



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2017년09월27일
(11) 등록번호 10-1782760
(24) 등록일자 2017년09월21일

(51) 국제특허분류(Int. Cl.)
G06N 3/04 (2006.01) G06F 17/18 (2006.01)
G06N 3/08 (2006.01)
(52) CPC특허분류
G06N 3/049 (2013.01)
G06F 17/18 (2013.01)
(21) 출원번호 10-2016-7012560
(22) 출원일자(국제) 2014년08월21일
심사청구일자 2017년02월17일
(85) 번역문제출일자 2016년05월12일
(65) 공개번호 10-2016-0071437
(43) 공개일자 2016년06월21일
(86) 국제출원번호 PCT/US2014/052157
(87) 국제공개번호 WO 2015/057305
국제공개일자 2015년04월23일
(30) 우선권주장
14/056,856 2013년10월17일 미국(US)
(56) 선행기술조사문헌
US20130073491 A1
Schrauwen, Benjamin, and Jan Van Campenhout.
"Extending spikeprop." Neural Networks, 2004.
Proceedings. 2004 IEEE International Joint
Conference on. Vol. 1. IEEE, 2004.
US20130073501 A1

(73) 특허권자
퀄컴 인코포레이티드
미국 92121-1714 캘리포니아주 샌 디에고 모어하우스 드라이브 5775
(72) 발명자
사라 앤서니
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
김벌 로버트 하워드
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
스피나 브라이언
미국 92121-1714 캘리포니아주 샌디에고 모어하우스 드라이브 5775
(74) 대리인
특허법인코리아나

전체 청구항 수 : 총 22 항

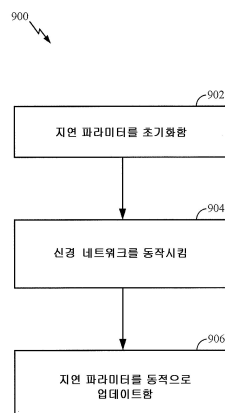
심사관 : 서광훈

(54) 발명의 명칭 시냅스 지연의 동적 할당 및 검사

(57) 요약

신경 네트워크에서 시냅스 지연들을 동적으로 변경하는 방법은 지연 파라미터를 초기화하는 것 및 신경 네트워크를 동작시키는 것을 포함한다. 본 방법은 또한 지연 파라미터를 포함한 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 동적으로 업데이트하는 것을 포함한다.

대표도 - 도9



(52) CPC특허분류
G06N 3/08 (2013.01)

명세서

청구범위

청구항 1

복수의 시냅스들을 포함하는 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법으로서, 상기 복수의 시냅스들 중 시냅스의 지연 파라미터를 초기화하는 단계;

상기 지연 파라미터를 설정하기 위한 함수를 이용하여 시뮬레이션 모드에서 상기 신경 네트워크를 동작시키는 단계;

상기 시냅스의 상기 지연 파라미터를 포함하는 상기 함수의 프로그램 스테이트먼트에 적어도 부분적으로 기초한 프로그램에 적어도 부분적으로 기초하여 상기 시뮬레이션 모드에서의 상기 신경 네트워크의 동작 동안 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 단계로서, 상기 지연 파라미터는 시냅스 유형, 뉴런 모델 유형, 또는 메모리 리소스들 중 적어도 하나에 기초하여 동적으로 업데이트되는, 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 단계; 및

상기 복수의 시냅스들의 서브세트에 대한 적어도 하나의 지연값을 동적으로 추출하는 단계를 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 2

제 1 항에 있어서,

상기 동적으로 업데이트하는 단계는, 상기 시냅스의 상기 지연 파라미터를 증분 또는 감분하는 단계를 더 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 3

제 1 항에 있어서,

상기 동적으로 업데이트하는 단계는, 상기 복수의 시냅스들의 패밀리에 대한 상기 지연 파라미터를 업데이트하는 단계를 더 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 4

제 1 항에 있어서,

상기 동적으로 업데이트하는 단계는, 상기 시냅스를 제거하도록 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 단계를 더 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 5

제 1 항에 있어서,

상기 시냅스의 상기 지연 파라미터가 특정된 지연 임계값보다 클 때, 상기 시냅스의 상기 지연 파라미터의 값을 잘라내는 (truncating) 단계를 더 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 6

제 1 항에 있어서,

동적 업데이트들의 수는 미리 정해진 임계값으로 제한되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 7

제 1 항에 있어서,

상기 동적으로 업데이트하는 단계는, 장래 시간에 발생하도록 상기 시냅스의 상기 지연 파라미터를 업데이트하는 단계를 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 8

제 1 항에 있어서,

상기 시냅스의 상기 지연 파라미터는 임의 함수 (arbitrary function) 에 적어도 부분적으로 기초하여 동적으로 업데이트되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 9

제 8 항에 있어서,

상기 임의 함수는 상기 복수의 시냅스들의 서브세트에 대한 확률 함수 또는 상기 복수의 시냅스들의 상기 서브세트 또는 상기 복수의 시냅스들의 특정 시냅스들에 대한 시간에 대한 함수인, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 10

제 9 항에 있어서,

상기 시냅스의 상기 지연 파라미터의 상기 업데이트는 확률론적 또는 결정론적인, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 구현 방법.

청구항 11

복수의 시냅스들을 포함하는 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치로서,

메모리; 및

적어도 하나의 프로세서를 포함하고,

상기 적어도 하나의 프로세서는:

상기 복수의 시냅스들 중 시냅스의 지연 파라미터를 초기화하고;

상기 지연 파라미터를 설정하기 위한 함수를 이용하여 시뮬레이션 모드에서 상기 신경 네트워크를 동작시키고;

상기 시냅스의 상기 지연 파라미터를 포함하는 상기 함수의 프로그램 스테이트먼트에 적어도 부분적으로 기초한 프로그램에 적어도 부분적으로 기초하여 상기 시뮬레이션 모드에서의 상기 신경 네트워크의 동작 동안 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 것으로서, 상기 지연 파라미터는 시냅스 유형, 뉴런 모델 유형, 또는 메모리 리소스들 중 적어도 하나에 기초하여 동적으로 업데이트되는, 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하고; 그리고

상기 복수의 시냅스들의 서브세트에 대한 적어도 하나의 지연값을 동적으로 추출하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 12

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 상기 시냅스의 상기 지연 파라미터를 증분 또는 감분하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 13

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 상기 복수의 시냅스들의 패밀리에 대한 상기 지연 파라미터를 동적으로

업데이트하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 14

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 상기 시냅스를 제거하도록 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 15

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한:

상기 시냅스의 상기 지연 파라미터가 특정된 지연 임계값보다 클 때, 상기 시냅스의 상기 지연 파라미터의 값을 잘라내도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 16

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 동적 업데이트들의 수를 미리 정해진 임계값으로 제한하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 17

제 11 항에 있어서,

상기 적어도 하나의 프로세서는 또한, 장래 시간에 발생하도록 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하도록 구성되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 18

제 11 항에 있어서,

상기 시냅스의 상기 지연 파라미터는 임의 함수에 적어도 부분적으로 기초하여 동적으로 업데이트되는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 19

제 18 항에 있어서,

상기 임의 함수는 상기 복수의 시냅스들의 서브세트에 대한 확률 함수 또는 상기 복수의 시냅스들의 상기 서브 세트 또는 상기 복수의 시냅스들의 특정 시냅스들에 대한 시간에 대한 함수인, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 20

제 19 항에 있어서,

상기 시냅스의 상기 지연 파라미터의 상기 업데이트는 확률론적 또는 결정론적인, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 21

복수의 시냅스들을 포함하는 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치로서,

상기 복수의 시냅스들 중 시냅스의 지연 파라미터를 초기화하는 수단;

상기 지연 파라미터를 설정하기 위한 함수를 이용하여 시뮬레이션 모드에서 상기 신경 네트워크를 동작시키는 수단;

상기 시냅스의 상기 지연 파라미터를 포함하는 상기 함수의 프로그램 스테이트먼트에 적어도 부분적으로 기초한

프로그램에 적어도 부분적으로 기초하여 상기 시뮬레이션 모드에서의 상기 신경 네트워크의 동작 동안 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 수단으로서, 상기 지연 파라미터는 시냅스 유형, 뉴런 모델 유형, 또는 메모리 리소스들 중 적어도 하나에 기초하여 동적으로 업데이트되는, 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 수단; 및

상기 복수의 시냅스들의 서브세트에 대한 적어도 하나의 지연값을 동적으로 추출하는 수단을 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 장치.

청구항 22

복수의 시냅스들을 포함하는 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 프로그램 코드를 저장한 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 프로그램 코드는 프로세서에 의해 실행되고, 그리고

상기 복수의 시냅스들 중 시냅스의 지연 파라미터를 초기화하는 프로그램 코드;

상기 지연 파라미터를 설정하기 위한 함수를 이용하여 시뮬레이션 모드에서 상기 신경 네트워크를 동작시키는 프로그램 코드;

상기 시냅스의 상기 지연 파라미터를 포함하는 상기 함수의 프로그램 스테이트먼트에 적어도 부분적으로 기초한 프로그램에 적어도 부분적으로 기초하여 상기 시뮬레이션 모드에서의 상기 신경 네트워크의 동작 동안 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 프로그램 코드로서, 상기 지연 파라미터는 시냅스 유형, 뉴런 모델 유형, 또는 메모리 리소스들 중 적어도 하나에 기초하여 동적으로 업데이트되는, 상기 시냅스의 상기 지연 파라미터를 동적으로 업데이트하는 프로그램 코드; 및

상기 복수의 시냅스들의 서브세트에 대한 적어도 하나의 지연값을 동적으로 추출하는 프로그램 코드를 포함하는, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하기 위한 프로그램 코드를 저장한 비일시적 컴퓨터 판독가능 저장 매체.

청구항 23

삭제

청구항 24

삭제

청구항 25

삭제

청구항 26

삭제

발명의 설명

기술 분야

[0001] 본 개시물의 특정 양태들은 일반적으로 신경 시스템 엔지니어링에 관한 것이고, 보다 구체적으로는, 스냅스 지연을 동적으로 할당하고 조사하는 시스템들 및 방법들에 관한 것이다.

배경 기술

[0002] 상호연결된 인공 뉴런들의 그룹 (즉, 뉴런 모델들) 을 포함할 수도 있는 인공 신경 네트워크는 계산 디바이스이거나 계산 디바이스에 의해 수행될 방법을 표현한다. 인공 신경 네트워크들은 생물학적 신경 네트워크들에 대응하는 구조 및/또는 기능을 가질 수도 있다. 그러나, 인공 신경 네트워크들은 소정의 응용들에 대해 혁신적이고 유용한 계산 기법들을 제공할 수도 있는데, 종래의 계산 기법들은 복잡하거나, 비현실적이거나, 부적합하다.

[0003] 인공 신경 네트워크들은 시냅스 지연의 영향들을 받는다. 최근에, 시냅스 지연은 특정 네트워크 특징들의 부상에 이용되어 왔다. 이것의 적용 동안, 시냅스 지연은 풍부한 피쳐들이 구현될 수 있게 할 수도 있음을 알아내었다. 그러나, 네트워크 모델들에 시냅스 지연을 할당하는 통상의 접근 방식은 애드 혹 (ad hoc) 이 최상의 것이었다. 추가로, 일부 네트워크 모델들만이 초기화 동안에 시냅스 지연의 할당을 허용할 뿐, 시뮬레이션 동안에 시냅스 지연들을 동적으로 변경하는 어떠한 방법도 제공하지 못한다. 따라서, 네트워크 모델에서 시냅스들에 대한 지연들을 동적으로 할당하고 검사하는 표준적이고 간단한 방법이 없다.

발명의 내용

과제의 해결 수단

[0004] 본 개시물의 일 양태에서, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 방법이 개시된다. 본 방법은 지연 파라미터를 초기화하는 것 및 신경 네트워크를 동작시키는 것을 포함한다. 본 방법은 또한 지연 파라미터를 포함한 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 동적으로 업데이트하는 것을 포함한다.

[0005] 본 개시물의 다른 양태에서, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 장치가 개시된다. 본 장치는 메모리 및 프로세서를 포함한다. 프로세서는 지연 파라미터를 초기화하고, 신경 네트워크를 동작시키도록 구성된다. 프로세서는 또한, 지연 파라미터를 포함한 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 동적으로 업데이트하도록 구성된다.

[0006] 본 개시물의 또 다른 양태에서, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 장치가 개시된다. 본 장치는 지연 파라미터를 초기화하는 수단을 포함한다. 본 장치는 또한 신경 네트워크를 동작시키는 수단을 포함한다. 또한, 본 장치는 지연 파라미터를 포함한 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 동적으로 업데이트하는 수단을 포함한다.

[0007] 본 개시물의 또 다른 양태에서, 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 컴퓨터 프로그램 제품이 개시된다. 프로그램 제품은 프로그램 코드를 그 위에 인코딩한 비일시적 컴퓨터 판독가능 매체를 포함한다. 프로그램 코드는 지연 파라미터를 초기화하는 프로그램 코드를 포함한다. 프로그램 코드는 또한 신경 네트워크를 동작시키는 프로그램 코드를 포함한다. 프로그램 코드는 또한, 지연 파라미터를 포함하는 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 동적으로 업데이트하는 프로그램 코드를 더 포함한다.

도면의 간단한 설명

[0008] 본 개시물의 특색들, 속성, 및 이점들은, 도면들과 연계하여 보는 경우, 하기에 제시된 상세한 설명으로부터 자명해질 것이며, 도면들에서, 유사한 도면 부호들은 그에 대응하는 것을 식별한다.

도 1 은 본 개시물의 특정 양태들에 따른 일 예시적인 뉴런들의 네트워크를 예시한다.

도 2 는 본 개시물의 특정 양태들에 따른 계산 네트워크 (신경 시스템 또는 신경 네트워크) 의 프로세싱 유닛 (뉴런) 의 일 예를 예시한다.

도 3 은 본 개시물의 특정 양태들에 따른 스파이크-타이밍 종속 가소성 (STDP) 곡선의 일 예를 예시한다.

도 4 는 본 개시물의 특정 양태들에 따른 뉴런 모델의 거동을 정의하기 위한 양의 체제 및 음의 체제의 일 예를 예시한다.

도 5 는 본 개시물의 특정 양태들에 따른 범용 프로세서를 이용하여 신경 네트워크를 설계하는 일 예시적인 구현을 예시한다.

도 6 은 본 개시물의 특정 양태들에 따른, 메모리가 개별 분산된 프로세싱 유닛들과 인터페이싱될 수도 있는, 신경 네트워크를 설계하는 일 예시적인 구현을 예시한다.

도 7 은 본 개시물의 특정 양태들에 따른, 분산된 메모리들 및 분산된 프로세싱 유닛들에 기초하여 신경 네트워크를 설계하는 일 예시적인 구현을 예시한다.

도 8 은 본 개시물의 특정 양태들에 따른 신경 네트워크의 일 예시적인 구현을 예시한다.

도 9 는 신경 네트워크에서 시냅스 지연들을 동적으로 변경하는 방법을 예시하는 흐름도이다.

발명을 실시하기 위한 구체적인 내용

- [0009] 첨부된 도면들과 연계하여 하기에 설명되는 상세한 설명은, 여러 구성들의 설명으로서 의도된 것이며 본원에서 설명되는 개념들이 실시될 수도 있는 구성들만을 나타내도록 의도된 것은 아니다. 상세한 설명은 여러 개념들의 완전한 이해를 제공하기 위한 목적으로 특정 세부사항들을 포함한다. 그러나, 이들 개념들이 이들 특정 세부사항들 없이 실시될 수도 있음이 당업자에게는 명백할 것이다. 일부 사례들에서, 이러한 개념들을 모호하게 하는 것을 방지하기 위해 공지의 구조들 및 컴포넌트들이 블록도의 형태로 도시된다.
- [0010] 독립적으로 또는 본 개시물의 임의의 다른 양태들과 결합하여 구현되는지 여부에 따라, 본 사상들에 기초하여, 당업자들은 본 개시물의 범위가 본원에 개시된 개시물들의 임의의 양태를 커버하고자 함을 이해해야 할 것이다. 예를 들어, 제시된 임의의 개수의 양태들을 이용하여 장치가 구현될 수도 있거나 방법이 실시될 수도 있다. 또한, 본 개시물의 범위는 본원에 제시된 개시물의 다양한 양태들에 더해 또는 그 외에 다른 구조, 기능성, 또는 구조와 기능성을 이용하여 실시되는 그러한 장치 또는 방법을 커버하고자 한다. 본원에 개시된 개시물의 임의의 양태는 청구항의 하나 이상의 요소들에 의해 구체화될 수도 있다.
- [0011] 단어 "예시적인" 은 본원에서 "일 예, 사례, 또는 실례의 역할을 하는" 것을 의미하기 위해 사용된다. "예시적" 으로 본원에서 설명된 임의의 실시형태는 반드시 다른 실시형태들보다 바람직하거나 이로운 것으로 해석되지는 않는다.
- [0012] 특정 양태들이 본원에서 설명되지만, 이러한 양태들의 많은 변형예들 및 치환예들이 본 개시물의 범위 내에 속한다. 바람직한 양태들의 일부 이득들 및 이점들이 언급되었지만, 본 개시물의 범위는 특정 이득들, 이용들, 또는 목적들로 제한되고자 하지 않는다. 오히려, 본 개시물들의 양태들은 상이한 기술들, 시스템 구성들, 네트워크들, 및 프로토콜들에 널리 적용되고자 하며, 본 개시물의 양태들 중 일부는 도면들에서 그리고 다음의 바람직한 양태들의 설명에서 예로서 예시된다. 상세한 설명 및 도면들은 제한하는 것이기 보다는 단지 본 개시물의 예시일 뿐이며, 본 개시물의 범위는 첨부된 청구항들 및 그의 등가물들에 의해 정의된다.
- [0013] **예시적인 신경 시스템, 트레이닝, 및 동작**
- [0014] 도 1 은 본 개시물의 특정 양태들에 따른 다수의 레벨들의 뉴런 (neuron) 들을 갖는 일 예시적인 인공 신경 시스템 (100) 을 도시한다. 신경 시스템 (100) 은 시냅스 연결들 (104) (즉, 피드-포워드 연결들) 의 네트워크를 통해 다른 레벨의 뉴런들 (106) 에 접속되는 일 레벨의 뉴런들 (102) 을 가질 수도 있다. 편의상, 오직 2 개의 레벨들의 뉴런들만이 도 1 에 도시되나, 보다 적거나 보다 많은 레벨들의 뉴런들이 신경 시스템에 존재할 수도 있다. 뉴런들 중 일부 뉴런은 측면 연결들을 통해 동일한 계층의 다른 뉴런들에 연결될 수도 있음에 유의해야 한다. 또한, 뉴런들 중 일부는 피드백 연결들을 통해 이전 계층의 뉴런에 다시 연결될 수도 있다.
- [0015] 도 1 에 도시된 바와 같이, 레벨 (102) 에서의 각각의 뉴런은 이전 레벨의 뉴런들 (도 1 에 도시 생략) 에 의해 생성될 수도 있는 입력 신호 (108) 를 수신할 수도 있다. 신호 (108) 는 레벨 (102) 의 뉴런의 입력 전류를 나타낼 수도 있다. 이러한 전류는 뉴런 막에 축적되어 막 전위를 충전할 수도 있다. 막 전위가 임계 값에 도달하는 경우, 뉴런은 다음 레벨의 뉴런들 (예를 들어, 레벨 106) 로 전달되도록 발화되어 출력 스파이크를 생성할 수도 있다. 일부 모델링 접근법들에서, 뉴런은 다음 레벨의 뉴런들로 신호를 지속적으로 전달할 수도 있다. 이러한 신호는 통상적으로 막 전위의 함수이다. 그러한 거동은 하기에 설명된 것들과 같은 아날로그 및 디지털 구현들을 포함하여, 하드웨어 및/또는 소프트웨어로 에뮬레이션 또는 시뮬레이션될 수 있다.
- [0016] 생물학적 뉴런들에서, 뉴런이 발화하는 경우에 생성된 출력 스파이크는 활동 전위라고 지칭된다. 이러한 전기 신호는 상대적으로 빠르고, 과도하고, 신경 자극적이며, 대략 100 mV 의 진폭 및 약 1 ms 의 지속기간을 갖는다. 일련의 연결된 뉴런들을 갖는 신경 시스템의 특정 실시형태 (예를 들어, 도 1 에서 일 레벨의 뉴런들에서 다른 레벨의 뉴런들로의 스파이크들의 전달) 에서, 모든 활동 전위는 기본적으로 동일한 진폭 및 지속기간을 가지고, 따라서, 신호에서의 정보는 진폭에 의해서 보다는, 주파수 및 스파이크들의 수, 또는 스파이크들의 시간에 의해서만 나타내어질 수도 있다. 활동 전위에 의해 이송되는 정보는 스파이크, 스파이킹된 뉴런, 및 다른 스파이크나 스파이크들에 대한 스파이크의 시간에 의해 결정될 수도 있다. 스파이크의 중요성은, 하기에 설명된 바와 같이, 뉴런들 사이의 연결에 적용된 가중치에 의해 결정될 수도 있다.
- [0017] 일 레벨의 뉴런들로부터 다른 레벨의 뉴런들로의 스파이크들의 전달은, 도 1 에 도시된 바와 같이, 시냅스 연결

들의 네트워크 (또는 단순히 "시냅스들")(104) 를 통해 달성될 수도 있다. 시냅스들 (104) 에 대해, 레벨 102 의 뉴런들은 시냅스-전 뉴런들이라고 여겨질 수도 있고, 레벨 106 의 뉴런들은 시냅스-후 뉴런들로 여겨질 수도 있다. 시냅스들 (104) 은 레벨 102 뉴런들로부터 출력 신호들 (즉, 스파이크들) 을 수신하며, 조정가

능한 시냅스 가중치들 ($w_1^{(i,i+1)}, \dots, w_P^{(i,i+1)}$) 에 따라 그러한 신호들을 스케일링할 수도 있으며, 여기서 P 는 레벨 102 와 레벨 106 의 뉴런들 사이의 시냅스 연결들의 전체 개수이고, i 는 뉴런 레벨의 표시자이다. 도 1 의 예에서, i 는 뉴런 레벨 102 를 나타내고 $i+1$ 은 뉴런 레벨 106 을 나타낸다. 또한, 스케일링된 신호들은 레벨 106 에서의 각각의 뉴런의 입력 신호로서 결합될 수도 있다. 레벨 106 에서의 각각의 뉴런은 대응하는 결합된 입력 신호에 기초하여 출력 스파이크들 (110) 을 생성할 수도 있다. 출력 스파이크들 (110) 은 다른 시냅스 연결들의 네트워크 (도 1 에 도시 생략) 를 이용하여 다른 레벨의 뉴런들로 전달될 수도 있다.

[0018] 생물학적 시냅스들은 시냅스-후 뉴런들에서 흥분성 또는 억제 (과분극) 활동들을 중재할 수 있고 또한 신경 신호들을 증폭시키는 역할을 할 수 있다. 흥분성 신호들은 막 전위를 탈분극한다 (즉, 정지 전위에 대해 막 전위를 증가시킨다). 임계치 위로 막 전위를 탈분극하도록 소정의 시간 기간 내에 충분한 흥분성 신호들이 수신되면, 활동 전위가 시냅스-후 뉴런에서 발생한다. 반면에, 억제 신호들은 일반적으로 막 전위를 과분극한다 (즉, 낮춘다). 억제 신호들은, 충분히 강하다면, 흥분성 신호들의 합에 반대로 작용하여 막 전위가 임계치에 도달하는 것을 방지할 수 있다. 시냅스 흥분에 반대로 작용하는 것에 더해, 시냅스 억제는 자발적 활성 뉴런들에 대해 강력한 제어를 발휘할 수 있다. 자발적 활성 뉴런은, 예를 들어, 그것의 역학 또는 피드백으로 인해, 추가적인 입력없이 스파이크하는 뉴런을 지칭한다. 이러한 뉴런들에서 활동 전위들의 자발적 생성을 억압함으로써, 시냅스 억제는 뉴런에서 발화하는 패턴을 형성할 수 있으며, 이는 일반적으로 조각 (sculpturing) 이라고 지칭된다. 다양한 시냅스들 (104) 은, 원하는 거동에 따라, 흥분성 시냅스 또는 억제 시냅스의 임의의 조합으로 작용할 수도 있다.

[0019] 신경 시스템 (100) 은 범용 프로세서, 디지털 신호 프로세서 (digital signal processor; DSP), 주문형 반도체 (application specific integrated circuit; ASIC), 필드 프로그램가능 게이트 어레이 (field programmable gate array; FPGA) 혹은 다른 프로그램가능한 로직 디바이스 (programmable logic device; PLD), 이산 게이트 혹은 트랜지스터 로직, 이산 하드웨어 컴포넌트들, 프로세서에 의해 실행되는 소프트웨어 모듈, 또는 그것들의 임의의 조합에 의해 에뮬레이션될 수도 있다. 신경 시스템 (100) 은 전기 회로에 의해 에뮬레이션되고, 이미지 및 패턴 인식, 머신 러닝, 모터 제어 등과 같은 광범위한 애플리케이션들에 활용될 수도 있다. 신경 시스템 (100) 에서 각각의 뉴런은 뉴런 회로로서 구현될 수도 있다. 출력 스파이크를 개시하는 임계 값으로 충전되는 뉴런 막은, 예를 들어, 뉴런 막을 통해 흐르는 전류를 통합하는 커패시터로서 구현될 수도 있다.

[0020] 일 양태에서, 커패시터는 뉴런 회로의 전류 통합 디바이스로서 제거될 수도 있고, 보다 작은 멤리스터 (memristor) 소자가 커패시터 대신에 이용될 수도 있다. 이러한 접근법은 뉴런 회로들, 뿐만 아니라 전류 통합기들로서 대형 커패시터들이 활용되는 다양한 다른 애플리케이션들에 적용될 수도 있다. 또한, 시냅스들 (104) 의 각각은 멤리스터 소자에 기초하여 구현될 수도 있으며, 여기서 시냅스 가중치 변화들은 멤리스터 저항의 변화들과 관련될 수도 있다. 나노미터 피쳐 크기의 멤리스터들로, 뉴런 회로 및 시냅스들의 영역이 실질적으로 감소될 수도 있으며, 이는 매우 큰 크기의 신경 시스템 하드웨어 구현예의 구현을 보다 실현가능하게 할 수도 있다.

[0021] 신경 시스템 (100) 을 에뮬레이션하는 신경 프로세서의 기능은 시냅스 연결들의 가중치들에 의존할 수도 있으며, 이는 뉴런들 사이의 연결들의 강도들을 제어할 수도 있다. 시냅스 가중치들은 전력 다운된 후에 프로세서의 기능을 보호하기 위해 비휘발성 메모리에 저장될 수도 있다. 일 양태에서, 시냅스 가중치 메모리는 메인 신경 프로세서 칩과는 별도로 외부 칩에 구현될 수도 있다. 시냅스 가중치 메모리는 대체가능한 메모리 카드로서 신경 프로세서 칩과는 별도로 패키징될 수도 있다. 이는 신경 프로세서에 다양한 기능들을 제공할 수도 있으며, 여기서 특정 기능은 신경 프로세서에 현재 접속된 메모리 카드에 저장된 시냅스 가중치들에 기초할 수도 있다.

[0022] 도 2 는 본 개시물의 특정 양태들에 따른 계산 네트워크 (예를 들어, 신경 시스템, 또는 신경 네트워크) 의 프로세싱 유닛 (예를 들어, 뉴런 또는 뉴런 회로) (202) 의 일 예시적인 도면 (200) 을 도시한다. 예를 들어, 뉴런 (202) 은 도 1 로부터의 레벨 102 및 레벨 106 의 뉴런들 중 임의의 뉴런에 대응할 수도 있다. 뉴런 (102) 은 다수의 입력 신호들 (204_1-204_N) 을 수신할 수도 있으며, 다수의 입력 신호들은 신경 시스템의 외부의 신호들, 또는 동일한 신경 시스템의 다른 뉴런들에 의해 생성된 신호들, 또는 양자 모두일 수도 있다. 입력

신호는 전류, 컨덕턴스, 전압, 실수값 및/또는 복소수 값일 수도 있다. 입력 신호는 고정-소수점 또는 부동-소수점 표현을 갖는 수치 값을 포함할 수도 있다. 이러한 입력 신호들은 조정가능한 시냅스 가중치들 ($206_1-206_N(W_1-W_N)$)에 따라 신호들을 스케일링하는 시냅스 연결들을 통해 뉴런 (202)에 전달될 수도 있으며, 여기서 N은 뉴런 (202)의 입력 연결들의 전체 개수일 수도 있다.

[0023] 뉴런 (202)은 스케일링된 입력 신호들을 결합하고 결합되어진 스케일링된 입력들을 이용해 출력 신호 (208) (즉, 신호 Y)를 생성할 수도 있다. 출력 신호 (208)는 전류, 컨덕턴스, 전압, 실수값 및/또는 복소수 값일 수도 있다. 출력 신호는 고정-소수점 또는 부동-소수점 표현을 갖는 수치 값일 수도 있다. 출력 신호 (208)는 그 다음에 동일한 신경 시스템의 다른 뉴런들에 입력 신호로서, 또는 동일한 뉴런 (202)에 입력 신호로서, 또는 신경 시스템의 출력으로서 전달될 수도 있다.

[0024] 프로세싱 유닛 (뉴런) (202)은 전기 회로에 의해 에뮬레이션될 수도 있고, 프로세싱 유닛의 입력 및 출력 연결들은 시냅스 회로들을 갖는 전기 연결부들에 의해 에뮬레이션될 수도 있다. 프로세싱 유닛 (202) 및 프로세싱 유닛의 입력 및 출력 연결들은 또한 소프트웨어 코드에 의해 에뮬레이션될 수도 있다. 프로세싱 유닛 (202)이 또한 전기 회로에 의해 에뮬레이션될 수도 있는 반면, 프로세싱 유닛의 입력 및 출력 연결들은 소프트웨어 코드에 의해 에뮬레이션될 수도 있다. 일 양태에서, 계산 네트워크에서 프로세싱 유닛 (202)은 아날로그 전기 회로일 수도 있다. 다른 양태에서, 프로세싱 유닛 (202)은 디지털 전기 회로일 수도 있다. 또 다른 양태에서, 프로세싱 유닛 (202)은 아날로그 및 디지털 컴포넌트들 양자 모두를 갖는 혼합-신호 전기 회로를 포함할 수도 있다. 계산 네트워크는 앞서 언급된 형태들 중 임의의 형태로 프로세싱 유닛들을 포함할 수도 있다. 그러한 프로세싱 유닛들을 이용하는 계산 네트워크 (신경 시스템 또는 신경 네트워크)은 광범위한 애플리케이션들, 예컨대, 이미지 및 패턴 인식, 머신 러닝, 모터 제어 등에 활용될 수도 있다.

[0025] 신경 네트워크를 트레이닝하는 과정 중에, 시냅스 가중치들 (예를 들어, 도 1로부터의 가중치들 ($w_1^{(i,i+1)}, \dots, w_P^{(i,i+1)}$) 및/또는 도 2로부터의 가중치들 (206_1-206_N))은 랜덤 값들로 초기화되고 학습 규칙에 따라 증가되거나 감소될 수도 있다. 학습 규칙의 예들은, 이로 제한되지는 않으나, 스파이크-타이밍-종속-소성 (spike-timing-dependent plasticity; STDP) 학습 규칙, Hebb 규칙, Oja 규칙, BCM (Bienenstock-Copper-Munro) 규칙 등을 포함한다는 것을 당업자들은 이해할 것이다. 특정 양태들에서, 가중치들은 2개의 값들 중 하나로 정해지거나 수렴할 수도 있다 (즉, 가중치들의 양분 분배). 이러한 결과는 각각의 시냅스 가중치에 대한 비트들의 수를 감소시키고, 시냅스 가중치들을 저장하는 메모리로부터의/메모리로의 판독 및 기록의 속도를 증가시키고, 시냅스 메모리의 전력 및/또는 프로세서 소비를 감소시키는데 활용될 수도 있다.

[0026] 시냅스 유형

[0027] 신경 네트워크들의 하드웨어 및 소프트웨어 모델들에서, 기능들과 관련된 시냅스의 프로세싱은 시냅스 유형에 기초할 수 있다. 시냅스 유형들은 비소성 시냅스들 (가중치 및 지연의 변화 없음), 가소성 시냅스들 (가중치가 변할 수도 있다), 구조적 지연 가소성 시냅스들 (가중치 및 지연이 변할 수도 있다), 완전 가소성 시냅스들 (가중치, 지연, 및 연결성이 변할 수도 있다), 및 그에 한 변형들 (예를 들어, 지연은 변할 수도 있으나, 가중치 또는 입력에서는 변화가 없을 수도 있다) 일 수도 있다. 이의 이점은 프로세싱이 세분될 수 있다는 것이다. 예를 들어, 비소성 시냅스들은 가소성 기능들이 실행되는 것 (또는 그러한 것이 완료되기를 기다리는 것)을 요구하지 않을 수도 있다. 유사하게, 지연 및 가중치 가소성은, 차례 차례로 또는 병렬로, 함께 또는 별도로 동작할 수도 있는 동작들로 세분될 수도 있다. 상이한 유형의 시냅스들은 적용되는 상이한 가소성 유형들의 각각에 대해 상이한 룩업 테이블들 또는 공식들 및 파라미터들을 가질 수도 있다. 따라서, 방법들은 시냅스의 유형에 대한 관련 테이블들, 공식들, 또는 파라미터들에 액세스할 것이다.

[0028] 스파이크-타이밍 종속 구조 가소성이 시냅스 가소성과 독립적으로 실행될 수도 있다는 추가적인 의미들이 있다. 구조 가소성 (즉, 지연 변화의 양)이 전-후 스파이크 차이의 직접적인 함수일 수도 있기 때문에, 구조적 가소성은 가중치 크기에 변화가 없는 경우 (예를 들어, 가중치가 최소 또는 최대 값에 도달한 경우, 또는 일부 다른 이유로 인해 변하지 않은 경우) 일지라도 구조 가소성이 실행될 수도 있다. 대안으로, 구조적 가소성은 가중치 변화 양의 함수로 또는 가중치들 혹은 가중치 변화들의 한계들과 관련되는 조건들에 기초하여 설정될 수도 있다. 예를 들어, 시냅스 지연은 가중치 변화가 발생하는 경우에만, 또는 가중치가 제로에 도달하나 최고 값에 있지 않은 경우에만 변할 수도 있다. 그러나, 이러한 프로세스들이 병렬로 되어 메모리 액세스들의 수 및 중첩을 감소시킬 수 있도록 독립적인 기능들을 가지는 것이 이로울 수 있다.

[0029] 시냅스 가소성의 결정

[0030] 신경 가소성 (또는 간단하게 "가소성") 은 새로운 정보, 감각 자극, 개발, 손상, 또는 장애에 응답하여 시냅스 연결들 및 거동을 변화시키는 뇌에서의 뉴런들 및 신경 네트워크들의 능력이다. 가소성은 생물학 뿐만 아니라 컴퓨터 신경 과학 및 신경 네트워크들에서의 학습 및 메모리에 있어 중요하다. (예를 들어, Hebbian 이론에 따른) 시냅스 가소성, 스파이크-타이밍-종속 가소성 (STDP), 비-시냅스 가소성, 활동-종속 가소성, 구조 가소성, 및 항상성 가소성과 같은 다양한 형태들의 가소성이 연구되었다.

[0031] STDP 는 뉴런들 사이의 시냅스 연결들의 강도를 조정하는 학습 프로세스이다. 연결 강도들은 특정 뉴런의 출력 및 수신된 입력 스파이크들의 상대적 타이밍 (즉, 활동 전위) 에 기초하여 조정된다. STDP 프로세스 하에서, 장기 강화 (long-term potentiation; LTP) 는 소정의 뉴런에 대한 입력 스파이크가, 평균적으로, 그 뉴런의 출력 스파이크 바로 전에 발생하려고 하면 발생할 수도 있다. 그 다음에, 그 특정 입력은 다소 더 강하게 된다. 반면에, 입력 스파이크가, 평균적으로, 출력 스파이크 바로 후에 발생하려고 하면, 장기 저하 (long-term depression; LTD) 가 발생할 수도 있다. 그 다음에, 그 특정 입력은 다소 약하게 되고, 따라서, 명칭이 "스파이크-타이밍-종속 가소성" 이다. 결과적으로, 시냅스-후 뉴런의 흥분을 야기할 수도 있는 입력들은 장래에 기여할 가능성이 더 크게 되고, 한편 시냅스-후 스파이크를 야기하지 않는 입력들은 장래에 기여할 가능성이 더 작아지게 된다. 프로세스는 연결들의 초기 세트의 서브세트가 남아있을 때까지 계속 되고, 한편 모든 다른 것들의 영향은 사소한 레벨로 감소된다.

[0032] 뉴런은 일반적으로 그것의 입력들 중 많은 입력이 짧은 기간 내에 발생하는 경우에 출력 스파이크를 생성하기 때문에 (즉, 출력을 야기하기에 충분하게 누적된다), 통상적으로 남아있는 입력들의 서브세트는 시간에 상관되는 경향이 있는 것들을 포함한다. 또한, 출력 스파이크 전에 발생하는 입력들이 강화되기 때문에, 가장 빠른 충분한 상관의 누적 표시를 제공하는 입력들이 결국 뉴런에 대한 최종 입력이 될 수도 있다.

[0033] STDP 학습 규칙은 시냅스-전 뉴런의 스파이크 시간 (t_{pre}) 과 시냅스-후 뉴런의 스파이크 시간 (t_{post}) 사이의 시간 차이의 함수 (즉, $t = t_{post} - t_{pre}$) 로서 시냅스-전 뉴런을 시냅스-후 뉴런에 연결하는 시냅스의 시냅스 가중치에 효과적으로 적용될 수도 있다. 통상적인 STDP 의 공식은 시간 차이가 양 (positive) 이면 (시냅스-전 뉴런이 시냅스-후 뉴런 전에 발화한다) 시냅스 가중치를 증가시키고 (즉, 시냅스를 강력하게 하고), 시간 차이가 음 (negative) 이면 (시냅스-후 뉴런이 시냅스-전 뉴런 전에 발화한다) 시냅스 가중치를 감소시키는 (즉, 시냅스를 억제하는) 것이다.

[0034] STDP 프로세스에서, 시간 경과에 따른 시냅스 가중치의 변화는 통상적으로 다음에서 주어진 지수함수형 감쇠 (exponential decay) 를 이용하여 달성된다:

[0035]
$$\Delta w(t) = \begin{cases} a_+ e^{-t/k_+} + \mu, & t > 0 \\ a_- e^{t/k_-}, & t < 0 \end{cases}, \quad (1)$$

[0036] 여기서 k_+ 및 k_- $\tau_{\text{sign}(\Delta t)}$ 은 각각 양 및 음의 시간 차이에 대한 시간 상수들이고, a_+ 및 a_- 은 대응하는 스케일링 크기들이고, μ 는 양의 시간 차이 및/또는 음의 시간 차이에 적용될 수도 있는 오프셋이다.

[0037] 도 3 은 STDP 에 따른 시냅스-전 스파이크와 시냅스-후 스파이크의 상대적 타이밍의 함수로서 시냅스 가중치 변화의 일 예시적인 도면 (300) 을 도시한다. 시냅스-전 뉴런이 시냅스-후 뉴런 전에 발화하면, 그래프 (300) 의 302 부분에서 도시된 바와 같이, 대응하는 시냅스 가중치가 증가될 수도 있다. 이러한 가중치 증가는 시냅스의 LTP 라고 지칭될 수 있다. LTP 의 양이 시냅스-전 스파이크 시간과 시냅스-후 스파이크 시간 사이의 차이의 함수로서 거의 기하급수적으로 감소할 수도 있다는 것이 그래프 부분 302 로부터 관찰될 수 있다. 그래프 (300) 의 부분 304 에 도시된 바와 같이, 역순 (reverse order) 의 발화는 시냅스 가중치를 감소시켜, 시냅스의 LTD 를 야기할 수도 있다.

[0038] 도 3 에서의 그래프 (300) 에 도시된 바와 같이, 음의 오프셋 (μ) 이 STDP 그래프의 LTP (원인) 부분 302 에 적용될 수도 있다. x-축의 교차 지점 (306) ($y=0$) 은 계층 i-1 로부터의 원인 입력들에 대한 상관관계를 고려하여 최대 시간 지연과 일치하게 구성될 수도 있다. 프레임-기반 입력 (즉, 스파이크들 또는 펄스들을 포함하는 특정 지속기간의 프레임의 형태인 입력) 의 경우에, 오프셋 값 (μ) 은 프레임 경계를 반영하도록 계산될 수 있다. 프레임에서의 제 1 입력 스파이크 (펄스) 는 직접적으로 시냅스-후 전위에 의해 모델링됨으로

써 또는 신경 상태에 대한 영향의 관점에서 시간이 경과함에 따라 감소하는 것으로 고려될 수도 있다. 프레임에서의 제 2 입력 스파이크 (펄스) 가 특정 시간 프레임과 상관되거나 관련있다고 고려되면, 관련 시간들에서의 값이 상이할 수도 있도록 (일 프레임보다 큰 것에 대해서는 음, 그리고 일 프레임보다 작은 것에 대해서는 양) 프레임 전후의 관련 시간들은 해당 시간 프레임 경계에서 분리되고 STDP 곡선의 하나 이상의 부분들을 오프셋함으로써 가소성의 면에서 상이하게 취급될 수도 있다. 예를 들어, 음의 오프셋 (μ) 은 프레임보다 큰 전-후 시간에서 곡선이 실제로 제로 아래로 가고 따라서 LTP 대신에 LTD 의 부분이도록 LTP 를 오프셋하도록 설정될 수도 있다.

[0039] 뉴런 모델들 및 동작

[0040] 유용한 스파이킹 뉴런 모델을 설계하기 위한 몇몇 일반적인 원리들이 있다. 훌륭한 뉴런 모델은 2 개의 계산 제도들: 일치 검출 및 함수적 계산의 측면에서 풍부한 잠재적 거동을 가질 수도 있다. 또한, 훌륭한 뉴런 모델은 시간 코딩을 가능하게 하도록 2 개의 요소들을 가져야 한다: 입력들의 도착 시간은 출력 시간에 영향을 주고 일치 검출은 좁은 시간 윈도우를 가질 수 있다. 마지막으로, 계산상으로 매력있도록, 훌륭한 뉴런 모델은 연속적인 시간에서의 폐쇄형 솔루션 및 근처의 어트랙터들 및 안장점들을 포함하는 안정적인 거동을 가질 수도 있다. 다시 말해서, 유용한 뉴런 모델은 실용적이고, 풍부하고, 사실적이고, 생물학적으로-일정한 거동들을 모델링하는데 이용되는 것뿐만 아니라 엔지니어 및 역 엔지니어 신경 회로들에서 이용될 수 있는 것이다.

[0041] 뉴런 모델은 입력 도착, 출력 스파이크와 같은 이벤트들, 또는 내부적이거나 외부적인 다른 이벤트에 의존할 수도 있다. 풍부한 거동 레퍼토리를 달성하기 위해서는, 복잡한 거동들을 보일 수 있는 상태 머신이 바람직할 수도 있다. (만약 있다면) 입력 기여와 별도로, 이벤트 자체의 발생이 상태 머신에 영향을 주고 이벤트에 후속하는 역학을 제약할 수 있다면, 시스템의 장래 상태는 상태 및 입력의 함수일 뿐만 아니라, 상태, 이벤트, 및 입력의 함수이다.

[0042] 일 양태에서, 뉴런 (n) 은 다음의 역학에 의해 통제되는 막 전압 ($v_n(t)$) 을 갖는 스파이킹 누출-통합-및-발화 뉴런으로 모델링될 수도 있다:

$$\frac{dv_n(t)}{dt} = \alpha v_n(t) + \beta \sum_m w_{m,n} y_m(t - \Delta t_{m,n}) \quad (2)$$

[0044] 여기서 α 및 β 는 파라미터들이고, $w_{m,n}$ 은 시냅스-전 뉴런 (m) 을 시냅스-후 뉴런 (n) 에 연결하는 시냅스에 대한 시냅스 가중치이고, $y_m(t)$ 은 뉴런 (n) 의 세포체 (soma) 에 도착할 때까지 $\Delta t_{m,n}$ 에 따라 수지상 (dendritic) 또는 축삭 (axonal) 지연될 수도 있는 뉴런 (m) 의 스파이킹 출력이다.

[0045] 시냅스-후 뉴런에 대한 충분한 입력이 확립된 때로부터 시냅스-후 뉴런이 실제로 발화할 때까지 지연이 있다는 것에 유의해야 한다. Izhikevich 의 단순 모델과 같은 동적 스파이크 뉴런 모델에서, 탈분극화 임계치 (v_r) 와 피크 스파이크 전압 (v_{peak}) 사이에 차이가 있으면 시간 지연이 초래될 수도 있다. 예를 들어, 단순 모델에서, 전압 및 복구에 대한 미분 방정식들의 쌍에 의해 뉴런 세포체 역학들이 통제될 수 있다, 즉:

$$\frac{dv}{dt} = (k(v - v_r)(v - v_r) - u + I) / C \quad (3)$$

$$\frac{du}{dt} = a(b(v - v_r) - u) \quad (4)$$

[0048] 여기서, v 는 막 전위이고, u 는 막 복구 변수이고, k 는 막 전위 (v) 의 시간 스케일을 설명하는 파라미터이고, a 는 복구 변수 u 의 시간 스케일을 설명하는 파라미터이고, b 는 막 전위 (v) 의 하위-임계 변동들에 대한 복구 변수 u 의 민감도를 설명하는 파라미터이고, v_r 은 막 휴지상태 전위이고, I 는 시냅스 전류이고, C 는 막의 커패시턴스이다. 이러한 모델에 따르면, 뉴런은 $v > v_{peak}$ 인 경우에 스파이킹하는 것으로

로 정의된다.

[0049] Hunzinger 콜드 (Cold) 모델

[0050] Hunzinger 콜드 뉴런 모델은 풍부하며 다양한 신경 거동들을 복제할 수 있는 최소 이중-체제 스파이킹 선형 동적 모델이다. 모델의 1- 또는 2-차원 선형 역학은 2 개의 체제들을 가질 수 있으며, 여기서 시간 상수 (및 연결) 는 체제에 의존할 수 있다. 하위-임계 체제에서, 규칙에 의해 음인 시간 상수는 일반적으로 생물학적으로-일관성있는 선형 방식으로 휴지상태로 셀을 반환하도록 작동하는 누수 채널 역학을 나타낸다. 규칙에 의해 양인 상위-임계 체제에서 시간 상수는 일반적으로 셀이 스파이킹하도록 구동하나 스파이크-생성에서 지연을 초래하는 누수 방지 채널 역학을 반영한다.

[0051] 도 4 에 도시된 바와 같이, 모델 (400) 의 역학은 2 개 (또는 그 보다 많은) 체제들로 나누어질 수도 있다. 이러한 체제들은 (LIF 뉴런 모델과 혼동되지 않게, 누수-통합-및-발화 (leaky-integrate-and-fire; LIF) 체제라고도 상호교환가능하게 지칭되는) 음의 체제 (402) 및 (ALIF 뉴런 모델과 혼동되지 않게, 누수-방지-통합-및-발화 (anti-leaky-integrate-and-fire; ALIF) 체제라고도 상호교환가능하게 지칭되는) 양의 체제 (404) 라고 불릴 수도 있다. 음의 체제 (402) 에서, 상태는 장래 이벤트 시에 휴지상태 (v_-) 쪽으로 향하는 경향이 있다. 이러한 음의 체제에서, 모델은 일반적으로 시간 입력 검출 속성들 및 다른 하위-임계 거동을 보인다. 양의 체제 (404) 에서, 상태는 스파이킹 이벤트 (v_s) 쪽으로 향하는 경향이 있다. 이러한 양의 체제에서, 모델은 후속하는 입력 이벤트들에 따라 스파이킹하는데 지연을 초래하는 것과 같은 계산 속성들을 보인다. 이러한 2 개의 체제들로의 역학의 이벤트들 및 분리의 면에서의 역학의 공식은 모델의 기본적인 특성들이다.

[0052] (상태들 (v 및 u) 에 대한) 선형 이중-체제 양방향-차원 역학은 다음과 같은 규칙에 의해 정의될 수도 있다:

$$\tau_{\rho} \frac{dv}{dt} = v + q_{\rho} \quad (5)$$

$$-\tau_u \frac{du}{dt} = u + r \quad (6)$$

[0055] 여기서 q_{ρ} 및 r 은 연결에 대한 선형 변환 변수들이다.

[0056] 심볼 ρ 는, 특정 체제에 대한 관계를 논의하거나 표현하는 경우, 각각 음의 체제 및 양의 체제에 대해 심볼 ρ 를 부호 "-" 또는 "+" 로 대체하도록 규칙에 따라 역학 체제를 지칭하기 위해 본원에서 이용된다.

[0057] 모델 상태는 막 전위 (전압) v 및 복구 전류 (recovery current) u 에 의해 정의된다. 기본 형태에서, 체제는 기본적으로 모델 상태에 의해 결정된다. 미묘하지만 중요한 정확도 및 일반 정의의 양태들이 있으나, 지금은, 전압 (v) 이 임계치 (v_+) 보다 높은 경우 양의 체제 (404) 에 있고 그렇지 않으면 음의 체제 (402) 에 있는 모델을 고려한다.

[0058] 체제-의존적인 시간 상수는 음의 체제 시간 상수인 τ_- 및 양의 체제 시간 상수인 τ_+ 를 포함한다. 복구 전류 시간 상수 (τ_u) 는 통상적으로 체제와 독립적이다. 편의를 위해, 음의 체제 시간 상수 (τ_-) 는 통상적으로 감쇠를 반영하도록 음의 양 (negative quantity) 으로 명시되어 전압 진전에 대한 동일한 표현이 양의 체제에 대해 이용될 수도 있으며, 여기서 지수 및 τ_+ 는 일반적으로 양이며 τ_u 도 그럴 것이다.

[0059] 2 개의 상태 엘리먼트들의 역학은 무연속변이 (null-cline) 들로부터 상태들을 오프셋하는 변환들에 의한 이벤트들에서 연결될 수도 있으며, 여기서 변환 변수들은 다음과 같다:

$$q_{\rho} = -\tau_{\rho} \beta u - v_{\rho} \quad (7)$$

$$r = \delta(v + \varepsilon) \quad (8)$$

[0062] 여기서 δ , ε , β 및 v_- , v_+ 은 파라미터들이다. v_ρ 에 대한 2 개의 값들은 2 개의 체제들에 대한 기준 전압들에 대한 베이스이다. 파라미터 v_- 는 베이스 전압이고, 막 전위는 일반적으로 음의 체제에서 v_- 쪽으로 감쇠한다. 파라미터 v_+ 는 양의 체제에 대한 베이스 전압이고, 막 전위는 양의 체제에서 일반적으로 v_+ 로부터 멀어지는 경향이 있다.

[0063] v 및 u 에 대한 무연속변이들은 각각 변환 변수들 q_ρ 및 r 의 음으로 주어진다. 파라미터 δ 은 u 무연속변이의 경사도를 제어하는 스케일 인자이다. 파라미터 ε 은 통상적으로 $-v_-$ 와 동일하게 설정된다. 파라미터 β 는 양 체제들에서 v 무연속변이들의 경사도를 제어하는 저항 값이다. τ_ρ 시간-상수 파라미터들은 각각의 체제에서 별도로 기하급수적 감쇠를 뿐만 아니라 무연속변이 경사도들도 제어한다.

[0064] 모델은 전압 (v) 이 값 (v_s) 에 도달하는 경우에 스파이킹하도록 정의될 수도 있다. 후속하여, 상태는 (스파이크 이벤트와 동일한 것일 수도 있는) 리셋 이벤트에서 리셋될 수도 있다:

[0065]
$$v = \hat{v}_- \quad (9)$$

[0066]
$$u = u + \Delta u \quad (10)$$

[0067] 여기서 $\hat{v}_-$ 및 Δu 는 파라미터들이다. 리셋 전압 ($\hat{v}_-$) 은 통상적으로 v_- 로 설정된다.

[0068] 순간적인 연결의 원리에 의해, (단일 지수 항을 갖는) 상태 뿐만 아니라 특정 상태에 도달하기 위해 시간에 대해 폐쇄 형태 해가 가능하다. 폐쇄 형태 상태 해들은 다음과 같다:

[0069]
$$v(t + \Delta t) = (v(t) + q_\rho) e^{\frac{\Delta t}{\tau_\rho}} - q_\rho \quad (11)$$

[0070]
$$u(t + \Delta t) = (u(t) + r) e^{\frac{\Delta t}{\tau_u}} - r \quad (12)$$

[0071] 따라서, 모델 상태는 입력 (시냅스-전 스파이크) 또는 출력 (시냅스-후 스파이크) 과 같은 이벤트들 시에만 업데이트될 수도 있다. 동작들은 또한 (입력 또는 출력이 있는지 여부에 상관없이) 임의의 특정 시간에 수행될 수도 있다.

[0072] 또한, 순간적인 연결 원리에 의해, 반복적 기법들 또는 수치 방법들 (예를 들어, Euler 수치 방법) 없이도 특정 상태에 도달하기 위한 시간이 미리 결정될 수도 있도록 시냅스-후 스파이크의 시간이 예상될 수도 있다. 이 전 전압 상태 (v_0) 를 고려하면, 전압 상태 (v_f) 에 도달되기까지의 시간 지연은 다음과 같이 주어진다:

[0073]
$$\Delta t = \tau_\rho \log \frac{v_f + q_\rho}{v_0 + q_\rho} \quad (13)$$

[0074] 전압 상태 (v) 가 v_s 에 도달하는 시점에 스파이크가 발생하는 것으로 정의되면, 전압이 주어진 상태 (v) 에 있는 시간에서부터 측정된 바와 같은 스파이크가 발생하기 전까지의 시간의 양 또는 상대적 지연에 대한 폐쇄형 해는 다음과 같다:

[0075]
$$\Delta t_s = \begin{cases} v > \hat{v}_+ \text{ 이면 } \tau_+ \log \frac{v_s + q_+}{v + q_+} \\ \text{그렇지 않으면 } \infty \end{cases} \quad (14)$$

[0076] 여기서 $\hat{v}_+$ 은 통상적으로 파라미터 v_+ 로 설정되나, 다른 변형들이 가능할 수도 있다.

- [0077] 모델 역학의 위의 정의들은 모델이 양의 체제 또는 음의 체제에 있는지 여부에 의존한다. 언급된 바와 같이, 연결 및 체제 (ρ) 는 이벤트들 시에 계산될 수도 있다. 상태 전파의 목적으로, 체제 및 연결 (변형) 은 마지막 (이전) 이벤트의 시점에서의 상태에 기초하여 정의될 수도 있다. 스파이크 출력 시간을 후속하여 예상하기 위한 목적으로, 체제 및 연결 변수는 다음 (현재) 이벤트 시점에서의 상태에 기초하여 정의될 수도 있다.
- [0078] 콜드 모델, 및 시뮬레이션, 에뮬레이션, 시간 모델을 실행하는 여러 가지의 가능한 구현들이 있다. 이는, 예를 들어, 이벤트-업데이트, 단계-이벤트 업데이트, 및 단계-이벤트 모드들을 포함한다. 이벤트 업데이트는 (특정 순간들에서) 이벤트들 또는 "이벤트 업데이트" 에 기초하여 상태들이 업데이트되는 업데이트이다. 단계 업데이트는 모델이 간격들 (예를 들어, 1 ms) 에서 업데이트되는 경우의 업데이트이다. 이는 반드시 반복적인 방법들 또는 수치 방법들을 요구하지는 않는다. 이벤트-기반 구현이 또한 오직 단계들에서 또는 단계들 사이에서 이벤트가 발생하는 경우에만 모델을 업데이트함으로써 또는 "단계-이벤트" 업데이트에 의해 단계-기반 시뮬레이터에서 제한된 시간 분해능에서 가능하다.
- [0079] *스냅스 지연의 동적 할당 및 검사*
- [0080] 본 개시물의 양태들은 신경 네트워크 모델링 언어와 연계하여 이용되는 함수들을 제공함으로써 스파이킹 신경 네트워크 모델들에서의 시냅스 지연들의 할당 및 검사에 대해 교시된다.
- [0081] HLND (High Level Network Description) 언어에 의해 모델링되는 네트워크에서의 각각의 시냅스는 PSP들 (post-synaptic potentials) 의 전달을 연기하는 연관 지연을 갖는다. 쉬운 설명을 위하여, HLND 이 이용되지만, 다른 프로그램 언어들이 또한, 신경 네트워크에서 시냅스들을 모델링하고 시냅스 지연을 동적으로 할당하는데 이용될 수도 있다.
- [0082] 시냅스 지연들은 탐색자에 의해 일정 값으로 초기화될 수도 있다. 네트워크의 시뮬레이션 동안에, 함수, 이를 테면, SetDelay() 함수가 시냅스 지연을 변경하는데 이용될 수도 있다. 함수
- [0083] SetDelay(int delay)
- [0084] 는 특정 시냅스에 할당되는 지연에 대응하는 정수값을 허용한다. 일부 양태들에서, 지연은 부동 소수점수일 수도 있다. 함수
- [0085] SetDelay(float delay)
- [0086] 는 부동 소수점수를 수락하며, 분수 지연들의 사양들을 허용한다. 부동 소수점수를 이용하여, 지연은 또한 초 단위로 설정될 수도 있다.
- [0087] *장래 시간에 대한 시냅스 지연의 할당*
- [0088] 일부 양태들에서, 지연 값은 장래의 시간에 대하여 연산될 수도 있다. 지연이 변경해야 하는 시간을 특정함으로써, 탐색자는 지연 변경을 스케줄링할 수도 있다. 예를 들어, 스냅스 지연은 하기 형태의 함수를 이용하여 할당될 수도 있다.
- [0089] SetDelay(int time, float delay)
- [0090] 이 함수는 장래에 특정되는 시간에 시냅스의 지연을 할당하는데 이용될 수도 있다. 지연 값은 부동 소수점수일 수도 있다. 물론, 이는 단지 예시에 불과하고, 지연의 정수값이 대안으로서 특정될 수도 있다.
- [0091] *스냅스 지연의 증분 및 감분*
- [0092] 일부 양태들에서, 지연은 또한 하기 형태의 함수들을 이용하여 증분 또는 감분될 수도 있다.
- [0093] IncrementDelay(float delta)
- [0094] DecrementDelay(float delta)
- [0095] 이는 지연을 먼저 검사함이 없이 지연의 변경을 가능하게 한다.
- [0096] *서브세트들에 대한 시냅스 지연의 할당*
- [0097] 위에 설명된 바와 같이, 단일의 시냅스들의 지연을 할당하는 것에 더하여, 시냅스들의 특정된 그룹 또는 서브세트에 대한 지연은 하기 형태의 함수를 이용하여 할당될 수도 있다.

- [0098] SetDelay(string tag, float delay)
- [0099] 이는 고유의 태그를 갖는 모든 스냅스들에 특정 지연을 할당하는데 이용될 수도 있다.
- [0100] *확률 분포들에 기초한 시냅스 지연의 할당*
- [0101] 일부 양태들에서, 시냅스들의 특정된 서브세트에 대한 랜덤 지연은 하기 형태의 함수를 이용하여 할당될 수도 있다.
- [0102] SetDelay(string tag, function distribution)
- [0103] 이는 특정된 확률 분포로부터 얻어지는 값들을 이용하여 고유의 태그를 갖는 모든 시냅스들에 지연을 할당하는데 이용될 수도 있다. 이는 신경 네트워크에서 다이버시티를 제공하기 때문에 유익할 수도 있다.
- [0104] *함수에 기초한 시냅스 지연의 할당*
- [0105] 일부 양태들에서, 시냅스들의 특정된 서브세트에 대한 임의 함수로부터 유도되는 지연은 하기 형태의 함수를 이용하여 할당될 수도 있다.
- [0106] SetDelay(string tag, function arbitrary)
- [0107] 이는 특정된 임의 함수로부터 얻어지는 값들을 이용하여 고유의 태그를 갖는 모든 시냅스들에 지연을 할당하는데 이용될 수도 있다.
- [0108] *뉴런 유형들에 기초한 시냅스 지연의 할당*
- [0109] 일부 양태들에서, 시냅스를 통하여 접속되는 뉴런들의 유형들은 적절한 지연을 결정하는데 이용될 수도 있다. 예를 들어, 이러한 시냅스들에 대한 지연들은 하기 형태의 함수를 이용하여 할당될 수도 있다.
- [0110] SetDelay(neuron_type pre, neuron_type post)
- [0111] 이는 특정된 스냅스-전 (pre) 및 스냅스-후 (post) 뉴런 유형들을 이용하여 스냅스에 지연을 할당한다.
- [0112] *스냅스의 제거*
- [0113] 일부 양태들에서, 예약된 지연값은 네트워크로부터 스냅스를 제거하는데 이용될 수도 있다. 예를 들어, 이 예약된 값 (예를 들어, -1) 에 지연을 설정함으로써 시냅스가 장래 프로세싱에 이용되지 않아야 함을 표시할 수도 있다.
- [0114] *스냅스 지연의 검사*
- [0115] SetDelay()함수에 상호보완적으로, GetDelay() 함수는 HLND 언어의 부분으로서 제공될 수도 있다. GetDelay() 함수는 지연 값을 동적으로 추출하는데 이용될 수도 있다. 또한, GetDelay() 함수는 또한, 시냅스에 대한 지연값을 검사하는데 이용될 수도 있다. 함수는 예를 들어, 하기에 의해 주어질 수도 있다
- [0116] int GetDelay() 또는
- [0117] float GetDelay()
- [0118] 이는 어떠한 인수도 허용하지 않을 수도 있지만, 특정 스냅스에 현재 할당되는 지연 값에 대응하는 정수 또는 부동 소수점수를 각각 리턴시킬 수도 있다.
- [0119] *서브세트들에 대한 시냅스 지연의 검사*
- [0120] 단일의 시냅스의 지연을 검사하는 것에 더하여, 스냅스들의 특정된 서브세트에 대한 지연이 검사될 수도 있다. 예를 들어, 시냅스들의 패밀리 또는 그룹이 특정 태그에 할당되었던 경우, 특정 패밀리에 대한 지연은 다음의 예시적인 함수들을 이용하여 검사될 수도 있다.
- [0121] int[] GetDelay(string tag) 또는
- [0122] float[] GetDelay(string tag)
- [0123] 이들 양쪽 모두는 특정된 태그를 갖는 모든 시냅스들에 대응하는 지연들의 배열을 리턴시킨다.
- [0124] 따라서, HLND 언어에 의한 SetDelay() 및 GetDelay() 함수들은 스파이킹 신경 네트워크에서의 임의의 스냅스의 지연을 (예를 들어, 시뮬레이션 동안에) 동적으로 할당 및 검사하기 위한 표준의 그리고 유연성있는 방법을 제

공한다.

- [0125] 도 5 는 본 개시물의 특정 양태들에 따른 범용 프로세서 (502) 를 이용하여 스냅스 지연들을 동적으로 업데이트하는 상술한 예시의 구현 (500) 을 도시한다. 변수들 (신경 신호들), 시냅스 가중치들, 계산 네트워크 (신경 네트워크) 와 연관된 시스템 파라미터들, 지연 파라미터들, 주파수 빈 정보, 및 시냅스 정보는 메모리 블록 (504) 에 저장될 수도 있고, 한편 범용 프로세서 (502) 에서 실행되는 명령들은 프로그램 메모리 (506) 로부터 로딩될 수도 있다. 본 개시물의 일 양태에서, 범용 프로세서 (502) 내로 로딩된 명령들은 지연 파라미터를 포함하는 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 초기화하고, 신경 네트워크를 동작시키고/시키거나 지연 파라미터를 동적으로 업데이트하는 코드를 포함할 수도 있다.
- [0126] 도 6 은 상술한 시냅스 지연들을 동적으로 업데이트하는 예시적인 구현에 (600) 를 예시하며, 여기서 메모리 (602) 는 본 개시물의 특정 양태들에 따라 계산 네트워크 (신경 네트워크) 의 개개의 (분산된) 프로세싱 유닛들 (신경 프로세서들) (606) 과 상호접속 네트워크 (604) 를 통해 인터페이싱될 수 있다. 변수들 (신경 신호들), 시냅스 가중치들, 계산 네트워크 (신경 네트워크) 지연들과 연관된 시스템 파라미터들, 지연 파라미터들, 주파수 빈 정보, 및 시냅스 정보는 메모리 (602) 에 저장될 수도 있고, 상호접속 네트워크 (604) 의 접속 (들)을 통해 메모리 (602) 로부터 각각의 프로세싱 유닛 (신경 프로세서) (606) 으로 로딩될 수도 있다. 본 개시물의 일 양태에서, 프로세싱 유닛 (606) 은 지연 파라미터를 포함하는 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 초기화하고, 신경 네트워크를 동작시키고/시키거나 지연 파라미터를 동적으로 업데이트하도록 구성될 수도 있다.
- [0127] 도 7 은 상술한 시냅스 지연들을 동적으로 업데이트하는 예시적인 구현에 (700) 를 예시한다. 도 7 에 도시된 바와 같이, 하나의 메모리 बैं크 (702) 는 계산 네트워크 (신경 네트워크) 의 하나의 프로세싱 유닛 (704) 과 직접적으로 인터페이싱될 수도 있다. 각각의 메모리 बैं크 (702) 는 변수들 (신경 신호들), 시냅스 가중치들, 및/또는 대응하는 프로세싱 유닛 (신경 프로세서) (704) 과 연관된 시스템 파라미터들, 지연 파라미터들, 주파수 빈 정보, 및 시냅스 정보를 저장할 수도 있다. 본 개시물의 일 양태에서, 프로세싱 유닛 (704) 은 지연 파라미터를 포함하는 스테이트먼트에 기초한 프로그램에 기초하여 지연 파라미터를 초기화하고, 신경 네트워크를 동작시키고/시키거나 지연 파라미터를 동적으로 업데이트하도록 구성될 수도 있다.
- [0128] 도 8 은 본 개시물의 특정 양태들에 따른 신경 네트워크 (800) 의 일 예시적인 구현을 도시한다. 도 8 에 도시된 바와 같이, 신경 네트워크 (800) 는 위에서 설명된 방법들의 다양한 동작들을 수행할 수도 있는 다수의 로컬 프로세싱 유닛들 (802) 을 가질 수도 있다. 각각의 로컬 프로세싱 유닛 (802) 은 로컬 상태 메모리 (804) 및 신경 네트워크의 파라미터들을 저장하는 로컬 파라미터 메모리 (806) 를 포함할 수도 있다. 또한, 로컬 프로세싱 유닛 (802) 은 로컬 모델 프로그램을 저장하기 위한 로컬 (뉴런) 모델 프로그램 (local model program; LMP) 메모리 (808), 로컬 학습 프로그램을 저장하기 위한 로컬 학습 프로그램 (local learning program; LLP) 메모리 (810), 및 로컬 연결 메모리 (812) 를 가질 수도 있다. 또한, 도 8 에 도시된 바와 같이, 각각의 로컬 프로세싱 유닛 (802) 은 로컬 프로세싱 유닛 (802) 의 로컬 메모리들에 대한 구성들을 제공하기 위한 구성 프로세서 유닛 (814) 과, 그리고 로컬 프로세싱 유닛들 (802) 사이의 라우팅을 제공하는 라우팅 유닛 (816) 과 인터페이싱될 수도 있다.
- [0129] 일 구성에서, 뉴런 모델은 시냅스 지연들을 동적으로 업데이트하도록 구성된다. 이 모델은 초기화하는 수단, 동작시키는 수단 및 업데이트하는 수단을 갖는다. 일 양태에서, 초기화하는 수단, 동작시키는 수단 및 업데이트하는 수단은 인용한 기능들을 수행하도록 구성된, 범용 프로세서 (502), 프로그램 메모리 (506), 메모리 블록 (504), 메모리 (602), 상호접속 네트워크 (604), 프로세싱 유닛들 (606), 프로세싱 유닛 (704), 로컬 프로세싱 유닛 (802), 및/또는 라우팅 접속 프로세싱 엘리먼트 (816) 일 수도 있다. 다른 구성에서, 앞서 언급된 수단은 앞서 언급된 수단에 의해 인용된 기능들을 수행하도록 구성된 임의의 모듈 또는 임의의 장치일 수도 있다.
- [0130] 본 개시물의 특정 양태들에 따르면, 각각의 로컬 프로세싱 유닛 (802) 은 원하는 신경 네트워크의 하나 이상의 기능적 피쳐들에 기초하여 신경 네트워크의 파라미터들을 결정하고, 결정된 파라미터들이 더 적응되고, 튜닝되고, 업데이트됨으로써 원하는 기능적 피쳐들을 향해 하나 이상의 기능적 피쳐들을 개발하도록 구성될 수도 있다.
- [0131] 도 9 는 신경 네트워크에서의 시냅스 지연들을 동적으로 변경하는 방법 (900) 을 예시하는 흐름도이다. 블록 902 에서, 신경 모델이 지연 파라미터를 초기화한다. 일부 양태들에서, 추가적인 지연 파라미터들 또는 지연 파라미터들의 그룹들이 초기화될 수도 있다. 블록 904 에서, 뉴런 모델은 신경 네트워크를

동작시킨다.

- [0132] 또한, 블록 906 에서, 뉴런 모델은 지연 파라미터를 동적으로 업데이트한다. 지연 파라미터는 지연 파라미터를 포함하는 스테이트먼트에 기초할 수도 있는 프로그램에 기초하여 업데이트될 수도 있다. 예를 들어, 일부 양태들에서, 지연 파라미터는 HLND 프로그램 SetDelay() 함수를 통하여 업데이트될 수도 있다. 일부 양태들에서, 시냅스들의 그룹 또는 패밀리에 대한 지연 파라미터가 업데이트될 수도 있다.
- [0133] 지연 파라미터는 지연 파라미터를 증분시키거나 또는 지연 파라미터를 감분시킴으로써 업데이트될 수도 있다. 일부 양태들에서, 지연 파라미터는 시냅스 유형, 뉴런 유형, 또는 시스템 리소스들 (예를 들어, 메모리 리소스들) 에 기초하여 설정 또는 업데이트될 수도 있다.
- [0134] 다른 양태에서, 지연 파라미터는 임의 함수에 기초하여 업데이트될 수도 있다. 임의 함수는 시냅스들 또는 특정 시냅스들의 그룹에 대하여 시냅스들의 모집단에 대한 확률 함수 또는 시간에 대한 함수일 수도 있다. 업데이트는 결정론적 또는 확률론적일 수도 있다.
- [0135] 또 다른 양태에서, 지연 파라미터는 뉴런 유형에 기초하여 업데이트될 수도 있다. 예를 들어, 지연 파라미터는 시냅스 전 뉴런의 유형, 시냅스-후 뉴런의 유형, 또는 양쪽 모두에 기초하여 업데이트될 수도 있다.
- [0136] 추가의 양태들에서, 본 방법은 지연 값 또는 지연값들의 패밀리를 동적으로 추출하는 것을 더 포함할 수도 있다. 일부 양태들에서, 본 방법은 무효 지연 파라미터를 결정하고 그 지연 파라미터를 잘라내는 것을 더 포함할 수도 있다.
- [0137] 또 다른 양태들에서, 지연 파라미터는 뉴런 모델로부터 시냅스를 제거하도록 업데이트될 수도 있다. 즉, 업데이트는 시냅스가 장래 프로세싱에 이용되지 않을 수도 있음을 표시하는데 이용될 수도 있다.
- [0138] 뉴런 모델은 추가로 동적 업데이트들의 수를 제한할 수도 있다. 이러한 제한의 구현에는 병목현상들을 회피하고 시스템 리소스들을 보존할 수도 있기 때문에 유익할 수도 있다.
- [0139] 상술된 방법들의 다양한 동작들은 대응하는 기능들을 수행할 수 있는 임의의 적합한 수단으로 수행될 수도 있다. 수단은 주문형 집적 회로 (ASIC), 또는 프로세서를 포함하여 다양한 하드웨어 및/또는 소프트웨어 컴포넌트(들) 및/또는 모듈(들)을 포함하나, 이로 제한되지는 않는다. 일반적으로, 도면들에 대응하는 동작들이 있는 경우, 이러한 동작들은 대응하는 상대 수단 + 동일한 번호를 갖는 기능 컴포넌트들을 가질 수도 있다.
- [0140] 본원에서 이용되는 바와 같이, 용어 "결정하기" 는 매우 다양한 액션들을 포괄한다. 예를 들어, "결정하기" 는 산출하기, 계산하기, 프로세싱하기, 도출하기, 조사하기, 검색하기 (예를 들어, 테이블, 데이터베이스, 또는 다른 데이터 구조에서 검색하기), 확인하기 등을 포함할 수도 있다. 또한, "결정하기" 는 수신하기 (예를 들어, 정보 수신하기), 액세스하기 (예를 들어, 메모리 내의 데이터에 액세스하기) 등을 포함할 수도 있다. 또한, "결정하기" 는 해결하기, 선택하기, 고르기, 설정하기 등을 포함할 수도 있다.
- [0141] 본원에서 이용되는 바와 같이, 아이템들의 리스트 중 "그 중 적어도 하나" 를 지칭하는 구절은 단일 구성부를 포함하여, 이러한 아이템들의 임의의 조합을 지칭한다. 예로서, "a, b, 또는 c" 중의 적어도 하나" 는 a, b, c, a-b, a-c, b-c, 및 a-b-c 를 포함하고자 한다.
- [0142] 본원 개시물과 연계하여 설명된 다양한 예증적인 논리 블록들, 모듈들, 및 회로들은 본원에서 개시된 기능들을 수행하도록 디자인된 범용 프로세서, 디지털 신호 프로세서 (DSP), 주문형 반도체 (ASIC), 필드 프로그램가능한 게이트 어레이 (FPGA) 또는 다른 프로그램가능한 로직 디바이스 (PLD), 이산 게이트 또는 트랜지스터 로직, 이산 하드웨어 컴포넌트들, 또는 이들의 임의의 조합에 의해 구현되거나 수행될 수도 있다. 범용 프로세서는 마이크로프로세서일 수도 있으나, 대안으로, 프로세서는 임의의 상업적으로 이용가능한 프로세서, 제어기, 마이크로제어기, 또는 상태 머신일 수도 있다. 프로세서는 또한 컴퓨팅 디바이스들의 조합, 예를 들어, DSP 와 마이크로프로세서의 조합, 복수의 마이크로프로세서들, DSP 코어와 연계한 하나 이상의 마이크로프로세서들, 또는 임의의 다른 그러한 구성으로 구현될 수도 있다.
- [0143] 본 개시물과 연계하여 설명된 방법의 단계들 또는 알고리즘은 하드웨어에서, 프로세서에 의해 실행되는 소프트웨어 모듈에서, 또는 이들 양자의 조합에서 직접적으로 구현될 수도 있다. 소프트웨어 모듈은 공지된 임의의 형태의 저장 매체 내에 있을 수도 있다. 이용될 수도 저장 매체들의 일부 예들은, 랜덤 액세스 메모리 (random access memory; RAM), 판독 전용 메모리 (read only memory; ROM), 플래시 메모리, 소거가능한 프로그램가능 판독 전용 메모리 (erasable programmable read-only memory; EPROM), 전기적으로 소거가능한 프로그램가능 판독 전용 메모리 (electrically erasable programmable read-only memory; EEPROM), 레지스터들, 하드

디스크, 이동식 디스크, CD-ROM 등을 포함한다. 소프트웨어 모듈은 단일 명령 또는 많은 명령들을 포함할 수도 있고, 상이한 프로그램들 사이에서 여러 상이한 코드 세그먼트들에 걸쳐, 그리고 다수의 저장 매체들에 걸쳐 분배될 수도 있다. 저장 매체는 프로세서에 연결되어, 프로세서가 저장 매체로부터 정보를 관독하거나 저장 매체에 정보를 기록할 수 있다. 대안에서, 저장 매체는 프로세서에 통합될 수도 있다.

[0144] 본원에 개시된 방법들은 설명된 방법을 달성하기 위한 하나 이상의 단계들 또는 액션들을 포함한다. 방법 단계들 및/또는 액션들은 청구항들의 범위를 벗어나지 않으면서 서로 상호 교환될 수도 있다. 다시 말해, 단계들 또는 액션들에 대한 특정 순서가 명시되지 않는 한, 특정 단계들 및/또는 액션들의 순서 및/또는 이용은 청구항들의 범위로부터 벗어남이 없이 수정될 수도 있다.

[0145] 설명된 기능들은 하드웨어, 소프트웨어, 펌웨어, 또는 이들의 임의의 조합으로 구현될 수도 있다. 하드웨어에서 구현된다면, 일 예시적인 하드웨어 구성은 디바이스에서의 프로세싱 시스템을 포함할 수도 있다. 프로세싱 시스템은 버스 아키텍처로 구현될 수도 있다. 버스는 프로세싱 시스템 및 전체 설계 제약들의 특정 애플리케이션들에 따라 임의의 개수의 상호접속하는 버스들 및 브리지들을 포함할 수도 있다. 버스는 프로세서, 머신-관독가능 매체들, 및 버스 인터페이스를 포함하여 다양한 회로들을 함께 링크할 수도 있다. 버스 인터페이스는 다른 것들 중에서 네트워크 어댑터를 버스를 통해 프로세싱 시스템에 연결하는데 이용될 수도 있다. 네트워크 어댑터는 신호 프로세싱 기능들을 구현하는데 이용될 수도 있다. 특정 양태들에서, 사용자 인터페이스 (예를 들어, 키보드, 디스플레이, 마우스, 조이스틱 등) 가 또한 버스에 연결될 수도 있다. 버스는 또한 다양한 다른 회로들, 예컨대, 타이밍 소스들, 주변기기를, 전압 조절기들, 전력 관리 회로들 등을 링크할 수도 있으며, 이는 공지되어 있으므로, 더 이상 설명되지 않을 것이다.

[0146] 프로세서는 컴퓨터 관독가능 매체 상에 저장된 소프트웨어의 실행을 포함하여 버스 및 범용 프로세싱을 관리하는 역할을 할 수도 있다. 프로세서는 하나 이상의 범용 및/또는 특수-목적용 프로세서들로 구현될 수도 있다. 예들은 마이크로프로세서들, 마이크로제어기들, DSP 제어기들, 및 소프트웨어를 실행할 수 있는 다른 회로부를 포함한다. 소프트웨어는 소프트웨어, 펌웨어, 미들웨어, 마이크로코드, 하드웨어 서술 언어, 또는 다른 것으로 지칭되더라도, 명령들, 데이터, 또는 이들의 임의의 조합을 의미하는 것으로 광범위하게 해석될 수 있다. 머신-관독가능 매체들은, 예로서, 랜덤 액세스 메모리 (RAM), 플래시 메모리, 관독 전용 메모리 (ROM), 프로그램가능한 관독 전용 메모리 (PROM), 소거가능한 프로그램가능 관독 전용 메모리 (EPROM), 전기적으로 소거가능한 프로그램가능 관독 전용 메모리 (EEPROM), 레지스터들, 자기 디스크들, 광학 디스크들, 하드 드라이브들, 또는 임의의 다른 적합한 저장 매체, 또는 이들의 임의의 조합을 포함할 수도 있다. 머신-관독가능 매체들은 컴퓨터-프로그램 제품으로 구체화될 수도 있다. 컴퓨터-프로그램 제품은 패키징 재료들을 포함할 수도 있다.

[0147] 하드웨어 구현에서, 머신-관독가능 매체들은 프로세서와 별개인 프로세싱 시스템의 일부일 수도 있다. 그러나, 머신-관독가능 매체들, 또는 이의 임의의 부분은 프로세싱 시스템의 외부에 있을 수도 있음을 당업자들은 쉽게 이해할 것이다. 예로서, 머신-관독가능 매체들은 디바이스와 별도의 컴퓨터 제품을 포함할 수도 있으며, 이 모두는 버스 인터페이스를 통해 프로세서에 의해 액세스가능하게 될 수도 있다. 대안으로, 또는 이에 더해, 머신-관독가능 매체들, 또는 이들의 임의의 부분은 프로세서에 통합될 수도 있으며, 그러한 경우에는 캐시 및/또는 범용 레지스터 파일들과 함께 있을 수도 있다. 논의된 다양한 컴포넌트들이 로컬 컴포넌트와 같이 특정 위치를 갖는 것으로 설명되었으나, 그것들은 또한 소정의 컴포넌트들이 분산 컴퓨팅 시스템의 일부로서 구성되는 것과 같이 다양한 방식으로 구성될 수도 있다.

[0148] 프로세싱 시스템은 프로세서 기능성을 제공하는 하나 이상의 마이크로프로세서들 및 적어도 일부분의 머신-관독가능 매체들을 제공하는 외부 메모리로 구현될 수도 있으며, 모두 외부 버스 아키텍처를 통해 다른 지원하는 회로부와 함께 링크된다. 대안으로, 프로세싱 시스템은 뉴런 모델들 및 본원에서 설명된 신경 시스템들의 모델들을 구현하기 위한 하나 이상의 뉴로모픽 프로세서들을 포함할 수도 있다. 다른 대안으로서, 프로세싱 시스템은 프로세서를 갖는 주문형 반도체 (ASIC), 버스 인터페이스, 사용자 인터페이스, 지원 회로부, 및 단일 칩 내에 통합되는 적어도 일부분의 머신-관독가능 매체들로, 또는 하나 이상의 필드 프로그램가능 게이트 어레이 (FPGA) 들, 프로그램가능 로직 디바이스 (PLD) 들, 제어기들, 상태 머신들, 게이트 로직, 이상 하드웨어 컴포넌트들, 또는 임의의 다른 적합한 회로부, 또는 본 개시물을 통해 설명된 다양한 기능성을 수행할 수 있는 회로들의 임의의 조합으로 구현될 수도 있다. 특정 응용 및 전체 시스템에 부과되는 전체 설계 제약들에 따라 본 개시물에 걸쳐 제시된 설명된 기능성을 가장 잘 구현하기 위한 방법을 당업자들은 인지할 것이다.

[0149] 머신-관독가능 매체들은 다수의 소프트웨어 모듈들을 포함할 수도 있다. 소프트웨어 모듈들은, 프로세서에

의해 실행되는 경우, 프로세싱 시스템으로 하여금 다양한 기능들을 수행하게 하는 명령들을 포함한다. 소프트웨어 모듈들은 송신 모듈 및 수신 모듈을 포함할 수도 있다. 각각의 소프트웨어 모듈은 단일 저장 디바이스에 있을 수도 있거나 다수의 저장 디바이스들에 걸쳐 분산될 수도 있다. 예로서, 소프트웨어 모듈은 트리거링 이벤트가 발생하는 경우 하드웨어 드라이브로부터 RAM 으로 로딩될 수도 있다. 소프트웨어 모듈의 실행 중에, 프로세서는 액세스 속도를 증가시키기 위해 명령들의 일부를 캐시 내로 로딩할 수도 있다. 하나 이상의 캐시 라인들은 그러면 프로세서에 의한 실행을 위해 범용 레지스터 파일 내로 로딩될 수도 있다. 하기에 소프트웨어 모듈의 기능성을 언급하는 경우, 그러한 기능성은 해당 소프트웨어 모듈로부터 명령들을 실행하는 경우 프로세서에 의해 구현된다는 것이 이해될 것이다.

[0150] 소프트웨어로 구현된다면, 기능들은 하나 이상의 명령들 또는 코드로서 컴퓨터 판독가능 저장 매체상에 저장될 수도 있다. 컴퓨터-판독가능 매체들은 컴퓨터-판독가능 저장 매체들 및 한 장소에서 다른 장소로 컴퓨터 프로그램의 전송을 가능하게 하는 임의의 매체를 포함하는 통신 매체들 양자 모두를 포함한다. 저장 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 이용 가능한 저장 매체일 수도 있다. 비제한적인 예로서, 이러한 컴퓨터-판독가능 저장 매체들은 RAM, ROM, EEPROM, CD-ROM 또는 다른 광학 디스크 스토리지, 자기 디스크 스토리지 또는 다른 자기 스토리지 디바이스들, 또는 요구되는 프로그램 코드를 명령들 또는 데이터 구조들의 형태로 저장하기 위해 사용될 수 있고 컴퓨터에 의해 액세스될 수 있는 임의의 다른 저장 매체를 포함할 수 있다. 또한, 임의의 연결부는 통신 매체라고 적절히 지칭된다. 예를 들어, 소프트웨어가 동축 케이블, 광섬유 케이블, 연선, 디지털 가입자 회선 (DSL), 또는 적외선 (IR), 무선, 및 마이크로파와 같은 무선 기술들을 사용하여 웹사이트, 서버, 또는 다른 원격 소스로부터 전송된다면, 동축 케이블, 광섬유 케이블, 연선, DSL, 또는 적외선, 무선, 및 마이크로파와 같은 무선 기술들은 통신 매체의 정의 내에 포함된다. 본원에서 사용된 디스크 (disk) 와 디스크 (disc) 는, 콤팩트 디스크 (CD), 레이저 디스크, 광학 디스크, 디지털 다기능 디스크 (DVD), 플로피디스크 및 블루레이® 디스크를 포함하며, 여기서 디스크 (disk) 는 통상 자기적으로 데이터를 재생하고, 디스크 (disc) 는 레이저를 이용하여 광학적으로 데이터를 재생한다. 따라서, 일부 양태들에서, 컴퓨터-판독가능 저장 매체들은 비일시적 컴퓨터-판독가능 저장 매체들 (예를 들어, 유형의 저장 매체들) 을 포함할 수도 있다. 또한, 다른 양태들에 있어서, 통신 매체들은 일시적 통신 매체들 (예를 들어, 신호) 을 포함할 수도 있다.

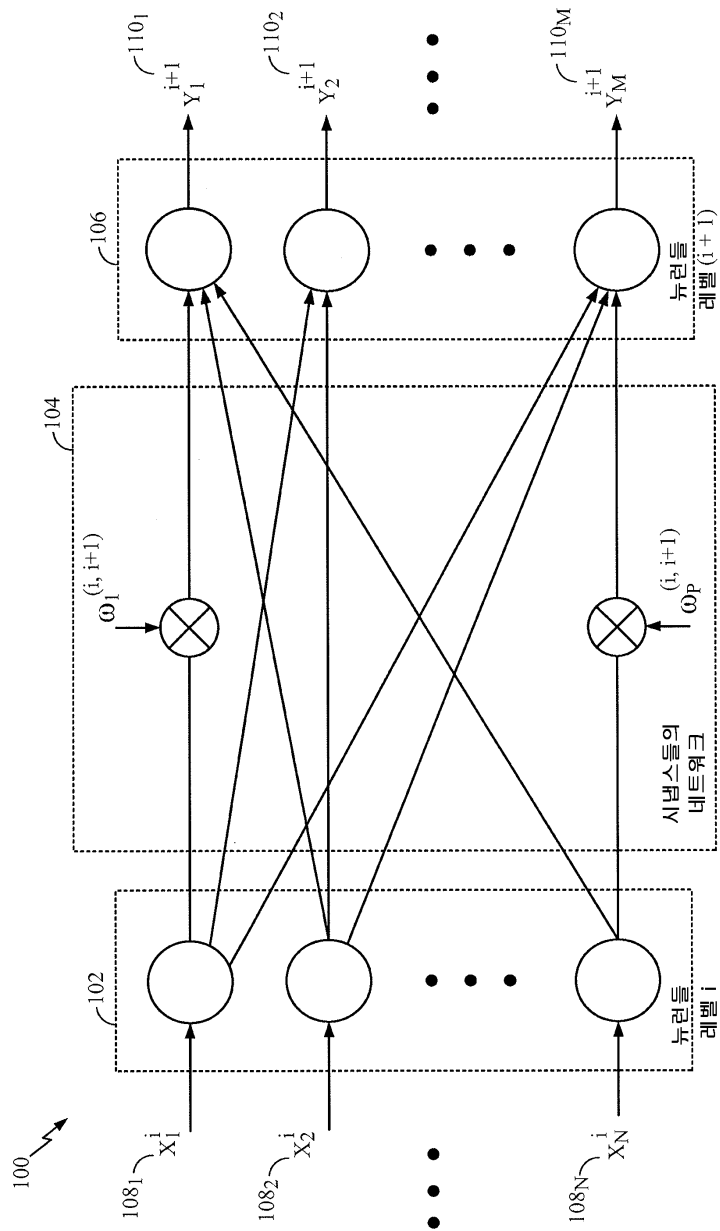
[0151] 따라서, 특정 양태들은 본원에 제시된 동작들을 수행하는 컴퓨터 프로그램 제품을 포함할 수도 있다. 예를 들어, 이러한 컴퓨터 프로그램 제품은 저장된 (및/또는 인코딩된) 명령들을 갖는 컴퓨터 판독가능 저장 매체를 포함할 수도 있으며, 명령들은 본원에 설명된 동작들을 수행하기 위해 하나 이상의 프로세서들에 의해 실행가능할 수도 있다. 특정 양태들에 있어서, 컴퓨터 프로그램 제품은 패키징 재료를 포함할 수도 있다.

[0152] 또한, 본원에 설명된 방법들 및 기법들을 수행하는 모듈들 및/또는 다른 적절한 수단은 다운로드될 수도 있고/있거나, 그렇지 않으면 가능한 적용가능한 사용자 단말 및/또는 기지국에 의해 획득될 수도 있다. 예를 들어, 본원에서 설명된 방법들을 수행하기 위한 수단의 전송을 용이하게 하기 위한 서버에 디바이스가 연결될 수도 있다. 대안으로, 본원에 설명된 다양한 방법들이 저장 수단 (예를 들어, RAM, ROM, 물리적 콤팩트 디스크 (CD) 나 플로피 디스크와 같은 물리적 저장 매체 등) 을 통해 제공될 수도 있어, 사용자 단말 및/또는 기지국은 디바이스에 연결할 시에 또는 디바이스에 저장 수단을 제공할 시에 다양한 방법들을 획득할 수 있다. 또한, 본원에서 설명된 방법들 및 기술들을 디바이스에 제공하기 위해 임의의 다른 적절한 기술들이 활용될 수 있다.

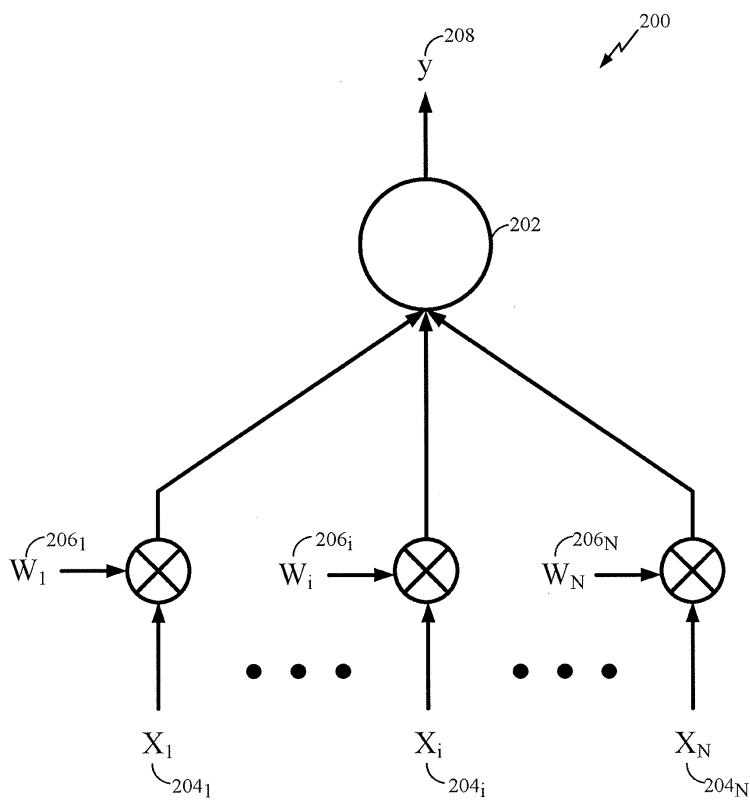
[0153] 청구항들은 위에서 예시된 정확한 구성 및 컴포넌트들로 제한되지 않는 것으로 이해되어야 한다. 청구항의 범위를 벗어나지 않으면서, 본원에서 설명된 시스템들, 방법들, 및 장치들의 배치, 동작 및 세부사항들에서 다양한 수정예들, 변경예들, 및 변형예들이 이루어질 수도 있다.

도면

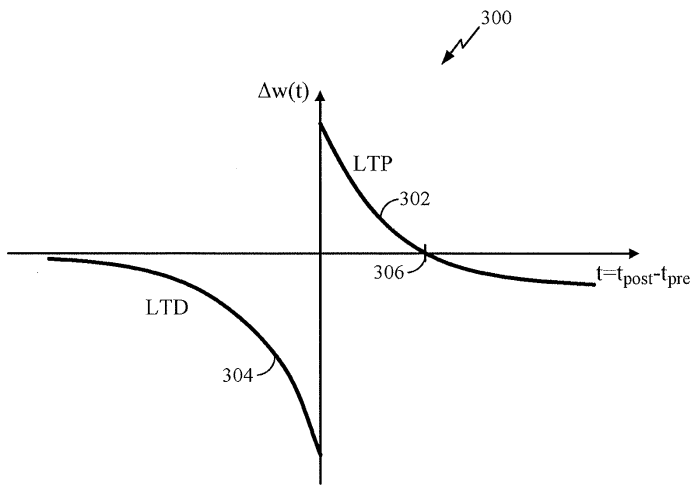
도면1



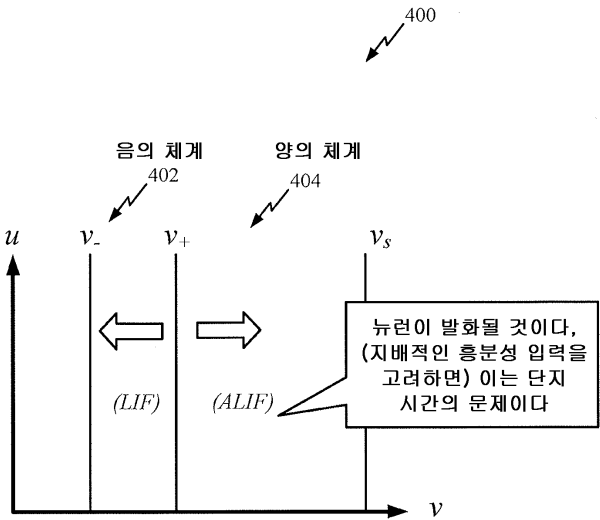
도면2



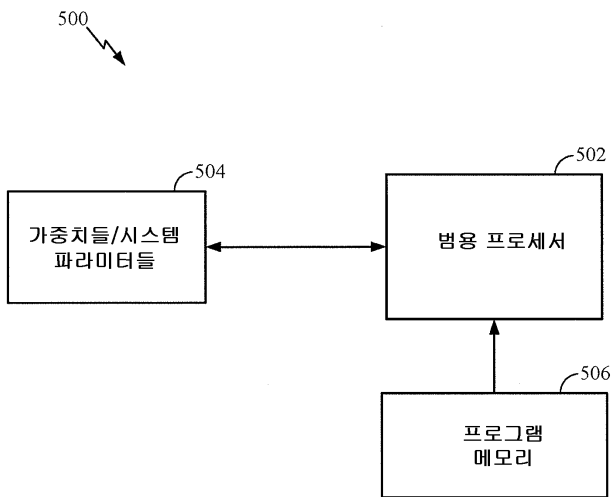
도면3



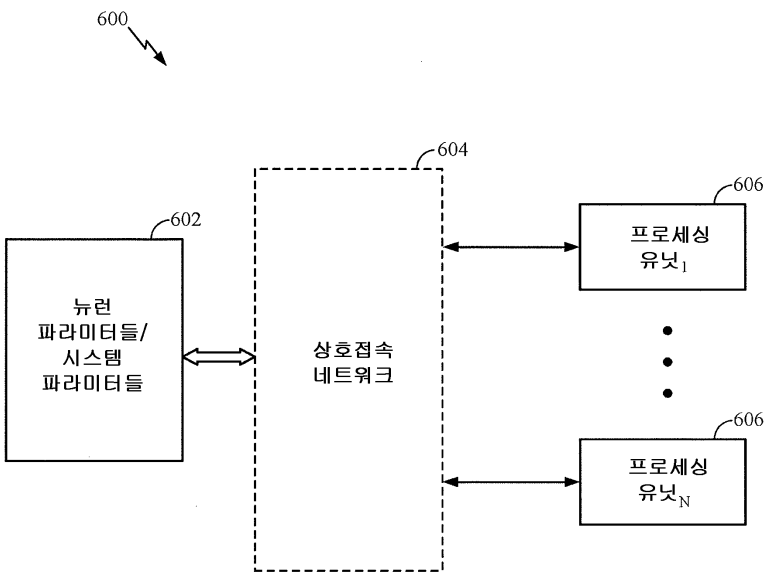
도면4



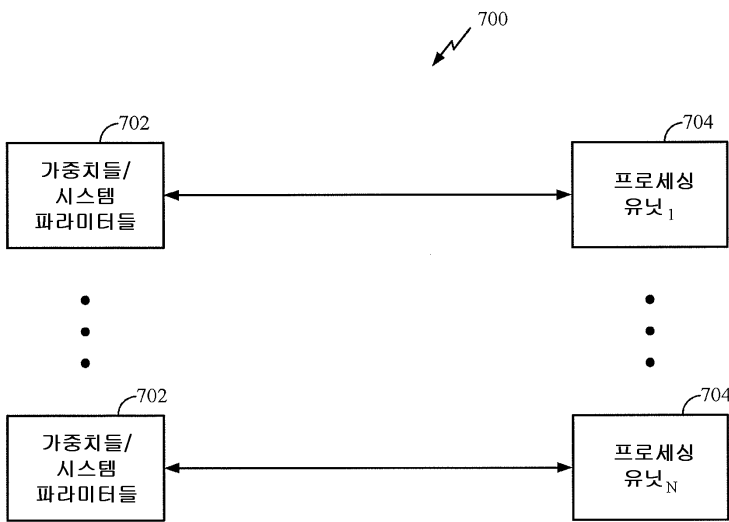
도면5



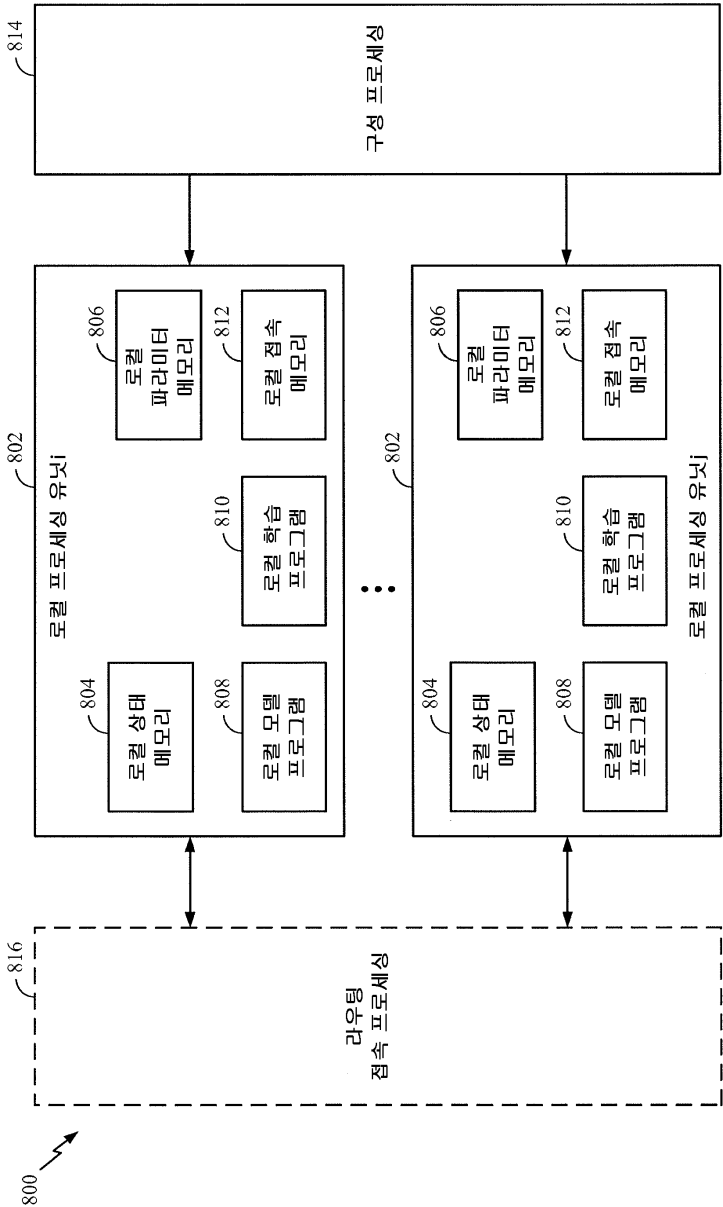
도면6



도면7



도면8



도면9

