



(12)发明专利申请

(10)申请公布号 CN 109902807 A

(43)申请公布日 2019.06.18

(21)申请号 201910148729.6

(22)申请日 2019.02.27

(71)申请人 电子科技大学

地址 611731 四川省成都市高新区(西区)  
西源大道2006号

(72)发明人 王海 肖涛 唐迪娅

(51)Int.Cl.

G06N 3/04(2006.01)

G06N 3/063(2006.01)

G06N 3/08(2006.01)

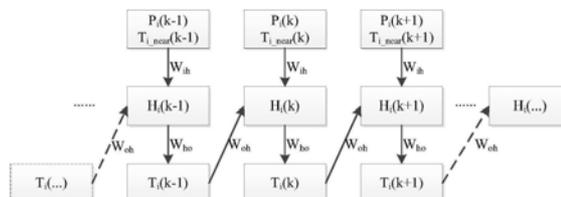
权利要求书1页 说明书3页 附图2页

(54)发明名称

一种基于循环神经网络的众核芯片分布式热建模方法

(57)摘要

本发明属于电子设计自动化领域,公开了一种基于循环神经网络的众核芯片分布式热建模方法。动态热管理可以非常有效地管理众核芯片的温度,而一个好的众核芯片热建模可以很好地辅助动态热管理。然而传统的众核芯片集总式热建模,随着芯片核心数的增加,它的计算开销呈指数增长。为了解决集总式热模型计算开销过大的问题,本发明提出了一种基于循环神经网络的众核芯片分布式热建模方法,它把芯片的每个核作为单独的计算单元,来建立循环神经网络模型,核与核之间进行有限的的数据交换。本发明能够以相当快的速度和很高的精度来模拟众核芯片的温度特性。



1. 一种基于循环神经网络的众核芯片分布式热建模方法,其特征在于:对众核芯片每个核进行热建模;利用循环神经网络来建立每个核的热模型;对众核芯片的每个核,其周围温度的选择方式固定以减少误差。

2. 根据权利要求1所述的基于循环神经网络的众核芯片分布式热建模方法,其特征在于:对众核芯片每个核进行热建模,芯片核心的位置不同,其热建模也不同;给定芯片核心的位置,它的温度可以根据自身的功率和周围核的温度计算出来。

3. 根据权利要求1所述的基于循环神经网络的众核芯片分布式热建模方法,其特征在于:可以利用循环神经网络很好地模拟非线性函数;可以利用循环神经网络处理向量序列的数据;循环神经网络的输入层,进行了权值拆解,以解决输入既有功率又有温度的问题。

4. 根据权利要求1所述的基于循环神经网络的众核芯片分布式热建模方法,其特征在于:对众核芯片的每个核,其周围温度的选择方式固定以减少误差,不是按照核的编号从小到大排列的,而是正上方为第一个,然后顺时针旋转,直到取完所有相邻核的温度,这样在处理最外围核的时候,才不容易出错,因为他们与外部环境相邻,并且相邻的位置还有区别。

5. 根据权利要求1所述的基于循环神经网络的众核芯片分布式热建模方法,其特征在于:循环神经网络的输入层,进行了权值拆解,以解决输入既有功率又有温度的温度,把输入权值矩阵拆成两部分,一部分给功率,另一部分给温度。

## 一种基于循环神经网络的众核芯片分布式热建模方法

### 技术领域

[0001] 本发明属于电子设计自动化领域,涉及深度学习技术领域,特别涉及一种基于循环神经网络的众核芯片分布式热建模方法。

### 背景技术

[0002] 随着半导体工艺的进步,芯片特征尺寸持续减小,到2018年,7nm的商业芯片已经开始量产。进入纳米量级后,由于漏电流的影响,芯片的主频很难再提高,因此芯片的高性能发展方向由增加芯片核心数代替提高主频,并取得显著成效。

[0003] 这种众核芯片的性能因为核心数的增加而得到很大的提升,但同时也带来了严重的芯片热可靠性问题,出现这个问题的主要原因是功率密度大导致的芯片温度过高。

[0004] 为了解决众核芯片热可靠性问题,动态热管理这个有效并且成本较低的方案被提出。这种方案基于自动控制理论,通过对功耗的准确估算和实时的调整,从而得到理想的温度分布。尤其当核心数目较少的时候,动态热管理技术能以较低的性能开销确保温度得到有效的管理。然而,当众核芯片核心数目过多,集总式热模型尺寸过大,计算开销随着芯片核心数的增加呈指数增长,这导致了热管理本身带来的处理器性能开销过大。

[0005] 针对以上问题,提出一种众核芯片分布式热建模是目前急待解决的问题之一。

### 发明内容

[0006] 为了解决上述技术中的问题,本发明提供了一种基于循环神经网络的众核芯片分布式热建模方法。这种建模方法把众核芯片热模型,分解为若干个小模型,更为极端的例子就是为众核芯片的每个核建立一个热模型,核与核之间进行有限的信息交换。这种方法首先搭建循环神经网络模型,然后通过离线的温度和功率数据来训练网络。训练完成的循环神经网络可以预测芯片上每个核的温度。

[0007] 本发明采用以下技术方案解决上述问题:

步骤一,从Hotspot中提取众核芯片热模型参数,主要是整个芯片上的热电容和热电阻参数,建立众核芯片热模型。

[0008] 步骤二,利用这些热模型获得了多组数据(多个时间节点,每个节点都有每个核的功率和温度信息),然后把这多组数据做成训练集和验证集,其中训练集将被用来训练循环神经网络,验证集只进行验证训练好的神经网络,验证集数据不用于训练。

[0009] 步骤三,把训练集送入还未训练的循环神经网络模型(里面的权值矩阵随机初始化),可以获得循环神经网络的输出。因为循环神经网络还未被训练,所以循环神经网络的输出和真实的输出还有较大的差距。为了得到一个精确的循环神经网络模型,可以通过调整权值矩阵,使它的温度输出尽可能接近训练集的输出。所以,目标就变成了,通过调整循环神经网络的权值矩阵来最小化损失函数,损失函数越小,循环神经网络的输出和真实的输出就越接近。

[0010] 步骤四,采取梯度下降优化算法来优化损失函数,设定好学习率分别计算损失函

数对循环神经网络的权值矩阵的偏导,然后对他们进行迭代更新。经过多次迭代,直到损失函数不再降低或达到我们设定的最大迭代次数,就表示训练完成,记录此时的损失函数值,即训练误差。同时,还会把验证集送进训练好的循环神经网络,在验证集上,不更新迭代,只记录其损失函数值,即验证误差。然后改变隐藏层的层数,以及隐藏层神经元的个数,重新训练新模型,记录其训练误差和验证误差。最终,从这些模型中,选出验证误差最小的那个,作为芯片核的热模型。对于众核芯片上的每一个核,都要为其训练一个热模型,组合起来,便是众核芯片分布式热模型。

[0011] 与现有技术相比,本发明的有益结果是:循环神经网络可以有效地拟合非线性函数,使得利用它建立的众核芯片分布式热模型能够以很高的准确率和很快的响应速度来模拟众核芯片的温度特性。

### 附图说明

[0012] 下面结合附图和实施例对本发明进一步说明。

[0013] 图1为16核芯片的布局、编号以及第6核的位置图。

[0014] 图2为第6核的热模型结构以及与其他相邻核的位置关系图。

[0015] 图3为循环神经网络结构图,它的循环结构是从输出层到隐藏层。

[0016] 图4为第6核热模型预测温度值和真实温度值的比较图。

### 具体实施方式

[0017] 为使本发明的目的、技术方案和优点更加清楚,下面将结合本发明实例中的附图,对本发明实例中的技术方案进行清楚、完整地描述,显然,所描述的实例是本发明一部分实例,而不是全部的实例。基于本发明中的实例,本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实例,都属于本发明保护的范围。

[0018] 图1为16核芯片的布局编号以及第6核的位置图。

[0019] 在本发明实例中,提供了一个16核的众核芯片,其编号如图1所示,整个众核芯片分布式热模型要给每个核都建一个模型,然后组合起来,就是整个众核芯片的分布式热模型。这里第6核颜色突出,将以第6核为例子,讲解建立模型的过程。

[0020] 图2为第6核的热模型结构以及与其他相邻核的位置关系图。

[0021] 画出了第6核的热模型结构以及与其他相邻核的位置关系,这里它和第2、5、7、10核相邻,用热电阻连接;还有接地热电容;还有外部输入功率。

[0022] 图3为循环神经网络结构图,考虑到非线性效应,本发明用神经网络来给每个核建立热模型,又因为温度在时间上有一系列的值,构成向量序列,而循环神经网络是一个专门从事序列建模的神经网络,它能够很好地处理向量序列形式的数据,所以这里最终采用循环神经网络来给每个核建立热模型。这里,输入为 $P_i(k)$ 和 $T_{i\_near}(k)$ ,分别表示 $k$ 时刻的第 $i$ 核的功率以及与第 $i$ 核相邻的核的温度,状态 $H_i(k)$ 在循环神经网络中也被叫做隐藏层,表示 $k$ 时刻第 $i$ 核的状态;输出 $T_i(k)$ 表示 $k$ 时刻第 $i$ 核的温度。 $W_{ih}$ 是输入层到隐藏层的权值矩阵, $W_{ho}$ 是隐藏层到输出层的权值矩阵, $W_{oh}$ 是输出层到隐藏层的权值矩阵。为了使循环神经网络能够更好地拟合函数,隐藏层可以有多层,这里展示的是只含一个隐藏层的循环神经网络。

[0023] 图4为第6核热模型预测温度值和真实温度值的比较图。

[0024] 将本发明的训练出来的第6核的基于循环神经网络的热模型用来预测第6核的温度,可以看到,模型预测的温度可以很好地拟合真实温度。

[0025] 本发明公开了一种基于循环神经网络的众核芯片分布式热模型方法,以上实例对本发明进行了详细的说明,但是并不局限于此,之后依然可以对之前实例所记载的技术方案进行修改,这并不会使相应技术方案的本质脱离本发明各实例技术方案的精神和范围。

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

图1

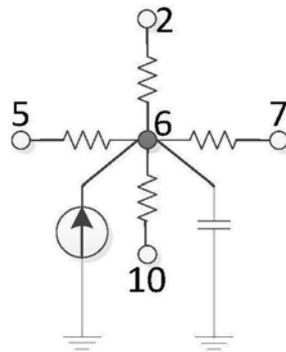


图2

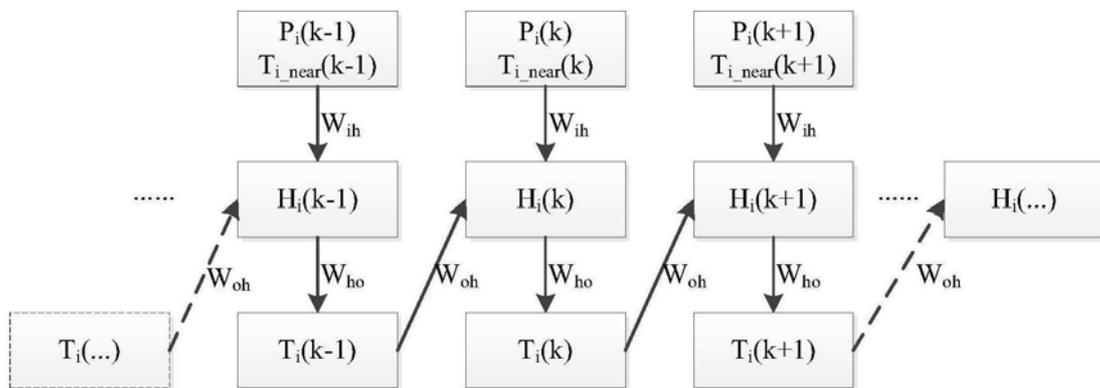


图3

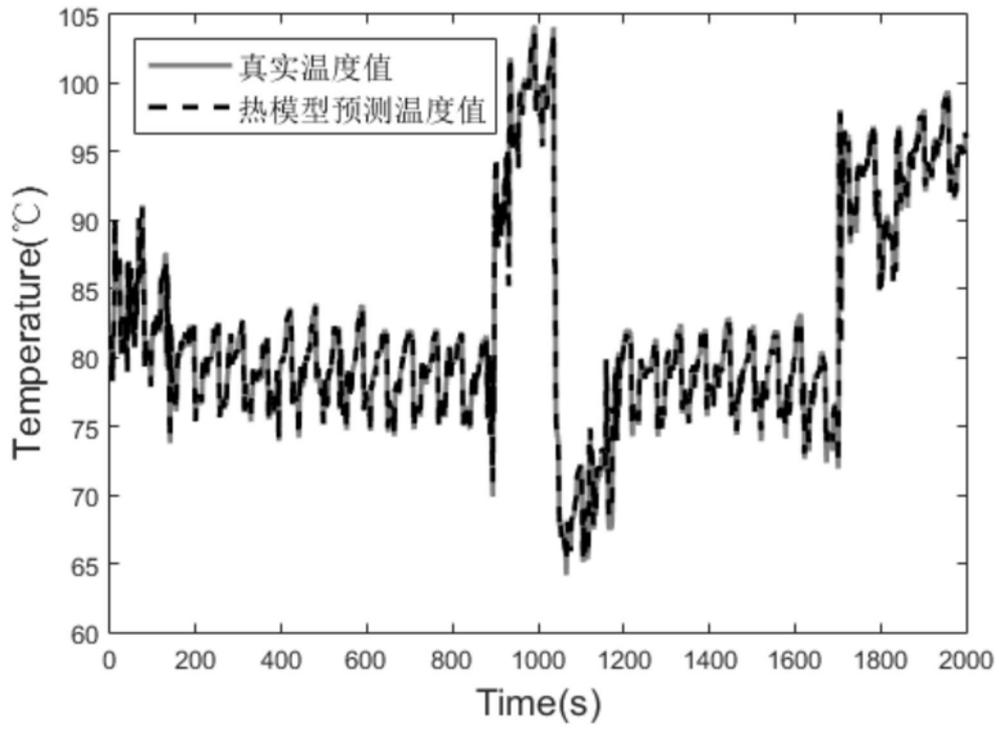


图4