서 노드일수록 클러스터 헤드에 적합한 센서 노드로 판단될 수 있다. 유틸리티 값 계산부(21)는 하기 수학적 식 2를 통해 유틸리티 값을 산출할 수 있다.

수학적 식 2

$$NU_j = w_1 v'_{1j} + w_2 v'_{2j} + w_3 v'_{3j} + \dots + w_n v'_{nj}$$

- [0088]
- [0089] 여기서, NU_j 는 클러스터 변수의 종류가 n 개 있을 때, j 번째 센서 노드의 유틸리티 값을 의미할 수 있다.
- [0090] 이처럼, 센서 노드(20)는 유틸리티 값 계산부(21)를 통해 자신과 이웃 센서 노드들의 클러스터 변수 정보 값의 특성을 고려하여, 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산할 수 있다.
- [0091] 이후, 클러스터 헤드 선정부(22)는 유틸리티 값 계산부(21)를 통해 산출된 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정할 수 있다. 이때, 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 센서 노드가 클러스터 헤드로 선정될 수 있다.
- [0092] 한편, 본원의 일 실시예에 따른 센서 노드(20)는 통신부(미도시)를 포함할 수 있으며, 통신부는 다른 이웃 센서 노드들과 클러스터 헤드 선정 및 클러스터 형성을 위한 관련 메시지(예를 들어, 헬로 메시지, 클러스터 헤드 알림 메시지, 가입 요청 메시지, 가입 응답 메시지 등)를 송수신할 수 있다.
- [0093] 자세히 살펴보면, 클러스터 헤드 선정부(22)는, 자신(즉, 센서 노드(20))과 이웃 센서 노드들의 유틸리티 값을 비교한 결과, 자신의 유틸리티 값이 가장 큰 경우, 자신을 클러스터 헤드로 선정할 수 있다. 이후, 센서 노드(20)는 통신부를 통해 센서 노드(20) 자신이 클러스터 헤드임을 알리는 클러스터 헤드 알림 메시지(CH-advertisement message)를 이웃 센서 노드들에게 브로드캐스팅 할 수 있다. 이때, 센서 노드(20)는 클러스터 헤드 알림 메시지에 자신의 유틸리티 변수 값(NU_j), 자신이 클러스터 헤드인지 여부를 나타내는 클러스터 헤드 변수 값(CH_Flag)과 자신의 ID를 포함시켜 브로드캐스팅 할 수 있다. 이후 센서 노드(20)는 이웃 센서 노드로부터 가입 요청 메시지(join-request message)를 수신하기 위해 가입 요청 수신 시간을 설정하고, 설정된 가입 요청 수신 시간이 만료될 때까지 기다릴 수 있다. 가입 요청 수신 시간이 만료된 이후, 센서 노드(20)는 가입 요청 수신 시간 동안에 가입 요청 메시지를 송신한 이웃 센서 노드들의 ID를 확인하고, 가입 요청 메시지를 송신한

이웃 센서 노드들에게 클러스터 멤버로 가입되었음을 알리는 가입 응답 메시지(join-response message)를 통신부를 통해 전송할 수 있다.

[0094] 한편, 클러스터 헤드 선정부(22)는, 자신(즉, 센서 노드(20))과 이웃 센서 노드들의 유틸리티 값을 비교한 결과, 자신의 유틸리티 값(NU_j) 보다 더 큰 유틸리티 값(NU_{N_j})이 존재하는 경우, 센서 노드(20)는 자신을 클러스터 멤버로 설정한 이후 자신의 유틸리티 값 보다 더 큰 유틸리티 값을 가지는 센서 노드인 클러스터 헤드로부터 클러스터 헤드 알림 메시지를 수신하기 위해, 클러스터 헤드 알림 수신 시간을 설정하고 설정된 시간이 만료될 때까지 기다릴 수 있다. 클러스터 헤드 알림 수신 시간 내에 클러스터 헤드 알림 메시지를 수신한 경우, 클러스터 멤버인 센서 노드(20)는 클러스터 헤드 알림 메시지를 전송한 클러스터 헤드로 가입 요청 메시지를 전송할 수 있다. 이후, 클러스터 헤드로부터 가입 응답 메시지를 수신하기 위해 센서 노드(20)는 가입 응답 수신 시간을 설정하고 기다릴 수 있다.

[0095] 만약에, 센서 노드(20)가 클러스터 헤드 알림 수신 시간 내에 2 개 이상의 클러스터 헤드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 센서 노드(20)는 2개 이상의 클러스터 헤드 중 가장 큰 유틸리티 값을 가지는 센서 노드를 자신의 클러스터 헤드로서 선정할 수 있다.

[0096] 또한, 센서 노드(20)는 이웃 센서 노드들과 메시지 교환을 위해 설정한 시간(예를 들어, 클러스터 헤드 알림 수신 시간, 가입 요청 수신 시간, 가입 응답 수신 시간 등) 동안에 어떤 메시지도 수신하지 못한 경우, 자신을 클러스터 헤드로 선정한 후, 싱크 노드(미도시)와 직접 통신을 수행할 수 있다.

[0097] 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서는, 복수의 센서 노드(1, 2, 3, 4, 5) 각각이 자신의 유틸리티 값을 계산한 이후, 복수의 센서 노드(1, 2, 3, 4, 5) 각각이 자신의 유틸리티 값과 이웃 센서 노드들의 유틸리티 값을 비교함으로써 분산적으로 클러스터 헤드를 선정할 수 있다.

[0098] 앞서 설명한 내용에 기초하여, 이하에서는 도 9를 참조하여 복수의 센서 노드가 클러스터 헤드를 선정하는 방법에 대해 설명하기로 한다.

[0099] 도 9는 본원의 일 실시예에 따른 클러스터 헤드 선정 방법을 통해 제안되는 변수 분류 클러스터링 알고리즘(VCCA)의 클러스터링 동작을 나타낸 도면이다.

[0100] 도 9를 참조하면, 클러스터링 동작은 노드 유틸리티 값 계산 단계(Node utility calculation phase)와 클러스터링 단계(Clustering phase)를 포함할 수 있다. 노드 유틸리티 값 계산 단계에서는, 복수의 센서 노드 각각이 유틸리티 값을 산출할 수 있다. 이때, 복수의 센서 노드 각각은 앞서 센서 노드(20)가 유틸리티 값 계산부(21)를 통해 유틸리티 값을 계산한 방법과 동일한 논리로 유틸리티 값을 산출할 수 있다. 클러스터링 단계에서는 복수의 센서 노드 각각이 자신의 유틸리티 값과 다른 이웃한 센서 노드의 유틸리티 값을 비교함으로써 분산적으로 클러스터 헤드를 선정할 수 있다. 이때, 복수의 센서 노드 각각은 앞서 센서 노드(20)가 클러스터 헤드 선정부(22)를 통해 클러스터 헤드를 선정하는 방법과 동일한 논리로 클러스터 헤드를 선정할 수 있다. 클러스터링 단계에서는 복수의 센서 노드 각각이, 복수의 센서 노드 각각에 의하여 계산된 유틸리티 값에 기초하여 클러스터 헤드를 선정하고, 클러스터를 형성할 수 있다.

[0101] 유틸리티 값 계산을 수행하기 이전에, 복수의 센서 노드 각각은, 두 개의 변수로서 NU_j 와 CH_Flag 를 0으로 초기화할 수 있다. 여기서, NU_j 는 j번째 센서 노드의 유틸리티 값을 나타내고, CH_Flag 는 자신이 클러스터 헤드인지 여부를 나타내는 것으로서, CH_Flag 값이 1이면 클러스터 헤드임을 의미하고, CH_Flag 값이 0이면 클러스터 멤버임을 의미할 수 있다.

[0102] 복수의 센서 노드 각각은, 두 개의 변수를 초기화한 이후 헬로 메시지 교환을 통해 이웃 센서 노드들의 ID와 클러스터 변수 정보 값을 수신하고, 제1 메트릭스를 생성하여 저장할 수 있다. 이후 복수의 센서 노드 각각은 클러스터 변수 정보 값에서 클러스터 변수의 특성에 따라 특정 카테고리로 분류하여 분류 함수를 적용하는 클러스터 변수 전처리 과정을 수행할 수 있으며, 이를 통해 전처리된 제1 메트릭스를 산출할 수 있다. 이후 복수의 센서 노드 각각은 전처리된 제1 메트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 메트릭스를 산출할 수 있다. 이후, 복수의 센서 노드 각각은 제2 메트릭스를 퍼지 추론 시스템의 퍼지 입력 변수로 사용함으로써 클러스터 변수의 종류 별로 가중치 값들의 행렬을 나타내는 제3 메트릭스를 산출할 수 있다. 이후 복수의 센서 노드 각각은 전처리된 제1 메트릭스와 가중치 행렬인 제3 메트릭스를 노드 유틸리티 계산식(즉, 수학식 2)의 입력 변수로 사용함으로써 노드 유틸리티 값을 산출할 수 있다.

- [0103] 이후, 복수의 센서 노드 각각은, 클러스터링 단계에서, 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교하여 클러스터 헤드를 선정할 수 있다. 가장 큰 유틸리티 값을 가진 센서 노드는 자신의 CH_Flag 변수를 1로 설정함으로써 자신을 클러스터 헤드로서 선정할 수 있다.
- [0104] 이후, 클러스터 헤드인 센서 노드는, 자신의 유틸리티 변수 값, 자신이 클러스터 헤드인지 여부를 나타내는 클러스터 헤드 변수 값(CH_Flag) 및 자신의 ID가 포함된 클러스터 헤드 알림 메시지를 이웃 센서 노드에게 브로드캐스팅하여 자신이 클러스터 헤드임을 알릴 수 있다. 이후 클러스터 헤드인 센서 노드는 이웃 센서 노드로부터 가입 요청 메시지를 수신하기 위해 가입 요청 수신 시간을 설정하고 기다릴 수 있다. 이후, 클러스터 헤드인 센서 노드는, 가입 요청 수신 시간이 만료된 이후, 가입 요청 메시지를 보낸 이웃 센서 노드들의 ID를 확인하고, 해당 이웃 센서 노드들에게 클러스터 멤버로 가입 되었음을 알리는 가입 응답 메시지를 전달할 수 있다.
- [0105] 이때, 만약 자신의 유틸리티 값보다 큰 유틸리티 값이 존재하는 것으로 판단되는 경우, 클러스터 헤드인 센서 노드는 자신의 CH_Flag 변수 값을 0으로 설정하여 자신을 클러스터 멤버로 설정할 수 있다. 이후, 클러스터 멤버로 설정된 센서 노드는, 주위에 존재하는 클러스터 헤드로부터 클러스터 헤드 알림 메시지를 수신하기 위해 클러스터 헤드 알림 수신 시간을 설정하고 기다릴 수 있다. 이후 클러스터 헤드 알림 메시지를 수신하면, 클러스터 멤버로 설정된 센서 노드는 가입 요청 메시지를 클러스터 헤드로 전송하고, 가입 응답 메시지를 수신하기 위해 가입 응답 수신 시간을 설정할 수 있다.
- [0106] 만약 이때, 클러스터 멤버로 설정된 센서 노드가, 2개 이상의 클러스터 헤드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 더 큰 유틸리티 값을 가진 센서 노드를 자신의 클러스터 헤드로 선택할 수 있다. 또한, 복수의 센서 노드 각각은 클러스터링을 위한 메시지들(예를 들어, 클러스터 헤드 알림 메시지, 가입 요청 메시지, 가입 응답 메시지 등)을 수신하기 위해 설정한 시간 동안에 어떠한 메시지도 수신하지 못한 경우, 자신을 클러스터 헤드로 선정하고, 싱크 노드(미도시)와 직접 통신을 수행할 수 있다.
- [0107] 이처럼, 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에서 제안하는 VCCA 기법은, 클러스터 헤드 선정을 위해, 다수의 클러스터 변수들을 특성에 따라 분류하고, 규칙 기반의 변수 맵핑을 통해 결과 값을 산출하는 퍼지 추론 시스템을 사용함으로써, 네트워크 역량을 의미하는 가장 큰 노드 유틸리티 값을 갖는 센서 노드를 클러스터 헤드로 선정할 수 있다. 이를 통해 본원의 일 실시예에 따른 클러스터 헤드 선정 방법은, 클러스터 헤드 선정 과정의 낮은 복잡도와 클러스터 변수에 대한 고확장성을 유지할 수 있다. 이러한 본원의 일 실시예에 따른 클러스터 헤드 선정 방법은, 종래의 클러스터 헤드 선정 기술 대비 처리율, 종단간 지연시간, 네트워크 수명 및 에너지 소비 측면에서 더 나은 성능을 나타낼 수 있다.
- [0108] 이하에서는 상기에 자세히 설명된 내용을 기반으로, 본원의 동작 흐름을 간단히 살펴보기로 한다.
- [0109] 도 10은 본원의 일 실시예에 따른 클러스터 헤드 선정 방법에 대한 동작 흐름도이다.
- [0110] 도 10에 도시된 클러스터 헤드 선정 방법은 앞서 설명된 센서 노드(20) 및 복수의 센서 노드(1, 2, 3, 4, 5) 각각에 의하여 수행될 수 있다. 따라서, 이하 생략된 내용이라고 하더라도 센서 노드(20) 및 복수의 센서 노드(1, 2, 3, 4, 5)에 대하여 설명된 내용은 도 10에도 동일하게 적용될 수 있다.
- [0111] 도 10을 참조하면, 우선, 단계S1010 이전에, 복수의 센서 노드 각각은, 자신의 유틸리티 변수 값과 클러스터 헤드인지 여부를 나타내는 클러스터 헤드 변수 값을 초기화시킬 수 있다.
- [0112] 다음으로, 단계S1010에서는 복수의 센서 노드 각각이 자신의 네트워크 역량을 나타내는 유틸리티 값을 계산할 수 있다.
- [0113] 단계S1010에서는, 복수의 센서 노드 각각이, 자신과 이웃 센서 노드의 클러스터 변수 정보 값을 이용하여 제1 메트릭스를 생성할 수 있다. 이때, 복수의 센서 노드 각각은, 클러스터 변수 정보 값으로서 클러스터 변수의 종류의 수를 고려한 클러스터 변수 각각의 순차 정보 및 이웃 센서 노드의 수를 고려한 이웃 센서 노드 각각의 순차 정보를 이용하여 제1 메트릭스를 생성할 수 있다. 이후, 복수의 센서 노드 각각은, 제1 메트릭스에 전처리를 수행함으로써 전처리된 제1 메트릭스를 산출할 수 있다. 이때, 복수의 센서 노드 각각은, 제1 메트릭스에 포함된 클러스터 변수의 종류별로 기정의된 카테고리 분류하고 기정의된 카테고리별로 대응하는 분류 함수를 적용함으로써 전처리된 제1 메트릭스를 산출할 수 있다. 이후, 복수의 센서 노드 각각은, 전처리된 제1 메트릭스에 기초하여 클러스터 변수의 종류별로 분산 값을 구함으로써 제2 메트릭스를 산출하고, 제2 메트릭스를 퍼지 추론 시스템의 퍼지 입력 변수로 사용함으로써 퍼지 출력 변수로서 클러스터 변수의 종류별 분산 값에 따른 가중치 값을 나타내는 제3 메트릭스를 산출하며, 전처리된 제1 메트릭스와 제3 메트릭스를 이용하여 유틸리티 값을 계

산할 수 있다.

- [0114] 다음으로, 단계S1020에서는, 복수의 센서 노드 각각이 자신의 유틸리티 값과 이웃 센서 노드의 유틸리티 값을 비교함으로써 클러스터 헤드를 선정할 수 있다.
- [0115] 이때, 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 제1 센서 노드가 클러스터 헤드로 선정될 수 있다.
- [0116] 또한, 단계S1020에서, 복수의 센서 노드 중 상기 클러스터 헤드로 선정된 제1 센서 노드는, 자신이 클러스터 헤드임을 알리는 클러스터 헤드 알림 메시지를 주변 이웃 센서 노드로 브로드캐스팅하고, 주변 이웃 센서 노드로부터 가입 요청 메시지를 수신하기 위한 가입 요청 시간을 설정하며, 설정된 가입 요청 시간이 만료된 것으로 판단되는 경우 가입 요청 메시지를 송신한 주변 이웃 센서 노드로 클러스터 멤버에 가입되었음을 알리는 가입 응답 메시지를 전송할 수 있다.
- [0117] 또한, 단계S1020에서는, 복수의 센서 노드 중 클러스터 헤드로 선정된 제1 센서 노드의 유틸리티 값 보다 큰 유틸리티 값을 가지는 제2 센서 노드가 존재하는 경우, 제1 센서 노드는 자신을 클러스터 멤버로 설정하고, 기설정된 클러스터 헤드 알림 수신 시간 동안에 제2 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 가입 요청 메시지를 제2 센서 노드로 전송할 수 있다. 이때, 자신을 클러스터 멤버로 설정한 제1 센서 노드가 기설정된 클러스터 헤드 알림 수신 시간 동안에 복수의 센서 노드로부터 클러스터 헤드 알림 메시지를 수신한 경우, 제1 센서 노드는 복수의 센서 노드 중 가장 큰 유틸리티 값을 가지는 센서 노드를 클러스터 헤드로서 선택할 수 있다.
- [0118] 상술한 설명에서, 단계 S1010 내지 S1020은 본원의 구현예에 따라서, 추가적인 단계들로 더 분할되거나, 더 적은 단계들로 조합될 수 있다. 또한, 일부 단계는 필요에 따라 생략될 수도 있고, 단계 간의 순서가 변경될 수도 있다.
- [0119] 본원의 일 실시 예에 따른 클러스터 헤드 선정 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 본 발명을 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 본 발명의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.
- [0120] 또한, 전술한 클러스터 헤드 선정 방법은 기록 매체에 저장되는 컴퓨터에 의해 실행되는 컴퓨터 프로그램 또는 애플리케이션의 형태로도 구현될 수 있다.
- [0121] 전술한 본원의 설명은 예시를 위한 것이며, 본원이 속하는 기술분야의 통상의 지식을 가진 자는 본원의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.
- [0122] 본원의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본원의 범위에 포함되는 것으로 해석되어야 한다.

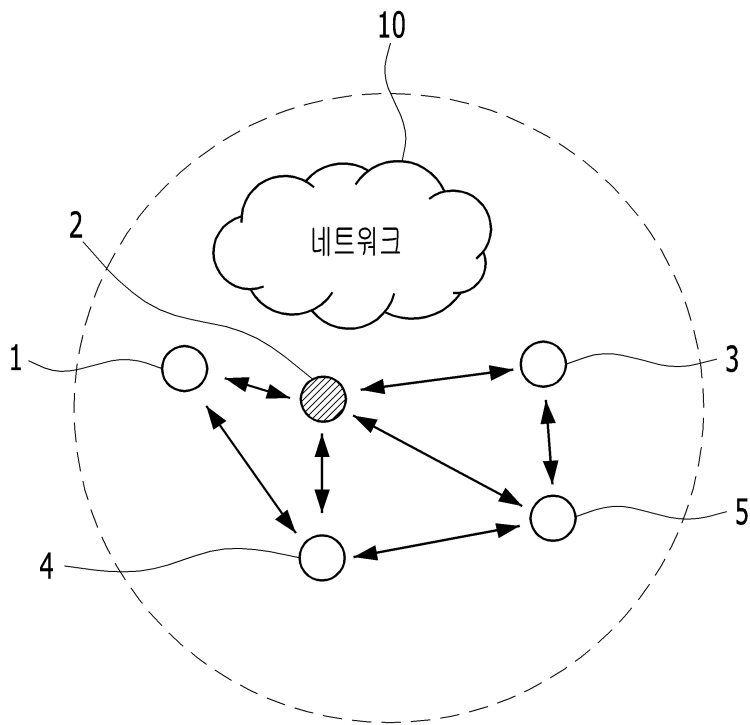
부호의 설명

- [0123] 1, 2, 3, 4, 5, 20: 센서 노드
21: 유틸리티 값 계산부
22: 클러스터 헤드 선정부

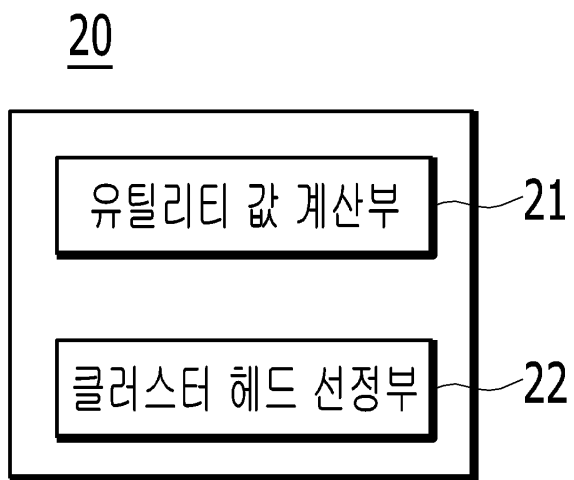
10: 네트워크

도면

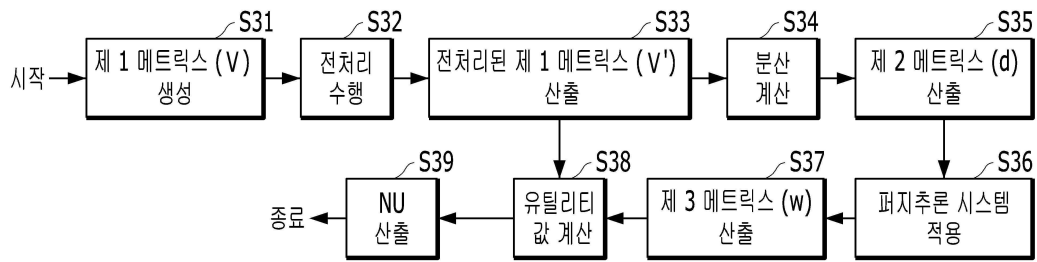
도면1



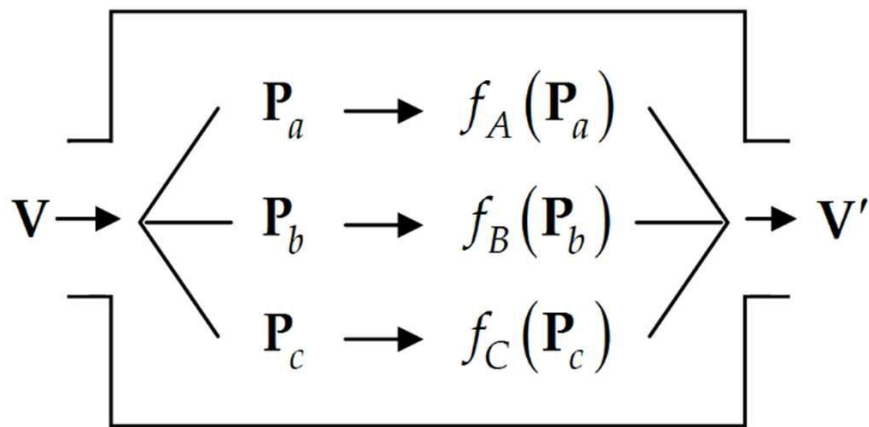
도면2



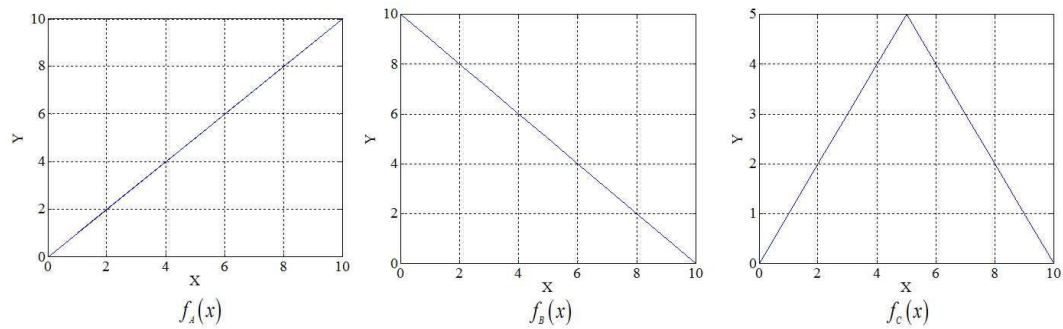
도면3



도면4



도면5



도면6

Table 1

Node ID	Cluster variable 1	Cluster variable 2	Cluster variable 3	Cluster variable 4
1	65	35.6	3.4	150.6
2	90	32.1	3.34	123.6
3	35	29.6	3.12	178.2

Table 2

Node ID	Cluster variable 1	Cluster variable 2	Cluster variable 3	Cluster variable 4
1	0.65	0.11	0.34	0.247
2	0.90	0.1975	0.334	0.382
3	0.35	0.26	0.312	0.109

Table 3

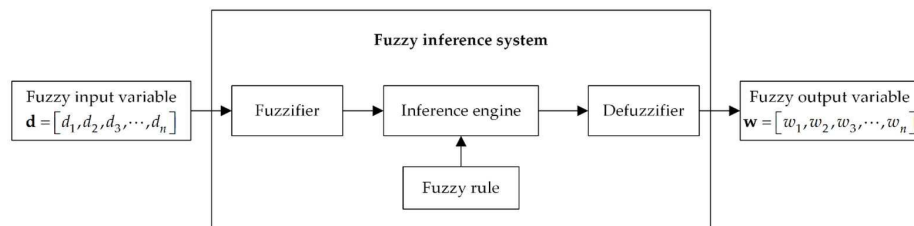
Node ID	Cluster variable 1	Cluster variable 2	Cluster variable 3	Cluster variable 4
1	0.65	0.11	0.34	0.247
2	0.90	0.1975	0.334	0.382
3	0.35	0.26	0.312	0.109

도면7

Table 4

	Cluster variable 1	Cluster variable 2	Cluster variable 3	Cluster variable 4
Cluster variable preprocessing	0.333333	0.279001	0.033807	0.371274
Normalization	0.333333	0.065091	0.033807	0.121132

도면8



도면9

Algorithm 1 Clustering operation

Node utility calculation phase:

- 1: **INITIALIZE** NU_i to 0, CH_Flag to 0
 - 2: Broadcast a hello message to neighbors
 - 3: Construct a neighbor table
 - 4: **FOR** each node, j , $j \in [1, n]$ // n is the number of nodes
 - 5: $V = [v_{ij}]$ // Cluster variable matrix creation
 - 6: $V' = [v'_{ij}] = \text{Preprocess}(V)$ // Cluster variable preprocessing
 - 7: $w = \text{FIS}(d) = \text{FIS}(\text{Dispersion}(V))$ // Dispersion calculation and Fuzzy inference system
 - 8: $NU_i = \text{NU}(v'_{ij}, w_i)$ // Node utility calculation
 - 9: **ENDFOR**
-

Clustering phase:

- 10: **FOR** each node, j , $j \in [1, n]$
 - 11: **IF** all neighbors of j , N_j , $N_j \in [1, n-1]$: $NU_i \geq NU_{N_i}$
 - 12: $CH_Flag \leftarrow 1$ // set as a cluster head
 - 13: Send a CH-advertisement message
 - 14: Start join-request timer
 - 15: **IF** receive join-request message
 - 16: Check node ID, send join-response message
 - 17: **ENDIF**
 - 18: **ELSE**
 - 19: $CH_Flag \leftarrow 0$ // set as a cluster member
 - 20: Start CH-advertisement timer
 - 21: **IF** receive CH-advertisement message
 - 22: Check CH ID, send join-request message
 - 23: Start join-response timer
 - 24: **IF** do not receive join-response message
 - 25: $CH_Flag \leftarrow 1$
 - 26: **ENDIF**
 - 27: **ELSE**
 - 28: $CH_Flag \leftarrow 1$
 - 29: **ENDIF**
 - 30: **ENDIF**
 - 31: **ENDFOR**
-

도면10

