



(12)发明专利

(10)授权公告号 CN 106104586 B

(45)授权公告日 2019.10.18

(21)申请号 201580013455.X

J・S・伯纳特 E・M・霍尔

(22)申请日 2015.03.10

M・E・罗梅拉乔利夫

(65)同一申请的已公布的文献号

C・M・维任斯基 M・E・斯穆特

申请公布号 CN 106104586 A

J・N・贝林格

(43)申请公布日 2016.11.09

(74)专利代理机构 上海专利商标事务所有限公司 31100

(30)优先权数据

代理人 唐杰敏

61/953,511 2014.03.14 US

(51)Int.Cl.

14/485,501 2014.09.12 US

G06N 3/04(2006.01)

(85)PCT国际申请进入国家阶段日

G06N 3/10(2006.01)

2016.09.12

(56)对比文件

(86)PCT国际申请的申请数据

US 2013073498 A1,2013.03.21,

PCT/US2015/019731 2015.03.10

US 2013262349 A1,2013.10.03,

(87)PCT国际申请的公布数据

US 2013073496 A1,2013.03.21,

W02015/138466 EN 2015.09.17

US 2005267721 A1,2005.12.01,

(73)专利权人 高通股份有限公司

US 2011182349 A1,2011.07.28,

地址 美国加利福尼亚州

审查员 夏雪

(72)发明人 A・K・科纳茨 B・F・贝哈巴迪

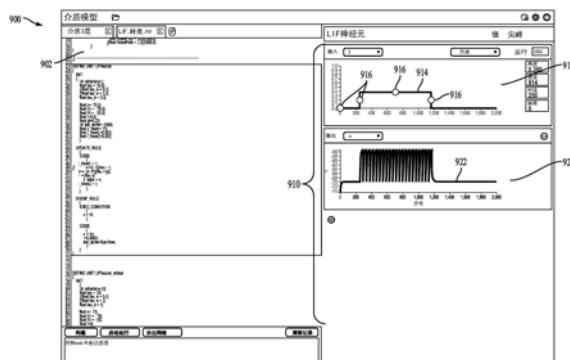
权利要求书2页 说明书16页 附图12页

(54)发明名称

神经元形态模型开发的上下文实时反馈

(57)摘要

一种方法包括生成神经元形态模型中的上下文反馈。该神经元形态模型包括在神经元形态模型的开发期间要被监视的一个或多个资产。该方法进一步包括显示互动上下文面板以便基于上下文反馈来示出表示。



1. 一种用于神经元形态模型开发的上下文实时反馈的方法,所述方法包括:  
生成神经元形态模型的上下文反馈,所述神经元形态模型包括在所述神经元形态模型的开发期间被监视的资产,所述资产包括神经元、突触或神经元网络中的至少一者;  
显示互动上下文面板以便至少部分地基于在所述神经元形态模型的开发期间的所述上下文反馈来示出表示;以及  
使用所述互动上下文面板来修改所述神经元形态模型的输入曲线。
2. 如权利要求1所述的方法,其特征在于,进一步包括基于执行所述模型来更新所述互动上下文面板。
3. 如权利要求1所述的方法,其特征在于,所述表示实时地发生。
4. 如权利要求1所述的方法,其特征在于,进一步包括操纵所述互动上下文面板以便更新所述神经元形态模型的代码。
5. 如权利要求1所述的方法,其特征在于,进一步包括更新所述神经元形态模型的代码以便更新所述互动上下文面板。
6. 如权利要求1所述的方法,其特征在于,所述互动上下文面板示出与所述神经元形态模型的动态相关的参数的上下文信息。
7. 如权利要求1所述的方法,其特征在于,所述表示包括所述神经元形态模型中的一个或多个神经元的布局或所述神经元形态模型中的神经元的连通性中的至少一者的视觉显示。
8. 如权利要求1所述的方法,其特征在于,所述上下文反馈包括相对于所述神经元形态模型的硬件布局的信息。
9. 如权利要求8所述的方法,其特征在于,所述信息包括功耗或计算负荷中的至少一者。
10. 一种用于神经元形态模型开发的上下文实时反馈的装备,所述装备包括:  
存储器;以及  
耦合至所述存储器的至少一个处理器,所述至少一个处理器被配置成:  
生成神经元形态模型的上下文反馈,所述神经元形态模型包括在所述神经元形态模型的开发期间被监视的资产,所述资产包括神经元、突触或神经元网络中的至少一者;  
显示互动上下文面板以便至少部分地基于在所述神经元形态模型的开发期间的所述上下文反馈来示出表示;以及  
使用所述互动上下文面板来修改所述神经元形态模型的输入曲线。
11. 如权利要求10所述的装备,其特征在于,所述至少一个处理器被进一步配置成基于执行所述模型来更新所述互动上下文面板。
12. 如权利要求10所述的装备,其特征在于,所述至少一个处理器被进一步配置成实时地显示所述表示。
13. 如权利要求10所述的装备,其特征在于,所述至少一个处理器被进一步配置成操纵所述互动上下文面板以更新所述神经元形态模型的代码。
14. 如权利要求10所述的装备,其特征在于,所述至少一个处理器被进一步配置成更新所述神经元形态模型的代码以更新所述互动上下文面板。
15. 如权利要求10所述的装备,其特征在于,所述互动上下文面板示出与所述神经元形

态模型的动态相关的参数的上下文信息。

16. 如权利要求10所述的装备,其特征在于,所述表示包括所述神经元形态模型中的一个或多个神经元的布局或所述神经元形态模型中的神经元的连通性中的至少一者的视觉显示。

17. 如权利要求10所述的装备,其特征在于,所述上下文反馈包括相对于所述神经元形态模型的硬件布局的信息。

18. 如权利要求17所述的装备,其特征在于,所述信息包括功耗或计算负荷中的至少一者。

19. 一种用于神经元形态模型开发的上下文实时反馈的装备,所述装备包括:

用于生成神经元形态模型的上下文反馈的装置,所述神经元形态模型包括在所述神经元形态模型的开发期间被监视的资产,所述资产包括神经元、突触或神经元网络中的至少一者;

用于显示互动上下文面板以便至少部分地基于在所述神经元形态模型的开发期间的所述上下文反馈来示出表示的装置;以及

用于使用所述互动上下文面板来修改所述神经元形态模型的输入曲线的装置。

20. 一种其上编码有程序代码的非瞬态计算机可读介质,所述程序代码由处理器执行并且包括:

用于生成神经元形态模型的上下文反馈的程序代码,所述神经元形态模型包括在所述神经元形态模型的开发期间被监视的资产,所述资产包括神经元、突触或神经元网络中的至少一者;

用于显示互动上下文面板以便至少部分地基于在所述神经元形态模型的开发期间的所述上下文反馈来示出表示的程序代码;以及

用于使用所述互动上下文面板来修改所述神经元形态模型的输入曲线的程序代码。

## 神经元形态模型开发的上下文实时反馈

[0001] 相关申请的交叉引用

[0002] 本申请要求于2014年3月14日提交且题为“CONTEXTUAL REAL-TIME FEEDBACK FOR NEUROMORPHIC MODEL DEVELOPMENT(神经元形态模型开发的上下文实时反馈)”的美国临时专利申请号61/953,511的权益,其公开内容通过援引全部明确纳入于此。

[0003] 背景

[0004] 领域

[0005] 本公开的某些方面一般涉及神经系统工程,并且尤其涉及用于神经元形态模型开发的上下文实时反馈的系统和方法。

[0006] 背景

[0007] 可包括一群互连的人工神经元(即,神经元模型)的人工神经网络是一种计算设备或者表示将由计算设备执行的方法。人工神经网络可具有生物学神经网络中的对应的结构和/或功能。然而,人工神经网络可为其中传统计算技术是麻烦的、不切实际的、或不胜任的某些应用提供创新且有用的计算技术。由于人工神经网络能从观察中推断出功能,因此这样的网络在因任务或数据的复杂度使得通过常规技术来设计该功能较为麻烦的应用中是特别有用的。

[0008] 概述

[0009] 在本公开的一方面,给出了一种方法。该方法包括生成神经元形态模型中的上下文反馈,该神经元形态模型包括在神经元形态模型的开发期间要被监视的一个或多个资产。该方法进一步包括显示互动上下文面板以便基于上下文反馈来示出表示。

[0010] 在本公开的另一方面,给出了一种装备。该装备包括存储器以及耦合至该存储器的一个或多个处理器。该(诸)处理器被配置成生成神经元形态模型中的上下文反馈,该神经元形态模型包括在神经元形态模型的开发期间要被监视的一个或多个资产。该(诸)处理器被进一步配置成显示互动上下文面板以便基于上下文反馈来示出表示。

[0011] 在本公开的另一方面,给出了一种装备。该装备包括用于生成神经元形态模型中的上下文反馈的装置,该神经元形态模型包括在神经元形态模型的开发期间要被监视的一个或多个资产。该装备进一步包括用于显示互动上下文面板以便基于上下文反馈来示出表示的装置。

[0012] 在本公开的另一方面,给出了一种计算机程序产品。该计算机程序产品包括其上编码有程序代码的非瞬态计算机可读介质。该程序代码包括用于生成神经元形态模型中的上下文反馈的程序代码,该神经元形态模型包括在神经元形态模型的开发期间要被监视的一个或多个资产。该程序代码进一步包括用于显示互动上下文面板以便基于上下文反馈来示出表示的程序代码。

[0013] 这已较宽泛地勾勒出本公开的特征和技术优势以便下面的详细描述可以被更好地理解。本公开的附加特征和优点将在下文描述。本领域技术人员应该领会,本公开可容易地被用作修改或设计用于实施与本公开相同的目的的其他结构的基础。本领域技术人员还应认识到,这样的等效构造并不脱离所附权利要求中所阐述的本公开的教导。被认为是本

公开的特性的新颖特征在其组织和操作方法两方面连同进一步的目的和优点在结合附图来考虑以下描述时将被更好地理解。然而,要清楚理解的是,提供每一幅附图均仅用于解说和描述目的,且无意作为对本公开的限定的定义。

[0014] 附图简述

[0015] 在结合附图理解下面阐述的详细描述时,本公开的特征、本质和优点将变得更加明显,在附图中,相同附图标记始终作相应标识。

[0016] 图1解说了根据本公开的某些方面的示例神经元网络。

[0017] 图2解说了根据本公开的某些方面的计算网络(神经系统或神经网络)的处理单元(神经元)的示例。

[0018] 图3解说了根据本公开的某些方面的尖峰定时依赖可塑性(STDP)曲线的示例。

[0019] 图4解说了根据本公开的某些方面的用于定义神经元模型的行为的正态相和负态相的示例。

[0020] 图5解说了根据本公开的某些方面的使用通用处理器来设计神经网络的示例实现。

[0021] 图6解说了根据本公开的某些方面的设计其中存储器可以与个体分布式处理单元对接的神经网络的示例实现。

[0022] 图7解说了根据本公开的某些方面的基于分布式存储器和分布式处理单元来设计神经网络的示例实现。

[0023] 图8解说了根据本公开的某些方面的神经网络的示例实现。

[0024] 图9是解说根据本公开的各方面的包括可调节输入曲线的示例性上下文面板的屏幕截图。

[0025] 图10A是解说根据本公开的各方面的示例性代码块的框图。

[0026] 图10B解说了根据本公开的各方面的可被包括在上下文面板中的示例性数据可视化特征。

[0027] 图11解说了根据本公开的各方面的示出用于生成神经元形态模型中的上下文反馈的架构的框图。

[0028] 图12解说了根据本公开的各方面的生成神经元形态模型中的上下文反馈的方法。

[0029] 详细描述

[0030] 以下结合附图阐述的详细描述旨在作为各种配置的描述,而无意表示可实践本文中所描述的概念的仅有的配置。本详细描述包括具体细节以便提供对各种概念的透彻理解。然而,对于本领域技术人员将显而易见的是,没有这些具体细节也可实践这些概念。在一些实例中,以框图形式示出众所周知的结构和组件以避免湮没此类概念。

[0031] 基于本教导,本领域技术人员应领会,本公开的范围旨在覆盖本公开的任何方面,不论其是与本公开的任何其他方面相独立地还是组合地实现的。例如,可以使用所阐述的任何数目的方面来实现装置或实践方法。另外,本公开的范围旨在覆盖使用作为所阐述的本公开的各个方面的补充或者与之不同的其他结构、功能性、或者结构及功能性来实践的此类装置或方法。应当理解,所披露的本公开的任何方面可由权利要求的一个或多个元素来实施。

[0032] 措辞“示例性”在本文中用于表示“用作示例、实例或解说”。本文中描述为“示例

性”的任何方面不必被解释为优于或胜过其他方面。

[0033] 尽管本文描述了特定方面,但这些方面的众多变体和置换落在本公开的范围之内。虽然提到了优选方面的一些益处和优点,但本公开的范围并非旨在被限定于特定益处、用途或目标。相反,本公开的各方面旨在能宽泛地应用于不同的技术、系统配置、网络、和协议,其中一些作为示例在附图以及以下对优选方面的描述中解说。详细描述和附图仅仅解说本公开而非限定本公开,本公开的范围由所附权利要求及其等效技术方案来定义。

#### [0034] 示例神经系统、训练及操作

[0035] 图1解说了根据本公开的某些方面的具有多级神经元的示例人工神经系统100。神经系统100可具有神经元级102,该神经元级102通过突触连接网络104(即,前馈连接)来连接到另一神经元级106。为简单起见,图1中仅解说了两级神经元,尽管神经系统中可能存在更少或更多级神经元。应注意,一些神经元可通过侧向连接来连接至同层中的其他神经元。此外,一些神经元可通过反馈连接来后向连接至先前层中的神经元。

[0036] 如图1所解说了,级102中的每一个神经元可以接收可由前级的神经元(未在图1中示出)生成的输入信号108。信号108可表示级102的神经元的输入电流。该电流可在神经元膜上累积以对膜电位进行充电。当膜电位达到其阈值时,该神经元可激发并生成输出尖峰,该输出尖峰将被传递到下一级神经元(例如,级106)。在一些建模办法中,神经元可以连续地向下一一级神经元传递信号。该信号通常是膜电位的函数。此类行为可在硬件和/或软件(包括模拟和数字实现,诸如以下所述那些实现)中进行仿真或模拟。

[0037] 在生物学神经元中,在神经元激发时生成的输出尖峰被称为动作电位。该电信号是相对迅速、瞬态的神经脉冲,其具有约为100mV的振幅和约为1ms的历时。在具有一系列连通的神经元(例如,尖峰从图1中的一级神经元传递至另一级神经元)的神经系统的特定实施例中,每个动作电位都具有基本上相同的振幅和历时,并且因此该信号中的信息可仅由尖峰的频率和数目、或尖峰的时间来表示,而不由振幅来表示。动作电位所携带的信息可由尖峰、发放了尖峰的神经元、以及该尖峰相对于一个或数个其他尖峰的时间来确定。尖峰的重要性可由向各神经元之间的连接所应用的权重来确定,如以下所解释的。

[0038] 尖峰从一级神经元向另一级神经元的传递可通过突触连接(或简称“突触”)网络104来达成,如图1中所解说了。相对于突触104,级102的神经元可被视为突触前神经元,而级106的神经元可被视为突触后神经元。突触104可接收来自级102的神经元的输出信号(即,尖峰),并根据可调节突触权重 $w_1^{(i,i+1)}$ 、 $\dots$ 、 $w_P^{(i,i+1)}$ 来按比例缩放那些信号,其中P是级102的神经元与级106的神经元之间的突触连接的总数,并且i是神经元级的指示符。在图1的示例中,i表示神经元级102并且i+1表示神经元级106。此外,经按比例缩放的信号可被组合以作为级106中每个神经元的输入信号。级106中的每个神经元可基于对应的组合输入信号来生成输出尖峰110。可使用另一突触连接网络(图1中未示出)将这些输出尖峰110传递到另一级神经元。

[0039] 生物学突触可以仲裁突触后神经元中的兴奋性或抑制性(超极化)动作,并且还可用于放大神经元信号。兴奋性信号使膜电位去极化(即,相对于静息电位增大膜电位)。如果在某个时间段内接收到足够的兴奋性信号以使膜电位去极化到高于阈值,则在突触后神经元中发生动作电位。相反,抑制性信号一般使膜电位超极化(即,降低膜电位)。抑制性信号

如果足够强则可抵消掉兴奋性信号之和并阻止膜电位到达阈值。除了抵消掉突触兴奋以外,突触抑制还可对自发活跃神经元施加强力的控制。自发活跃神经元是指在没有进一步输入的情况下(例如,由于其动态或反馈而)发放尖峰的神经元。通过压制这些神经元中的动作电位的自发生成,突触抑制可对神经元中的激发模式进行定形,这一般被称为雕刻。取决于期望的行为,各种突触104可充当兴奋性或抑制性突触的任何组合。

[0040] 神经系统100可由通用处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或其他可编程逻辑器件(PLD)、分立的门或晶体管逻辑、分立的硬件组件、由处理器执行的软件模块、或其任何组合来仿真。神经系统100可用在大范围的应用中,诸如图像和模式识别、机器学习、电机控制、及类似应用等。神经系统100中的每一神经元可被实现为神经元电路。被充电至发起输出尖峰的阈值的神经元膜可被实现为例如对流经其的电流进行积分的电容器。

[0041] 在一方面,电容器作为神经元电路的电流积分器件可被除去,并且可使用较小的忆阻器元件来替代它。这种办法可应用于神经元电路中,以及其中大容量电容器被用作电流积分器的各种其他应用中。另外,每个突触104可基于忆阻器元件来实现,其中突触权重变化可与忆阻器电阻的变化有关。使用纳米特征尺寸的忆阻器,可显著地减小神经元电路和突触的面积,这可使得实现大规模神经系统硬件实现更为切实可行。

[0042] 对神经系统100进行仿真的神经处理器的功能性可取决于突触连接的权重,这些权重可控制神经元之间的连接的强度。突触权重可存储在非易失性存储器中以在掉电之后保留该处理器的功能性。在一方面,突触权重存储器可实现在与主神经处理器芯片分开的外部芯片上。突触权重存储器可与神经处理器芯片分开地封装成可更换的存储卡。这可向神经处理器提供多种多样的功能性,其中特定功能性可基于当前附连至神经处理器的存储卡中所存储的突触权重。

[0043] 图2解说了根据本公开的某些方面的计算网络(例如,神经系统或神经网络)的处理单元(例如,神经元或神经元电路)202的示例性示图200。例如,神经元202可对应于来自图1的级102和106的任何神经元。神经元202可接收多个输入信号204<sub>1</sub>-204<sub>N</sub>,这些输入信号可以是该神经系统外部的信号、或是由同一神经系统的其他神经元所生成的信号、或这两者。输入信号可以是电流、电导、电压、实数值的和/或复数值的。输入信号可包括具有定点或浮点表示的数值。可通过突触连接将这些输入信号递送到神经元202,突触连接根据可调节突触权重206<sub>1</sub>-206<sub>N</sub>(W<sub>1</sub>-W<sub>N</sub>)对这些信号进行按比例缩放,其中N可以是神经元202的输入连接总数。

[0044] 神经元202可组合这些经按比例缩放的输入信号,并且使用组合的经按比例缩放的输入来生成输出信号208(即,信号Y)。输出信号208可以是电流、电导、电压、实数值的和/或复数值的。输出信号可以是具有定点或浮点表示的数值。随后该输出信号208可作为输入信号传递至同一神经系统的其他神经元、或作为输入信号传递至同一神经元202、或作为该神经系统的输出来传递。

[0045] 处理单元(神经元)202可由电路来仿真,并且其输入和输出连接可由具有突触电路的电连接来仿真。处理单元202及其输入和输出连接也可由软件代码来仿真。处理单元202还可由电路来仿真,而其输入和输出连接可由软件代码来仿真。在一方面,计算网络中的处理单元202可以是模拟电路。在另一方面,处理单元202可以是数字电路。在又一方面,

处理单元202可以是具有模拟和数字组件两者的混合信号电路。计算网络可包括任何前述形式的处理单元。使用这样的处理单元的计算网络(神经系统或神经网络)可用在大范围的应用中,诸如图像和模式识别、机器学习、电机控制、及类似应用等。

[0046] 在神经网络的训练过程期间,突触权重(例如,来自图1的权重 $w_1^{(i,i+1)}, \dots, w_P^{(i,i+1)}$

和/或来自图2的权重 $206_1-206_N$ )可用随机值来初始化并根据学习规则而被增大或减小。本领域技术人员将领会,学习规则的示例包括但不限于尖峰定时依赖可塑性(STDP)学习规则、Hebb规则、Oja规则、Bienenstock-Copper-Munro (BCM) 规则等。在一些方面,这些权重可稳定或收敛至两个值(即,权重的双峰分布)之一。该效应可被用于减少每个突触权重的位数、提高从/向存储突触权重的存储器读取和写入的速度、以及降低突触存储器的功率和/或处理器消耗。

[0047] 突触类型

[0048] 在神经网络的硬件和软件模型中,突触相关功能的处理可基于突触类型。突触类型可以是非可塑突触(权重和延迟没有改变)、可塑突触(权重可改变)、结构化延迟可塑突触(权重和延迟可改变)、全可塑突触(权重、延迟和连通性可改变)、以及基于此的变型(例如,延迟可改变,但在权重或连通性方面没有改变)。多种类型的优点在于处理可以被细分。例如,非可塑突触不会使用待执行的可塑性功能(或等待此类功能完成)。类似地,延迟和权重可塑性可被细分成可一起或分开地、顺序地或并行地运作的操作。不同类型的突触对于适用的每一种不同的可塑性类型可具有不同的查找表或公式以及参数。因此,这些方法将针对该突触的类型来访问相关的表、公式或参数。

[0049] 还进一步牵涉到以下事实:尖峰定时依赖型结构化可塑性可独立于突触可塑性地来执行。结构化可塑性即使在权重幅值没有改变的情况下(例如,如果权重已达最小或最大值、或者其由于某种其他原因而不被改变)也可被执行,因为结构化可塑性(即,延迟改变的量)可以是pre-post(前-后)尖峰时间差的直接函数。替换地,结构化可塑性可被设为权重变化量的函数或者可基于与权重或权重变化的界限有关的条件来设置。例如,突触延迟可仅在权重变化发生时或者在权重到达0的情况下才改变,但在这些权重为最大值时则不改变。然而,具有独立函数以使得这些过程能被并行化从而减少存储器访问的次数和交叠可能是有利的。

[0050] 突触可塑性的确定

[0051] 神经元可塑性(或简称“可塑性”)是大脑中的神经元和神经网络响应于新的信息、感官刺激、发展、损坏、或机能障碍而改变其突触连接和行为的能力。可塑性对于生物学中的学习和记忆、以及对于计算神经元科学和神经网络是重要的。已经研究了各种形式的可塑性,诸如突触可塑性(例如,根据Hebbian理论)、尖峰定时依赖可塑性(STDP)、非突触可塑性、活跃性依赖可塑性、结构化可塑性和自稳态可塑性。

[0052] STDP是调节神经元之间的突触连接的强度的学习过程。连接强度是基于特定神经元的输出与收到输入尖峰(即,动作电位)的相对定时来调节的。在STDP过程中,如果至某个神经元的输入尖峰平均而言倾向于紧挨在该神经元的输出尖峰之前发生,则可发生长期增强(LTP)。于是使得该特定输入在一定程度上更强。另一方面,如果输入尖峰平均而言倾向于紧接在输出尖峰之后发生,则可发生长期抑压(LTD)。于是使得该特定输入在一定程度上

更弱,并由此得名“尖峰定时依赖可塑性”。因此,使得可能是突触后神经元兴奋原因的输入甚至在将来作出贡献的可能性更大,而使得不是突触后尖峰的原因的输入在将来作出贡献的可能性更小。该过程继续,直至初始连接集合的子集保留,而所有其他连接的影响减小至无关紧要的水平。

[0053] 由于神经元一般在其许多输入都在一短时段内发生(即,累积性足以引起输出)时产生输出尖峰,因此通常保留下来的输入子集包括倾向于在时间上相关的那些输入。另外,由于在输出尖峰之前发生的输入被加强,因此提供对相关性的最早充分累积性指示的那些输入将最终变成至该神经元的最后输入。

[0054] STDP学习规则可因变于突触前神经元的尖峰时间 $t_{pre}$ 与突触后神经元的尖峰时间 $t_{post}$ 之间的时间差(即, $t=t_{post}-t_{pre}$ )来有效地适配将该突触前神经元连接到该突触后神经元的突触的突触权重。STDP的典型公式化是若该时间差为正(突触前神经元在突触后神经元之前激发)则增大突触权重(即,增强该突触),以及若该时间差为负(突触后神经元在突触前神经元之前激发)则减小突触权重(即,抑压该突触)。

[0055] 在STDP过程中,突触权重随时间推移的改变可通常使用指数式衰退来达成,如由下式给出的:

$$[0056] \Delta w(t) = \begin{cases} a_+ e^{-t/k_+} + \mu, & t > 0 \\ a_- e^{t/k_-}, & t < 0 \end{cases}, \quad (1)$$

[0057] 其中 $k_+$ 和 $k_-$ 是针对正和负时间差的时间常数, $a_+$ 和 $a_-$ 是对应的比例缩放幅值,并且 $\mu$ 是可应用于正时间差和/或负时间差的偏移。

[0058] 图3解说了根据STDP,突触权重作为突触前(presynaptic)和突触后(postsynaptic)尖峰的相对定时的函数而改变的示例性示图300。如果突触前神经元在突触后神经元之前激发,则对应的突触权重可被增大,如曲线图300的部分302中所解说的。该权重增大可被称为该突触的LTP。从曲线图部分302可观察到,LTP的量可因变于突触前和突触后尖峰时间之差而大致呈指数式地下降。相反的激发次序可减小突触权重,如曲线图300的部分304中所解说的,从而导致该突触的LTD。

[0059] 如图3中的曲线图300中所解说的,可向STDP曲线图的LTP(因果性)部分302应用负偏移 $\mu$ 。x轴的交越点306( $y=0$ )可被配置成与最大时间滞后重合以考虑到来自层 $i-1$ 的各因果性输入的相关性。在基于帧的输入(即,呈特定历时的包括尖峰或脉冲的帧的形式的输入)的情形中,可计算偏移值 $\mu$ 以反映帧边界。该帧中的第一输入尖峰(脉冲)可被视为要么如直接由突触后电位所建模地随时间衰退,要么在对神经状态的影响的意义上随时间衰退。如果该帧中的第二输入尖峰(脉冲)被视为与特定时间帧相关或有关,则该帧之前和之后的有关时间可通过使STDP曲线的一个或多个部分偏移以使得这些有关时间中的值可以不同(例如,对于大于一个帧为负,而对于小于一个帧为正)来在该时间帧边界处被分开并在可塑性意义上被不同地对待。例如,负偏移 $\mu$ 可被设为偏移LTP以使得曲线实际上在大于帧时间的pre-post时间处变得低于零并且它由此为LTD而非LTP的一部分。

[0060] 神经元模型及操作

[0061] 存在一些用于设计有用的尖峰发放神经元模型的一般原理。良好的神经元模型在以下两个计算态相(regime)方面可具有丰富的潜在行为:重合性检测和功能性计算。此外,

良好的神经元模型应当具有允许时间编码的两个要素:输入的抵达时间影响输出时间,以及重合性检测能具有窄时间窗。最后,为了在计算上是有吸引力的,良好的神经元模型在连续时间上可具有闭合形式解,并且具有稳定的行为,包括在靠近吸引子和鞍点之处。换言之,有用的神经元模型是可实践且可被用于建模丰富的、现实的且生物学一致的行为并且可被用于对神经电路进行工程设计和反向工程两者的神经元模型。

[0062] 神经元模型可取决于事件,诸如输入抵达、输出尖峰或其他事件,无论这些事件是内部的还是外部的。为了达成丰富的行为库,能展现复杂行为的状态机可能是期望的。如果事件本身的发生在撇开输入贡献(若有)的情况下能影响状态机并约束该事件之后的动态,则该系统的将来状态并非仅是状态和输入的函数,而是状态、事件和输入的函数。

[0063] 在一方面,神经元n可被建模为尖峰带漏泄积分激发神经元,其膜电压 $v_n(t)$ 由以下动态来支配:

$$[0064] \frac{dv_n(t)}{dt} = \alpha v_n(t) + \beta \sum_m w_{m,n} y_m(t - \Delta t_{m,n}), \quad (2)$$

[0065] 其中 $\alpha$ 和 $\beta$ 是参数,  $w_{m,n}$ 是将突触前神经元m连接至突触后神经元n的突触的突触权重,以及 $y_m(t)$ 是神经元m的尖峰发放输出,其可根据 $\Delta t_{m,n}$ 被延迟达树突或轴突延迟才抵达神经元n的胞体。

[0066] 应注意,从建立了对突触后神经元的充分输入的时间直至该突触后神经元实际上激发的时间存在延迟。在动态尖峰发放神经元模型(诸如Izhikevich简单模型)中,如果在去极化阈值 $v_t$ 与峰值尖峰电压 $v_{peak}$ 之间有差量,则可引发时间延迟。例如,在该简单模型中,神经元胞体动态可由关于电压和恢复的微分方程对来支配,即:

$$[0067] \frac{dv}{dt} = (k(v - v_t)(v - v_r) - u + I)/C, \quad (3)$$

$$[0068] \frac{du}{dt} = a(b(v - v_r) - u), \quad (4)$$

[0069] 其中v是膜电位,u是膜恢复变量,k是描述膜电位v的时间尺度的参数,a是描述恢复变量u的时间尺度的参数,b是描述恢复变量u对膜电位v的阈下波动的敏感度的参数, $v_r$ 是膜静息电位,I是突触电流,以及C是膜的电容。根据该模型,神经元被定义为在 $v > v_{peak}$ 时发放尖峰。

[0070] Hunzinger Cold模型

[0071] Hunzinger Cold神经元模型是能再现丰富多样的各种神经行为的最小双态尖峰发放线性动态模型。该模型的一维或二维线性动态可具有两个态相,其中时间常数(以及耦合)可取决于态相。在阈下态相中,时间常数(按照惯例为负)表示漏泄通道动态,其一般作用于以生物学一致的线性方式使细胞返回到静息。阈上态相中的时间常数(按照惯例为正)反映抗漏泄通道动态,其一般驱动细胞发放尖峰,而同时在尖峰生成中引发等待时间。

[0072] 如图4中所解说的,该模型400的动态可被划分成两个(或更多个)态相。这些态相可被称为负态相402(也可互换地称为带漏泄积分激发(LIF)态相,勿与LIF神经元模型混淆)以及正态相404(也可互换地称为抗漏泄积分激发(ALIF)态相,勿与ALIF神经元模型混淆)。在负态相402中,状态在将来事件的时间趋向于静息( $v_-$ )。在该负态相中,该模型一般

展现出时间输入检测性质及其他阈下行为。在正态相404中,状态趋向于尖峰发放事件( $v_s$ )。在该正态相中,该模型展现出计算性质,诸如取决于后续输入事件而引发发放尖峰的等待时间。在事件方面对动态进行公式化以及将动态分成这两个态相是该模型的基础特性。

[0073] 线性双态相二维动态(对于状态v和u)可按照惯例定义为:

$$[0074] \tau_\rho \frac{dv}{dt} = v + q_\rho \quad (5)$$

$$[0075] -\tau_u \frac{du}{dt} = u + r, \quad (6)$$

[0076] 其中 $q_\rho$ 和 $r$ 是用于耦合的线性变换变量。

[0077] 符号 $\rho$ 在本文中用于标示动态态相,在讨论或表达具体态相的关系时,按照惯例对于负态相和正态相分别用符号“-”或“+”来替换符号 $\rho$ 。

[0078] 模型状态由膜电位(电压) $v$ 和恢复电流 $u$ 来定义。在基本形式中,态相在本质上是由模型状态来决定的。该精确和通用的定义存在一些细微却重要的方面,但目前考虑该模型在电压 $v$ 高于阈值( $v_+$ )的情况下处于正态相404中,否则处于负态相402中。

[0079] 态相相关时间常数包括负态相时间常数 $\tau_-$ 和正态相时间常数 $\tau_+$ 。恢复电流时间常数 $\tau_u$ 通常是与态相无关的。出于方便起见,负态相时间常数 $\tau_-$ 通常被指定为反映衰退的负量,从而用于电压演变的相同表达式可用于正态相,在正态相中指数和 $\tau_+$ 将一般为正,正如 $\tau_u$ 那样。

[0080] 这两个状态元素的动态可在发生事件之际通过使状态偏离其零倾线(nullcline)的变换来耦合,其中变换变量为:

$$[0081] q_\rho = -\tau_\rho \beta u - v_\rho \quad (7)$$

$$[0082] r = \delta (v + \epsilon), \quad (8)$$

[0083] 其中 $\delta$ 、 $\epsilon$ 、 $\beta$ 和 $v_-$ 、 $v_+$ 是参数。 $v_\rho$ 的两个值是这两个态相的参考电压的基数。参数 $v_-$ 是负态相的基电压,并且膜电位在负态相中一般将朝向 $v_-$ 衰退。参数 $v_+$ 是正态相的基电压,并且膜电位在正态相中一般将趋向于背离 $v_+$ 。

[0084]  $v$ 和 $u$ 的零倾线分别由变换变量 $q_\rho$ 和 $r$ 的负数给出。参数 $\delta$ 是控制 $u$ 零倾线的斜率的比例缩放因子。参数 $\epsilon$ 通常被设为等于 $-v_-$ 。参数 $\beta$ 是控制这两个态相中的 $v$ 零倾线的斜率的电阻值。 $\tau_\rho$ 时间常数参数不仅控制指数式衰退,还单独地控制每个态相中的零倾线斜率。

[0085] 该模型可被定义为在电压 $v$ 达到值 $v_s$ 时发放尖峰。随后,状态可在发生复位事件(其可以与尖峰事件完全相同)之际被复位:

$$[0086] v = \hat{v}_- \quad (9)$$

$$[0087] u = u + \Delta u \quad (10)$$

[0088] 其中 $\hat{v}_-$ 和 $\Delta u$ 是参数。复位电压 $\hat{v}_-$ 通常被设为 $v_-$ 。

[0089] 依照瞬时耦合的原理,闭合形式解不仅对于状态是可能的(且具有单个指数项),而且对于到达特定状态的时间也是可能的。闭合形式状态解为:

$$[0090] v(t + \Delta t) = (v(t) + q_\rho) e^{\frac{\Delta t}{\tau_\rho}} - q_\rho \quad (11)$$

$$[0091] \quad u(t + \Delta t) = (u(t) + r) e^{-\frac{\Delta t}{\tau_u}} - r. \quad (12)$$

[0092] 因此,模型状态可仅在发生事件之际被更新,诸如在输入(突触前尖峰)或输出(突触后尖峰)之际被更新。还可在任何特定时间(无论是否有输入或输出)执行操作。

[0093] 而且,依照瞬时耦合原理,突触后尖峰的时间可被预计,因此到达特定状态的时间可提前被确定而无需迭代技术或数值方法(例如,欧拉数值方法)。给定了先前电压状态 $v_0$ ,直至到达电压状态 $v_f$ 之前的时间延迟由下式给出:

$$[0094] \quad \Delta t = \tau_\rho \log \frac{v_f + q_\rho}{v_0 + q_\rho}. \quad (13)$$

[0095] 如果尖峰被定义为发生在电压状态 $v$ 到达 $v_s$ 的时间,则从电压处于给定状态 $v$ 的时间起测量的直至发生尖峰前的时间量或即相对延迟的闭合形式解为:

$$[0096] \quad \Delta t_s = \begin{cases} \tau_+ \log \frac{v_s + q_+}{v + q_+} & \text{如果 } v > \hat{v}_+ \\ \infty & \text{其他} \end{cases} \quad (14)$$

[0097] 其中 $\hat{v}_+$ 通常被设为参数 $v_+$ ,但其他变型可以是可能的。

[0098] 模型动态的以上定义取决于该模型是在正态相还是负态相中。如所提及的,耦合和态相 $\rho$ 可基于事件来计算。出于状态传播的目的,态相和耦合(变换)变量可基于在上一(先前)事件的时间的状态来定义。出于随后预计尖峰输出时间的目的,态相和耦合变量可基于在下一(当前)事件的时间的状态来定义。

[0099] 存在对该Cold模型、以及在时间上执行模拟、仿真、或建模的若干可能实现。这包括例如事件-更新、步阶-事件更新、以及步阶-更新模式。事件更新是其中基于事件或“事件更新”(在特定时刻)来更新状态的更新。步阶更新是以间隔(例如,1ms)来更新模型的更新。这不一定利用迭代方法或数值方法。通过仅在事件发生于步阶处或步阶间的情况下才更新模型或即通过“步阶-事件更新”,基于事件的实现以有限的时间分辨率在基于步阶的模拟器中实现也是可能的。”

[0100] 神经元形态模型开发的上下文实时反馈

[0101] 科学家们开发大脑功能和行为的计算模型以描述神经网络的结构、连通性和行为。此过程是艰苦的,并且在模型定义与结果之间提供反馈之前存在一长时间段。为了查看是否达成期望的行为,用户可定义、构建以及运行该模型,并且此后分析该模型的行为。在一些情形中,可能花费若干小时以便寻找甚至一个简单的错误并且花费长得多的时间来寻找更复杂的错误。

[0102] 本公开的各方面涉及实时地提供上下文信息。例如,在一些方面,实时的视觉化和测试结果可在神经元形态模型的创建期间被显示。

[0103] 图5解说了根据本公开的某些方面的前述使用通用处理器502来生成神经元形态模型中的上下文反馈的示例实现500。与计算网络(神经网络)相关联的变量(神经信号)、突触权重、系统参数,延迟,频槽信息资产定义,群定义,连通性信息以及上下文信息可被存储在存储器块504中,而在通用处理器502处执行的指令可从程序存储器506中加载。在本公开的一方面,加载到通用处理器502中的指令可包括代码,该代码用于生成神经元形态模型(包括在模型的开发期间要被监视的资产)中的上下文反馈,和/或显示互动上下文面板以

便基于上下文反馈来示出表示。

[0104] 图6解说了根据本公开的某些方面的前述在神经元形态模型中生成上下文反馈的示例实现600,其中存储器602可以经由互连网络604与计算网络(神经网络)的个体(分布式)处理单元(神经处理器)606对接。与计算网络(神经网络)相关联的变量(神经信号)、突触权重、系统参数,延迟,频槽信息,资产定义,群定义,连通性信息和上下文信息可被存储在存储器602中,并且可从存储器602经由互连网络604的连接被加载到每个处理单元(神经处理器)606中。在本公开的一方面,处理单元606可被配置成生成神经元形态模型(包括在模型的开发期间要被监视的资产)中的上下文反馈,和/或被配置成显示互动上下文面板以便基于上下文反馈来示出表示。

[0105] 图7解说了前述生成神经元形态模型中的上下文反馈的示例实现700。如图7中所解说了,一个存储器组702可与计算网络(神经网络)的一个处理单元704直接对接。每个存储器组702可存储与对应的处理单元(神经处理器)704相关联的变量(神经信号)、突触权重、和/或系统参数,延迟,频槽信息,资产定义,群定义,连通性信息和上下文信息。在本公开的一方面,处理单元704可被配置成生成神经元形态模型(包括在模型的开发期间要被监视的资产)中的上下文反馈,和/或被配置成显示互动上下文面板以便基于上下文反馈来示出表示。

[0106] 图8解说了根据本公开的某些方面的神经网络800的示例实现。如图8中所解说了,神经网络800可具有多个局部处理单元802,它们可执行本文所描述的方法的各种操作。每个局部处理单元802可包括存储该神经网络的参数的局部状态存储器804和局部参数存储器806。另外,局部处理单元802可具有用于存储局部模型程序的局部(神经元)模型程序(LMP)存储器808、用于存储局部学习程序的局部学习程序(LLP)存储器810、以及局部连接存储器812。此外,如图8中所解说了,每个局部处理单元802可与用于提供对局部处理单元的局部存储器的配置的配置处理器单元814对接,并且与提供各局部处理单元802之间的路由的路由连接元件816对接。

[0107] 在一种配置中,神经元模型被配置成用于生成神经元形态模型(包括在模型的开发期间要被监视的资产)中的上下文反馈,和/或显示互动上下文面板以便基于上下文反馈来示出表示。神经元模型包括生成装置和显示装置。在一个方面,该生成装置和/或显示装置可以是被配置成执行所叙述的功能的通用处理器502、程序存储器506、存储器块504、存储器602、互连网络604、处理单元606、处理单元704、局部处理单元802、和/或路由连接处理单元816。在另一种配置中,前述装置可以是被配置成执行由前述装置所叙述的功能的任何模块或任何装置。

[0108] 根据本公开的某些方面,每个局部处理单元802可被配置成基于神经网络的期望的一个或多个功能性特征来确定神经网络的参数,以及随着所确定的参数被进一步适配、调谐、和更新来使这一个或多个功能性特征朝着期望的功能性特征发展。

[0109] 本公开涉及在神经元形态模型开发过程的所有阶段期间提供实时信息的上下文面板。在一些方面,该上下文面板可以是连同代码编辑器一起提供的用户界面。该上下文面板可被配置成随着用户输入描述(以创建)神经元形态模型的程序代码而显示实时的视觉化和测试结果。在一些方面,该上下文面板可被配置成使得响应于用户输入(选择运行一用户指定的运行时间)而在任何时间可选择性地显示视觉化和测试结果信息。如此,可向开发

者提供对神经元形态模型的实时分析,这可减少调试和开发时间。

[0110] 该上下文面板可在模型开发过程的所有阶段期间提供相关信息。在一些方面,开发过程可被划分为用于视觉化和评估上下文的三个阶段:

[0111] 1. 定义资产(例如,神经元、突触);

[0112] 2. 创建集群(例如,神经元群和突触群);以及

[0113] 3. 连接集群(例如,经由突触的群连接)。

[0114] 当然,这仅是示例性的并且不是限定性的。

[0115] 用于资产定义的上下文面板

[0116] 上下文面板可提供相对于神经元形态模型的所定义资产的视觉化。在一些方面,开发环境可自动检测资产(诸如举例而言,神经元、突触或小型网络)的定义。进而,上下文面板可基于神经元形态模型的对应代码来启动或激活具有相关的互动视觉化。在一些方面,上下文面板可被配置为一个或多个界面元素。

[0117] 上下文面板可提供一种或多种形式的上下文信息。在一些方面,上下文面板可提供与模型的动态和/或统计相关的上下文信息。上下文信息可包括迹线、图形表示、或一个或多个变量或参数随时间的值的另一指示。例如,在一些方面,上下文面板可包括:驱动神经元的输入曲线的图表、以及膜电位 $v$ 和膜恢复变量 $u$ 的标绘。当然,在上下文面板中可显示附加的或更少的变量或参数的上下文信息。

[0118] 在一些方面,视觉化可以是可调节的。在一个示例中,输入曲线可通过图表的拖放移动、基于文本的输入、以及输入操纵方案、或其他用户输入来调节。在另一示例中,用户可选择不同的输入变量、输入类型、以及输入波形类型。在调节输入图表时,可实时地调节并显示输出。由此,较大网络模型的个体神经组件可被互动地调整和验证而无需切换到用于个体神经元的独立“测试平台(test-bench)”。

[0119] 在一些方面,可基于执行神经元形态模型来更新上下文面板。例如,上下文面板可被配置成提供与模型的操作相关的统计信息(例如,神经元激发速率)。

[0120] 图9是解说根据本公开的各方面的包括可调节输入曲线的示例性上下文面板的屏幕截图900。参照图9,可经由代码编辑器902来进入定义神经网络模型(例如,资产定义)的程序代码。上下文面板910可被配置有数据视觉化特征以显示数据和仿真结果的实时视觉化。例如,如图9中所示出的,上下文面板可包括输入栏912以及输出栏920。输入栏912可包括可调节的输入曲线914。在图9的示例中,输入曲线914可通过选择和操纵输入曲线914的一个或多个指定点916来调节。当然,调节点的形式、类型、以及数目是示例性而非限制性的。

[0121] 在一些方面,输出栏920(例如,输出曲线922)可被实时地更新以反映对输入曲线的调节。

[0122] 在一些方面,双向互动可被用于提高的设计效率。例如,上下文面板中的上下文信息的基于视觉或基于测试的操纵可在代码中反映。在另一方面,可经由上下文面板中所提供的上下文信息来反映代码更新。

[0123] 用于创建集群的上下文面板

[0124] 在一些方面,上下文面板可提供与神经元集群相关的视觉化。例如,上下文面板可提供关于空间中的神经元集群的布局或布置的上下文信息(例如,统计)。在一些方面,开发

环境可自动检测集群创建并且可启动具有相关的互动视觉化的上下文面板。此外,与集群对应的代码可经由界面来显示或可包括在文件中。在一个示例中,可在用户访问(例如,工作在)代码的相应区段时(例如,当特定代码区段中存在光标或编辑器的提示时、当代码的区段集中显示在视野中时等等)显示上下文信息。

[0125] 上下文面板可显示例如3维(3-D)空间中的位置,并且可包括每个新创建的集群的标记或标签。在一些方面,可在1-D、2-D或3-D空间中定义集群/网络。标记或标签可标识以及修改集群的一部分的参数。在一个示例中,标记可修改仿真中使用的模型神经元(例如,COLD神经元或LIF神经元)。在另一示例中,标记可修改神经元集群的一部分的神经元参数。

[0126] 可通过更新定义模型的程序代码来更新模型。可经由视觉化或通过操纵视觉化来更新和反映上下文面板中的信息。类似地,还可通过操纵视觉化来更新定义模型的代码。例如,如果神经元集群中的神经元的(诸)参数被操纵或以其他方式修改,则相应的代码可被更新以反映(诸)参数中的变化。

[0127] 在一些方面,可在上下文面板中提供神经元集群的空间布局。以此方式,可视觉地验证神经元网络(例如,神经元集群)的空间布局而无需切换到另一工具。在一些方面,神经网络的空间布局可用对定义神经网络的代码的调节中所反映的对空间布局的调节来操纵。在一些方面,空间布局和对其的调节可实时地进行。

[0128] 在一些方面,上下文面板还可提供关于与神经元集群相关的硬件布局的信息。以此方式,上下文面板可提供与用于实现模型的硬件相关的统计信息和性能度量。此外,上下文面板可基于集群定义或视觉化的操纵来提供性能估计和折衷信息。在一个示例中,上下文面板可提供与功耗相关的视觉化,该功耗与神经元集群或其一部分相关。在又一示例中,上下文面板可提供与因神经元集群引起的计算负荷相关的视觉化。有了此信息,可通过操纵视觉化或通过更新程序代码区段来修改集群以便改进系统或模型效率。

[0129] 用于连接集群的上下文面板

[0130] 在一些方面,上下文面板可提供与神经元集群的连通性相关的视觉化。例如,开发环境可自动检测在神经元形态模型的各部分之间的连接(例如,突触)。此外,上下文面板可被启动具有相关互动视觉化以便实时地显示连通性信息。当然,这仅是示例性的且上下文面板可独立于具体的集群或连接定义地启动此外,上下文面板可用于视觉地定义神经网络的集群和连接并且这些定义可进而在代码(例如,在程序代码区段902中的代码)中反映。

[0131] 在一些方面,对于神经元形态模型的集群和/或其他部分的连接的代码可经由界面来显示或可包括在文件中。在一个示例中,可在用户访问(例如,工作在)代码的相应区段时(例如,当特定代码区段中存在光标或编辑器的提示时、当代码的区段集中显示在视野中时等等)显示上下文信息。

[0132] 例如,上下文面板可示出具有神经元的连接的典型神经元,并且该上下文面板可被配置用于互动连接操纵。这可以按有效且便利的方式来使得能够对神经元集群之间的连通性模式进行探察和测试。在一些方面,可实时地提供互动连接显示和连接操纵。

[0133] 图10A是解说根据本公开的各方面的示例性代码块1000的框图。框1010提供了用于定义诸如神经元之类的资产的示例。在此示例性代码块中,神经元可以是抑制性COLD神经元或兴奋性COLD神经元。当然,这仅仅是示例性的以便于解释,并且可使用任何类型的神经元。

[0134] 在框1012中提供了创建神经元集群的示例性代码。在框1012中,可使用例如在框1010中定义的神经元来创建两个不同类型的集群(例如,抑制性集群和兴奋性集群)。如此,可用空间对准来定义神经元的网格。

[0135] 框1014包括用于连接神经元集群的示例性代码。在此示例中,可以按一对一方式来连接各集群。然而,其它连接配置也是可能的。例如,可以按1到10或1到所有的配置来连接各集群。

[0136] 在一些方面,在神经元形态模型的开发期间,各代码块可被选择用于仿真和数据视觉化。在选择之际,可显示上下文面板以便提供相应的上下文信息。

[0137] 图10B解说了根据本公开的各方面的可包括在上下文面板中的示例性数据视觉化特征1050。如图10B中所示的,数据视觉化特征可以是提供时间关系中的尖峰发生的图形表示的光栅标绘1052。在另一示例中,数据视觉化特征可以是示出神经元形态模型中的神经元的瞬时活动(例如,尖峰)的活动地图1054,和/或可提供神经活动的时间平均示图的热图1056。

[0138] 在一些方面,数据视觉化特征可以是连通性地图1058。连通性地图1058可图形地解说神经元形态模型(例如,如在代码块1010、1012、和/或1014中所定义的)中的神经元(例如,1060)或一部分神经元集群的布局和连接。为了便于解说,连通性地图1058以2-D来示出。然而,这仅仅是示例性的,并且可同样地使用3-D或另一形式的视觉化。通过使用连通性地图1058,神经元形态模型可从各种视角来视觉地观察。例如,通过选择连通性地图的元素(例如,神经元),可显示神经元或神经元集群的扇入和/或扇出。

[0139] 在一些方面,神经元或至神经元的连接可经由数据视觉化特征来调节。例如,可经由连通性地图来选择神经元、神经元集群、或连接神经元的突触以便调节所选模型元素的参数。在一些方面,可禁用所选模型元素(1060)以便仿真神经元形态模型的操作而无需例如神经元或神经元集群的类型。当然,这些仅仅是数据视觉化特征的示例性形式并且也可利用视觉化特征的其他类型和/或组合。

[0140] 此外,通过操纵数据视觉化特征,可更新相应代码。在一些方面,数据视觉化和代码更新可实时地进行。

[0141] 图11解说了根据本公开的各方面的示出用于生成神经元形态模型中的上下文反馈的架构1100的框图。该架构包括集成开发引擎(IDE)1102、编译器1104、服务器1106以及执行引擎1108。IDE 1102可用于生成定义神经元形态模型的模型种类集合。可经由编译器1104来编译该模型种类集合。在一些方面,要被调谐的单个种类实例也可被提供给编译器1104。

[0142] 经编译对象可被提供给服务器1106。经编译对象的仿真可被加载在执行引擎1108上。随着仿真被加载,可调节神经元形态模型的参数。进而,经更新的参数可被提供给服务器1106并且实时地在仿真中反映。

[0143] 在一些方面,仿真的结果可被提供给服务器1106。在其他方面,仿真结果可被提供给IDE 1102以用于定义神经元形态模型的模型种类的调节。

[0144] 图12解说了生成神经元形态模型中的上下文反馈的方法1200。在框1202中,神经元模型生成神经元形态模型中的上下文反馈,该神经元形态模型包括在模型的开发期间要被监视的资产。此外,在框1204中,神经元模型显示互动上下文面板以便基于上下文反馈来

示出表示。

[0145] 在一些方面,可实时地提供该表示。在其他方面,该方法可进一步包括基于执行模型来更新互动上下文面板。在又一些方面,该方法可进一步包括操纵上下文面板以更新对应于模型的代码。在又一些方面,该方法可包括更新对应于模型的代码以更新上下文面板。

[0146] 以上所描述的方法的各种操作可由能够执行相应功能的任何合适的装置来执行。这些装置可包括各种硬件和/或软件组件和/或模块,包括但不限于电路、专用集成电路(ASIC)、或处理器。一般而言,在附图中有解说的操作的场合,那些操作可具有带相似编号的相应配对装置加功能组件。

[0147] 如本文所使用的,术语“确定”涵盖各种各样的动作。例如,“确定”可包括演算、计算、处理、推导、研究、查找(例如,在表、数据库或其他数据结构中查找)、查明、及类似动作。另外,“确定”可包括接收(例如接收信息)、访问(例如访问存储器中的数据)、及类似动作。而且,“确定”可包括解析、选择、选取、确立及类似动作。

[0148] 如本文所使用的,引述一列项目中的“至少一个”的短语是指这些项目的任何组合,包括单个成员。作为示例,“a、b或c中的至少一个”旨在涵盖:a、b、c、a-b、a-c、b-c、以及a-b-c。

[0149] 结合本公开描述的各种解说性逻辑框、模块、以及电路可用设计成执行本文中描述的功能的通用处理器、数字信号处理器(DSP)、专用集成电路(ASIC)、现场可编程门阵列(FPGA)或其他可编程逻辑器件(PLD)、分立的门或晶体管逻辑、分立的硬件组件、或其任何组合来实现或执行。通用处理器可以是微处理器,但在替换方案中,处理器可以是任何市售的处理器、控制器、微控制器、或状态机。处理器还可以被实现为计算设备的组合,例如DSP与微处理器的组合、多个微处理器、与DSP核心协同的一个或多个微处理器、或任何其它此类配置。

[0150] 结合本公开所描述的方法或过程的步骤可直接在硬件中、在由处理器执行的软件模块中、或在这两者的组合中体现。软件模块可驻留在本领域所知的任何形式的存储介质中。可使用的存储介质的一些示例包括随机存取存储器(RAM)、只读存储器(ROM)、闪存、可擦除可编程只读存储器(EPROM)、电可擦除可编程只读存储器(EEPROM)、寄存器、硬盘、可移动盘、CD-ROM等等。软件模块可包括单条指令、或许多条指令,且可分布在若干不同的代码段上,分布在不同的程序间以及跨多个存储介质分布。存储介质可被耦合到处理器以使得该处理器能从/向该存储介质读写信息。在替换方案中,存储介质可以被整合到处理器。

[0151] 本文所公开的方法包括用于实现所描述的方法的一个或多个步骤或动作。这些方法步骤和/或动作可以彼此互换而不会脱离权利要求的范围。换言之,除非指定了步骤或动作的特定次序,否则具体步骤和/或动作的次序和/或使用可以改动而不会脱离权利要求的范围。

[0152] 所描述的功能可在硬件、软件、固件或其任何组合中实现。如果以硬件实现,则示例硬件配置可包括设备中的处理系统。处理系统可以用总线架构来实现。取决于处理系统的具体应用和整体设计约束,总线可包括任何数目的互连总线和桥接器。总线可将包括处理器、机器可读介质、以及总线接口的各种电路链接在一起。总线接口可用于尤其将网络适配器等经由总线连接至处理系统。网络适配器可用于实现信号处理功能。对于某些方面,用户接口(例如,按键板、显示器、鼠标、操纵杆,等等)也可以被连接到总线。总线还可以链接

各种其他电路,诸如定时源、外围设备、稳压器、功率管理电路以及类似电路,它们在本领域中是众所周知的,因此将不再进一步描述。

[0153] 处理器可负责管理总线和一般处理,包括执行存储在机器可读介质上的软件。处理器可用一个或多个通用和/或专用处理器来实现。示例包括微处理器、微控制器、DSP处理器、以及其他能执行软件的电路系统。软件应当被宽泛地解释成意指指令、数据、或其任何组合,无论是被称作软件、固件、中间件、微代码、硬件描述语言、或其他。作为示例,机器可读介质可包括随机存取存储器(RAM)、闪存、只读存储器(ROM)、可编程只读存储器(PROM)、可擦式可编程只读存储器(EPROM)、电可擦式可编程只读存储器(EEPROM)、寄存器、磁盘、光盘、硬驱动器、或者任何其他合适的存储介质、或其任何组合。机器可读介质可被实施在计算机程序产品中。该计算机程序产品可以包括包装材料。

[0154] 在硬件实现中,机器可读介质可以是处理系统中与处理器分开的一部分。然而,如本领域技术人员将容易领会的,机器可读介质或其任何部分可在处理系统外部。作为示例,机器可读介质可包括传输线、由数据调制的载波、和/或与设备分开的计算机产品,所有这些都可由处理器通过总线接口来访问。替换地或补充地,机器可读介质或其任何部分可被集成到处理器中,诸如高速缓存和/或通用寄存器文件可能就是这种情形。虽然所讨论的各种组件可被描述为具有特定位置,诸如局部组件,但它们也可按各种方式来配置,诸如某些组件被配置成分布式计算系统的一部分。

[0155] 处理系统可以被配置为通用处理系统,该通用处理系统具有一个或多个提供处理器功能性的微处理器、以及提供机器可读介质中的至少一部分的外部存储器,它们都通过外部总线架构与其他支持电路系统链接在一起。替换地,该处理系统可以包括一个或多个神经元形态处理器以用于实现本文所述的神经元模型和神经系统模型。作为另一替换方案,处理系统可以用带有集成在单块芯片中的处理器、总线接口、用户接口、支持电路系统、和至少一部分机器可读介质的专用集成电路(ASIC)来实现,或者用一个或多个现场可编程门阵列(FPGA)、可编程逻辑器件(PLD)、控制器、状态机、门控逻辑、分立硬件组件、或者任何其他合适的电路系统、或者能执行本公开通篇所描述的各种功能性的电路的任何组合来实现。取决于具体应用和加诸于整体系统上的总设计约束,本领域技术人员将认识到如何最佳地实现关于处理系统所描述的功能性。

[0156] 机器可读介质可包括数个软件模块。这些软件模块包括当由处理器执行时使处理系统执行各种功能的指令。这些软件模块可包括传送模块和接收模块。每个软件模块可以驻留在单个存储设备中或者跨多个存储设备分布。作为示例,当触发事件发生时,可以从硬驱动器中将软件模块加载到RAM中。在软件模块执行期间,处理器可以将一些指令加载到高速缓存中以提高访问速度。随后可将一个或多个高速缓存行加载到通用寄存器文件中以供处理器执行。在参照以下述及软件模块的功能性时,将理解此类功能性是在处理器执行来自该软件模块的指令时由该处理器来实现的。

[0157] 如果以软件实现,则各功能可作为一条或多条指令或代码存储在计算机可读介质上或藉其进行传送。计算机可读介质包括计算机存储介质和通信介质两者,这些介质包括促成计算机程序从一地向另一地转移的任何介质。存储介质可以是能被计算机访问的任何可用介质。作为示例而非限定,此类计算机可读介质可包括RAM、ROM、EEPROM、CD-ROM或其他光盘存储、磁盘存储或其他磁存储设备、或能用于携带或存储指令或数据结构形式的期望

程序代码且能被计算机访问的任何其他介质。另外,任何连接也被正当地称为计算机可读介质。例如,如果软件是使用同轴电缆、光纤电缆、双绞线、数字订户线(DSL)、或无线技术(诸如红外(IR)、无线电、以及微波)从web网站、服务器、或其他远程源传送而来,则该同轴电缆、光纤电缆、双绞线、DSL或无线技术(诸如红外、无线电、以及微波)就被包括在介质的定义之中。如本文中所使用的盘(disk)和碟(disc)包括压缩碟(CD)、激光碟、光碟、数字多用碟(DVD)、软盘、和蓝光<sup>®</sup>碟,其中盘(disk)常常磁性地再现数据,而碟(disc)用激光来光学地再现数据。因此,在一些方面,计算机可读介质可包括非瞬态计算机可读介质(例如,有形介质)。另外,对于其他方面,计算机可读介质可包括瞬态计算机可读介质(例如,信号)。上述的组合应当也被包括在计算机可读介质的范围内。

[0158] 因此,某些方面可包括用于执行本文中给出的操作的计算机程序产品。例如,此类计算机程序产品可包括其上存储(和/或编码)有指令的计算机可读介质,这些指令能由一个或多个处理器执行以执行本文中所描述的操作。对于某些方面,计算机程序产品可包括包装材料。

[0159] 此外,应当领会,用于执行本文中所描述的方法和技术的模块和/或其它恰适装置能由用户终端和/或基站在适用的场合下载和/或其他方式获得。例如,此类设备能被耦合至服务器以促成用于执行本文中所描述的方法的装置的转移。替换地,本文所述的各种方法能经由存储装置(例如,RAM、ROM、诸如压缩碟(CD)或软盘等物理存储介质等)来提供,以使得一旦将该存储装置耦合至或提供给用户终端和/或基站,该设备就能获得各种方法。此外,可利用适于向设备提供本文所描述的方法和技术的任何其他合适的技术。

[0160] 将理解,权利要求并不被限定于以上所解说的精确配置和组件。可在上述方法和装置的布置、操作和细节上作出各种修改、变更和变型而不会背离权利要求的范围。

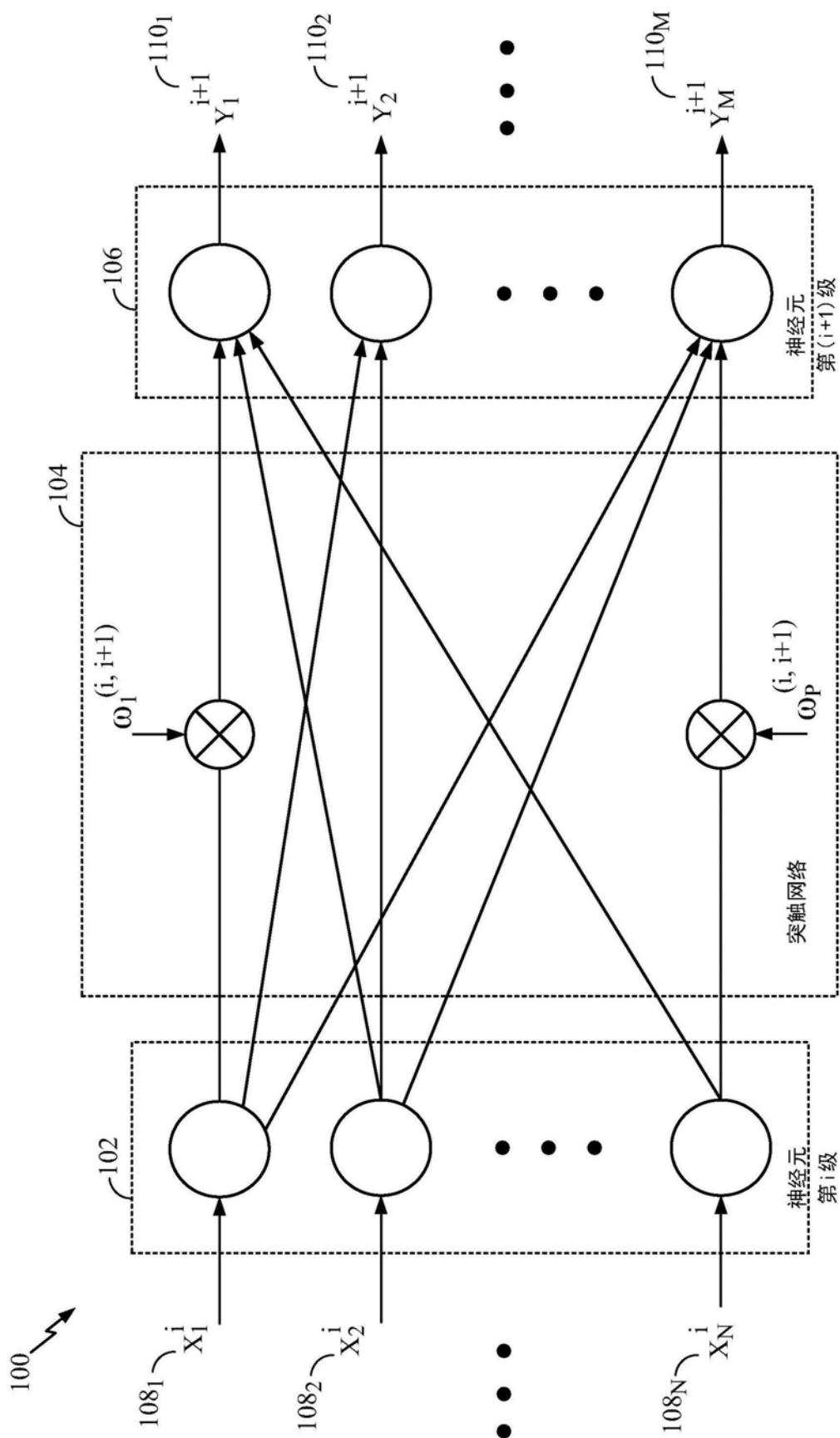


图1

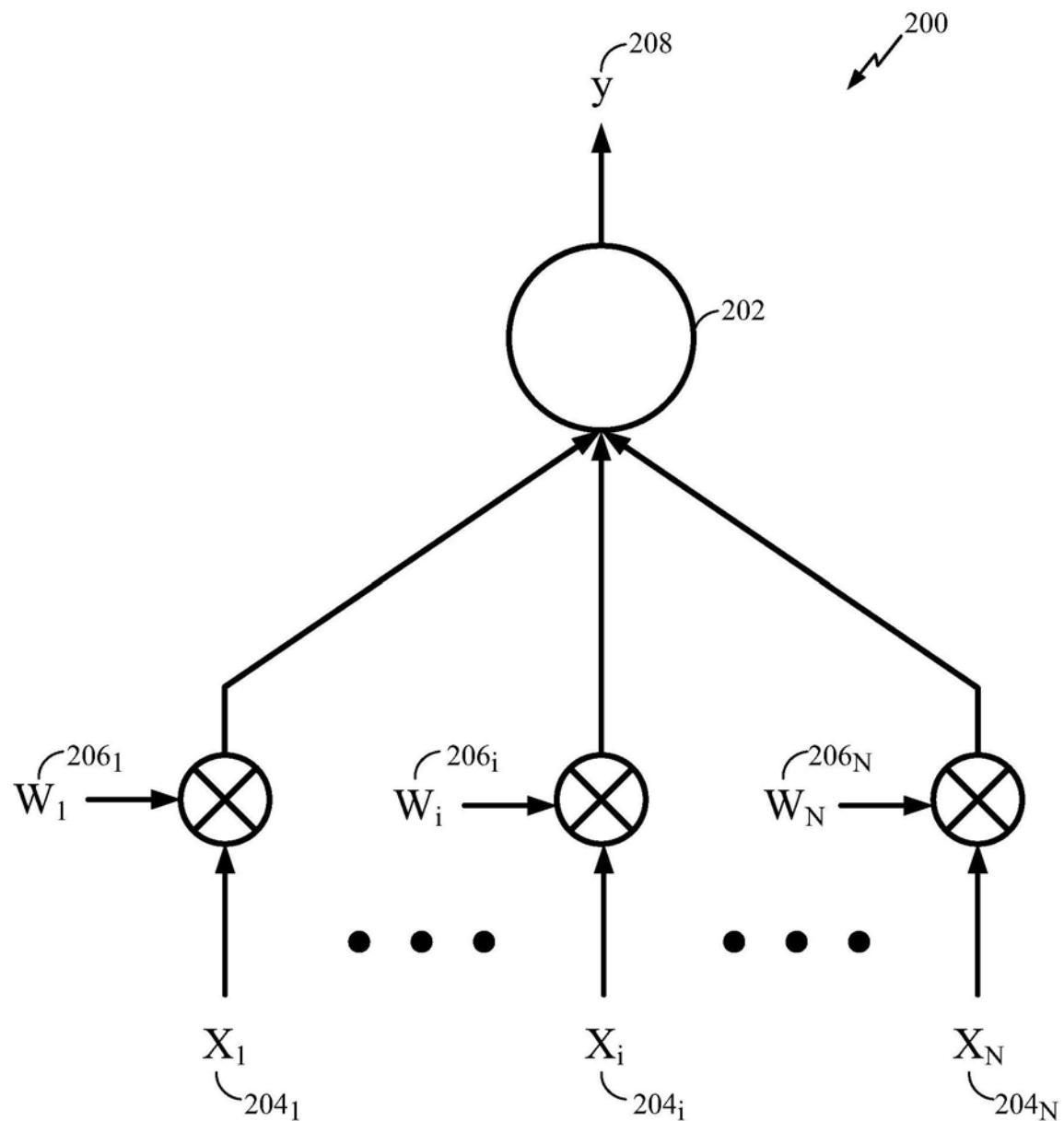


图2

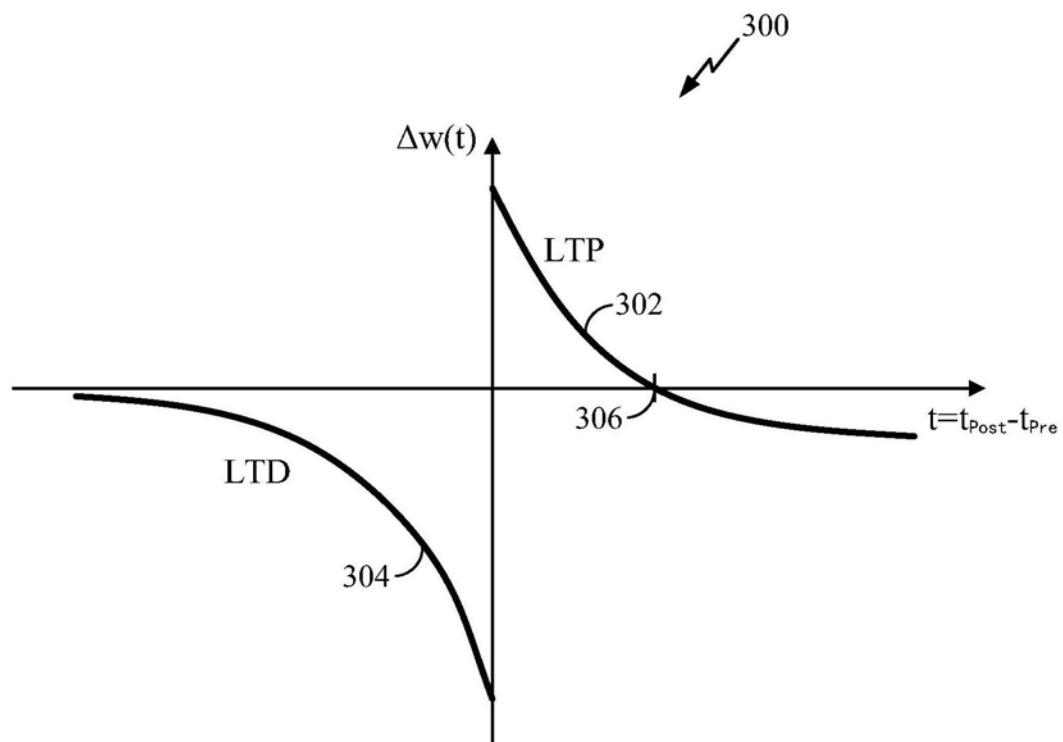


图3

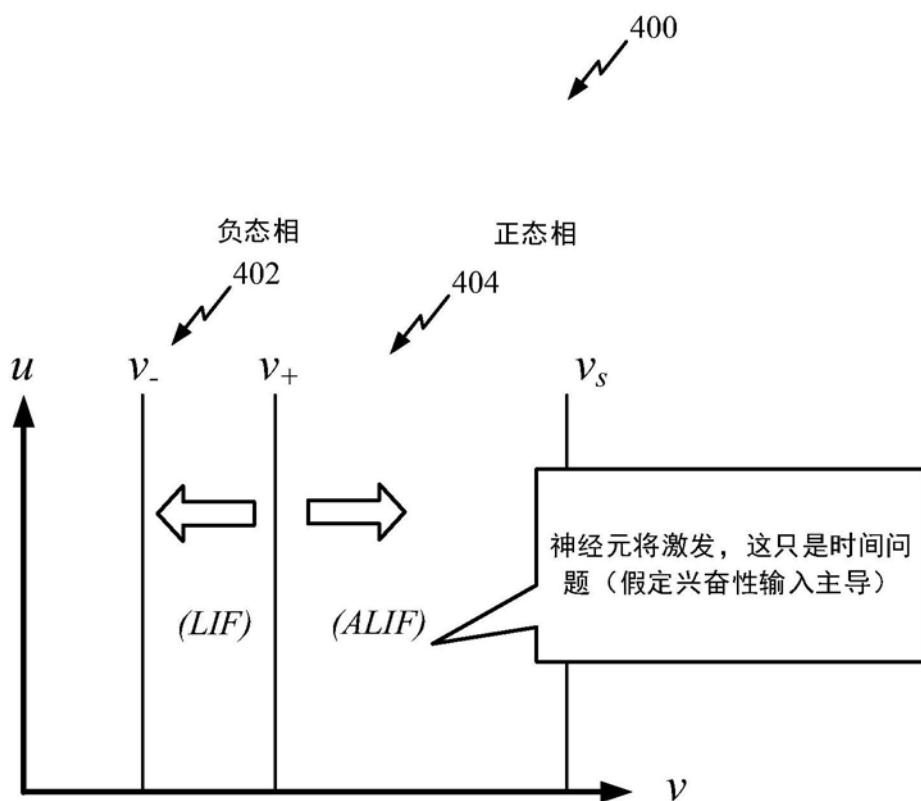


图4

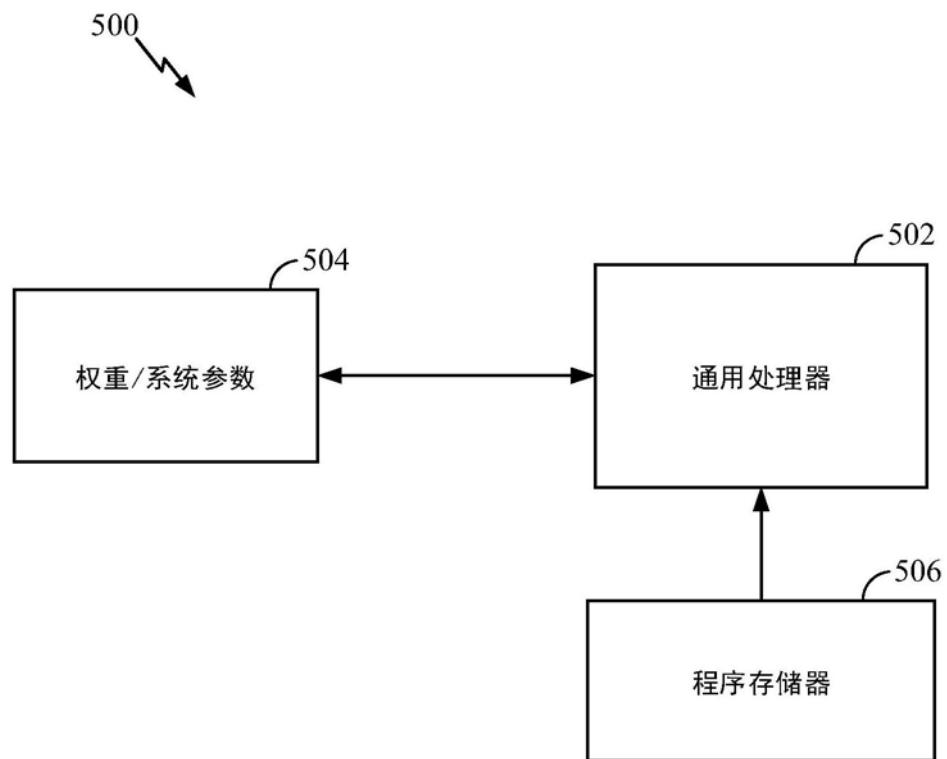


图5

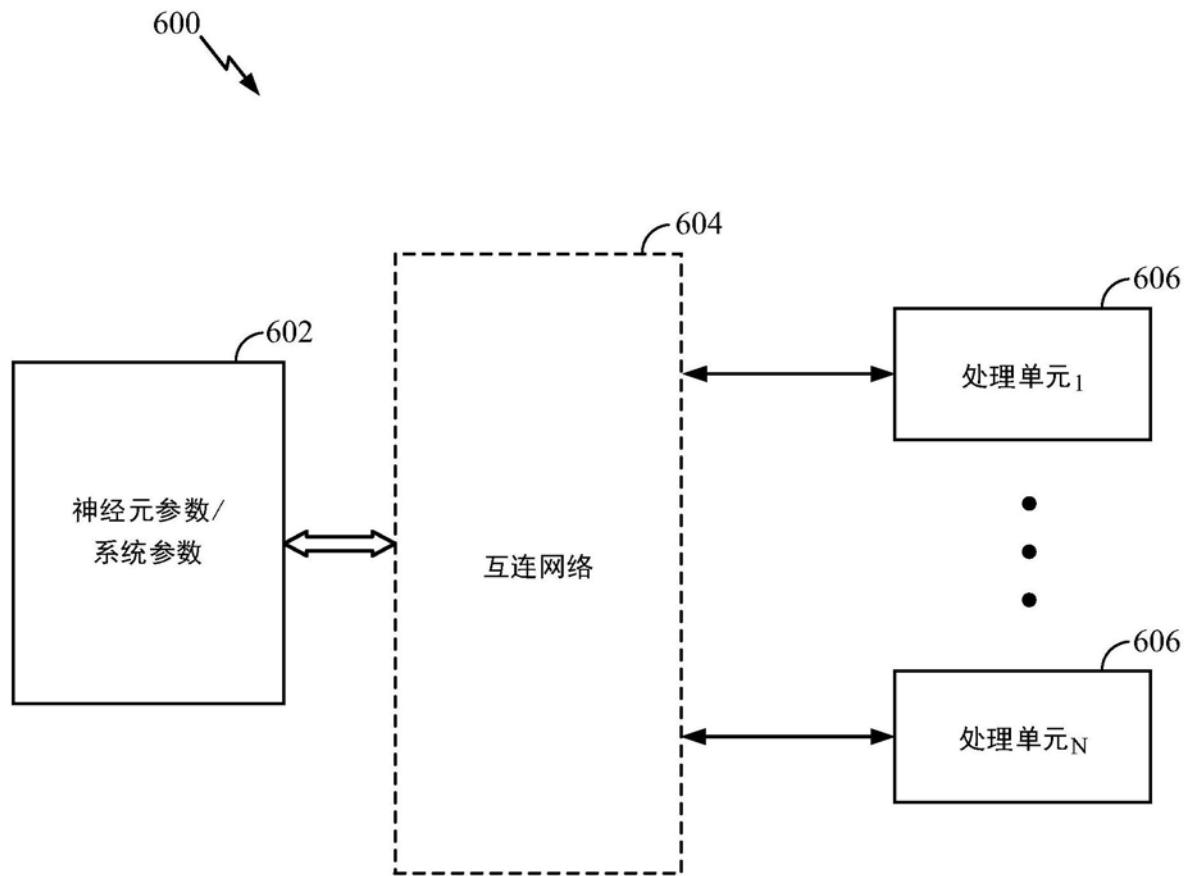


图6

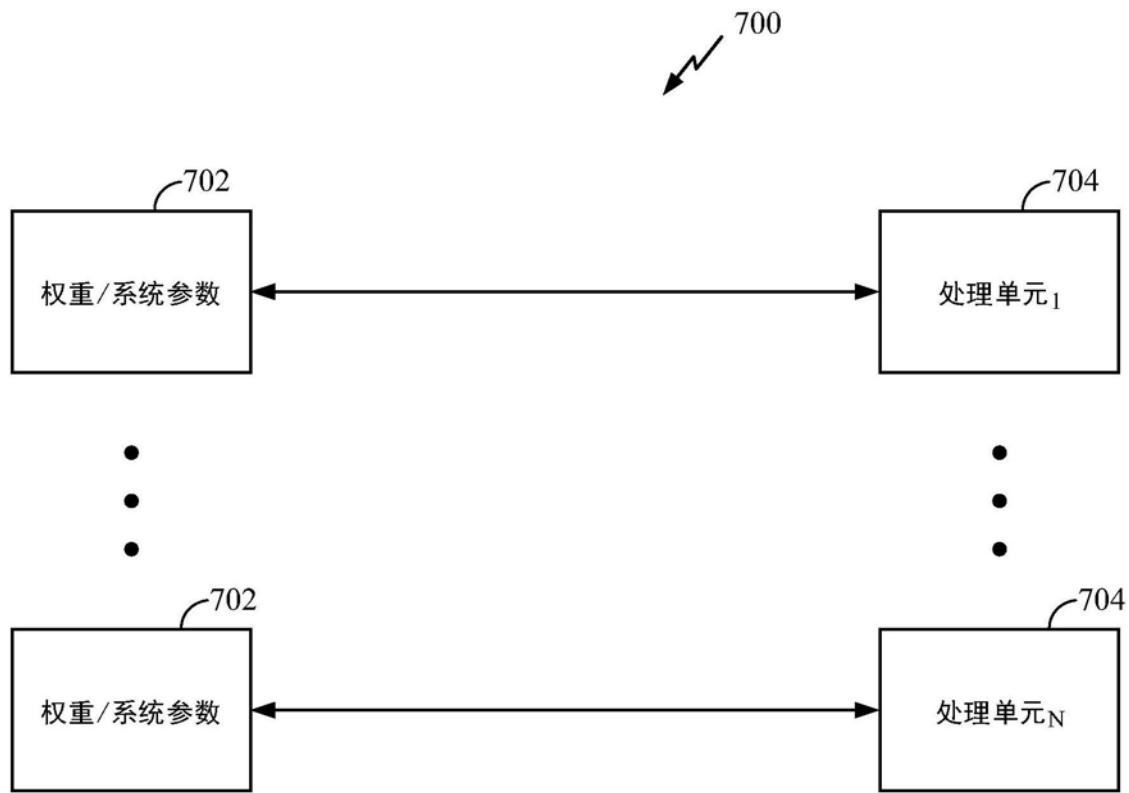


图7

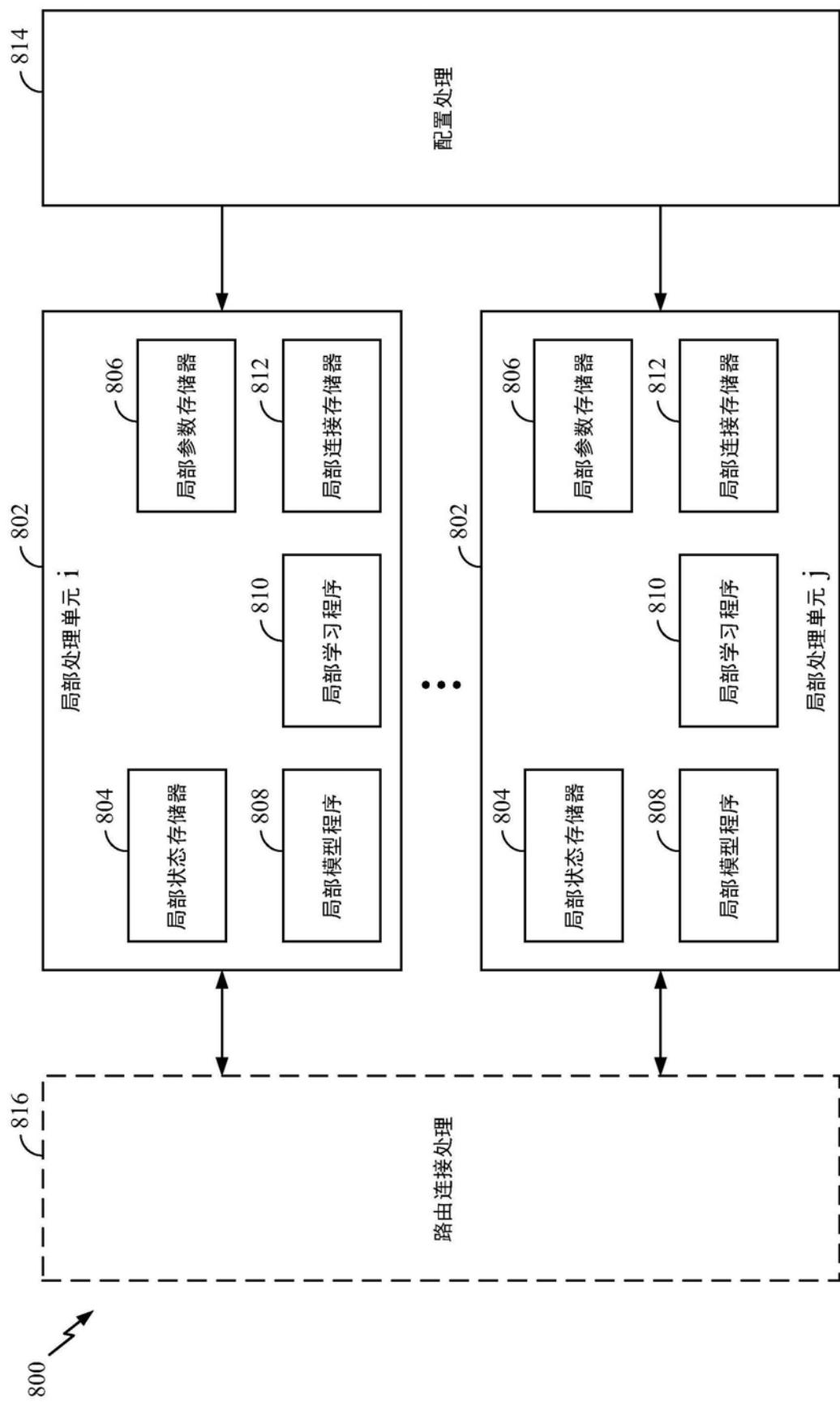


图8

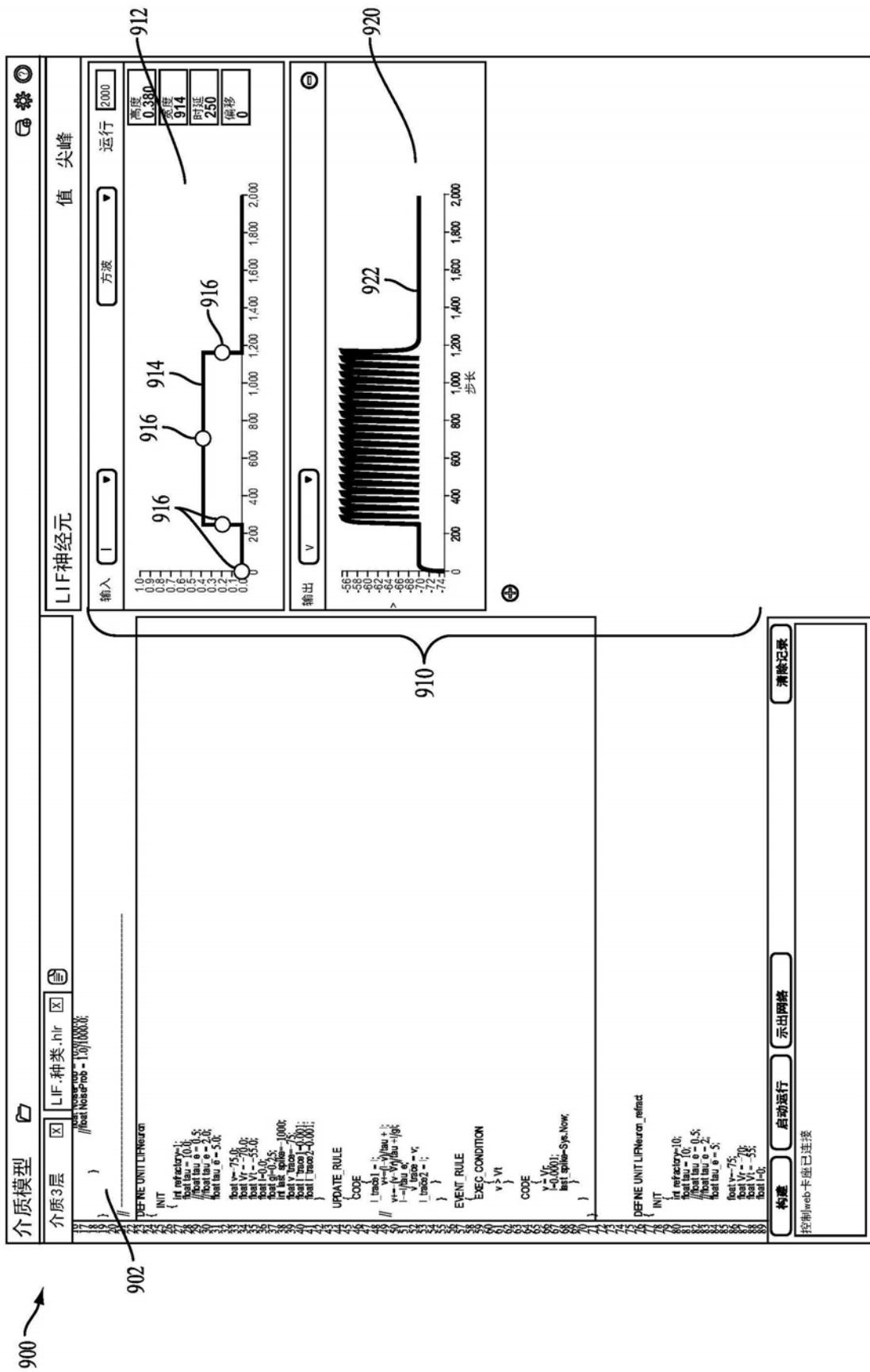


图9

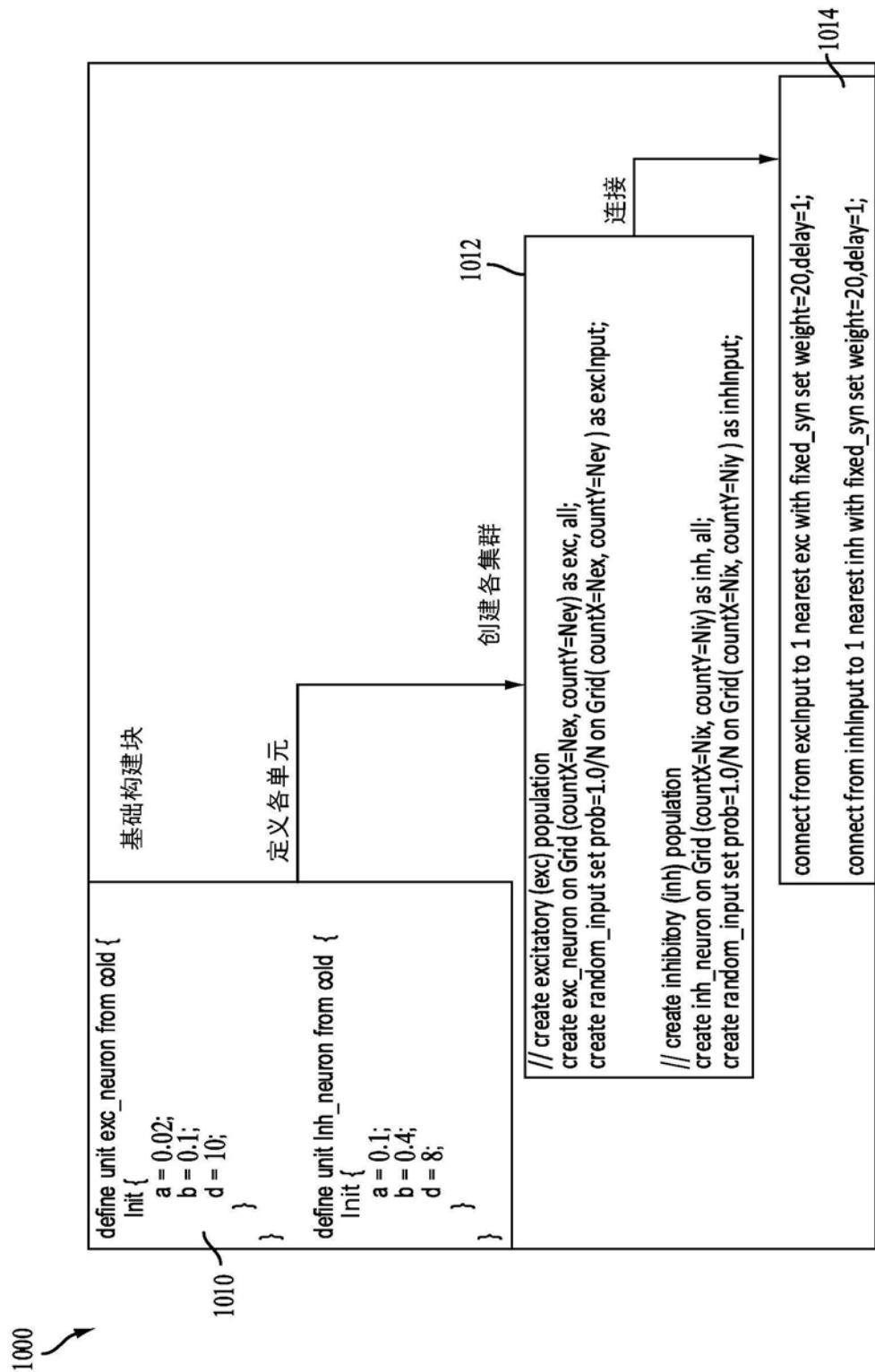
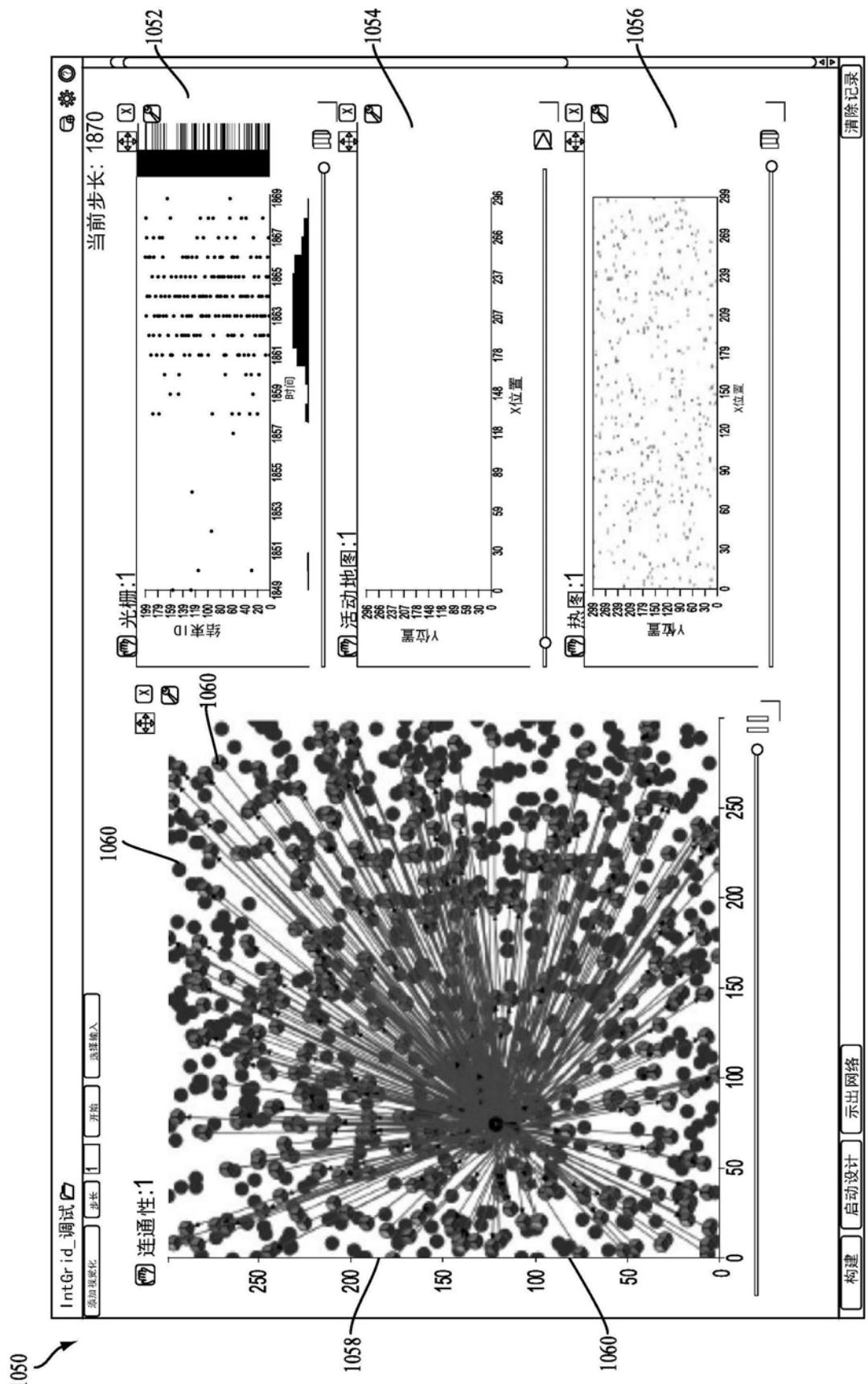


图10A



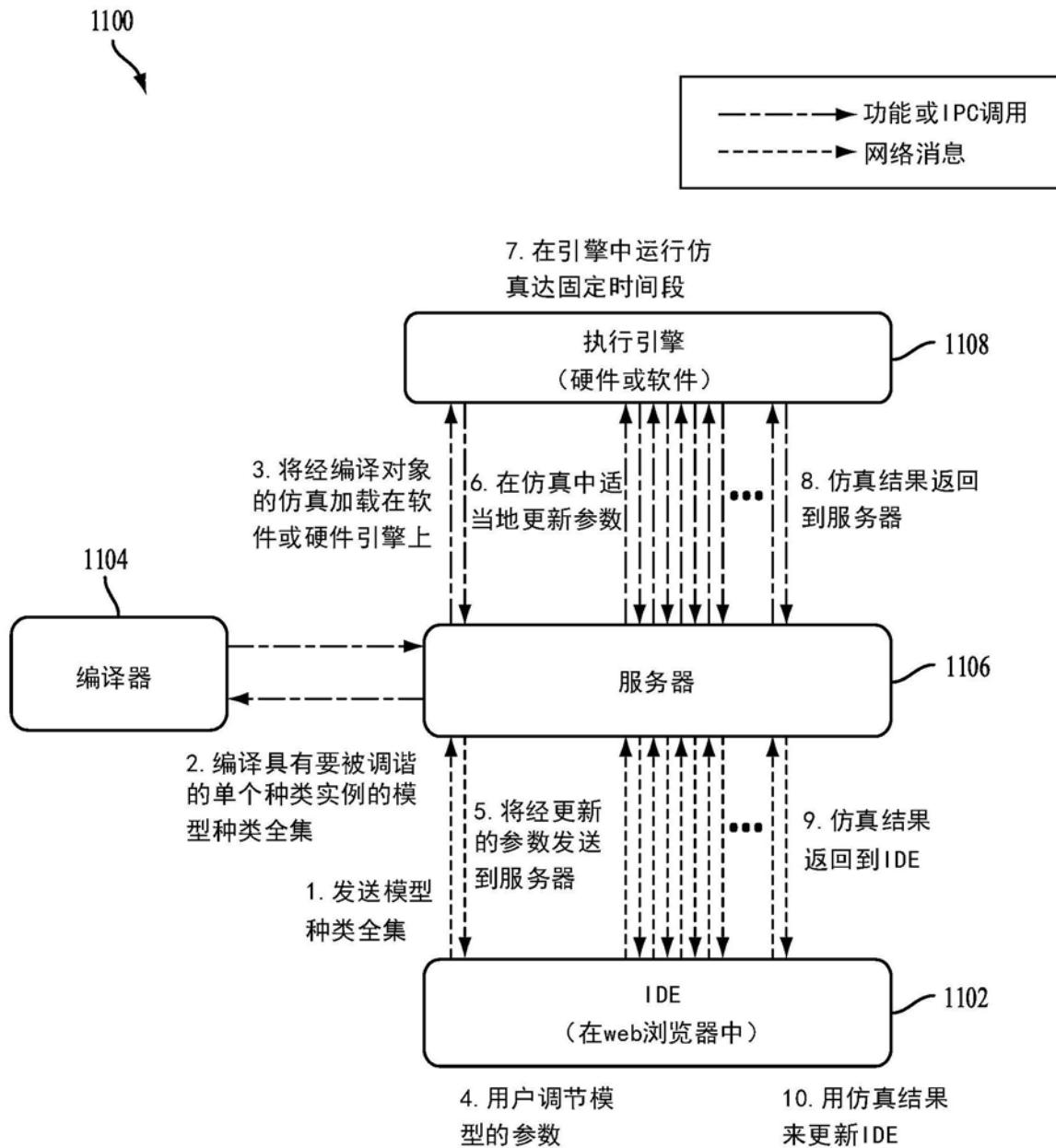


图11

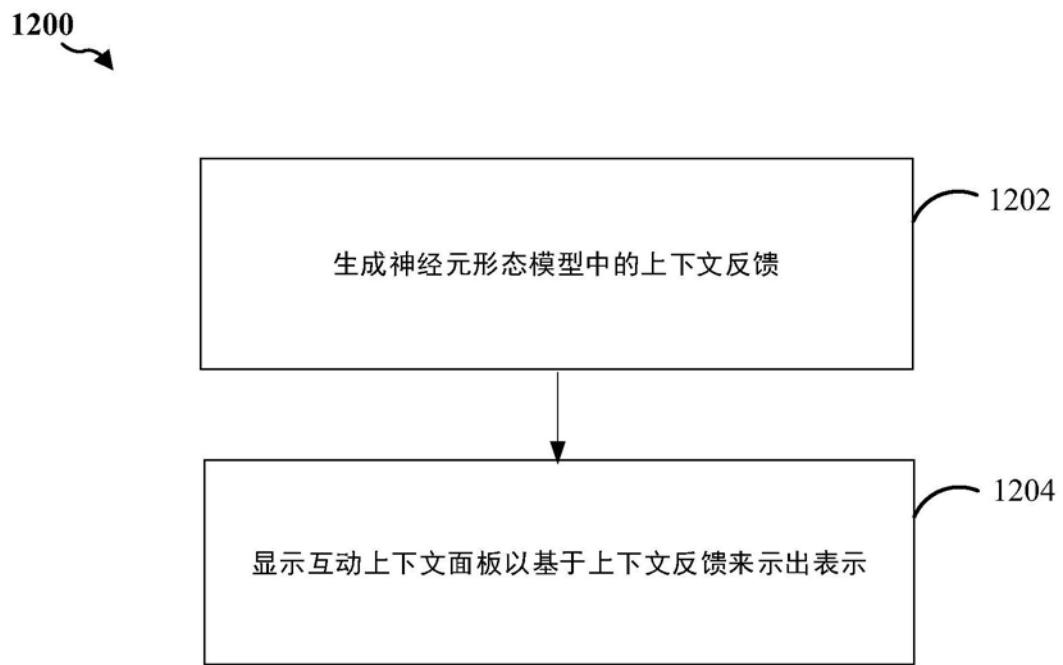


图12