(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2017년03월24일
(11) 등록번호 10-1719714
(24) 등록일자 2017년03월20일

(51) 국제특허분류(Int. Cl.)
G06F 17/30 (2006.01) *G06Q 50/00* (2006.01)
G06Q 50/30 (2012.01)
(52) CPC특허분류
G06F 17/30958 (2013.01)
G06F 17/30336 (2013.01)
(21) 출원번호 10-2015-0171988
(22) 출원일자 2015년12월04일
심사청구일자 2015년12월04일
(56) 선행기술조사문헌
JP2012133522 A*
JP2013062735 A
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
경희대학교 산학협력단
경기도 용인시 기흥구 덕영대로 1732 (서천동, 경희대학교 국제캠퍼스내)
(72) 발명자
이영구
대전광역시 서구 청사서로 11, 106동 302호 (월평동, 무지개아파트)
박성찬
경기도 용인시 기흥구 덕영대로 1732, 349호 (서천동, 경희대학교 국제캠퍼스 전자정보대학)
김현욱
경기도 용인시 기흥구 동탄기흥로 864, 105동 1502호 (공세동, 성원상떼레이크뷰)
(74) 대리인
인비전 특허법인

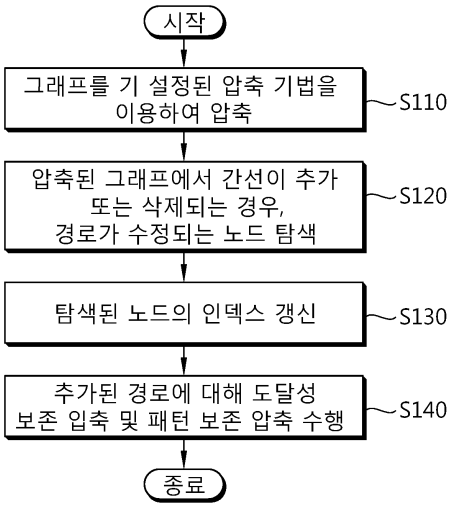
전체 청구항 수 : 총 3 항 심사관 : 홍경아

(54) 발명의 명칭 **동적 환경 소셜 네트워크 그래프에서 다중 제약 조건 그래프 패턴 매칭을 위한 효율적인 인덱스 갱신 방법 및 장치**

(57) 요 약

동적 그래프 압축 방법 및 장치를 제공한다. 그래프 압축 장치에 의한 그래프 압축 방법은 그래프를 기 설정된 압축 기법을 이용하여 압축하는 단계, 상기 압축된 그래프에서 제1 노드와 제2 노드 간의 간선이 추가 또는 삭제되는 경우, 상기 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색하는 단계 및 상기 탐색된 노드의 인덱스를 갱신하는 단계를 포함할 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06Q 50/01 (2013.01)
G06Q 50/30 (2015.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711026767
부처명 미래창조과학부
연구관리전문기관 정보통신기술진흥센터
연구사업명 대학 ICT 연구센터육성 지원사업
연구과제명 실시간 모바일 클라우드 서비스 플랫폼 연구개발
기 여 율 1/1
주관기관 경희대학교
연구기간 2015.01.01 ~ 2015.12.31

명 세 서

청구범위

청구항 1

그래프 압축 장치에 의한 인덱스 갱신 방법에 있어서,

그래프를 기 설정된 압축 기법을 이용하여 압축하는 단계;

상기 압축된 그래프에서 제1 노드와 제2 노드 간의 간선이 추가 또는 삭제되는 경우, 상기 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색하는 단계; 및

상기 탐색된 노드의 인덱스를 갱신하는 단계를 포함하고,

상기 갱신하는 단계는,

상기 제1 노드와 상기 제2 노드간의 간선이 추가되고 상기 경로가 수정되는 노드가 상기 제1 노드를 자식 노드로 가지고 있는 노드의 집합인 제1 노드 집합에 포함되는 경우, 상기 제1 노드 집합에 속한 노드의 인덱스에 상기 제2 노드를 부모 노드로 가지고 있는 노드를 자식경로로 추가하는 단계;

상기 제1 노드와 상기 제2 노드 간에 간선이 추가되고 상기 경로가 수정되는 노드가 상기 제2 노드를 부모 노드로 가지고 있는 노드의 집합인 제2 노드 집합에 포함되는 경우, 상기 제2 노드 집합에 속한 노드의 인덱스에 상기 제1 노드를 자식 노드로 가지고 있는 노드를 부모경로로 추가하는 단계; 및

상기 제1 노드와 상기 제2 노드 간의 간선이 삭제되고 해당 경로가 패턴 보존 압축이 수행되었던 경로인 경우, 상기 제1 노드의 외향 간선으로 연결된 이웃 노드 집합의 노드가 갖는 인덱스에서 상기 제2 노드로의 경로가 존재하면 해당 경로를 상기 제1 노드의 인덱스에 추가하는 단계를 포함하는 것을 특징으로 하는 인덱스 갱신 방법.

청구항 2

제1항에 있어서,

상기 탐색하는 단계는,

상기 압축된 그래프에 대해 너비 우선 탐색을 수행함으로써 상기 경로가 수정되는 노드를 탐색하는 단계인 것을 특징으로 하는 인덱스 갱신 방법.

청구항 3

삭제

청구항 4

삭제

청구항 5

제1항에 있어서,

상기 갱신하는 단계 이후에,

상기 제1 노드와 상기 제2 노드간의 경로에 대해 도달성 보존압축 및 패턴압축을 수행하는 단계를 더 포함하는 것을 특징으로 하는 인덱스 갱신 방법.

발명의 설명

기 술 분 야

[0001] 본 발명의 실시예들은 동적으로 변화하는 소셜 네트워크 그래프(Social Network Graph)에서 다중 제약 조건 그

래프 패턴 매칭을 위하여 그래프 압축을 수행하는 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 최근 정보화 기기의 발달로 인해 소셜 네트워크(Social Network)를 사용하는 사용자가 증가하고 있다. 이에 따라 소셜 네트워크에서 사용자들의 기호분석, 특정 분야의 인력탐색 등에 필요한 의미있는 정보를 얻기 위해 다중제약 그래프에서의 그래프 패턴 매칭(graph pattern matching) 기술이 제안되었으며, 다중제약 그래프를 압축하여 효율적으로 그래프 패턴 매칭을 수행하는 기법이 제안되었다.

[0003] 그래프 압축 기법은 그래프의 특성을 유지하면서 노드(node)와 경로의 수를 줄인다. 그래프 압축 기법 중 하나로서, 한국공개특허공보 제10-2015-0125511호 (공개일: 2015년 11월 9일) "그래프 압축 처리 방법 및 장치"에는 그래프 상에 존재하는 노드의 차수를 이용하여 허브 노드를 추출하고, 허브 노드 간의 연결을 고려하여 차수가 높은 순으로 허브 노드를 제거하고, 제거한 허브 노드로부터 분해되는 스포크(spoke) 노드를 식별하고, 제거한 허브 노드의 차수를 고려하여 그래프 상에 상기 스포크 노드를 배열하고, 상기 스포크 노드의 배열에 따른 그래프의 압축을 처리하는 것이 기재되어 있다.

[0004] 그러나 사용자 간의 관계는 실시간으로 변화하기 때문에 소셜 네트워크 또한 동적으로 변화하지만, 기존의 그래프 압축 기법은 소셜 네트워크의 동적인 환경을 고려하지 않아 비효율적이다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 기술적 과제는 압축된 그래프에서 간선이 추가되거나 삭제되는 경우에 상기 압축된 그래프에 이를 효율적으로 반영할 수 있는 동적 환경 소셜 네트워크 그래프에서 다중 제약 조건 그래프 패턴 매칭을 위한 그래프 압축 방법을 제공함에 있다.

[0006] 본 발명의 다른 기술적 과제는 압축된 그래프에서 간선이 추가되거나 삭제되는 경우에 상기 압축된 그래프에 이를 효율적으로 반영할 수 있는 동적 환경 소셜 네트워크 그래프에서 다중 제약 조건 그래프 패턴 매칭을 위한 그래프 압축 장치를 제공함에 있다.

과제의 해결 수단

[0007] 본 발명의 일 양태에 따르면, 그래프 압축 장치에 의한 그래프 압축 방법은 그래프를 기 설정된 압축 기법을 이용하여 압축하는 단계, 상기 압축된 그래프에서 제1 노드와 제2 노드 간의 간선이 추가 또는 삭제되는 경우, 상기 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색하는 단계 및 상기 탐색된 노드의 인덱스를 갱신하는 단계를 포함할 수 있다.

[0008] 일측에 따르면, 상기 탐색하는 단계는 상기 압축된 그래프에 대해 너비 우선 탐색을 수행함으로써 상기 경로가 수정되는 노드를 탐색하는 단계일 수 있다.

[0009] 다른 측면에 다르면, 상기 갱신하는 단계는 상기 제1 노드와 상기 제2 노드간의 간선이 추가되고 상기 경로가 수정되는 노드가 상기 제1 노드를 자식 노드로 가지고 있는 노드의 집합에 포함되는 경우, 상기 노드의 집합에 속한 노드의 인덱스에 상기 제2 노드를 부모 노드로 가지고 있는 노드를 자식경로로 추가하는 단계를 포함할 수 있다.

[0010] 또 다른 측면에 다르면, 상기 갱신하는 단계는 상기 제1 노드와 상기 제2 노드간의 간선이 추가되고 상기 경로가 수정되는 노드가 상기 제2 노드를 부모 노드로 가지고 있는 노드의 집합에 포함되는 경우, 상기 노드의 집합에 속한 노드의 인덱스에 상기 제1 노드를 자식 노드로 가지고 있는 노드를 부모경로로 추가하는 단계를 포함할 수 있다.

[0011] 또 다른 측면에 다르면, 상기 갱신하는 단계 이후에 상기 제1 노드와 상기 제2 노드간의 경로에 대해 도달성 보존압축 및 패턴압축을 수행하는 단계를 더 포함할 수 있다.

발명의 효과

[0012] 그래프의 동적인 변화에 대하여 압축된 그래프에서 경로가 수정된 노드의 인덱스만 갱신하기 때문에 적은 비용으로 효율적인 연산을 수행할 수 있다.

도면의 간단한 설명

[0013] 도 1은 본 발명의 일실시예에 따른 그래프 압축 방법을 나타내는 흐름도이다.

도 2는 본 발명의 일실시예에 따라 압축된 그래프에서 간선이 추가되는 경우를 설명하기 위한 도면이다.

도 3은 본 발명의 일실시예에 따른 동적 그래프 압축 알고리즘을 나타내는 도면이다.

도 4는 본 발명의 일실시예에 따른 그래프 압축 장치를 나타내는 블록도이다.

발명을 실시하기 위한 구체적인 내용

[0014] 아래에서는 첨부한 도면을 참고로 하여 본 발명의 실시예에 대하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.

[0015] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 "~부" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0016] 도 1은 본 발명의 일실시예에 따른 그래프 압축 방법을 나타내는 흐름도이다.

[0017] 도 1을 참조하면, 본 발명에 따른 그래프 압축 장치는 그래프 패턴 매칭을 위하여 그래프를 기 설정된 압축 기법을 이용하여 압축할 수 있다(S110). 여기서, 상기 그래프는 다중 제약 그래프일 수 있다. 또한 상기 그래프는 동적으로 구조가 변화하는 동적 그래프일 수 있다. 일 예로, 상기 그래프는 소셜 네트워크 그래프일 수 있다. 압축된 그래프에서 각각의 노드들은 부모 노드들로 가는 경로와 자식 노드들로 가는 경로를 인덱스 값으로 저장할 수 있다.

[0018] 그래프가 압축되면, 본 발명에 따른 그래프 압축 장치는 압축된 그래프에서 제1 노드와 제2 노드 간의 간선이 추가 또는 삭제되는 경우, 상기 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색하고(S120), 탐색된 노드의 인덱스를 갱신할 수 있다(S130). 즉, 본 발명에 따른 그래프 압축 장치는 효율적인 그래프 패턴 매칭을 수행하기 위해서 압축된 그래프의 전체 노드가 아닌 갱신이 필요한 특정 노드들에 대해서만 인덱스를 갱신할 수 있다.

[0019] 예를 들어, 본 발명에 따른 그래프 압축 장치는 압축된 그래프에 대해 너비 우선 탐색(BFS: Breadth First Search)을 수행함으로써 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색할 수 있다. 탐색 결과, 제1 노드와 제2 노드 간에 간선이 추가되고 경로가 수정되는 노드가 제1 노드를 자식 노드로 가지고 있는 노드의 집합(제1 노드 집합)에 포함되는 경우, 상기 제1 노드 집합에 속한 노드의 인덱스에 상기 제2 노드를 부모 노드로 가지고 있는 노드를 자식경로로 추가할 수 있다. 한편 제1 노드와 제2 노드 간에 간선이 추가되고 경로가 추가되는 노드가 제2 노드를 부모 노드로 가지고 있는 노드의 집합(제2 노드 집합)에 포함되는 경우, 상기 제2 노드 집합에 속한 노드의 인덱스에 상기 제1 노드를 자식 노드로 가지고 있는 노드(제1 노드 집합에 속한 노드)를 부모경로로 추가할 수 있다.

[0020] 반면, 제1 노드와 제2 노드 간의 간선이 삭제되는 경우, 제1 노드 집합에 속한 노드의 인덱스에서 제2 노드 집합에 속한 모든 노드를 자식경로에서 삭제할 수 있다. 이때 삭제된 경로가 패턴 보존 압축이 수행되었던 경로라면, 기존 그래프(압축되기 전의 그래프)의 패턴이 소실될 수 있다. 따라서, 이 경우 제1 노드의 외향 간선으로 연결된 이웃 노드 집합을 검사하여 만일 상기 이웃 노드 집합의 노드가 갖는 인덱스에서 제2 노드로의 경로가 존재한다면, 해당 경로를 제1 노드의 인덱스에 모두 추가할 수 있다.

[0021] 이후, 본 발명에 따른 그래프 압축 장치는 제1 노드와 제2 노드 간에 추가된 경로에 대해 도달성 보존 압축(reachability compress) 및 패턴 보존 압축(pattern compress)을 수행할 수 있다(S140).

[0022] 도 2는 본 발명의 일실시예에 따라 압축된 그래프에서 간선이 추가되는 경우를 설명하기 위한 도면이다.

[0023] 도 2를 참조하면, 일 예로 압축된 그래프에서 새로운 유향 간선(210)이 추가되는 경우 생성되는 노드 집합이 도

시되어 있다. 도 2에서 V_i는 노드 A(v_a)를 자식 경로로 갖고 있는 모든 노드들의 집합을 나타내고, V_o는 노드 B(v_b)를 부모 경로로 갖고 있는 모든 노드들의 집합을 나타낸다.

[0024] 노드 A로부터 노드 B로의 간선(210)이 추가되는 경우, 그래프 압축 장치는 V_i에 속한 각각의 노드가 갖는 인덱스에 대해 V_o의 모든 노드를 자식경로로 추가하고, V_o에 속한 각각의 노드가 갖는 인덱스에 대해 V_o의 모든 노드를 부모경로로 추가할 수 있다. V_i에서 V_o로의 경로는 다음의 수학식 1과 같다.

수학식 1

[0025] $$v_i \in V_i,\ v_o \in V_o,\ path(v_i, v_o) = index(v_i) :: path(v_i, v_a) + index(v_o) :: path(v_b, v_o)$$

[0026] 만일 V_i에서 V_o로의 경로를 이미 갖고 있다면, 기존의 경로와 새로운 경로 path(v_i, v_o)에 대해 도달성 보존 압축과 패턴 보존 압축을 수행할 수 있다.

[0027] 한편, 압축된 그래프에 존재했던 노드 A(v_a)로부터 노드 B(v_b)로의 간선(210)이 삭제되는 경우, 그래프 압축 장치는 V_i에 속한 각각의 노드가 갖는 인덱스에 대해 V_o의 모든 노드를 자식경로에서 삭제할 수 있다. 이때, 삭제된 경로가 패턴 보존 압축이 수행되었던 경로라면, 기존 그래프(압축되기 전의 그래프)의 패턴이 소실될 수 있다. 따라서, 이 경우 노드 A(V_a)의 외향 간선으로 연결된 이웃 노드 집합(V_j)을 검사할 수 있다. 그리고 만일 상기 이웃 노드 집합(V_j)의 노드(v_j)가 갖는 인덱스에서 노드 B(v_b)로의 경로가 존재한다면, 해당 경로를 노드 A(v_a)의 인덱스에 모두 추가할 수 있다. 이웃 노드 집합(V_j)의 노드(v_j)를 거치는 경로는 다음의 수학식 2와 같다.

수학식 2

[0028] $$v_i \in V_i,\ v_o \in V_o,\ v_j \in V_j,\ path(v_i, v_j, v_o) = index(v_i) :: path(v_i, v_j) + index(v_j) :: path(v_j, v_o)$$

[0029] 그리고 본 발명에 따른 그래프 압축 장치는 모든 추가된 경로 path(vi, vj, vo)에 대해 도달성 보존 압축과 패턴 보존 압축을 수행할 수 있다.

[0030] 만일, 삭제되는 경로가 여러 경로를 압축하고 있는 경로라면 기존 그래프의 특성이 소실된다. 예를 들어 경로 3개 경로(P_1, P_2, P_3)의 압축된 인덱스를 갖는 경로(P3)와 경로 P_1, P_2 포함하여 총 3개의 경로 (P_1, P_2, P_3(P_3+P_4+P_5))를 갖는 압축된 그래프(G)에 대해서, 경로 P_3이 삭제될 경우, 압축된 그래프(G)에서는 P_4, P_5에 대한 정보가 함께 소실된다. 따라서 삭제된 간선을 포함하는 경로 외의 경로를 보존하기 위해 삭제된 경로를 제외한 다른 경로가 존재하는지 탐색할 수 있다. 그리고 탐색된 경로들에 대해 다시 도달성 보존 압축 및 패턴 보존 압축을 수행함으로써 그래프의 특성을 보존할 수 있다. 따라서 본 발명에 따른 그래프 압축 장치 및 방법은 기존 그래프의 특성을 유지할 수 있으며, 기존에 구축된 인덱스를 사용하여 경로를 효율적으로 탐색할 수 있다.

[0031] 도 3은 본 발명의 일실시예에 따른 동적 그래프 압축 알고리즘을 나타내는 도면이다.

[0032] 도 3에는 일 예로, 압축된 그래프에서 간선이 변경되었을 때 경로가 수정되는 노드들의 인덱스 정보를 갱신하는 알고리즘이 도시되어 있다. 도 3을 참조하면, 간선이 추가되는 경우 해당 간선으로 인해 새로 추가되는 경로에 노드를 인덱스에 추가하고 그래프 탐색의 효율성을 위하여 그래프 압축을 수행한다(line4-11). 그리고, 간선이 삭제되는 경우, 해당 간선을 포함하는 경로를 모두 제거하고 간선을 제거한다(line12-17). 그러나 간선을 삭제하는 경우 다른 간선을 이용한 경로가 존재할 수 있다. 따라서 이를 탐색하기 위해 이웃 노드 인덱스를 이용하여 탐색한다(line18-24).

[0033] 도 4는 본 발명의 일실시예에 따른 그래프 압축 장치를 나타내는 블록도이다.

[0034] 도 4를 참조하면, 본 발명에 따른 그래프 압축 장치(400)는 판단부(410), 갱신부(420) 및 압축부(430)를 포함할 수 있다.

[0035] 판단부(410)는 그래프 패턴 매칭을 위하여 기 설정된 압축 기법으로 압축된 그래프에서 간선이 추가 또는 삭제되는지를 판단할 수 있다. 여기서, 상기 그래프는 다중 제약 그래프일 수 있으며, 동적으로 구조가 변화하는 소셜 네트워크 그래프일 수 있다. 상기 압축된 그래프는 SSC(Strong Social Component) 그래프라 정의될 수 있다.

[0036] 갱신부(420)는 판단부(410)에 의해 제1 노드와 제2 노드 간의 간선이 추가 또는 삭제된 것으로 판단되면, 상기 간선의 추가 또는 삭제로 인해 경로가 수정되는 노드를 탐색하고 탐색된 노드의 인덱스를 갱신한다.

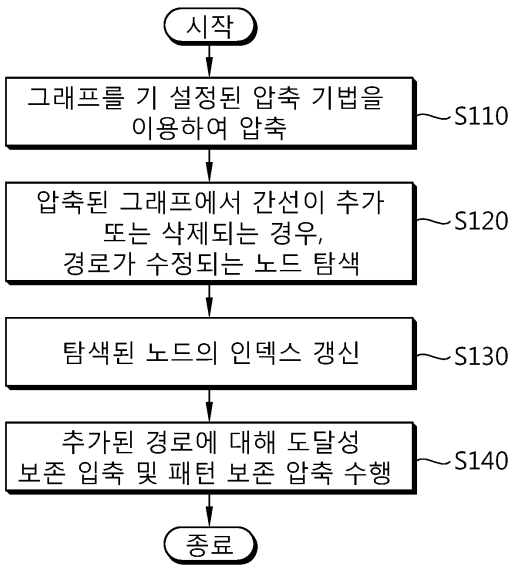
[0037] 예를 들어, 갱신부(420)는 제1 노드와 제2 노드 간에 간선이 추가되는 경우, 너비 우선 탐색을 수행함으로써 제1 노드가 속한 집합에 포함된 노드의 인덱스에 제2 노드가 속한 집합에 포함된 노드를 자식경로로 추가하는 한편, 제2 노드가 속한 집합에 포함된 노드의 인덱스에 제1 노드가 속한 집합에 속한 노드를 부모경로로 추가할 수 있다. 반면, 갱신부(420)는 제1 노드와 제2 노드 간의 간선이 삭제되는 경우, 제1 노드 집합에 속한 노드의 인덱스에서 제2 노드 집합에 속한 모든 노드를 자식경로에서 삭제할 수 있다. 이때 삭제된 경로가 패턴 보존 압축이 수행되었던 경로라면, 제1 노드의 외향 간선으로 연결된 이웃 노드 집합을 검사하여 만일 상기 이웃 노드 집합의 노드가 갖는 인덱스에서 제2 노드로의 경로가 존재한다면, 해당 경로를 제1 노드의 인덱스에 모두 추가할 수 있다.

[0038] 압축부(430)는 제1 노드와 제2 노드 간에 추가된 경로에 대해 도달성 보존 압축 및 패턴 보존 압축을 수행할 수 있다.

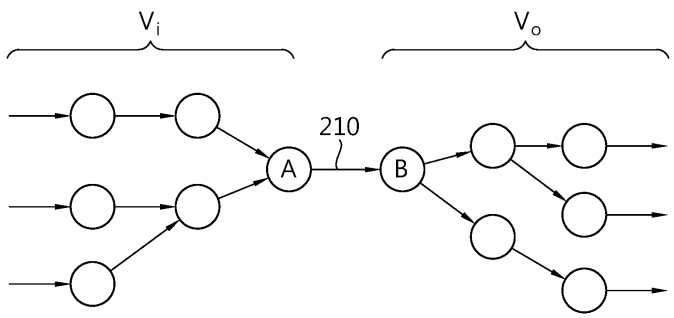
[0039] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 다양한 수정 및 변형이 가능할 것이다. 따라서, 본 발명에 개시된 실시예들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다.

도면

도면1



도면2



도면3

Dynamic SSC Compress Algorithm

Input: A SSC graph G_s, Changed Edge $e_n(v_a, v_b)$
Output: Renewal SSC graph G_s^r

1: $V_i = v_a \cup$ Parent Node of v_a;
2: $V_o = v_b \cup$ Child Node of v_b;
3: **for each** *node* $v_i \in V_i$
4: **if** Case of Add $e_n(v_a, v_b)$
5: $G_s^r = G_s \cup e_n(v_a, v_b)$;
6: **for each** *node* $v_o \in V_o$
7: $index(v_i) = index(v) \cup path(v_i, v_o)$;
8: $index(v_o) = index(v) \cup path(v_i, v_o)$;
9: $index(v_i)$, $index(v_o)$ $Compress$;
10: $index(v_i) \rightarrow G_s^r$; // Changed v_i Apply to G_s
11: $index(v_o) \rightarrow G_s^r$; // Changed v_o Apply to G_s
12: **else** // Case of Delete $e_n(v_a, v_b)$
13: **for each** *node* $v_o \in V_o$
14: $index(v_i) = index(v) - path(v_i, v_o)$;
15: $index(v_o) = index(v) - path(v_i, v_o)$;
16: $G_s^r = G_s - e_n(v_a, v_b)$;
17: V_j = Neighbor Nod Set of v_a
18: **for each** *node* $v_j \in V_j$
19: **if** When $path(v_j, v_o)$ Exist
20: $index(v_i) = index(v) \cup path(v_i, v_j, v_o)$;
21: $index(v_o) = index(v) \cup path(v_i, v_j, v_o)$;
22: $index(v_i)$, $index(v_o)$ $Compress$;
23: $index(v_i) \rightarrow G_s^r$; // Changed v_i Apply to G_s
24: $index(v_o) \rightarrow G_s^r$; // Changed v_o Apply to G_s
25: **Return** G_s^r;

도면4

