



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2018년07월06일
(11) 등록번호 10-1875925
(24) 등록일자 2018년07월02일

(51) 국제특허분류(Int. Cl.)

G06F 17/11 (2006.01)

(52) CPC특허분류

G06F 17/11 (2013.01)

(21) 출원번호 10-2017-0017416

(22) 출원일자 2017년02월08일

심사청구일자 2017년02월08일

(56) 선행기술조사문헌

US20140278305 A1

WO2012123505 A2

(73) 특허권자

울산과학기술원

울산광역시 울주군 언양읍 유니스트길 50

(72) 발명자

정원기

울산광역시 울주군 언양읍 유니스트길 50 울산과학기술대학교

홍수민

울산광역시 울주군 언양읍 유니스트길 50 울산과학기술대학교

(74) 대리인

제일특허법인

전체 청구항 수 : 총 7 항

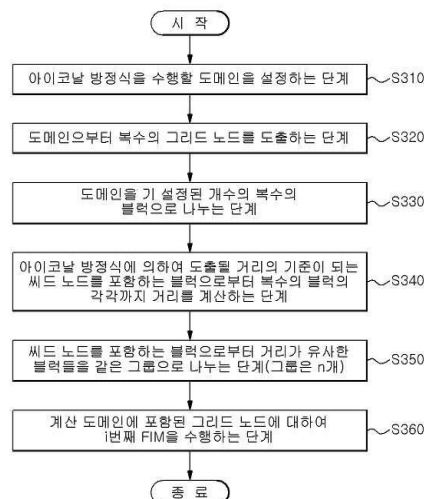
심사관 : 홍경아

(54) 발명의 명칭 그룹 순서 기반 고속 반복 계산 방법 및 장치

(57) 요약

본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 방법은 아이코날 방정식을 계산할 도메인을 설정하는 단계, 도메인으로부터 복수의 그리드 노드(grid node)를 도출하는 단계, 도메인을 기 설정된 개수의 복수의 블록으로 나누는 단계, 아이코날 방정식에 의하여 계산할 거리의 기준이 되는 씨드 노드(seed node)를 포함하는 블록으로부터 복수의 블록의 각각까지 거리를 계산하는 단계, 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹(n개)으로 분류하는 단계, 씨드 노드로부터 거리가 i번째(i는 1부터 n의 자연수)로 가까운 i번째 그룹의 블록에 포함된 그리드 노드 중 일부를 액티브 리스트에 포함하는 제1 단계 및 액티브 리스트에 포함된 그리드 노드를 기준으로 i번째 FIM(fast iterative method)을 수행하는 제2 단계를 포함하고, 이때 제1 단계 및 제2 단계는 i가 1부터 n이 될 때까지 반복하여 수행된다.

대표도 - 도3



이 발명을 지원한 국가연구개발사업

과제고유번호 R0190-15-2012

부처명 교육과학기술부

연구관리전문기관 정보통신기술진흥센터

연구사업명 SW컴퓨팅산업원천기술개발사업

연구과제명 빅데이터 처리 고도화 핵심 기술개발 사업 총괄 및 고성능 컴퓨팅 기술을 활용한 성능 가속화 기술 개발

기 여 율 1/1

주관기관 울산과학기술원

연구기간 2016.06.01 ~ 2017.05.31

공지예외적용 : 있음

명세서

청구범위

청구항 1

아이코날 방정식을 계산할 도메인을 설정하는 단계;

상기 도메인으로부터 복수의 그리드 노드(grid node)를 도출하는 단계;

상기 도메인을 기 설정된 개수의 복수의 블록으로 나누는 단계;

상기 아이코날 방정식에 의하여 계산할 거리의 기준이 되는 씨드 노드(seed node)를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리를 계산하는 단계;

상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 단계 -상기 그룹은 n 개임(n은 2 이상의 자연수)-;

상기 씨드 노드를 포함하는 블록으로부터 거리가 i번째(i는 1부터 n의 자연수)로 가까운 i번째 그룹의 블록에 포함된 그리드 노드 중 일부를 액티브 리스트에 포함시키는 제1 단계; 및

상기 액티브 리스트에 포함된 그리드 노드를 기준으로 i번째 FIM(fast iterative method)을 수행하는 제2 단계를 포함하고,

상기 제1 단계 및 상기 제2 단계는 상기 i가 1부터 n이 될 때까지 반복하여 수행되는

그룹 순서 기반 고속 반복 계산 방법.

청구항 2

제1항에 있어서,

상기 동일 그룹으로 분류하는 단계는,

상기 씨드 노드를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리에 대하여 k-means clustering 알고리즘을 적용하여 상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 단계를 포함하는

그룹 순서 기반 고속 반복 계산 방법.

청구항 3

제1항에 있어서,

상기 제1 단계는,

i가 2 이상인 경우, i번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 i-1번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드를 상기 액티브 리스트에 포함시키는 단계를 포함하는

그룹 순서 기반 고속 반복 계산 방법.

청구항 4

제1항에 있어서,

상기 제1 단계는,

i가 2 이상인 경우, i번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 i-1번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드에 대하여 아이코날 방정식의 해를 계산하는 단계; 및

상기 해가 기 설정된 값에 수렴하는 그리드 노드를 액티브 리스트에 포함시키는 단계를 포함하는

그룹 순서 기반 고속 반복 계산 방법.

청구항 5

아이코날 방정식을 계산할 도메인을 설정하는 도메인 설정부;

상기 도메인으로부터 복수의 그리드 노드(grid node)를 도출하는 그리드 노드 도출부;

상기 도메인을 기 설정된 개수의 복수의 블록으로 나누고, 상기 아이코날 방정식에 의하여 도출될 거리의 기준이 되는 씨드 노드(seed node)를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리를 계산하는 거리 계산부;

상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 그룹 분류부-상기 그룹은 n개임(n은 2 이상의 자연수)-; 및

상기 씨드 노드를 포함하는 블록으로부터 거리가 i번째(i는 1부터 n의 자연수)로 가까운 i번째 그룹의 블록에 포함된 그리드 노드를 액티브 리스트에 포함하는 제1 과정 및 상기 액티브 리스트에 포함된 그리드 노드를 기준으로 i번째 FIM(fast iterative method)을 수행하는 제2 과정을 수행하되, 상기 제1 과정 및 상기 제2 과정은 상기 i가 1부터 n이 될 때까지 반복하여 수행하는 FIM 계산부를 포함하는

그룹 순서 기반 고속 반복 계산 장치.

청구항 6

제1항 내지 제4항 중 어느 한 항의 방법을 프로세서가 수행하도록 하는 컴퓨터 판독 가능 기록매체에 저장된 프로그램.

청구항 7

제1항 내지 제4항 중 어느 한 항의 방법을 프로세서가 수행하게 하는 명령어를 포함하는 프로그램이 기록된 컴퓨터 판독 가능 기록매체.

발명의 설명

기술 분야

[0001] 본 발명은 그룹 순서 기반 고속 반복 계산 방법 및 장치에 관한 것으로서, 보다 자세하게는 아이코날 방정식(eikonal equation)을 컴퓨터 상에서 수행함에 있어서 그룹 단위의 그리드 노드(grid node)간 종속성을 이용하여 FIM(fast iterative method)을 수행하는 그룹 순서 기반 고속 반복 계산 방법 및 장치에 관한 것이다.

배경 기술

[0002] 아이코날 방정식은 두 점간 광학적 최단 거리 또는 기하학적 최단 거리를 도출할 수 있도록 하는 방정식이다.

[0003] 아이코날 방정식을 컴퓨터 상에서 수행하기 위해 가장 널리 쓰이던 FMM(fast marching method)은 아이코날 방정식을 풀기 위하여 그리드 노드(grid node) 간의 종속성에 따라 계산을 수행함으로써, 중복 계산의 가능성을 원천 봉쇄하였지만 종속성에 따라 그리드 노드를 순차적으로 계산하여야 한다는 특성상 복수의 스레드가 병렬적으로 계산을 수행하도록 하기 어려운 문제가 있었다.

[0004] 이를 해결하기 위하여 FIM(fast iterative method)은 아이코날 방정식의 병렬 계산을 위해 그리드 노드간의 종속성과 상관없이 액티브 리스트(Active List)에 속한 그리드 노드들의 병렬적인 계산 방법을 제시하였다. (Won-Ki Jeong and Ross T. Whitaker, A fast iterative method for eikonal equations, SIAM J. Sci. Comput. Vol.30 No.5, pages 2512-2534, 2008.)

[0005] 그러나 FIM은 액티브 리스트에 속한 그리드 노드들의 종속성을 제대로 고려하지 않기 때문에 도메인의 특성에 따라 많은 중복 계산을 허용한다는 문제를 가지고 있다.

선행기술문헌

특허문헌

[0006] (특허문헌 0001) 미국등록특허 제8,762,442호(2011.02.24 공고)

발명의 내용

해결하려는 과제

[0007] 본 발명의 실시예에서 해결하고자 하는 과제는 FIM을 수행함에 있어 도메인의 특성에 구애 받지 않고 중복 계산을 최소화하면서 병렬 계산이 가능하도록 하는 기술을 제공하는 것이다.

[0008] 다만, 본 발명의 실시예가 이루고자 하는 기술적 과제는 이상에서 언급한 과제로 제한되지 않으며, 이하에서 설명할 내용으로부터 통상의 기술자에게 자명한 범위 내에서 다양한 기술적 과제가 도출될 수 있다.

과제의 해결 수단

[0009] 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 방법은 아이코날 방정식을 계산할 도메인을 설정하는 단계, 상기 도메인으로부터 복수의 그리드 노드(grid node)를 도출하는 단계, 상기 도메인을 기 설정된 개수의 복수의 블록으로 나누는 단계, 상기 아이코날 방정식에 의하여 계산할 거리의 기준이 되는 씨드 노드(seed node)를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리를 계산하는 단계, 상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 단계 -상기 그룹은 n 개임(n 은 2 이상의 자연수)-, 상기 씨드 노드를 포함하는 블록으로부터 거리가 i 번째(i 는 1부터 n 의 자연수)로 가까운 i 번째 그룹의 블록에 포함된 그리드 노드 중 일부를 액티브 리스트에 포함시키는 제1 단계 및 상기 액티브 리스트에 포함된 그리드 노드를 기준으로 i 번째 FIM(fast iterative method)을 수행하는 제2 단계를 포함하고, 상기 제1 단계 및 상기 제2 단계는 상기 i 가 1부터 n 이 될 때까지 반복하여 수행될 수 있다.

[0010] 이때 상기 동일 그룹으로 분류하는 단계는 상기 씨드 노드를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리에 대하여 k -means clustering 알고리즘을 적용하여 상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 단계를 포함할 수 있다.

[0011] 또한 상기 제1 단계는 i 가 2 이상인 경우, i 번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 $i-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드를 상기 액티브 리스트에 포함시키는 단계를 포함할 수 있다.

[0012] 더하여 상기 제1 단계는 i 가 2 이상인 경우, i 번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 $i-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드에 대하여 아이코날 방정식의 해를 계산하는 단계 및 상기 해가 기 설정된 값에 수렴하는 그리드 노드를 액티브 리스트에 포함시키는 단계를 포함할 수 있다.

[0013] 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 장치는 아이코날 방정식을 계산할 도메인을 설정하는 도메인 설정부, 상기 도메인으로부터 복수의 그리드 노드(grid node)를 도출하는 그리드 노드 도출부, 상기 도메인을 기 설정된 개수의 복수의 블록으로 나누고, 상기 아이코날 방정식에 의하여 도출될 거리의 기준이 되는 씨드 노드(seed node)를 포함하는 블록으로부터 상기 복수의 블록의 각각까지 거리를 계산하는 거리 계산부, 상기 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류하는 그룹 분류부-상기 그룹은 n 개임(n 은 2 이상의 자연수)- 및 상기 씨드 노드를 포함하는 블록으로부터 거리가 i 번째(i 는 1부터 n 의 자연수)로 가까운 i 번째 그룹의 블록에 포함된 그리드 노드를 액티브 리스트에 포함하는 제1 과정 및 상기 액티브 리스트에 포함된 그리드 노드를 기준으로 i 번째 FIM(fast iterative method)을 수행하는 제2 과정을 수행하되, 상기 제1 과정 및 상기 제2 과정은 상기 i 가 1부터 n 이 될 때까지 반복하여 수행하는 FIM 계산부를 포함한다.

발명의 효과

[0014] 본 발명의 실시예에 따르면, 복수의 그리드 노드로 이루어진 그룹 단위의 종속성에 따라 하나의 그룹의 연산이 끝나기 전에 다른 그룹으로의 진행을 허용하지 않으면서, 동일 그룹에 속한 그리드 노드 간에는 FIM의 계산을 허용하여 중복 계산을 최소화할 수 있다.

[0015] 또한 중복 계산을 최소화함에 따라 효율적인 계산이 가능하기 때문에 도메인의 특성에 구애 받지 않고 범용적으로 활용할 수 있으며, 복잡하고 대용량인 데이터를 빠르게 연산할 수 있다.

도면의 간단한 설명

- [0016] 도 1은 기존 FIM의 계산 과정을 설명하기 위한 예시도이다.
- 도 2는 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 장치의 기능 블록도이다.
- 도 3은 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 방법의 프로세스를 도시하는 흐름도이다.
- 도 4a 내지 도 4c는 본 발명의 일 실시예에 따라 씨드 노드로부터 각 블록간의 거리를 계산하는 것을 설명하기 위한 예시도이다.
- 도 5a 내지 도 5c는 본 발명의 일 실시예에 따라 각 그룹의 순번에서 FIM을 수행하기 위한 그리드 노드의 시작 지점을 설명하기 위한 예시도이다.
- 도 6a 내지 도 6d는 그리드 노드 간의 종속성으로 인하여 수행되는 중복 계산을 설명하기 위한 예시도이다.
- 도 7a 내지 도 7d는 그리드 노드 간의 종속성을 고려하여 계산하는 경우 계산이 간소화됨을 설명하기 위한 예시도이다.

발명을 실시하기 위한 구체적인 내용

- [0017] 본 발명의 목적과 기술적 구성 및 그에 따른 작용 효과에 관한 자세한 사항은 본 발명의 명세서에 첨부된 도면에 의거한 이하의 상세한 설명에 의해 보다 명확하게 이해될 것이다. 첨부된 도면을 참조하여 본 발명에 따른 실시예를 상세하게 설명한다.
- [0018] 본 명세서에서 개시되는 실시예들은 본 발명의 범위를 한정하는 것으로 해석되거나 이용되지 않아야 할 것이다. 이 분야의 통상의 기술자에게 본 명세서의 실시예를 포함한 설명은 다양한 응용을 갖는다는 것이 당연하다. 따라서, 본 발명의 상세한 설명에 기재된 임의의 실시예들은 본 발명을 보다 잘 설명하기 위한 예시적인 것이며 본 발명의 범위가 실시예들로 한정되는 것을 의도하지 않는다.
- [0019] 도면에 표시되고 아래에 설명되는 기능 블록들은 가능한 구현의 예들일 뿐이다. 다른 구현들에서는 상세한 설명의 사상 및 범위를 벗어나지 않는 범위에서 다른 기능 블록들이 사용될 수 있다. 또한, 본 발명의 하나 이상의 기능 블록이 개별 블록들로 표시되지만, 본 발명의 기능 블록들 중 하나 이상은 동일 기능을 실행하는 다양한 하드웨어 및 소프트웨어 구성들의 조합일 수 있다.
- [0020] 또한, 어떤 구성 요소들을 포함한다는 표현은 개방형의 표현으로서 해당 구성 요소들이 존재하는 것을 단순히 지칭할 뿐이며, 추가적인 구성 요소들을 배제하는 것으로 이해되어서는 안 된다.
- [0021] 나아가 어떤 구성 요소가 다른 구성 요소에 연결되어 있다거나 접속되어 있다고 언급될 때에는, 그 다른 구성 요소에 직접적으로 연결 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 한다.
- [0022] 또한 '제1, 제2' 등과 같은 표현은 복수의 구성들을 구분하기 위한 용도로만 사용된 표현으로써, 구성들 사이의 순서나 기타 특징들을 한정하지 않는다.
- [0023] 이하에서는 도면들을 참조하여 본 발명의 실시예들에 대해 설명하도록 한다.
- [0024] 아이코날 방정식은 두 점간 광학적 최단 거리 또는 기하학적 최단 거리를 도출할 수 있도록 하는 방정식으로 아래 수학적 1과 같다.

수학적 1

[0025]
$$H(\mathbf{x}, \nabla \phi) = |\nabla \phi(\mathbf{x})|^2 - \frac{1}{f^2(\mathbf{x})} = 0 \quad \forall \mathbf{x} \in \Omega,$$

- [0026] 이때 Ω 은 계산을 수행할 대상이 되는 도메인을 의미하며, $\emptyset$ 는 도메인 상에서 특정 지점까지의 거리를 계산할 기준이 되는 지점이고, $\mathbf{x}$ 는 도메인 상에서의 특정 지점으로, $\nabla \emptyset(\mathbf{x})$ 는 지점 $\emptyset$ 로부터 지점 $\mathbf{x}$ 까지의 최단 거리를 의미하며, $f(\mathbf{x})$ 는 Ω 에서의 속도 함수이고, $H(\mathbf{x}, \nabla \emptyset(\mathbf{x}))$ 를 0으로 두었을 때 위 방정식의 해 $\nabla \emptyset(\mathbf{x})$ 를 구하여 지점 $\emptyset$ 로부터 지점 $\mathbf{x}$ 까지의 최단 거리를 구할 수 있다.

[0027] 이러한 아이코날 방정식을 컴퓨터 상에서 수행하기 위해서는 연속적으로 이루어진 도메인을 불연속적으로 나누어야 하기 때문에, 예를 들면 기존에 사용되는 Godunov upwind difference scheme을 통해 도메인을 불연속적으로 이루어진 그리드 노드(grid node)로 분할할 수 있다.

[0028] Godunov upwind difference scheme을 통해 도메인이 불연속적인 그리드 노드로 분할된 경우 도메인이 3차원이라면 아이코날 방정식은 아래 수학식 2와 같이 나타낼 수 있다.

수학식 2

$$g(\mathbf{x}) = \left[\frac{(U(\mathbf{x}) - U(\mathbf{x})^{x\min})^+}{h_x} \right]^2 + \left[\frac{(U(\mathbf{x}) - U(\mathbf{x})^{y\min})^+}{h_y} \right]^2 + \left[\frac{(U(\mathbf{x}) - U(\mathbf{x})^{z\min})^+}{h_z} \right]^2 - \frac{1}{f(\mathbf{x})^2},$$

[0029]

[0030] 이때 $U(\mathbf{x})$ 는 도메인 상에서 거리를 계산할 기준이 되는 지점 $\emptyset$ 로부터 그리드 노드 $\mathbf{x}=(i, j, k)$ 까지의 최단 거리를 의미하며, $U(\mathbf{x})^{p\min}$ 은 p 축($p \in \{x, y, z\}$)상에서 $U(\mathbf{x})$ 와 인접한 두 개의 그리드 노드 중 더 짧은 거리에 있는 그리드 노드의 $U(\mathbf{x})$ 값을 의미한다. 또한 h_p 는 p 축에 있어서 그리드 노드 간의 공간이며, $f(\mathbf{x})$ 는 속도 함수이고, $(n)^+ = \max(n, 0)$ 을 의미한다.

[0031] FIM(fast iterative method)은 아이코날 방정식을 컴퓨터 상에서 병렬적으로 계산하기 위해 액티브 리스트(active list)를 생성한다. 액티브 리스트에는 아이코날 방정식을 계산하고 있거나 계산해야 할 그리드 노드의 정보를 글로벌 메모리(global memory)에 저장하여, 복수의 스레드들이 액티브 리스트를 참조하여 액티브 리스트에 포함된 그리드 노드에 대해 아이코날 방정식을 계산함으로써 병렬적인 연산이 가능하도록 한다.

[0032] 도 1a 내지 도 1c는 기존 FIM의 계산 과정을 설명하기 위한 예시도이다.

[0033] 도 1a에 도시된 바와 같이, FIM 모델은 도메인 상에서 거리를 계산할 기준이 되는 씨드 노드(seed node)(101)와 인접한 그리드 노드(102, 103)를 액티브 리스트에 포함시켜 첫번째 연산을 통해 액티브 리스트에 포함된 그리드 노드의 아이코날 방정식을 계산하고, 도 1b와 같이 첫번째 연산이 끝나면 첫번째 연산에서 계산된 그리드 노드(102, 103)와 인접한 그리드 노드(104, 105, 106)를 액티브 리스트에 추가하여 액티브 리스트에 있는 그리드 노드에 대하여 아이코날 방정식을 계산하도록 한다. 이때 아이코날 방정식의 계산을 통해 특정 값에 수렴하게 되는 그리드 노드는 액티브 리스트에서 제외시키며, 이 과정을 모든 그리드 노드가 수렴하게 될 때까지 반복하여 아이코날 방정식의 계산을 완료한다.

[0034] 아래 표 1은 아이코날 방정식을 컴퓨터 상에서 계산하기 위한 기존 FIM의 수도 코드를 나타낸 것이다.

표 1

Algorithm 1: FIM

Input: Grid Ω , Solution U , Active list L

/ Initialization */*

```

1 forall  $x \in \Omega$  do
2   if  $x$  is a source node then
3      $U(x) \leftarrow 0$ 
4     add  $x$  to  $L$ 
5   else
6      $U(x) \leftarrow \infty$ 
7   /* Compute new solutions for  $L$  */
8   while  $L$  is not empty do
9     forall  $x \in L$  do
10       $p \leftarrow U(x)$ 
11       $q \leftarrow \text{solution of } g(x) = 0$ 
12      /* If not converged */
13      if  $p > q$  then
14         $U(x) \leftarrow q$ 
15      /* If converged */
16      else
17        /* Check adjacent neighbor nodes
18        for reactivation */
19        forall  $x_{nb}$  adjacent to  $x$  do
20          if  $U(x_{nb}) > U(x)$  and  $x_{nb} \notin L$  then
21             $p \leftarrow U(x_{nb})$ 
22             $q \leftarrow \text{solution of } g(x_{nb}) = 0$ 
23            if  $p > q$  then
24               $U(x_{nb}) \leftarrow q$ 
25              add  $x_{nb}$  to  $L$ 
26      remove  $x$  from  $L$ 

```

[0035]

[0036] 표 1에 표기된 source node는 도메인 ?에서 특정 그리드 노드까지의 거리를 계산할 기준이 되는 씨드 노드(seed node)를 의미한다. 따라서 씨드 노드 자체의 $U(x)$ 는 0 이며, 그 외의 그리드 노드의 아이코날 방정식의 해를 ∞ 로 초기 설정한 후, 씨드 노드로부터 인접한 그리드 노드를 액티브 리스트에 추가하면서 액티브 리스트에 포함된 그리드 노드에 대한 아이코날 방정식을 수행한다. 이때 이전에 계산한 아이코날 방정식의 해인 p 와 가장 최근 계산한 아이코날 방정식의 해 q 를 비교하여 특정 그리드 노드에서 $q-p < 0$ 인 경우 해당 그리드 노드에서의 아이코날 방정식의 해가 수렴한 것으로 간주하여 액티브 리스트에서 해당 그리드 노드를 제외하는 방식으로 모든 그리드 노드가 수렴할 때까지 반복하여 아이코날 방정식을 계산한다.

[0037] 한편, 아이코날 방정식의 특성상 최단 거리를 구하기 위해서는 씨드 노드에서 거리가 가까운 그리드 노드부터 거리를 계산하고, 그 다음에서야 거리가 계산된 그리드 노드에서 거리가 가까운 그리드 노드를 계산할 수 있다. 이러한 특성상 그리드 노드간에는 계산의 종속성이 존재한다.

[0038] 그러나 기존 FIM은 액티브 리스트에 속한 그리드 노드들의 종속성을 고려하지 않기 때문에 도메인의 특성에 따라 많은 중복 계산을 허용한다는 문제를 가지고 있다. 이에, 본 발명의 실시예는 FIM을 수행함에 있어 도메인의 특성에 구애 받지 않고 중복 계산을 최소화하면서 병렬 계산이 가능하기 위한 기술로서 아래 도 2 내지 도 8과 함께 설명하도록 한다.

[0039] 도 2는 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 장치(200)의 기능 블록도이다.

[0040] 도 2에 도시된 바와 같이, 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 장치(200)는 도메인 설정부(210), 그리드 노드 도출부(220), 거리 계산부(230), 그룹 분류부(240) 및 FIM 계산부(250)를 포함한다.

[0041] 도메인 설정부(210)는 아이코날 방정식을 계산할 도메인을 설정한다. 예를 들어, 특정 지역에서 지진파가 전달되는 거리 또는 속도를 알기 위해 특정 지역의 지리학적 정보를 도메인으로 설정할 수 있다.

[0042] 그리드 노드 도출부(220)는 도메인으로부터 복수의 그리드 노드를 도출한다. 예를 들어, 그리드 노드 도출부(22

0)는 Godunov upwind difference scheme을 통해 도메인을 불연속적으로 이루어진 그리드 노드(grid node)로 분할할 수 있다.

- [0043] 거리 계산부(230)는 도메인을 기 설정된 개수의 복수의 블록으로 나누고, 아이코날 방정식에 의하여 도출될 거리의 기준이 되는 씨드 노드를 포함하는 블록으로부터 복수의 블록의 각각까지 거리를 계산한다. 이때 블록 간의 거리는 각 블록 내에 포함된 어느 하나의 그리드 노드 간의 거리를 통해 산출할 수 있다. 또는 블록의 중점에 위치하는 그리드 노드 간의 거리를 통해 산출할 수 있다. 한편, 씨드 노드를 포함하는 블록 자체는 씨드 노드를 포함하고 있으므로 거리의 값이 0이라고 설정할 수 있다.
- [0044] 그룹 분류부(240)는 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류할 수 있다.
- [0045] FIM 계산부(250)는 나누어진 그룹의 개수가 n개라고 할 경우, 씨드 노드를 포함하는 블록으로부터 거리가 i번째 (i는 1부터 n의 자연수)로 가까운 i번째 그룹의 블록에 포함된 그리드 노드를 액티브 리스트에 포함하는 제1 과정 및 액티브 리스트에 포함된 그리드 노드를 기준으로 i번째 FIM을 수행하는 제2 과정을 수행하되, 제1 과정 및 제2 과정은 i가 1부터 n이 될 때까지 반복하여 수행할 수 있다.
- [0046] 이때 그룹 순서 기반 고속 반복 계산 장치(200)의 각 구성이 동작하는 구체적인 과정을 도 3과 함께 설명하기로 한다.
- [0047] 한편 상술한 실시예가 포함하는 도메인 설정부(210), 그리드 노드 도출부(220), 거리 계산부(230), 그룹 분류부(240) 및 FIM 계산부(250)는 이들의 기능을 수행하도록 프로그램된 명령어를 포함하는 메모리, 및 이들 명령어를 수행하는 마이크로프로세서를 포함하는 연산 장치에 의해 구현될 수 있다.
- [0048] 도 3은 본 발명의 일 실시예에 따른 그룹 순서 기반 고속 반복 계산 방법의 프로세스를 도시하는 흐름도이다. 도 3에 따른 그룹 순서 기반 고속 반복 계산 방법의 각 단계는 도 2를 통해 설명된 그룹 순서 기반 고속 반복 계산 장치(200)에 의해 수행될 수 있으며, 각 단계를 설명하면 다음과 같다.
- [0049] 우선, 도메인 설정부(210)는 아이코날 방정식을 계산할 도메인을 설정하고(S310), 그리드 노드 도출부(220)는 도메인으로부터 복수의 그리드 노드를 도출한다(S320).
- [0050] 이후, 거리 계산부(230)는 도메인을 기 설정된 개수의 복수의 블록으로 나누고, 아이코날 방정식에 의하여 도출될 거리의 기준이 되는 씨드 노드를 포함하는 블록으로부터 복수의 블록의 각각까지 거리를 계산한다(S330).
- [0051] 예를 들어, 도 4a 내지 도 4c를 참조할 때 도메인으로부터 도출된 그리드 노드가 15x15개라고 할 때, 도 4a와 같이 도메인을 5x5개의 블록으로 나누면, 한 블록에는 가로 3개, 세로 3개의 그리드 노드를 포함한다. 이때 각 블록 간의 종속성을 알기 위해 블록 단위로 씨드 노드가 포함된 블록으로부터 다른 블록 간의 거리를 측정한다. 이때 거리는 예를 들어 아이코날 방정식의 해가 될 수 있고, 아이코날 방정식을 풀기 위해 FMM(fast matching method) 또는 FIM의 모델을 사용할 수 있다. 한편, 블록의 개수는 도메인의 크기 또는 그리드 노드의 개수에 따라 사용자가 설정할 수 있다.
- [0052] 기존 FIM은 노드 간의 종속성에 대한 고려 없이 중복하여 계산하는 방식으로 그리드 노드의 거리를 계산하였다. 이와 달리, 본 발명의 일 실시예에서는 종속성을 고려하되, 모든 그리드 노드의 종속성을 분석하는 것이 아니라 그리드 노드를 블록 단위로 구분하여 블록 간 종속성을 계산하여 종속성에 따라 블록 단위로 FIM을 수행한다.
- [0053] 이때 모든 그리드 노드에 대한 종속성을 구하는 것은 결국 아이코날 방정식을 수행하는 것이기 때문에 매우 오랜 시간이 걸리지만, 예를 들어 15x15 = 225개의 그리드 노드를 5x5 = 25개의 블록 단위로 압축하여 미리 거리를 구하는 계산은 매우 빠르게 연산이 가능하다.
- [0054] 즉, 도 4b에 도시된 바와 같이 아이코날 방정식에 의하여 도출될 거리의 기준이 되는 씨드 노드를 도메인 정중앙의 블록에 포함된 그리드 노드로 설정하였다면, 거리 계산부(230)는 씨드 노드를 포함하는 블록으로부터 25개의 블록까지의 거리를 계산할 수 있다. 이때 블록 간의 거리는 각 블록 내에 포함된 어느 하나의 그리드 노드 간의 거리를 통해 산출할 수 있다. 또는 블록의 중점에 위치하는 그리드 노드 간의 거리를 통해 산출할 수 있다. 한편, 씨드 노드를 포함하는 블록 자체는 씨드 노드를 포함하고 있으므로 거리의 값이 0이라고 설정할 수 있다.
- [0055] 이후, 그룹 분류부(240)는 도 4c에 도시된 바와 같이 씨드 노드를 포함하는 블록으로부터 거리가 유사한 블록들을 동일 그룹으로 분류할 수 있다(S340).

- [0056] 한편, 도 4b에 도시된 블록에는 씨드 노드로부터의 거리가 동일한 블록들이 도시되어 있으나, 실제 분석에서는 각 블록마다 씨드 노드로부터 도출된 거리가 상이할 것이다. 따라서 거리가 유사한 블록들을 그룹으로 구분 짓기 위해 k-means clustering 알고리즘을 사용할 수 있다.
- [0057] 이때 나누어진 그룹의 개수가 n 개라고 할 경우, FIM 계산부(250)는 씨드 노드를 포함하는 블록으로부터 거리가 i 번째(i 는 1부터 n 의 자연수)로 가까운 i 번째 그룹의 블록에 포함된 그리드 노드를 액티브 리스트에 포함하는 제1 단계(S350) 및 액티브 리스트에 포함된 그리드 노드를 기준으로 i 번째 FIM을 수행하는 제2 단계(S360)를 수행하되, 제1 단계 및 제2 단계는 i 가 1부터 n 이 될 때까지 반복하여 수행할 수 있다.
- [0058] 예를 들어, 도 5a 내지 도 5c에 도시된 바와 같이 그룹 1, 2, 3이 있을 때, 도 5a와 같이 1번째 그룹을 대상으로 1번째 FIM을 수행한다. 이후, 1번째 그룹에 대하여 FIM에 따라 계산이 끝나면, 도 5b와 같이 1번째 그룹 및 2번째 그룹을 대상으로 2번째 FIM을 수행한다. 이후, 1번째 그룹 및 2번째 그룹에 대하여 FIM에 따라 계산이 끝나면, 도 5c와 같이 1번째 그룹, 2번째 그룹 및 3번째 그룹을 대상으로 3번째 FIM을 수행한다. 이와 같은 방식으로 모든 그리드 노드들에 대하여 종속성을 고려한 그룹의 순서대로 FIM을 수행하여 중복계산을 최소화할 수 있다.
- [0059] 이때 도 5b 및 도 5c와 같이, 2번째 이상의 FIM 수행 시, i 번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 $i-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드를 i 번째 FIM을 수행함에 있어서의 최초 액티브 리스트에 포함할 수 있다. 이에 따라, i 번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 $i-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드가 이미 계산된 그리드 노드와 인접하고 있으므로 이미 계산된 그리드 노드와의 종속성이 가장 크기 때문에, 이들을 먼저 계산하여 중복 연산을 줄일 수 있다.
- [0060] 한편, 도 6a 내지 도 6d는 그리드 노드 간의 종속성으로 인하여 수행되는 중복 계산을 설명하기 위한 예시도이다. 도 6a에서 가장 오른쪽에 있는 그리드 노드들이 화살표 방향으로 종속성이 있다고 할 때, 가장 오른쪽에 있는 그리드 노드들을 i 번째 FIM 수행 시의 최초 액티브 리스트에 포함시켰다면, 액티브 리스트에 포함된 그리드 노드에 대하여 동시에 아이코날 방정식을 계산하여도, 종속성에 의하여 오른쪽 최하단의 그리드 노드에 대한 아이코날 방정식의 해가 수렴한 이후, 나머지 그리드 노드들에 대하여 중복하여 아이코날 방정식을 계산하여야 한다.
- [0061] 따라서 도 7a 내지 도 7d에 도시된 바와 같이, i 번째 FIM을 수행하는 데에 있어서 가장 오른쪽에 있는 그리드 노드 중 종속성의 우선 순위가 가장 높은 오른쪽 최하단의 그리드 노드만을 최초의 액티브 리스트에 포함시키면 중복 계산 없이 순차적으로 아이코날 방정식을 계산할 수 있다.
- [0062] 이를 위해, 2번째 이상의 FIM 수행 시, i 번째 그룹에 포함된 그리드 노드 중 1번째 그룹부터 $i-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드에 대하여 아이코날 방정식의 해를 계산하고, 해가 계산된 그리드 노드 중 해가 기 설정된 값에 수렴하는 그리드 노드만을 i 번째 FIM 수행 시의 최초 액티브 리스트에 포함하여 중복 계산을 줄일 수 있다.
- [0063] 상술한 실시예에 따른 단계를 수도코드로 나타내면 아래 표 2와 같다.

표 2

Algorithm 2: GROUP-ORDERED FIM	
Input: Grid Ω , Solution U , Active list L	
/* Coarse grid level */	
1	Generate and initialize coarse grid $\tilde{\Omega}$ from Ω
2	Run FIM on $\tilde{\Omega}$ to assign distance per block
3	Cluster blocks in $\tilde{\Omega}$ into k groups (G_1 to G_k)
/* Fine grid level */	
4	Initialize U and L as line 1 to 6 in Algorithm 1
5	$G = \emptyset$
6	$n \leftarrow$ number of threads
7	$N = \{0, 1, 2, \dots, n-1\}$
8	for $v = 1$ to k do
9	$G = G \cup G_v$
10	if v is 1 then
11	$L_{upper} \leftarrow L$
12	else
13	Get L_{upper} from G and G_v (see Equation 3)
14	foreach $i \in N$ do in parallel
15	Get tight active list $\tilde{L}_{tight}$ from L_i (see Equation 4)
16	$L_i \leftarrow \tilde{L}_{tight}$
17	while L_j is not empty for some $j \in N$ do
18	Solve FIM for L_j

[0064]

[0065] 이때 표 2의 L_{upper} 는 아래 수학식 3과 같이 표현할 수 있다.

수학식 3

[0066]

$$L_{upper} = \{x | x \in B \subset G_v \text{ and } \exists x_{nb} \in B' \subset G \setminus G_v\}$$

[0067]

G_v 는 v 번째 그룹, B 는 v 번째 그룹 내에 포함된 블록, x 는 v 번째 그룹 내에 포함된 그리드 노드이면서 1번째 그룹부터 $v-1$ 번째 그룹에 포함된 그리드 노드에 인접하는 그리드 노드를 의미한다.

[0068]

또한 표 2의 $\tilde{L}_{tight}$ 는 아래 수학식 4와 같이 표현할 수 있다.

수학식 4

[0069]

$$L_{tight} = \{x | x \in B \subset G_v \text{ and } \forall \text{ its upwind neighbor nodes } x_{nb} \text{ belong to } G \setminus G_v\}$$

[0070]

G_v 는 v 번째 그룹, B 는 v 번째 그룹 내에 포함된 블록, x 는 v 번째 그룹 내에 포함된 그리드 노드이면서 중 1번째 그룹부터 $v-1$ 번째 그룹에 포함된 그리드 노드로부터 종속성이 가장 우선 순위인 그리드 노드를 의미한다.

[0071]

이때 $\tilde{L}_{tight}$ 를 구하는 과정을 수도코드로 나타내면 아래 표 3과 같다.

표 3

Algorithm 3: COMPUTE TIGHT BOUND

```

Input:  $L_{upper}, \tilde{L}_{tight}$ 
/* Update distance of  $L_{upper}$  using a
   Jacobi update */
1 foreach  $x \in L_{upper}$  do in parallel
2    $p \leftarrow U(x)$ 
3    $q \leftarrow \text{solution of } g(x) = 0$ 
4    $U(x) \leftarrow q$ 
/* Update distance of  $L_{upper}$  again and
   collect converged nodes */
5 foreach  $x \in L_{upper}$  do in parallel
6    $p \leftarrow U(x)$ 
7    $q \leftarrow \text{solution of } g(x) = 0$ 
8   if  $p \leq q$  then
9      $\mid$  add  $x$  to  $\tilde{L}_{tight}$ 

```

[0072]

[0073] 이처럼 본 발명의 일 실시예에 따르면, 복수의 그리드 노드로 이루어진 그룹 단위의 종속성에 따라, 하나의 그룹의 연산이 끝나기 전에 다른 그룹으로의 진행을 허용하지 않으면서 동일 그룹에 속한 그리드 노드 간에는 FIM의 계산을 허용하여 중복 계산을 최소화할 수 있다.

[0074] 또한 중복 계산을 최소화함에 따라 효율적인 계산이 가능하기 때문에 도메인의 특성에 구애 받지 않고 범용적으로 활용할 수 있으며, 복잡하고 대용량인 데이터를 빠르게 연산할 수 있다.

[0075] 상술한 본 발명의 실시예들은 다양한 수단을 통해 구현될 수 있다. 예를 들어, 본 발명의 실시예들은 하드웨어, 펌웨어(firmware), 소프트웨어 또는 그것들의 결합 등에 의해 구현될 수 있다.

[0076] 하드웨어에 의한 구현의 경우, 본 발명의 실시예들에 따른 방법은 하나 또는 그 이상의 ASICs(Application Specific Integrated Circuits), DSPs(Digital Signal Processors), DSPDs(Digital Signal Processing Devices), PLDs(Programmable Logic Devices), FPGAs(Field Programmable Gate Arrays), 프로세서, 컨트롤러, 마이크로 컨트롤러, 마이크로 프로세서 등에 의해 구현될 수 있다.

[0077] 펌웨어나 소프트웨어에 의한 구현의 경우, 본 발명의 실시예들에 따른 방법은 이상에서 설명된 기능 또는 동작들을 수행하는 모듈, 절차 또는 함수 등의 형태로 구현될 수 있다. 소프트웨어 코드는 메모리 유닛에 저장되어 프로세서에 의해 구동될 수 있다. 상기 메모리 유닛은 상기 프로세서 내부 또는 외부에 위치하여, 이미 공지된 다양한 수단에 의해 상기 프로세서와 데이터를 주고 받을 수 있다.

[0078] 이와 같이, 본 발명이 속하는 기술분야의 당업자는 본 발명이 그 기술적 사상이나 필수적 특징을 변경하지 않고서 다른 구체적인 형태로 실시될 수 있다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적인 것이 아닌 것으로서 이해해야만 한다. 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 등가개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

[0079] 200: 그룹 순서 기반 고속 반복 계산 장치

210: 도메인 설정부

220: 그리드 노드 도출부

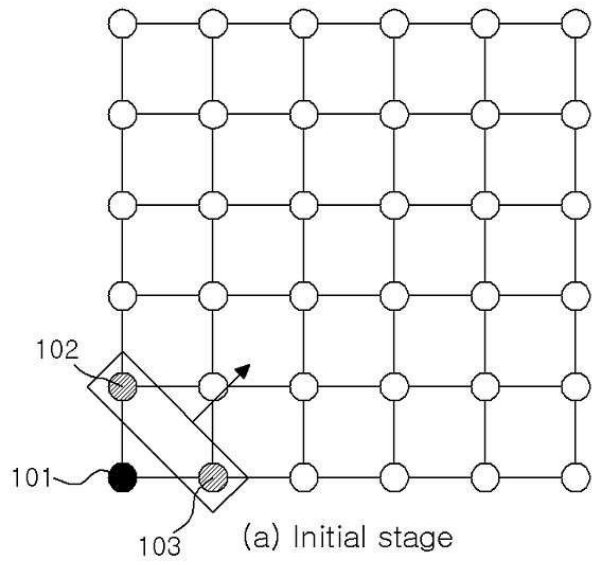
230: 거리 계산부

240: 그룹 분류부

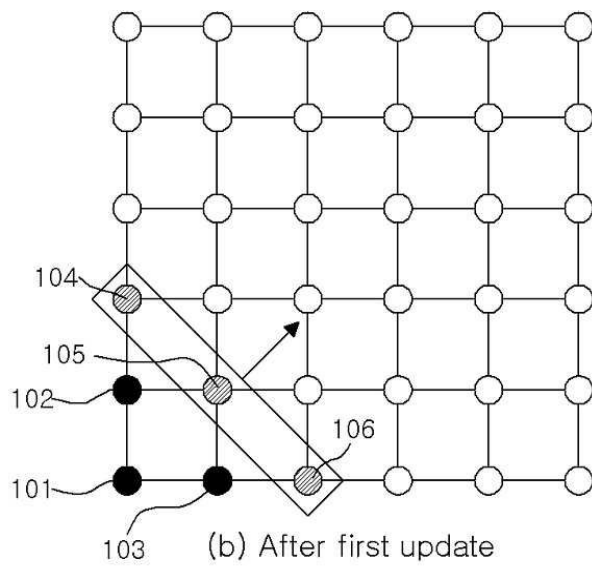
250: FIM 계산부

도면

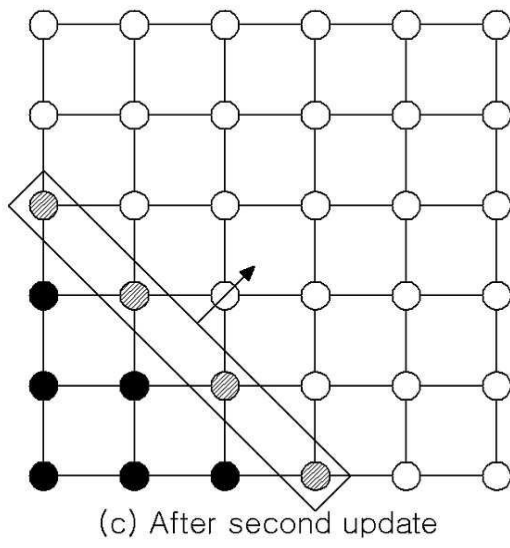
도면1a



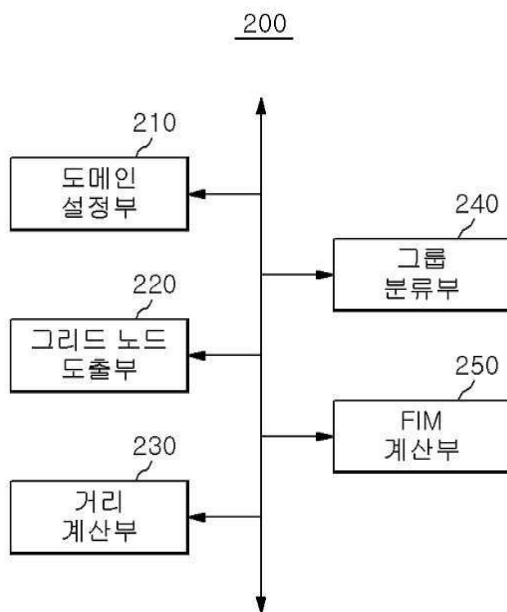
도면1b



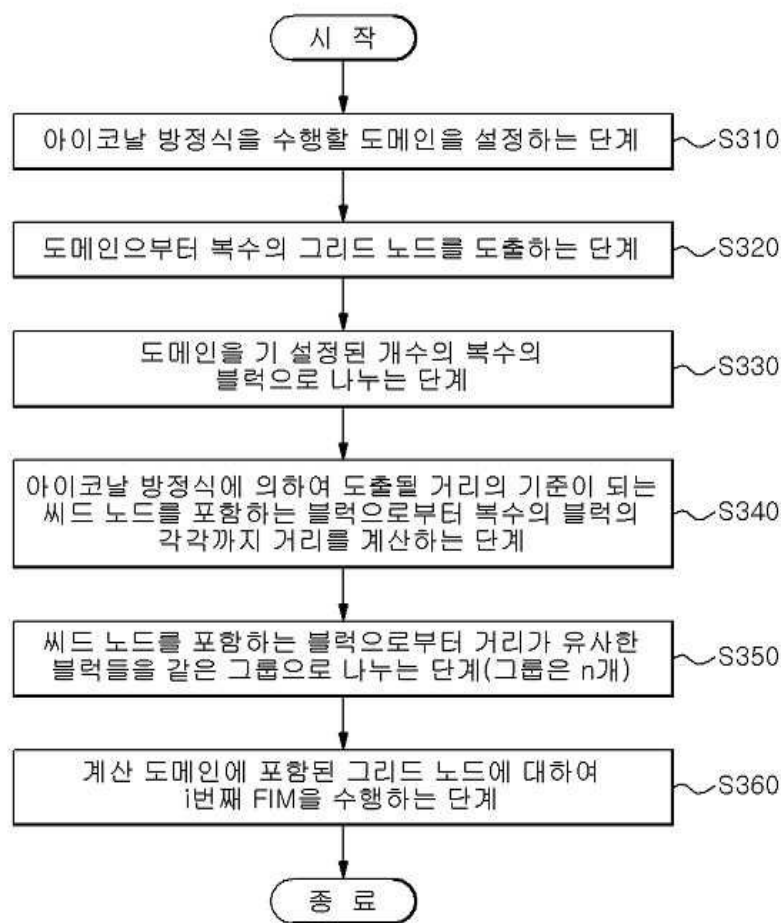
도면1c



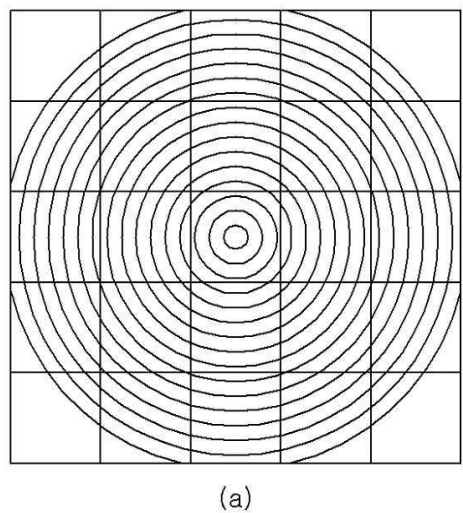
도면2



도면3



도면4a



도면4b

3.252	2.545	2	2.545	3.252
2.545	1.707	1	1.707	2.545
2	1	0	1	2
2.545	1.707	1	1.707	2.545
3.252	2.545	2	2.545	3.252

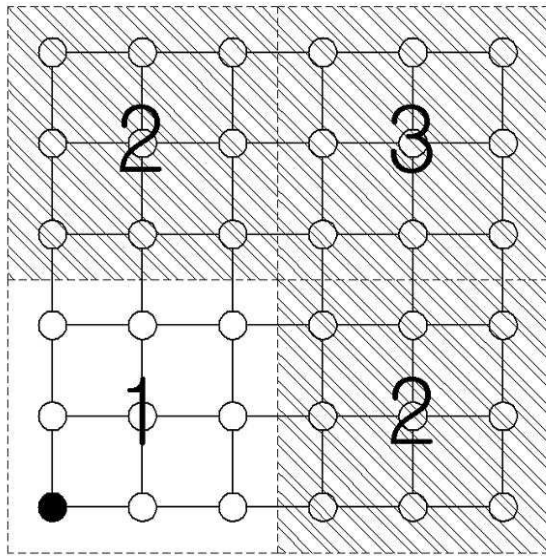
(b)

도면4c

6	5	4	5	6
5	3	2	3	5
4	2	1	2	4
5	3	2	3	5
6	5	4	5	6

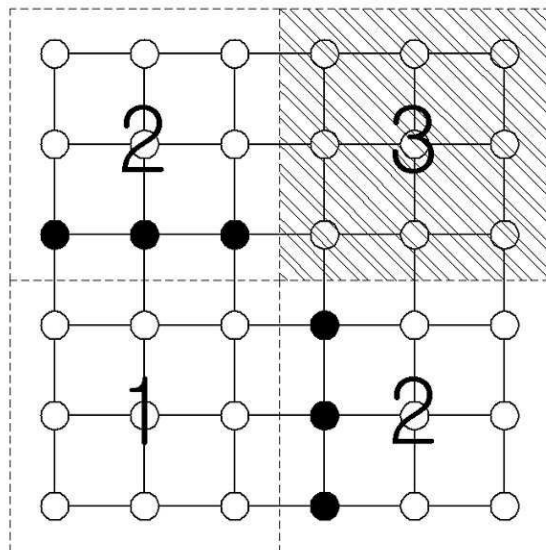
(c)

도면5a



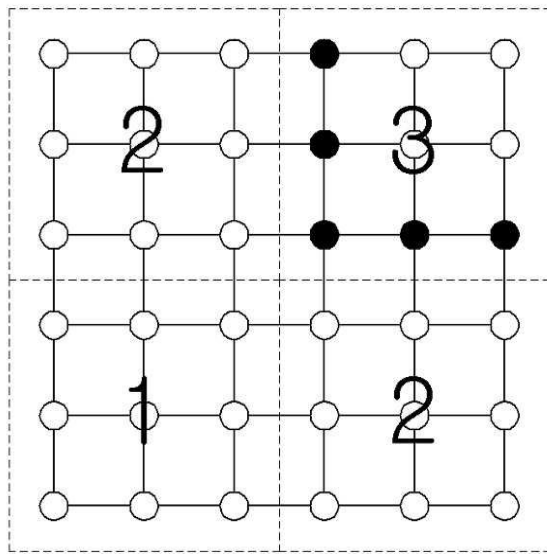
(a)

도면5b



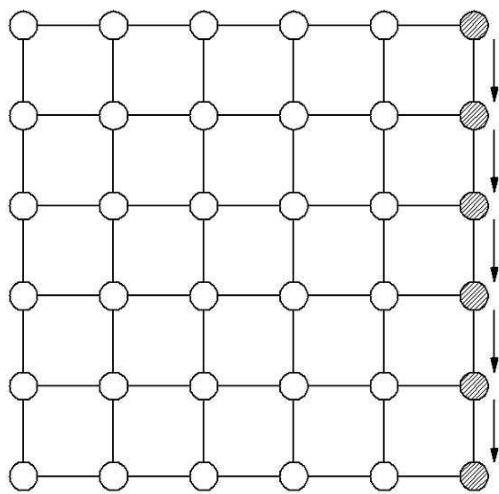
(b)

도면5c



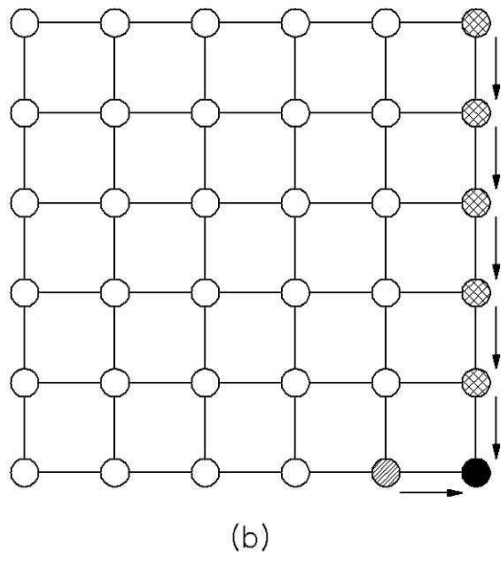
(c)

도면6a

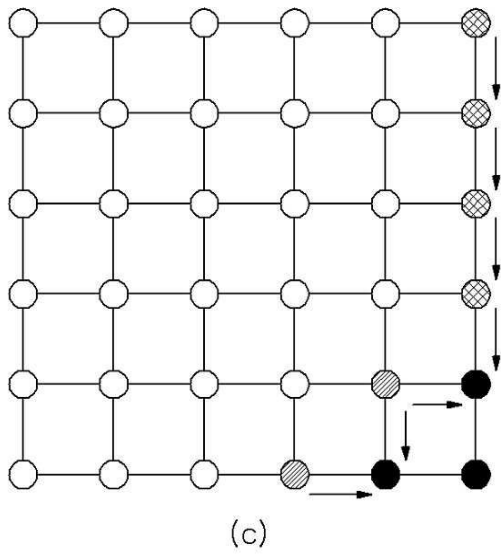


(a)

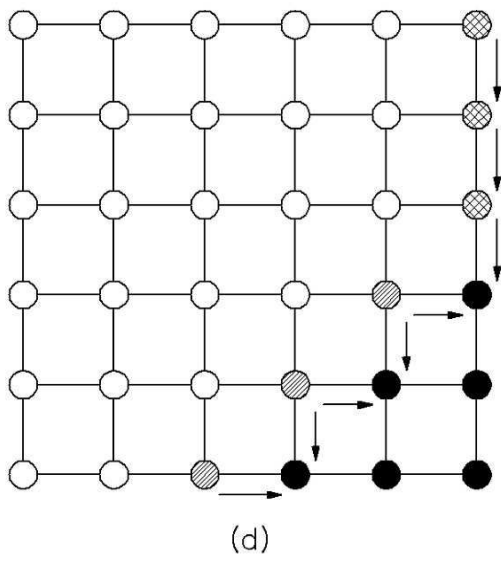
도면6b



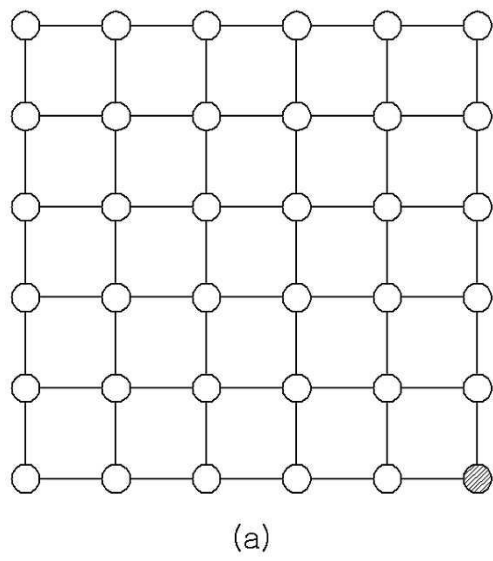
도면6c



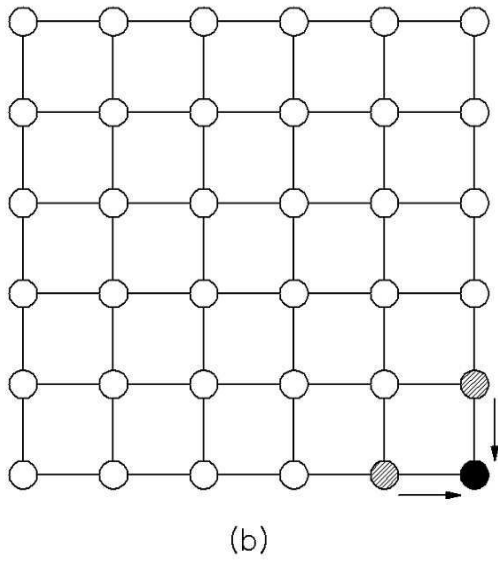
도면6d



도면7a



도면7b



도면7c

