(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2017-0113937
(43) 공개일자 2017년10월13일

(51) 국제특허분류(Int. Cl.)
H04L 12/803 (2013.01)
(52) CPC특허분류
H04L 47/125 (2013.01)
G06N 3/126 (2013.01)
(21) 출원번호 10-2016-0037906
(22) 출원일자 2016년03월29일
심사청구일자 2016년03월29일

(71) 출원인
인하대학교 산학협력단
인천광역시 남구 인하로 100, 인하대학교 (용현동)
(72) 발명자
권구인
인천광역시 연수구 해돋이로120번길 16 송도풍림아이원2단지아파트 203동 301호
강승보
경상북도 포항시 남구 지곡로 294 223동 902호 (지곡동,효자그린2차아파트)
(74) 대리인
이은철

전체 청구항 수 : 총 5 항

(54) 발명의 명칭 **유전자 알고리즘을 이용한 부하 분배 장치 및 그 방법**

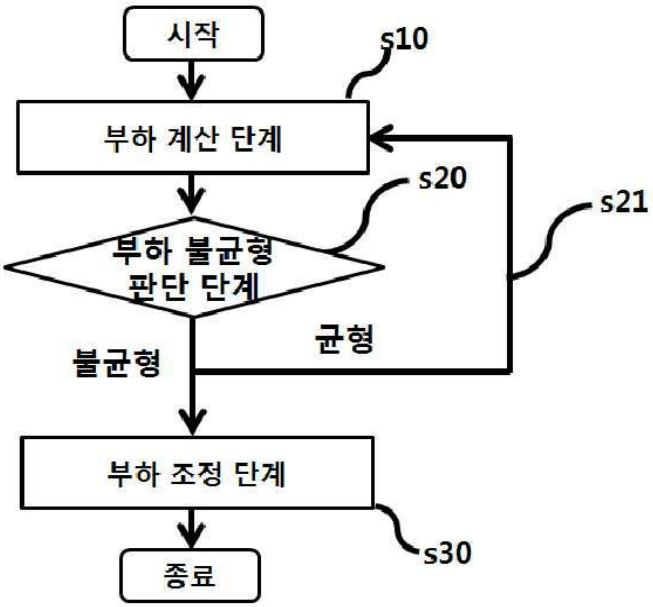
(57) 요 약

본 발명은 유전자 알고리즘을 이용한 SDN상의 부하 분배 장치 및 방법을 제시하고 있다.

본 발명에 따른 유전자 알고리즘을 이용한 SDN상의 부하 분배 장치는 SDN상의 각 컨트롤러에 부여되는 부하의 양을 및 상기 부하의 양의 표준편차를 산출하고 상기 산출된 값을 부하 불균형 판단부에 제공하는 부하 계산부;

(뒷면에 계속)

대 표 도 - 도3



상기 부하의 양 및 표준편차를 토대로 상기 각 컨트롤러에 부여되는 부하의 양이 균형한지 불균형한지 판단하여 상기 판단결과를 부하 조정부에 제공하는 부하 불균형 판단부;

상기 부하 불균형 판단부에서 상기 각 컨트롤러에 부여되는 부하의 양이 불균형하다고 판단된 경우 상기 각 컨트롤러에 부여되는 부하의 양을 조정하는 부하 조정부;를 포함할 수 있다.

본 발명에 의하면, 다수의 컨트롤러를 사용하는 SDN 환경에서 각 컨트롤러에 주어지는 부하의 양을 수치로 표현할 수 있다.

또한, 부하간 표준편차를 이용하여 컨트롤러간 부여되는 부하량의 불균형 즉 부하 불균형을 감지할 수 있으며, 부하 분배 알고리즘을 통하여 부하가 과도하게 주어진 컨트롤러의 부하의 양을 조정할 수 있다.

최종적으로 부하 불균형에서 기인한 컨트롤러와 스위치간 연결 실패 현상을 방지함으로서 네트워크의 신뢰성을 보장할 수 있다.

(52) CPC특허분류
H04L 43/16 (2013.01)
H04L 67/1002 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 2015R1A2A2A01003501
부처명 미래창조과학부
연구관리전문기관 한국연구재단
연구사업명 중견연구자지원사업
연구과제명 NFV 환경의 자원할당 최적화를 위한 네트워크 서비스 체인 연구
기 여 율 1/1
주관기관 인하대학교 산학협력단
연구기간 2015.05.01 ~ 2018.04.30

명 세 서

청구범위

청구항 1

SDN상의 각 컨트롤러에 부여되는 부하의 양을 및 상기 부하의 양의 표준편차를 산출하고 상기 산출된 값을 부하 불균형 판단부에 제공하는 부하 계산부;

상기 부하의 양 및 표준편차를 토대로 상기 각 컨트롤러에 부여되는 부하의 양이 균형한지 불균형한지 판단하여 상기 판단결과를 부하 조정부에 제공하는 부하 불균형 판단부;

상기 부하 불균형 판단부에서 상기 각 컨트롤러에 부여되는 부하의 양이 불균형하다고 판단된 경우 상기 각 컨트롤러에 부여되는 부하의 양을 조정하는 부하 조정부;

를 포함하는 것을 특징으로 하는 유전자 알고리즘을 이용한 부하 분배 장치.

청구항 2

제1항에 있어서, 상기 부하 불균형 판단부는,

상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하고, 상기 산출된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단하는 것을 특징으로 하는 유전자 알고리즘을 이용한 부하 분배 장치.

청구항 3

제2항에 있어서, 상기 부하 조정부는,

상기 부하 불균형 판단부에서 부하 불균형으로 판단된 경우 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하를 상기 기 지정된 임계값 미만인 표준편차를 갖는 컨트롤러에 분배하는 것을 특징으로 하는 유전자 알고리즘을 이용한 부하 분배 장치.

청구항 4

a)부하 계산부가 SDN상의 각 컨트롤러에 부여되는 부하의 양 및 상기 부하의 양의 표준편차를 산출하고 부하 불균형 판단부에 제공하는 단계;

b)상기 부하 불균형 판단부가 상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하고, 상기 산출된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단하여 판단결과를 부하 분배부에 제공하는 단계;

c)상기 b)단계에서 부하 불균형으로 판단되면, 상기 부하 분배부가 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하를 상기 기 지정된 임계값 미만의 표준편차를 갖는 컨트롤러에 분배하는 단계;

를 포함하는 유전자 알고리즘을 이용한 부하 분배 방법.

청구항 5

제4항에 있어서, b)단계 이후에,

c-1)상기 b)단계에서 부하 균형으로 판단되면 상기 다시 a)단계로 돌아가는 단계;를 포함하는 것을 특징으로 하

는 유전자 알고리즘을 이용한 부하 분배 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 유전자 알고리즘을 이용한 부하 분배 장치 및 그 방법으로, 보다 상세하게는 SDN상의 각 컨트롤러에 부여되는 부하의 양이 불균형한 경우 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 부하를 분산시키는 기술에 관한 것이다.

배 경 기 술

[0002] 기존의 컴퓨터 네트워크는 매우 복잡하고 관리에 어려움이 따르는 구조를 이루어 왔다. 이러한 네트워크는 라우터, 스위치, 미들박스(방화벽, 침입 탐지 시스템 등)등의 많은 장비들로 구성되어 있으며 라우터와 스위치는 복잡하고 분산된 제어 소프트웨어이며 장비 제조사인 벤더 중심의 폐쇄적인 형태를 띠고 있다.

[0003] 네트워크 관리자는 표준화되지 않은 벤더별로 상이한 장비와 그에 특화된 관리 도구를 사용할 수밖에 없다.

[0004] 폐쇄적인 특성으로 인해 현재의 네트워크는 새로운 서비스를 도입하기 힘들뿐만 아니라 분산된 구조로 인해 자동화된 중앙 관리에도 어려움을 겪고 있다.

[0005] 이러한 문제점들을 극복하기 위해 개방형 인터페이스를 제공하는 SDN(Software Defined Network) 기술과 그를 지원하는 프로토콜인 OpenFlow가 제안되었다.

[0006] 새로운 네트워크 기술 분야인 SDN(Software Defined Network)에서 스위치 및 라우터들의 라우팅 테이블을 관리하는 컨트롤러 장비를 사용한다.

[0007] 즉 컨트롤러는 다수의 스위치들의 라우팅 테이블의 업데이트를 위하여 다수의 스위치들을 관리한다.

[0008] SDN 기술은 제어 평면(control plane)과 전송 평면(data plane)을 분리하는데 핵심을 둔다.

[0009] 분리된 두 평면 중 제어 평면은 소프트웨어로 이루어진 컨트롤러(controller)가 담당하며 전송 평면은 하드웨어인 스위치가 담당한다.

[0010] 네트워크 관리자는 스위치의 행동을 제어하는 제어 평면, 즉 컨트롤러를 직접 프로그래밍 함으로서 라우팅 연산 및 부하 분배 등 필요에 맞는 서비스를 네트워크에 적용시킬 수 있다. 현재의 컨트롤러(Floodlight, Opendaylight 등)들은 라우팅 연산뿐만 아니라 토폴로지 처리(topology processing), 방화벽, VPN 등 많은 서비스들을 제공한다.

[0011] SDN 환경에서 스위치는 컨트롤러로부터 전송해야할 패킷에 대한 플로우 테이블을 받기 위해 packet_in 메시지와 함께 해당 패킷을 상기 컨트롤러에게 전송한다.

[0012] 상기 컨트롤러는 packet_in 메시지를 수신하면 해당 패킷에 대한 라우팅 연산을 실시하여 플로우 테이블을 구성한다.

[0013] 상기 플로우 테이블은 packet_out 메시지와 함께 상기 스위치로 전송되고 상기 스위치는 이에 따라 패킷을 전송하게 된다.

[0014] 이 때 상기 컨트롤러가 수신하는 packet_in 메시지의 양과 각종 서비스에 필요한 연산을 처리하는데 있어 가능한 처리량 즉 상기 컨트롤러에 부여되는 부하량이 기 설정된 상기 컨트롤러가 처리 가능한 부하량을 넘어서게 되면 상기 컨트롤러가 스위치가 요청하는 사항에 대하여 처리하지 못하게 되는 현상이 발생한다.

[0015] 따라서 다수의 컨트롤러를 이용할 때 상기 다수의 컨트롤러에 부여되는 부하가 상기 다수의 컨트롤러간에 균형있게 분배되어야 할 필요성이 있다.

발명의 내용

해결하려는 과제

[0016] 따라서 본 발명은, SDN환경에서 다수의 컨트롤러에 부여되는 부하량의 불균형이 일어난 경우 유전자 알고리즘을 적용한 부하 분배 알고리즘을 통해 부하를 조정하여 SDN 환경에서 스위치가 컨트롤러에 요청하는 사항에 대하여

상기 컨트롤러가 처리하지 못하는 현상을 방지하고자 한다.

과제의 해결 수단

[0017] 본 발명에 따른 유전자 알고리즘을 이용한 부하 분배 장치는,

[0018] SDN상의 각 컨트롤러에 부여되는 부하의 양을 및 상기 부하의 양의 표준편차를 산출하고 상기 산출된 값을 부하 불균형 판단부에 제공하는 부하 계산부;

[0019] 상기 부하의 양 및 표준편차를 토대로 상기 각 컨트롤러에 부여되는 부하의 양이 균형한지 불균형한지 판단하여 상기 판단결과를 부하 조정부에 제공하는 부하 불균형 판단부;

[0020] 상기 부하 불균형 판단부에서 상기 각 컨트롤러에 부여되는 부하의 양이 불균형하다고 판단된 경우 상기 각 컨트롤러에 부여되는 부하의 양을 조정하는 부하 조정부;를 포함할 수 있다.

[0021] 바람직하게는, 상기 부하 불균형 판단부는,

[0022] 상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하고, 상기 산출된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단할 수 있다.

[0023] 바람직하게는, 상기 부하 조정부는,

[0024] 상기 부하 불균형 판단부에서 부하 불균형으로 판단된 경우 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하를 상기 기 지정된 임계값 미만인 표준편차를 갖는 컨트롤러에 분배할 수 있다.

[0025] 본 발명에 따른 유전자 알고리즘을 이용한 부하 분배 방법은,

[0026] a)부하 계산부가 SDN상의 각 컨트롤러에 부여되는 부하의 양 및 상기 부하의 양의 표준편차를 산출하고 부하 불균형 판단부에 제공하는 단계;

[0027] b)상기 부하 불균형 판단부가 상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하고, 상기 산출된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단하여 판단결과를 부하 분배부에 제공하는 단계;

[0028] c)상기 b)단계에서 부하 불균형으로 판단되면, 상기 부하 분배부가 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하를 상기 기 지정된 임계값 미만의 표준편차를 갖는 컨트롤러에 분배하는 단계;를 포함할 수 있다.

[0029] 또한, 상기 b)단계 이후,

[0030] c-1)상기 b)단계에서 부하 균형으로 판단되면 상기 다시 a)단계로 돌아가는 단계;를 포함할 수 있다.

발명의 효과

[0031] 본 발명에 의하면, 다수의 컨트롤러를 사용하는 SDN 환경에서 각 컨트롤러에 주어지는 부하의 양을 수치로 표현할 수 있다.

[0032] 또한, 부하간 부하량의 표준편차를 이용하여 컨트롤러간 부여되는 부하량의 불균형 즉 부하 불균형을 감지할 수 있으며, 부하 분배 알고리즘을 통하여 부하가 과도하게 주어진 컨트롤러의 부하의 양을 조정할 수 있다.

[0033] 최종적으로 부하 불균형에서 기인한 컨트롤러와 스위치간 연결 실패 현상을 방지함으로서 네트워크의 신뢰성을 보장 할 수 있다.

도면의 간단한 설명

[0034] 본 명세서에서 첨부되는 다음의 도면들은 본 발명의 바람직한 실시 예를 예시하는 것이며, 후술하는 발명의 상세한 설명과 함께 본 발명의 기술사상을 더욱 이해시키는 역할을 하는 것이므로, 본 발명은 그러한 도면에 기재된 사항에만 한정되어 해석되어서는 아니 된다.

도 1은 본 발명에 따른 유전자 알고리즘을 이용한 부하 분배 장치 및 방법에 사용되는 수식을 정리한 표를 나타낸 도이다.

도 2는 본 발명의 실시 예에 따른 유전자 알고리즘을 이용한 부하 분배 장치의 세부적인 구성을 나타낸 도이다.

도 3은 본 발명의 실시 예에 따른 유전자 알고리즘을 이용한 부하 분배 방법의 과정을 나타낸 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0035] 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시 예를 예시하는 첨부 도면 및 도면에 기재된 내용을 참조하여야 한다.

[0036] 본 발명의 실시 예에서 제시되는 특정한 구조 내지 기능적 설명들은 단지 본 발명의 개념에 따른 실시 예를 설명하기 위한 목적으로 예시된 것으로, 본 발명의 개념에 따른 실시 예들은 다양한 형태로 실시될 수 있다.

[0037] 또한 본 명세서에 설명된 실시 예들에 한정되는 것으로 해석되어서는 아니 되며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경물, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0038] 또한, 본 명세서에서 사용되는 제1, 제2 등과 같이 서수를 포함하는 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다.

[0039] 상기 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제1 구성요소는 제2 구성 요소로 명명될 수 있고, 유사하게 제2 구성 요소도 제1 구성 요소로 명명될 수 있다.

[0040] 본 명세서에서 사용하는 용어는 단지 특정한 실시 예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다.

[0041] 이하 첨부된 도면을 참조로 본 발명의 여러 실시 예들을 상세히 설명하기 전에, 다음의 상세한 설명에 기재되거나 도면에 도시된 구성요소들의 구성 및 배열들의 상세로 그 응용이 제한되는 것이 아니라는 것을 알 수 있을 것이다. 본 발명은 다른 실시예들로 구현되고 실시될 수 있고 다양한 방법으로 수행될 수 있다.

[0042] 또한, "제 1(first)", "제 2(second)"와 같은 용어는 설명을 위해 본 발명 및 첨부 청구항들에 사용되고 상대적인 중요성 또는 취지를 나타내거나 의미하는 것으로 의도되지 않는다.

[0043] 이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시 예를 설명함으로써, 본 발명을 상세히 설명한다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0044] 유전자 알고리즘(Genetic Algorithm)은 자연 세계의 진화 과정을 컴퓨터 상에서 시뮬레이션하여 복잡한 실세계의 문제를 해결하고자 하는 계산 모델이다.

[0045] 이러한 유전자 알고리즘의 기본적인 원리는, 사용자가 얻고자 하는 결과를 생물체의 유전 정보를 담고 있는 염색체의 형태를 가지는 자료구조로 나타낸 다음, 세대를 거듭하면서 염색체 교환과 돌연변이 현상을 표현하는 유전 연산자를 염색체에 적용시켜 최적의 결과물을 얻도록 하는 것이다.

[0046] 이하 도1에 나타낸 본 발명을 설명하는데 필요한 수식을 참조하여 본 발명을 설명한다.

[0047] 본 발명의 부하 분배 알고리즘은 유전자 알고리즘을 이용한다. 유전자 알고리즘을 적용시키기에 앞서 SDN상의 스위치의 집합은 $S = \{S_1, S_2, \cdots, S_i\}$, 컨트롤러의 집합은 $C = \{C_1, C_2, \cdots, C_N\}$, 그리고 스위치를 컨트롤러에 배치하는 방법의 집합을 $M = \{M_1, M_2, \cdots, M_N\}$로 정의한다.

[0048] 컨트롤러의 부하는 CPU 사용량, 메모리 사용량, 네트워크 I/O, 그리고 각 스위치부터의 메시지 수신량으로 측정될 수 있다.

[0049] 이는 컨트롤러가 제어 평면으로서 수행하는 라우팅 연산, 스위치들과의 통신, 플로우 테이블의 전송에 쓰이는 자원들이며 이러한 자원들의 상태를 조사하여 부하를 계산할 수 있다.

[0050] 일반적으로 컨트롤러의 처리량(throughput)에 있어 병목이 발생하는 요소는 라우팅 연산에 사용되는 CPU의 활용도(utilization)로 알려져 있으며 이는 라우팅 연산을 요구하는 메시지인 packet_in 메시지의 수신량에도 영향을 받는다.

[0051] Packet_in 메시지를 많이 보내는 스위치는 컨트롤러의 부하를 증가시키는 것으로 간주할 수 있으므로, 컨트롤러에서는 이러한 스위치들이 보내는 packet_in 메시지의 합으로 부하를 계산할 수 있다.

[0052] 현재로부터 과거 시간 동안 컨트롤러 에 전하는 평균적인 packet_in 메시지의 양을 측정하기 위하여 스위치 가 컨트롤러에 전하는 메시지의 양인 $S_i(T)$를 다음과 같이 표현할 수 있다.

[0053] $$\overline{S_i(T)} = \frac{S_i(T)}{T}$$

[0054] 컨트롤러의 부하는 앞서 계산한 스위치로부터의 packet_in 메시지 수신량의 합과 CPU 활용량(utilization)을 이용하여 계산할 수 있다.

[0055] $$C(T) = \frac{w \times \sum_{i=1}^{N} \overline{S_i(T)}}{C_{MAX}(T)} \times 100 + U$$

[0056] U는 CPU 활용량으로서 0부터 100사이의 값을 가진다. $C_{MAX}T$는 컨트롤러가 시간동안 처리할 수 있는 최대 packet_in 메시지의 양의 값을 가진다.

[0057] T시간동안 해당 컨트롤러로 전송되는 packet_in 메시지의 양과 최대 처리 양의 비율을 계산하여 0부터 100사이의 값을 갖도록 100을 곱해준다.

[0058] w는 0보다 크고 1보다 작은 가중치로, 컨트롤러에 부여되는 부하 계산시 packet_in 메시지의 양과 CPU 활용량에 대하여 서로 다른 가중치를 두고자 할 때 사용한다.

[0059] packet_in 메시지의 수신시 컨트롤러는 해당 목적지에 대한 경로 정보가 기존에 존재하지 않아 새로운 연산이 필요할 수 있으며, 기존 경로 정보가 존재할 경우에는 해당 목적지로의 경로 연산이 필요 없어 CPU의 사용량에 큰 영향을 미치지 않는다.

[0060] 따라서 컨트롤러의 부하는 packet_in 메시지의 수신량과 CPU 사용량을 함께 고려한다.

[0061] 현재 네트워크에서 특정 스위치가 전송 요청을 많이 보내거나 특정 컨트롤러가 처리하는 연산량이 급격히 늘어나 병목 현상이 일어나게 되면 컨트롤러들간의 부하 불균형이 발생한다.

[0062] 이때의 부하의 균형을 맞추기 위하여 각 컨트롤러의 부하를 $C_i(T)'$라고 두고 배치 방법 의 컨트롤러간 부하의 시간 동안의 표준편차 $\sigma(T)$를 구할 수 있다.

[0063] $$\overline{C(T)'} = \frac{1}{N}\sum_{i=1}^{N} C_i(T)'$$

[0064] $$\sigma(T) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\overline{C(T)'} - C_i(T)')^2}$$

[0065] 본 발명은 전술한 방식으로 여러 컨트롤러간 부하의 표준 편차를 계산하여 사전에 주어진 값을 넘어서는 경우 불균형으로 간주하여 부하 분배 알고리즘을 적용한다.

[0066] 컨트롤러들간의 부하에 대한 표준편차가 작을수록 좋은 적합도를 가지며, 좋은 적합도를 가지는 배치 방법들이 선택되는 확률을 높게 하고자 하는 것이 본 부하 분배 유전자 알고리즘의 첫 번째 과정이며, 이를 위한 적합도 함수 $f(M_i, T)$는 아래에 정의되어 있다.

[0067] 또한 선택된 2개의 배치방법들을 이용하여 복제 및 교배를 통하여 부하가 더 적은 새로운 배치 방법을 찾아내는 것이 본 부하분배 유전자 알고리즘의 최종 목적이다.

[0068] 부하의 분배에 유전자 알고리즘을 적용하기 위하여 트리 구조로 표현된 컨트롤러와 스위치의 배치 방법들의 집합을 최초의 해집단으로서 설정한다.

[0069] 그리고 T시간 동안의 배치 방법 M_i가 가지는 적합도 값을 $f(M_i, T)$로 정의하고 아래와 같은 적합도 함수로 각 배치 방법들의 적합도를 계산 할 수 있다.

[0070] $$f(M_i,T)=\frac{1}{A+B\times f_{H_i}}$$

[0071] $$f_{H_i}=\Phi(\sigma_i(M_i,T)-\sigma_0),\quad \Phi(X)=\begin{cases}1, X\le 0\\ r, X>0, where\ r>1\end{cases}$$

[0072] σ_0는 미리 정의된 기준이 되는 부하의 표준 편차이며 f_{H_i}는 배치 방법 M_i에 대한 컨트롤러들간 부하의 표준편차와 σ_0의 차이에 대한 값이다. A와 B는 가중치를 의미하며, 적합도 계산시 네트워크 상황에 따른 표준편차의 차이에 대한 민감도를 조정하는 것이 필요 할 때 사용 될 수 있다.

[0073] 현재 M_i의 표준편차 σ_i가 미리 정의된 σ_0보다 작다면 f_{H_i}는 1이 되고, 크다면 f_H는 r이 되며 r은 1보다 큰 값을 갖는다.

[0074] 즉 컨트롤러간 부하의 표준편차가 미리 정의된 σ_0(기 지정한 임계값)보다 크면 적합도는 낮은 값을 가질 것이며, 반대로 부하의 표준편차가 σ_0(기 지정한 임계값)보다 작다면 적합도는 높은 값을 가질 것이다.

[0075] 적합도가 계산된 개체들은 선택의 과정을 거치게 된다. 각 개체의 적합도가 더 높을수록 다음 세대에 유전자, 즉 컨트롤러에 배치된 스위치에 관한 정보를 전할 확률이 높아진다.

[0076] 개체의 선택에는 적합도 비례 선택방법을 이용한다. 적합도 비례 선택 방법은 각 개체의 적합도가 전체 개체의 적합도에서 차지하는 비율에 따라 선택되는 확률이 달라지는 방법이다. 본 발명에서는 각 개체가 선택될 확률을 다음과 같이 계산한다.

[0077] $$p(M_i)=\frac{f(M_i,T)}{\sum_{i=1}^{N} f(M_i,T)}$$

[0078] $f(M_i,T)$는 각 배치 방법들의 적합도를 나타내며 N는 전체 개체, 즉 배치 방법들의 수를 의미한다. 적합도가 높은 배치 방법이 선택될 확률은 높게 된다.

[0079] 각각의 적합도에 따른 확률로 선택된 개체들은 교배 연산의 수행으로 더 나은 다음 세대의 개체를 생성하게 된다.

[0080] 유전자 알고리즘에서는 복제와 교배 연산을 통해 선택된 개체의 유전자를 재조합하여 더 나은 해를 얻는다. 본 발명에서는 각 배치 방법의 적합도를 계산한 뒤 선택 전략에 입각하여 두 배치 방법 M_1, M_2를 선택하여 두 개체의 배치를 조합하여 M_0를 도출한다.

[0081] 만약 컨트롤러 C_i 에 스위치 S_j가 두 배치 방법 모두 포함되어 있다면, 조합이 이루질 때 M_0의 배치에는 서로 공통되는 배치 방법은 그대로 사용하여 초기의 M_0를 작성한다.

[0082] 이때 배치되지 않고 남아 있는 스위치들은 아래와 같은 방법으로 M_0의 배치 방법을 완성한다.

[0083] 남아 있는 스위치들 중에서 컨트롤러의 부하에 가장 영향을 크게 미칠것으로 예상되는 스위치를 현재 부하가 가장 적은 컨트롤러에게 배치한다.

[0084] 즉 배치되지 않은 스위치들은 전체 스위치 부하의 합에서 차지하는 각각의 비율 ξ를 구하여 ξ값이 큰 스위치부터 부하가 적은 컨트롤러에 차례대로 배치한다.

[0085] 전술한 내용을 정리하면,

[0086] 컨트롤러의 부하 불균형은 각 컨트롤러들의 부하 표준 편차를 계산함으로서 감지 할 수 있으며,

[0087] 불균형으로 인지하는 표준 편차의 기준은 네트워크의 성능과 요구 사항에 따라 그 값이 정해질 수 있다(예로써 50). 또한 불균형을 감지하면 앞서 언급한 컨트롤러 부하의 수학적 모델과 유전자 알고리즘에 대한 내용을 이용하여 다음과 같은 순서에 따라 부하 분배 알고리즘을 적용할 수 있다.

[0088] 1)최초의 네트워크가 형성될 때 컨트롤러와 스위치는 요청을 주고받으며 임의의 연결이 생성된다.

[0089] 2)네트워크가 형성되면 스위치는 컨트롤러에게 전송 요청을 전달하게 되고 컨트롤러에는 부하가 주어지게 될 것이다. 이 때 컨트롤러에 부여된 부하 양의 표준편차를 계산하여 기 지정된 임계값에 넘어서게 되면 부하 불균형으로 간주하고 부하 분배 알고리즘을 수행한다.

[0090] 3)스위치가 컨트롤러에 배치될 수 있는 방법들 중 무작위로 N개를 뽑아 최초의 해집단을 구성한 후, 각 개체의 적합도를 계산한 뒤 선택 전략에 입각하여 두 배치 방법 M_1, M_2을 선택하여 교배연산을 통해 새로운 배치 방법 M_0를 얻어낸다. 이 과정에서 일정한 확률 P_m에 따라 변이 연산을 수행하여 생성될 자손의 다양성을 높인다.

[0091] 4)M_0로 배치하였을 때 컨트롤러 부하의 표준편차를 계산하여 정해진 기준에 미치지 못하면 다시 과정 2)로 돌아가서 알고리즘을 실시하고 만족한다면 그 결과대로 스위치를 재배치한다.

[0092] 도 1은 본 발명의 실시 예에 따른 유전자 알고리즘을 이용한 부하 분배 장치의 세부적인 구성을 나타낸 도이다.

[0093] 본 발명에 따른 유전자 알고리즘을 이용한 SDN상의 부하 분배 장치(100)는 부하 계산부(10), 부하 불균형 판단부(20), 부하 조정부(30)를 포함할 수 있다.

[0094] 부하 계산부(10)는, 전술한 수식을 이용하여 SDN상의 각 컨트롤러에 부여되는 부하의 양 및 상기 부하의 양의 표준편차를 산출할 수 있으며, 상기 산출된 값을 부하 불균형 판단부(20)에 제공할 수 있다.

[0095] 부하 불균형 판단부(20)는, 상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하고, 상기 계산된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단할 수 있으며, 상기 판단 결과를 부하 조정부(30)에 제공할 수 있다.

[0096] 부하 조정부(30)는, 상기 부하 불균형 판단부(20)에서 부하 불균형으로 판단한 경우, 상기 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 조정할 수 있다.

[0097] 부하 조정부(30)가 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 조정하는 방법으로 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 상기 기 지정된 임계값 미만의 표준편차를 갖는 컨트롤러에 부여할 수 있다.

[0098] 예를 들면 기 지정된 임계값(표준편차)이 50이라면 부하 계산부(10)가 계산한 제1컨트롤러의 표준편차가 70이고 제2컨트롤러의 표준편차가 30이라면 부하 불균형 판단부(20)는 상기 제1컨트롤러를 부하 불균형으로, 상기 제2컨트롤러를 부하 균형으로 판단한다.

[0099] 그리고 부하 조정부(30)는 상기 제1컨트롤러에 부여된 부하를 상기 제2컨트롤러에 분배할 수 있다.

[0100] 상기 제2컨트롤러는 다시 제2컨트롤러에 부여되는 부하량을 계산하는 단계를 거치게 된다.

[0101] 이 때 상기 제2컨트롤러에 분배하는 상기 제1컨트롤러의 부하량을 몇으로 정할 것인지는 상기 제1컨트롤러의 표준편차가 기 지정된 임계값 50이하가 될 때까지 또는 상기 제2컨트롤러의 표준편차가 기 지정된 임계값 50을 초과하기 전까지 등의 방법을 통해 정할 수 있다.

[0102] 상기 기 지정된 임계값은 가변적인 수치이고, 네트워크 관리자가 설정할 수 있으며, SDN상의 각 컨트롤러에 부여되는 부하의 편차를 적게 하여 컨트롤러들의 성능을 균등하게 관리하고 싶다면 상기 기 지정된 임계값의 수치를 낮춰서 설정하는 것이 바람직하다.

[0103] 도 2는 본 발명의 실시 예에 따른 유전자 알고리즘을 이용한 SDN상의 부하 분배 방법의 과정을 나타낸 순서도이다.

[0104] 본 발명에 따른 유전자 알고리즘을 이용한 부하 분배 방법은 부하 계산 단계(s10), 부하 불균형 판단 단계(s20), 부하 조정 단계(s30)를 포함할 수 있다.

[0105] 부하 계산 단계(s10)는, 전술한 수식을 이용하여 SDN상의 각 컨트롤러에 부여되는 부하의 양 및 상기 부하의 양의 표준편차를 산출하는 절차, 상기 산출된 값을 부하 불균형 판단부(20)에 제공하는 절차를 포함할 수 있다.

[0106] 부하 불균형 판단 단계(s20)는, 상기 산출된 표준편차가 기 지정된 임계값 이상이면 부하 불균형으로 판단하는 절차, 상기 계산된 표준편차가 상기 기 지정된 임계값 미만이면 부하 균형으로 판단하는 절차, 상기 판단 결과

를 부하 조정부(30)에 제공하는 절차를 포함할 수 있다.

[0107] 부하 조정 단계(s30)는, 상기 부하 불균형 판단 단계(s20)에서 부하 불균형으로 판단한 경우, 상기 유전자 알고리즘을 이용한 부하 분배 알고리즘을 적용하여 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 조정할 수 있다.

[0108] 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 조정하는 방법으로 상기 기 지정된 임계값 이상의 표준편차를 갖는 컨트롤러에 부여된 부하의 양을 상기 기 지정된 임계값 미만의 표준편차를 갖는 컨트롤러에 부여하는 절차를 포함할 수 있다.

[0109] 상기 부하 불균형 판단 단계(s20)에서 부하 균형으로 판단한 경우, 다시 부하 계산 단계(s20)로 돌아가는 단계(s21)를 포함할 수 있다.

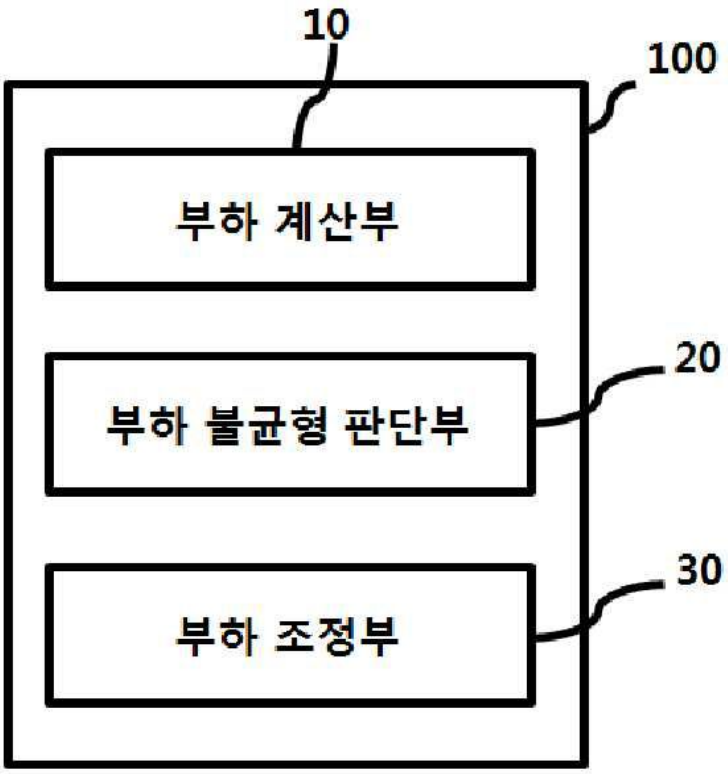
[0110] 지금까지 본 발명을 바람직한 실시 예를 참조하여 상세히 설명하였다. 그러나 본 발명이 상기한 실시 예에 한정되는 것은 아니며, 이하의 특허청구범위에서 청구하는 본 발명의 요지를 벗어남이 없이 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자라면 누구든지 다양한 변형 또는 수정이 가능한 범위까지 본 발명의 기술적 사상이 미친다 할 것이다.

도면

도면1

기호	의미
$S=\{S_1, S_2, \cdots, S_x\}$	스위치 S_i들의 집합
$C=\{C_1, C_2, \cdots, C_y\}$	컨트롤러 C_i들의 집합
$M=\{M_1, M_2, \cdots, M_N\}$	어느 스위치들을 어느 컨트롤러에 어떻게 배치하는지에 대한 방법인 M_i 들의 집합
$S_i(T)$	스위치 i가 시간 T 동안 컨트롤러에게 송신하는 packet_in 메시지 양
$\overline{S_i(T)}$	스위치 i가 시간당 송신하는 packet_in 메시지 양
$C(T)$	컨트롤러의 T시간동안 부하 (스위치들로부터 수신되는 packet_in 메시지의 양과 CPU 사용량을 고려하여 결정.
$w\times\sum_{i=1}^{N}\overline{S_i(T)}$	컨트롤러가 T시간동안 모든 스위치들로부터 수신하는 packet_in 메시지 양
$C_{MAX}(T)$	컨트롤러가 T시간동안 처리할 수 있는 최대 packet_in 메시지 양
U	컨트롤러의 CPU 사용량
$C_i(T)'$	컨트롤러 i의 부하
$\sigma(T)$	컨트롤러들 부하의 표준편차
M_i	스위치들을 어느 컨트롤러에 어떻게 배치하는지에 대한 방법. 여러 가지 방법 중 i 번째 배치 방법
$f(M_i,T)$	T시간 동안의 배치 방법 M_i가 가지는 적합도 값
σ_0	부하분배 시작 기준이 되는 미리 정의된 부하들의 표준편차 값
$f_{R_i}=\Phi(\sigma_i(M_i,T)-\sigma_0)$	현재 배치 방법에 대한 부하와 표준편차와의 차이 값
$\Phi(X)=\begin{cases}1, X\le 0\\ r, X>0\end{cases}$	X 의 값이 0 보다 작으면 1, 0 보다 크면 r의 값 이때
$p(M_i)$	배치 방법 M_i가 선택될 확률 적합도가 높은 배치 방법은 높은 확률 값을 가짐.

도면2



도면3

