

基于大规模训练神经网络的微小故障在线检测

司文杰¹ 杨飞飞²

(华南理工大学机械与汽车工程学院 广州 510641)¹ (华南理工大学材料科学与工程学院 广州 510641)²

摘要 神经网络已经广泛应用于系统建模和模式识别领域。但为了逼近未知的参数或者系统动态,需要大量的神经元达到足够的逼近精度,因此导致了计算负荷的增大。运算量制约着大规模神经网络计算,无法使其应用到实际的在线系统中。CPU处理无法保证在线数据的同步运算,需要借助图形处理单元GPU(Graphic Processing Unit)来解决实时性同步运算问题。首先,利用RBF神经网络的持续激励PE(Persistent Excitation)特性对系统输入进行分析,减少神经元的数目且优化设计算法,从而提高逼近精度。其次,基于LabVIEW平台,利用LabVIEW的GPU高性能分析工具包实现神经网络算法和并行计算。最后,在一台航空低速轴流压气机中开发基于大规模训练神经网络的LabVIEW系统。实验结果表明,提出的方法可以实现对系统的在线实时运行,满足航空失速检测的要求。

关键词 神经网络,持续激励,LabVIEW,GPU,大规模计算,在线实验

中图分类号 TP183 文献标识码 A DOI 10.11896/j.issn.1002-137X.2017.02.039

Online Detection of Incipient Fault Based on Large-scale Neural Networks

SI Wen-jie¹ YANG Fei-fei²

(School of Mechanical and Automotive Engineering, South China University of Technology, Guangzhou 510641, China)¹

(School of Materials Science and Engineering, South China University of Technology, Guangzhou 510641, China)²

Abstract Neural networks have been widely used for the system modeling and pattern recognition. However, in order to approximate the unknown parameters or system dynamics, it needs enough neurons to achieve sufficiently accurate approximation, which leads to increase of the computational cost. The computation would restrict the online application of the large-scale neural networks. Because CPU processing cannot keep pace with online data capture, the commonly available graphics processors are used for the bulk of data processing in online systems. First, the input of the system was analyzed by persistent excitation characteristics of RBF neural network, reducing the number of neurons and optimizing design optimization algorithm to improve the approximation error. Secondly, LabVIEW and LabVIEW GPU analysis toolkit were used to achieve algorithm implementation and parallel computing. Finally, online experiment of stall detection was conducted in a low speed axial compressor based on LabVIEW. Experimental results show that the proposed method can meet compressors stall detection of online operating system.

Keywords Neural network, Persistent excitation, LabVIEW, GPU, Large-scale computing, Online experiment

1 引言

人工神经网络(Artificial Neural Networks)是基于人脑的并行处理能力、由大量计算单元(computational units)相互耦合连接组成的非线性、自适应信息处理的并行分布式系统。神经网络模拟生物大脑神经网络的并行处理、记忆存储的方式,用学习代替了编程来解决非线性、复杂的未知系统问题。由于广泛的用途和处理复杂问题的能力,神经网络成为研究的热点,主要应用领域包括航空(飞行控制系统、飞机部件的故障检测、无人机的路径规划)、汽车制造(汽车的自动导航系统、汽油喷射控制、自动刹车系统)、银行业务和金融服务(汇率预测、信贷申请评估、现金预测、金融投资风险预测)以及电

子制造(芯片寿命分析、语言合成、非线性建模)等^[1-2]。

LabVIEW是NI(National Instruments)公司研制的用于仪器控制、数据采集和分析的图形编程环境。LabVIEW的应用开发利用VIs(Virtual Instruments)连接的框图,其输入和输出的交互在一个名为前面板(Front Panel)的窗口上运行。基于源码的图标比较贴切及形象地诠释了“软件即仪器”的观点,消除了繁琐的编程。LabVIEW可以直接交互设备、传感器和执行器。这种虚拟的计算环境已经被广泛应用到很多控制系统,如发酵监测控制^[3]、AC伺服电机控制^[4]、粮食湿度监控系统^[5]和本文的神经网络建模与识别系统。基于大型数据集快速并行处理的需求,GPU的计算优势受到越来越多人的关注。除了专业图形应用外,GPU还被用于解决大量的通

到稿日期:2015-12-24 返修日期:2016-04-13 本文受国家自然科学基金重点项目(60934001)资助。

司文杰(1985-),男,博士后,主要研究方向为航空发动机旋转失速等不稳定流动故障检测、模式识别等;杨飞飞(1985-),女,博士后,主要研究方向为神经网络自适应控制、动态模式识别、基于模式的控制等,E-mail:yfflgw@163.com(通信作者)。

用计算问题。不同于 CPU 串行代码的设计, GPU 是将大量的晶体管用作 ALU(Arithmetic-Logic Unit) 计算单元, 适合高强度计算的应用^[6]。同时, 类似大脑的神经网络芯片受到各种关注, 目标是创造出能够“学习”的计算机。高通的 Zeroth 芯片被称作“神经处理装置”, 由上百万个处理芯片组成, 且具有并行处理的能力。虽然不同于现有硬件结构, 但其还是无法与人脑的自动协调处理能力相比。

为了保证机械的安全运行, 需要在设备没有产生重大故障之前进行检测, 所以微小故障的检测受到人们广泛的关注^[7], 如列车悬挂系统故障检测^[8]和电子机械制动系统的故障检测^[9]。其中, 基于神经网络学习的非线性故障检测方法可以利用学习和存储的记忆去逼近未知的不确定项和故障函数, 以增加故障检测策略的鲁棒性和灵敏度, 因此开始得到人们的重视。但是在学习的方法中有一个主要的限制因素: 系统动态地准确逼近需要满足持续激励条件。

如今的科技研究中, 大多数的神经网络算法都无法应用于实际场景。算法的运算速度一直是神经网络应用的制约因素: 针对不同的系统输入, 需要设计不同的神经网络布局; 微小信号的检测需要在满足持续激励条件下保证神经网络权值能收敛到真值或者最优值, 准确的建模是神经网络准确预测和检测的前提; 训练得到的模式库会因各种条件的不同导致规模增大, 会直接影响检测的计算负荷。本文结合美国国家仪器(NI)公司的 LabVIEW 开发平台与 GPU, 并优化神经网络数量以及神经网络算法来实现在线的大规模神经网络计算。

本文的主要创新点以及难点: 1) 首先利用未知系统动态学习的理论^[11], 根据 RBF 神经网络的输入是回归轨迹, 保证由径向基函数构成的子向量可以满足持续激励条件, 使沿轨迹的部分神经网络权值收敛到真值或最优值; 2) 在此基础上, 提出了微小信号的故障检测方法, 利用准确的建模结果构建模式库, 基于动态模式识别的方法提出系统内部动态相似性度量方法来衡量系统的内部差异; 3) 最后, 利用 LabVIEW 的 GPU 高性能分析工具包实现神经网络算法和并行计算的混合编程, 在在线航空发动机实验中, 可以在毫秒级的失速先兆发生之前给出预警信号并能预留时间进行制动控制。

2 预备知识

本节主要介绍美国国家仪器(NI)公司的 LabVIEW 软件以及 GPU 工具包和 RBF 神经网络的持续激励条件。

2.1 LabVIEW 及其 GPU 分析工具包

LabVIEW 是 NI 公司研制的图形化编程环境。与传统的根据语句和指令顺序运行的编程语言不同, LabVIEW 采用数据流的形式进行编程, 按照每个程序框图中的连接线的走向决定函数的执行顺序。

LabVIEW 提供了与仪器外观类似的控件, 可以方便地拖放器件进而构建用户界面。器件之间的连接需要内部的连线和图标, 这就是 LabVIEW 的图形化源代码。

GPU 是不同于 CPU 的图形处理器: 在结构上, GPU 利用更多的晶体管作为执行单元, 具有更快的运算速度; 在浮点计算能力上, GPU 由同一个运算单元执行整数和浮点计算; 在内存带宽上, GPU 是显存上的内存子系统, 支持对双精度

的运算; 在延迟与带宽上, GPU 的高显存带宽和处理能力提供了巨大的数据吞吐量, 访问的显存延迟可达到数百乃至上千时钟周期。

在 LabVIEW 环境中进行 GPU 计算利用了 LabVIEW 的 GPU 分析工具包, 这适应于占用内存比例较高的强度运算, 并显著提高了运算速度。通过该 GPU 分析工具包, GPU 并行架构能够得到充分利用。该工具包提供的 GPU 接口主要由两个 LabVIEW 库组成: lvcuda, lvlib 和 lvcublas, lvlib, 它们包含 LabVIEW 数据类型和子 VI。该工具包利用 NVIDIA 的 CUDA(Compute Unified Device Architecture)工具包的功能以及 CUBLAS 和 CUFFT 方法的核心资源进行 LabVIEW 应用程序编写, 从而将计算任务转移到图像处理单元(GPU)。在利用 GPU 分析工具包时, 无法直接对 LabVIEW 代码进行编译, 需要借助封装好的 CUDA 函数。在 CUDA 的设备环境中, 利用其核心的程序运算, 在 LabVIEW 执行过程中调用自定义的 CUDA 内核代码, 这样就保证了大规模的数据计算成功移植到 GPU 上进行快速运算。

2.2 RBF 神经网络及其持续激励条件

径向基神经网络(Radical Basis Function, RBF)是一种前馈神经网络, 发展于 20 世纪 80 年代, 广泛应用于模式识别、函数逼近等领域。RBF 神经网络是线性参数化模型, 表达式如下:

$$f(Z) = \sum_{i=1}^N w_i s_i(Z) = W^T S(Z) \quad (1)$$

其中, Z 是神经网络输入向量; $W = [w_1, w_2, \dots, w_N]^T$ 是神经网络权值; $s_i(\cdot)$ 是径向基函数, 这里用高斯函数, $S(Z) = [s_1(\|Z - \xi_1\|), \dots, s_N(\|Z - \xi_N\|)]^T$, 表达式如下:

$$s_i(\|Z - \xi_i\|) = \exp\left[-\frac{(Z - \xi_i)^T (Z - \xi_i)}{\eta_i^2}\right], i = 1, 2, \dots, N \quad (2)$$

其中, $\xi_i (i = 1, \dots, N)$ 是神经网络中心点, η 是神经网络宽度。当 $\|Z\| \rightarrow \infty$ 时, $s_i(\|Z - \xi_i\|) \rightarrow 0$, 其中 $\|\cdot\|$ 代表向量的欧氏范数。

当 RBF 神经网络的神经元的个数 N 足够大时, 可以以任意精度逼近非线性函数 $f(Z)$ 。对于局部化的 RBF 神经网络, 沿着轨迹的神经元能够近似逼近非线性函数 $f(Z)$:

$$f(Z) = W_\zeta^T S_\zeta(Z) + \epsilon_\zeta \quad (3)$$

其中, ϵ 是局部神经网络的逼近误差, $W_\zeta \in R^{N_\zeta}$ 是沿着系统轨迹的激励权值, $N_\zeta < N$ 。

持续激励条件可以保证神经网络参数估计收敛到真值或者最优值, 是系统辨识领域中的一个核心概念, 起到重要作用。Kurdila 等^[10]给出了适用于离散和连续的持续激励条件, 指出当系统的轨迹在每一个时间段 $[t_0, t_0 + T_0]$ 内遍历所有神经元足够小的 ϵ 领域, 并且停留足够的时间, 则回归向量 $S(Z(t))$ ($S(\cdot)$ 是高斯函数) 满足持续激励。这里的 ϵ 领域定义如下:

$$0 < \epsilon < h = \frac{1}{2} \min_{i \neq j} \|\xi_i - \xi_j\| \quad (4)$$

Wang 等^[11]放宽了系统输入必须遍历所有神经元足够小的 ϵ 领域的限制: $\epsilon \geq \sqrt{qh} = \frac{\sqrt{q}}{2} \min_{i \neq j} \|\xi_i - \xi_j\| > 0$, 保证沿着轨

迹的神经元中心领域内的回归子向量 $S_{\xi}(Z)$ 满足持续激励条件。

文献[11]提出了对未知系统动态进行局部准确建模的方法,其基本要素包括:1)使用径向基函数神经网络;2)具有回归状态轨迹的系统可以满足部分持续激励条件;3)在周期(或回归)轨迹的邻域内实现对非线性系统动态的局部准确神经网络逼近(局部准确建模);4)所学的知识以时不变且空间分布的方式表达,以常值神经网络权值的方式存储,并且可以在动态环境下用于动态模式的快速识别。

3 基于大规模神经网络的系统辨识及 LabVIEW 的 GPU 算法

本节首先介绍文献[11]提出的针对未知系统的动态学习的理论,基于此,根据 RBF 神经网络复杂度的持续激励条件和高斯函数的局部特性,设计减少神经网络的方案并保证准确建模;其次,给出基于 LabVIEW 平台的 GPU 编程算法;最后,利用学习到的系统动态构建模式库,在 LabVIEW 平台上实现在线检测算法,并解决实时快速性问题。

3.1 未知非线性系统的学习

未知系统动态学习的理论^[11]研究未知动态环境下的知识获取、表达、存储和再利用。周期或者回归轨迹的神经元函数组成的子向量满足部分 PE 条件,可以保证沿着系统轨迹的神经元收敛到真值或者最优值,对系统内部动态进行局部准确的建模。

考虑一个非线性系统 $\dot{x} = F(x; p)$, $x(t_0) = x_0$, x 是可测量的系统状态,这里指的是 LabVIEW 系统中经过滤波的系统输入。在真实的测量现场中,系统是未知的;并且,通常系统状态都是一致有界的,起始于 x_0 的系统状态轨迹可以保证是回归轨迹。

构造如下的神经网络系统用于估计系统状态:

$$\dot{\hat{x}} = -A(\hat{x} - x) + \hat{W}^T S(x) \quad (5)$$

其中, \hat{x} 是神经网络的状态, $A = \text{diag}\{a_1, a_2, \dots, a_n\}$ 是对角矩阵, a_i 是设计常值。神经网络 $\hat{W}^T S(x)$ 用来逼近未知系统动态。

这里采用如下的更新率:

$$\dot{\hat{W}}_i = -\Gamma_i S_i(x) \tilde{x}_i - \sigma_i \Gamma_i \hat{W}_i \quad (6)$$

其中, $\tilde{x}_i = \hat{x}_i - x_i$ 是 LabVIEW 系统输入状态的估计误差, \hat{W}_i 是最优权值 W_i^* 的估计值, $\hat{W} = \hat{W} - W^*$, $\Gamma_i = \Gamma_i^T > 0$, $\sigma_i > 0$ 是很小的设计参数。

需要注意的是,基于未知系统动态学习的理论, LabVIEW 系统输入的系统状态轨迹是回归轨迹,神经网络中沿着系统状态轨迹的神经元函数构成的子向量可以满足持续激励条件。因此,这里只需采用沿着系统状态轨迹的神经元,这样可以大幅减轻学习的负担,并且可以保证神经网络权值收敛到真值或最优值。为了说明这一点,给出系统状态估计误差系统:

$$\begin{aligned} \dot{\tilde{x}}_i &= -a_i \tilde{x}_i + \hat{W}_i^T S_i(x) - f_i(x, p) \\ &= -a_i \tilde{x}_i + \hat{W}_{\xi_i}^T S_{\xi_i}(x) + \hat{W}_{\zeta_i}^T S_{\zeta_i}(x) - f_i(x, p) \\ &= -a_i \tilde{x}_i + \hat{W}_{\xi_i}^T S_{\xi_i}(x) - \varepsilon_{\xi_i} \end{aligned} \quad (7)$$

权值估计误差系统:

$$\dot{\hat{W}}_i = \dot{\tilde{W}}_i = -\Gamma_i S_i(x) \tilde{x}_i - \sigma_i \Gamma_i \hat{W}_i \quad (8)$$

其中, $(\cdot)_{\xi_i}$ 和 $(\cdot)_{\zeta_i}$ 分别代表系统状态轨迹领域和远离轨迹的区域。 S_{ξ_i} 是子向量。 $\hat{W} = \hat{W} - W^*$, \hat{W} 是 W^* 的估计值。 $\tilde{x} = \hat{x} - x$ 。

基于 RBF 神经网络的局部特性,任何周期或者回归轨迹 $\varphi_{\xi}(x_0)$ 能保证径向基函数子向量 $S_{\xi_i}(\varphi_{\xi})$ 满足持续激励条件,进而可以保证状态估计误差指数收敛到小领域内。同样,可以知道 $S_{\zeta_i}(\varphi_{\zeta})$ 值很小,又因为 \tilde{x} 是有界的,所以 $\dot{\hat{W}}_{\zeta_i}$ 值将会很小。这里可以考虑如下的 Lyapunov 函数:

$$V = \frac{1}{2} \sum_{i=1}^n (\tilde{x}_i^2 + \hat{W}_i^T \Gamma_i^{-1} \hat{W}_i) \quad (9)$$

结合式(7)和式(8), V 的导数为:

$$\begin{aligned} \dot{V} &= \sum_{i=1}^n (\tilde{x}_i \dot{\tilde{x}}_i + \hat{W}_i^T \Gamma_i^{-1} \dot{\hat{W}}_i) \\ &= \sum_{i=1}^n (-a_i \tilde{x}_i^2 - \tilde{x}_i \varepsilon_i - \sigma_i \hat{W}_i^T \hat{W}_i) \end{aligned} \quad (10)$$

令 $a_i = a_{i1} + a_{i2}$, 且大于零,得出最后结果:

$$\dot{V} \leq \sum_{i=1}^n \left(-a_{i1} \tilde{x}_i^2 - \frac{\sigma_i}{2} \frac{\|\tilde{W}_i\|^2}{2} + \frac{\sigma_i}{2} \frac{\|\tilde{W}_i^*\|^2}{2} + \frac{\varepsilon_i^{*2}}{4a_{i2}} \right) \quad (11)$$

可以看出,当 $|\tilde{x}_i| \leq \frac{\varepsilon_i^*}{2\sqrt{a_{i1} a_{i2}}} + \sqrt{\frac{\sigma}{2a_{i1}}} \|W_i^*\|$ 时, \dot{V} 小

于零。可以保证闭环系统中所有信号有界。状态误差可以收敛到小的领域内,并且 ε^* , $a_i, \sqrt{\sigma} \|W_i^*\|$ 决定了领域的大小。由于 RBF 神经网络的局部特性,使得 $S_{\xi_i}(\varphi)$ 很小,由于 $|\tilde{x}_i|$ 收敛到零区域,导致 $S_{\zeta_i}(\varphi)$ 更小;又由于权值 W_{ζ_i} 在远离系统状态轨迹并没有触发学习,最终会使 $\hat{W}_{\zeta_i}^T S_{\zeta_i}(x)$ 保存一个很小的值。可以得出,沿着系统轨迹的 φ_{ξ} 逼近的未知动态为:

$$\begin{aligned} f_i(x; p) &= \bar{W}_{\xi_i}^T S_{\xi_i}(x) + \xi_i'(t) \\ &= \bar{W}_{\xi_i}^T S_{\xi_i}(x) + \bar{W}_{\zeta_i}^T S_{\zeta_i}(x) + \xi_i(t) \\ &= \bar{W}_i^T S_i(x) + \xi_i(t), \forall x \in \Omega_{\xi} \end{aligned} \quad (12)$$

其中, $\bar{W}_i = \text{mean}_{t \in [t_a, t_b]} \hat{W}_i$, $t \in [t_a, t_b]$ 是权值常值。 Ω_{ξ} 是沿系统轨迹的一个局部区域,因为高斯函数具有局部特性,所以在该区域的神经元才会被激励, $|\xi_i(t)| < \xi^*$, $\xi^* > 0$ 是一个小常数,这就实现了对非线性系统的未知动态进行局部准确的建模。

3.2 检测算法

系统 $\dot{x} = F(x; p)$, $x(t_0) = x_0$ 的动态已通过上面的方法学习。这里给出相似性检测策略:动态模式系统轨迹是相似的,且沿着轨迹,两个模式的系统动态差异很小,即:

$$\max_{x \in \varphi_{\xi}(x_0, p')} |f_i'(x; p') - \bar{W}_i^T S_i(x)| \leq \varepsilon_i^* + \xi^* \quad (13)$$

其中, ξ^* 是上一步的建模误差, ε_i^* 是实际的两个模式的内部系统差异: $\max |f_i'(x; p') - f_i(x; p)| \leq \varepsilon_i^*$ 。这里的 $\varepsilon_i^* + \xi^* > 0$ 被称为学习模式的相似性度量。

下文给出快速识别的方法。通过上一步的学习,得出不同系统的系统内部动态,以不变式 RBF 神经网络权值进行存储,即构建模式库。针对每个实际测量的系统,与存储的一组模式库比较得出如下的残差系统(构建的估计器与学习的动态作差):

$$\dot{\tilde{x}}_i^k = -b_i \tilde{x}_i^k + \bar{W}_i^{kT} S_i(x) - f_i'(x_i; p') \quad (14)$$

其中, $\tilde{x}_i^k = \bar{x}_i^k - x_i$, 这里的头标 k 代表模式库中不同模式与被测系统的比较。基于文献[11]提出的理论和动态模式识别方法, 将相似性估计误差的大小作为衡量被测系统和模式库训练模式相似性的一个标准。

注: 本文提出的微小振动故障检测方法可以在故障函数幅值大于两倍神经网络逼近误差时检测出故障, 比文献[12]提出的检测方法更加敏感。由于在学习阶段提前进行了模式库的构建, 因此减轻了识别阶段的计算负荷, 提高了检测的灵敏度。识别的准确性以准确建模为基础, 这里的准确建模指运用线性时变系统的稳定性分析方法, 深入分析系统设计参数与算法性能的内在关系, 通过改变算法参数以及神经网络布局来提高微小振动信号的持续激励水平, 最终实现对微小振动故障的近似准确动力学建模。

3.3 LabVIEW 的 GPU 算法

LabVIEW GPU 分析工具包如图 1 所示, 其主要分为 LVCUDA, LVCUBLAS, LVCUFFT 和 LVGPU SDK 4 个部分。

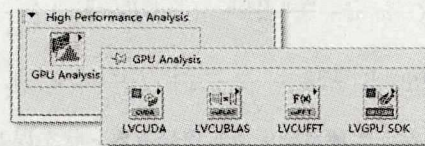


图 1 LabVIEW GPU 工具包的函数选板

本文介绍 LabVIEW 下的 GPU 编程的一般步骤。一般情况下, 使用 LabVIEW GPU 分析工具包与 GPU 通信, 可以细分为 3 个阶段:

- 1) 初始化 GPU 资源。在该步骤中所需要初始化的资源包括 GPU 设备以及相关函数库。
- 2) 在 GPU 上执行相应的计算。利用函数执行相应的计算, 包括 CUBLAS 库和 FFT 库。
- 3) 释放 GPU 资源。该过程也是利用相关函数来实现的, 包括释放内存、库以及设备。

本文利用 LabVIEW 的 GPU 进行神经网络计算, 最终实现实时的在线检测算法。

4 仿真和结果

本节是对第 3 节提出的算法进行量化分析以及试验台验证。首先, 利用 LabVIEW 的 GPU 进行算法实现, 并且针对 LabVIEW 在 CPU 与 GPU 的条件下开展量化研究。其次, 利用 Duffing 振子来验证算法建模的准确性, 并对 RBF 神经网络在 LabVIEW 的 GPU 进行实现。最后, 利用一台低速航空发动机进行在线的试验台验证, 结果证明基于 LabVIEW 的大规模神经网络检测系统可以提前检测到失速故障信号的发生。

4.1 GPU 实现

按照 3.3 小节介绍的 GPU 编程流程进行程序设计: 利用两个矩阵的相乘运算展示 LabVIEW 环境下的 GPU 编程, 并且验证 GPU 在加速大规模矩阵运算的加速效果。这里用到的两个矩阵的维数(即矩阵中元素个数大小)从 $10 \times 10, 100 \times$

$100, 1000 \times 1000, 3000 \times 1000$ 到 3000×2000 依次变化, 展示程序的最终运行时间, 量化 LabVIEW 的 GPU 功能在大规模计算中的加速性能。

1) 初始化设备和库。由于本文所有的例子都是对两个矩阵的相乘进行演示, 因此在这里初始化 CUBLAS 库。

2) 分配内存, 然后通过 CPU 上传所要计算的矩阵到 GPU 显存中, 为接下来的 GPU 运算做准备。

3) 执行矩阵相乘运算, 使用 GPU 工具包下的常规矩阵与矩阵的乘积 VI(SGEMM), 如图 2 所示。这一部分在 GPU 内执行, 并通过程序在 CUDA 核心上自动分配线程, 并行线程的运行达到最优, 使 GPU 显卡满负荷工作。

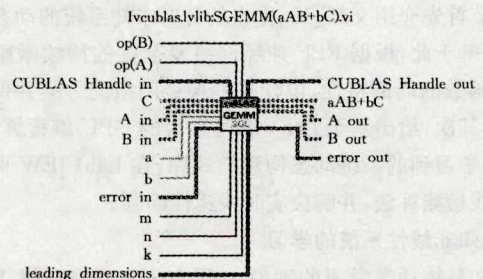


图 2 SGEMM 接线图

4) 计算完成后, 为避免内存泄漏以及资源占用, 还需要对设备以及相关库进行释放, 可通过 Free Memory 子 VI、Release Library 子 VI 以及 Release Device 子 VI 实现。

同样, 在 CPU 上重新计算这两个矩阵的运算, 并计算时间。两种情况下, GPU 与 CPU 的运行效果比较如表 1 所列。

表 1 运算结果的时间比较

A(或 B)矩阵维数 大小/行数 * 列数	GPU 上的运算时间/s	CPU 上的运算时间/ s
10 * 10	0.162105	0.022996
100 * 100	0.193420	0.000470
1000 * 1000	0.210642	0.258205
3000 * 1000	0.388196	2.306360
3000 * 2000	0.573712	4.811110

从实验运行的数据可知, 随着矩阵维数的增加, 与 CPU 相比, GPU 在计算上占据更大的优势。本例所运行的 CUDA 核心数为 96, 刚开始在矩阵维数较小时, 与 CPU 相比, GPU 没有太大优势, 大部分时间花在 CPU 向 GPU 的传输过程中; 当矩阵维数增加, 元素增加到一定数目即 CUDA 核心数两倍以上时, 在 GPU 条件下的程序能得到很好的加速, GPU 运算的优势更加明显。

4.2 Duffing 振子

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -p_2 x_1 - p_3 x_1^3 - p_1 x_2 + q \cos(\omega t) \end{cases} \quad (15)$$

其中, $x = [x_1, x_2]$ 是系统状态, p_1, p_2, p_3, ω 和 q 是常值参数。系统动态 $f_2 = -p_2 x_1 - p_3 x_1^3 - p_1 x_2 + q \cos(\omega t)$ 是未知、平滑的非线性函数。这里用 Duffing 振子作为参考模型来产生回归轨迹, 并验证所提方法建模的准确性和 LabVIEW 的神经网络实现。

选择状态初值 $x(0) = [x_1(0), x_2(0)]^T = [0.0, -1.8]^T$, 其他的参数设定为: $p_1 = 0.55, p_2 = -1.1, p_3 = 1.0, \omega = 1.8$ 和 $q = 1.498$, 这里构建的 RBF 神经网络在规则的网格中如图

3 所示,设神经网络宽度 $\eta=0.2$,神经网络的设计参数 $a=5$, $\Gamma=2, \sigma=0.001$,权值的初值 $\hat{W}(0)=0.0$ 。

从图 3—图 6 可以看出,系统动态被局部准确辨识,以常值 RBF 神经网络的形式存储。图 5 给出了未知系统动态 f_2 与 NN 的逼近结果以及它们之间的误差。从图 4 中可以看出,神经网络权值在有限时间内收敛到常值。在图 6 中神经网络近似发生在状态轨迹的小邻域内,而对于其他区域,学习并不发生,对应于图中为零的区域,即权值为零的位置。

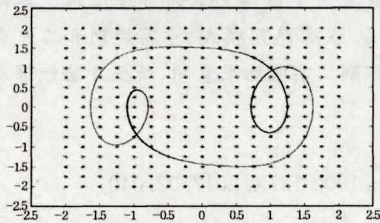


图 3 系统状态轨迹及其神经网络布局

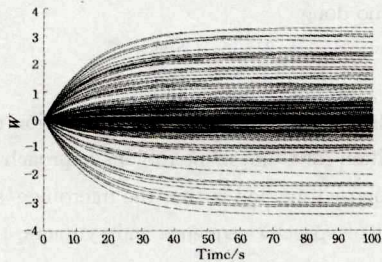


图 4 权值收敛图

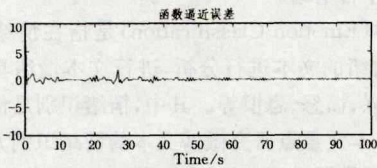
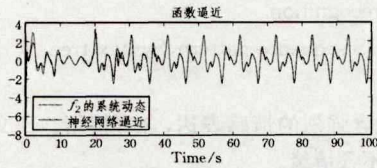


图 5 系统动态逼近及其逼近误差

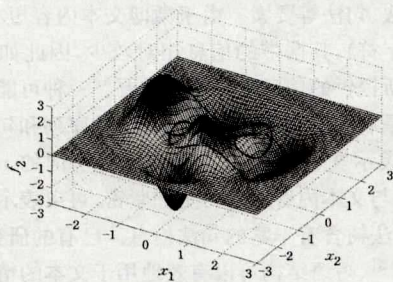


图 6 系统的局部准确近似

从对 Duffing 振子的仿真研究中可以发现,只有沿着系统状态轨迹的神经元才会被激励,而远离轨迹的其他区域没有进行学习,如图 6 所示。因此,在进行 LabVIEW 在线计算时,在对轨迹的微小邻域内布置神经网络,这样可以减轻神经网络的计算负荷。

4.3 航空发动机的在线检测系统

针对在线实验测试,需要利用 GPU 强大的计算能力来解决本文方法(即文献[11]提出的理论及动态模式识别算法)带来的计算负荷较大的问题,以大幅度缩短神经网络建模时间,增强所提方法的实用性。

低速压气机试验台的数据采集系统(上位机)由 NI 公司生产,利用其旗下的 LabVIEW 并行机制开发能实现上位机和下位机的实时通讯。基于 LabVIEW 软件设计的轴流压气机旋转失速检测系统涉及到数据滤波、训练、识别以及参数设置。

试验台如图 7 所示。



图 7 试验台轮廓

具体的试验台参数如表 2 所列。

表 2 低速轴流压气机主要涉及的参数

设计参数	符号及单位	参数值
外径	$D2/\text{mm}$	450
设计转速	$n/r \cdot \text{min}^{-1}$	3000
转子叶尖切向速度	$u/\text{m} \cdot \text{s}^{-1}$	70.70
轮毂比	d	0.75
转子叶片数目	z_r	19
静子叶片数目	z_s	13
转子叶尖间隙绝对值	$\bar{\delta}_i = \delta_i/h/\%$	0.89

图 8 是基于 LabVIEW 平台的旋转失速在线检测示意图。进行实时在线监测时,该检测系统接收上位机的实时数据并将其作为神经网络径向基函数的输入,构造一系列嵌入了常值 RBF 神经网络的动态估计器进行并行计算并生成了残差,然后根据同步状态估计误差的最小原则进行失速实时检测。

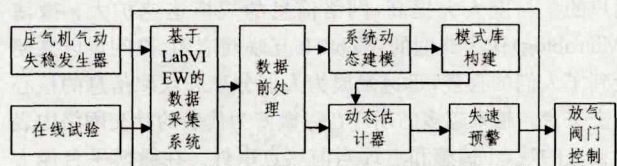


图 8 基于 LabVIEW 平台的旋转失速在线检测流程图

图 9 所示为实际运行时的系统界面截图。

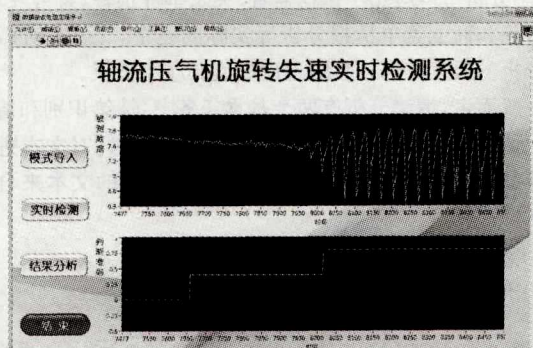


图 9 基于 LabVIEW 的在线检测系统界面

- Scotland, UK, 2011(10):24-28.
- [6] LI Q Q, GU J F. Activity driven modelling of online social network [J]. Journal of Systems Engineering, 2015(1):9-15. (in Chinese)
李倩倩, 顾基发. 用户行为驱动的在线社交网络建模[J]. 系统工程学报, 2015(1):9-15.
- [7] ZHANG B, ZHANG Y, GAO K N, et al. Combining Relation and Content Analysis for Social Tagging Recommendation [J]. Journal of Software, 2012, 23(3):476-488. (in Chinese)
张斌, 张引, 高克宁, 等. 融合关系与内容分析的社会标签推荐[J]. 软件学报, 2012, 23(3):476-488.
- [8] KRISHNAMURTHY B, GILL P, ARLITT M. A New Chirps About Twitter[C]//Proceedings of the 1st Workshop on Online Social Networks. 2008:19-24.
- [9] LU P, ZHANG S S, GAO Q Y. Research on BBV Mode with Limited Node Strength Based on Common Neighbors [J]. Computer Science, 2014, 41(4):49-52. (in Chinese)
逯鹏, 张姗姗, 高庆一. 基于共同邻居的点权有限 BBV 模型研究[J]. 计算机科学, 2014, 41(4):49-52.
- [10] QIN H Y. Research on Correlative Techniques of Hot-topic Discovery About Internet Public Opinion [D]. Habin: Harbin Engineering University, 2010. (in Chinese)
秦宏宇. 网络舆情热点发现相关技术研究[D]. 哈尔滨: 哈尔滨工程大学, 2010.
- [11] XU Z M, LI D, LIU T, et al. Measuring Similarity between Microblog Users and Its Application [J]. Chinese Journal of Computers, 2014, 37(1):207-218. (in Chinese)
徐志明, 李栋, 刘挺, 等. 微博用户的相似性度量及其应用[J]. 计算机学报, 2014, 37(1):207-218.
- [12] WU S F, XU J M, WU X B. Similarity Measurement of Microblogging Users Merging User Tags and Relationships [J]. Journal of Intelligence, 2014(12):170-173, 126. (in Chinese)
吴树芳, 徐建民, 武晓波. 融合用户标签和关系的微博用户相似性度量[J]. 情报杂志, 2014(12):170-173, 126.
- [13] SONG D, HEFLIN J. Domain-independent entity reference in RDF graphs[C]//Proceedings of the 19th ACM International Conference on Information and Knowledge Management. ACM, 2010:1821-1824.
- [14] GUO J Y, ZHANG Z B, SUN Q Y. Study and Applications of Analytic Hierarchy Process [J]. China Safety Science Journal, 2008, 18(5):148-153. (in Chinese)
郭金玉, 张忠彬, 孙庆云. 层次分析法的研究与应用 [J]. 中国安全科学学报, 2008, 18(5):148-153.
- [15] SONG J F, LI G H. Analysis and Improvement on Evaluation of Information Retrieval Algorithm [J]. Journal of Chinese Computer Systems, 2003, 24(10):1800-1804. (in Chinese)
宋峻峰, 李国辉. 信息检索算法评价指标的分析与改进 [J]. 小型微型计算机系统, 2003, 24(10):1800-1804.

(上接第 243 页)

在进行本次测试时, 试验台上压气机的信号经放大后用磁带机记录, 进行同步采样, 采样频率为 5kHz。在服务器端进行信号的转换和滤波处理, 为了减轻传输的负担和客户端识别算法的运算负荷, 进行了重采样。从图 9 可以看出, 第一个开关信号作为失速故障信号的预警信号, 第二个信号是失速故障发生信号, 有 1 秒多钟的时间裕度, 这为进一步的控制提供了一定的预警时间, 有助于保障飞行安全。

结束语 本文利用 LabVIEW 的 GPU 解决了神经网络在线实现的计算问题。首先, 介绍了 RBF 神经网络及其持续激励条件, 这是保证神经网络能够准确逼近系统动态的前提。其次, 基于 LabVIEW 的 GPU 特性进行混合编程, 提出了针对神经网络建模和检测的具体方法, 为在线实现提供了解决途径。通过沿着系统状态轨迹布置神经网络实现了系统的局部准确建模, 利用持续激励水平和神经网络建模准确性的关系, 优化系统参数。最后, 利用 Duffing 振说明了沿着系统轨迹布局神经网络的依据, 并在一台低速航空发动机中实现了基于 LabVIEW 的失速检测系统。在线失速试验说明提出的方法可以保证实时的大规模神经网络计算。

参 考 文 献

- [1] ANDERSON J A. An introduction to neural networks[M]. MIT press, 1995.
- [2] HAGAN M T, DEMUTH H B, BEALA M H. Neural network design[M]. Boston: Pws Pub., 1996.
- [3] ZENG D, GAO L, LIN L, et al. Application of LabVIEW in online monitoring and automatic control of fermentation process [J]. Control & Computer, 2006(22):48-50.
- [4] HORNG J H. Hybrid MATLAB and LabVIEW with neural network to implement a SCADA system of AC servo motor[J]. Advances in Engineering Software, 2008, 39(3):149-155.
- [5] LIU J M, XU Z Z, SUN D H. Virtual instrument and neural network in application of grain moisture detection[C]//Proceedings of the 2014 International Conference on Control Engineering and Information Systems. 2014:155-158.
- [6] LU F S, SONG J Q, YIN F K, et al. Survey of CPU/GPU synergetic parell computing[J]. Computer Science, 2011, 38(3):5-9. (in Chinese)
卢风顺, 宋君强, 银福康, 等. CPU-GPU 协同并行计算研究综述[J]. 计算机科学, 2011, 38(3):5-9.
- [7] GAO Z, CECATI C, DING S X. A survey of fault diagnosis and fault-tolerant techniques-Part I: fault diagnosis With model-based and signal-based approaches[J]. IEEE Transactions on Industrial Electronics, 2015, 62(6):3757-3767.
- [8] WU Y, JIANG B, LU N, et al. ToMFIR-based incipient fault detection and estimation for high-speed rail vehicle suspension system[J]. Journal of the Franklin Institute, 2015, 352(4):1672-1692.
- [9] HWANG W, HUH K. Fault Detection and Estimation for Electromechanical Brake Systems Using Parity Space Approach[J]. Journal of Dynamic Systems, Measurement, and Control, 2015, 137(1):014504.
- [10] KURDILA A J, NARCOWICH F J, WARD J D. Persistency of excitation in identification using radial basis function approximants[J]. SIAM Journal on Control and Optimization, 1995, 33(2):625-642.
- [11] WANG C, HILL D J. Deterministic learning theory for identification, recognition, and control[M]. CRC Press, 2009.
- [12] POLYCARPOU M M, TRUNOV A B. Learning approach to nonlinear fault diagnosis: detectability analysis[J]. IEEE Transactions on Automatic Control, 2000, 45(4):806-812.