

DOI: 10.19650/j.cnki.cjsi.J2210840

多计算任务调度优化的机载智能状态监测单元*

康守强¹, 李 坤¹, 王本宽², 刘大同²

(1. 哈尔滨理工大学测控技术与通信工程学院 哈尔滨 150080; 2. 哈尔滨工业大学电子与信息工程学院 哈尔滨 150001)

摘要:为保障无人机的飞行安全,机载智能状态监测得到重点关注。然而,受机载计算资源约束,监测多个关键参数的多任务场景给机载实时计算带来较大挑战。因此,本文提出有限现场可编程门阵列(FPGA)资源下多计算任务调度优化的机载智能状态监测单元,首先,基于堆叠长短期记忆网络建立不同监测参数对应尺度的监测模型,并利用FPGA构建各模型的定制计算加速单元。然后,建立FPGA资源和模型计算时间联合约束的多任务调度优化方法,获取定制计算加速单元部署和计算调度策略,最小化总任务完成时间。最后,根据上述策略在机载计算平台中部署指定尺度的加速单元,完成多计算任务调度。采用实际飞行数据对机载智能状态监测单元进行验证。结果表明,该单元能够高效完成多个状态监测任务的实时计算,较其它基于FPGA的多任务计算方法,计算效率提升优于16.08%。

关键词: 机载状态监测;多计算任务调度;FPGA;无人机

中图分类号: V247.1 TH707 **文献标识码:** A **国家标准学科分类代码:** 510.4030

Airborne intelligent condition monitoring unit based on multiple computing tasks scheduling optimization

Kang Shouqiang¹, Li Kun¹, Wang Benkuan², Liu Datong²

(1. School of Measurement Control and Communication Engineering, Harbin University of Science and Technology, Harbin 150080, China; 2. School of Electronics and Information Engineering, Harbin Institute of Technology, Harbin 150001, China)

Abstract: To ensure the flight safety of unmanned aerial vehicles (UAV), the airborne intelligent condition monitoring has received a lot of attention. However, constrained by airborne computing resources, multi-task scenarios that monitor multiple key parameters pose a greater challenge to airborne real-time computing. To address this issue, this article proposes an airborne intelligent condition monitoring unit with multi-task scheduling optimization under limited field programmable gate array (FPGA) resource. Firstly, the monitoring models of different scales corresponding to different monitoring parameters are established based on stacked long short-term memory, and the custom computing acceleration units for each model are constructed by using FPGA. Secondly, a multi-task scheduling optimization method with joint constraints of FPGA resources and model computing time is proposed to obtain customized computing acceleration unit deployment and computing scheduling strategies, which minimizes the completion time of all tasks. Finally, according to the above strategies, the acceleration units of specified scales are deployed in the airborne computing platform to complete multi-task scheduling and computing. Real flight data are used to verify the proposed method. The results show that the unit can efficiently perform real-time calculation for multiple condition monitoring tasks. The results show that the unit can efficiently perform real-time calculation for multiple condition monitoring tasks. Comparing with other FPGA-based computing methods, the efficiency is improved by 16.08%.

Keywords: airborne condition monitoring; multiple computing tasks scheduling; FPGA; UAV

0 引言

随着无人机技术的发展,无人机凭借运行成本低、适

应性强和机动性高等特点,广泛应用于民用和军事领域^[1]。与有人驾驶飞机相比,由于缺乏人的实时控制,加上研制成本压缩,其发生事故的风险明显增加^[2]。为提高无人机的飞行安全性和可靠性,对状态监测技术提出

了迫切需求^[3]。状态监测主要是指通过飞行数据对无人机异常或故障的状态进行检测。无人机状态监测方法主要分为两类:基于机理模型的方法和基于数据驱动的方法。基于机理模型的方法需要建立无人机的动力学模型,过程较为复杂^[4],多用于准确机理模型已构建的场景。基于数据驱动的方法利用无人机运行状态监测数据进行建模,方法相对灵活,但其性能依赖于历史飞行数据的覆盖程度^[5]。鉴于无人机飞行试验积累了大量的历史数据,数据驱动的方法逐渐受到状态监测研究人员的重点关注^[6-7]。其中,基于长短期记忆网络(long short-term memory, LSTM)的状态监测方法因具备较好的时序和多维空间关联特征提取能力,适用于处理高维、时空关联的飞行数据,而得到广泛应用^[8]。

然而,基于 LSTM 的状态监测方法在获取优异性能的同时带来了高计算复杂度,在有限的机载计算资源下,模型实时推理的需求难以保障。为缓解上述问题,满足机载计算的实时、低延迟和低功耗场景特点,基于现场可编程逻辑门阵列(field programmable gate array, FPGA)构建具有可定制、并行、低延时等优势的边缘计算单元成为主要解决方式之一^[9-10]。文献[11]将最小二乘支持向量机部署在 FPGA 中,实现无人机垂直速度指示器的实时异常检测。文献[12]提出基于 FPGA 的可用、响应迅速且不显眼的单元(realizable, responsive, and unobtrusive unit, R2U2),并在其中部署贝叶斯网络,以对 GPS 故障进行检测。文献[13]利用 FPGA 对堆叠 LSTM 神经网络进行加速和部署,完成对滚转角速率参数的实时监测。但是,上述研究仅考虑了单个飞行参数的监测任务,而实际应用中需要同时执行对多个监测任务。在有限的 FPGA 计算资源下,如何实现多个监测任务实时推理成为机载状态监测研究的一项新挑战。

针对多计算任务在 FPGA 中的执行,一方面可利用 FPGA 的动态重构功能对计算资源重复利用,实现多任务运行,如文献[14-15]对 FPGA 部分资源的功能进行重构,采用最优替换策略和提前预取技术,优化任务的最大完成时间;另一方面可采用多任务调度,合理匹配计算资源和计算任务,来提高多任务的运行效率。如文献[16-17]在 CPU+FPGA 的异构实时边缘计算系统中,分别使用改进的基于预配置策略算法和一种动态部分重构平台的任务调度框架,优化流计算任务的总执行时间。文献[18]提出一种新的特征感知任务调度方法,根据加速估计和运行时负载均衡策略将任务分配给合适的处理器。文献[19]结合动态重构和多任务调度,提出一种基于并行参数约束的分层流水线和多核可重构计算的卷积神经网络处理器,在计算效率和检测精度之间实现了折中。上述方法在解决有限 FPGA 计算资源下多计算任务执行问题时,基于动态重构的方法未考虑重构时间过长而占用任务

执行时间,导致实时处理需求无法满足的问题;而基于多任务调度的方法忽略了有限 FPGA 资源下模型部署对多任务计算效率的影响,难以获取更优的执行效率。

综上,本文针对无人机多状态监测任务机载实时计算的需求,提出有限 FPGA 资源下多计算任务调度优化的机载智能状态监测单元。首先,建立基于堆叠 LSTM 的智能状态监测模型,及其对应的 FPGA 定制计算加速单元,提升单个状态监测任务的计算效率;然后,以 FPGA 计算资源和各任务计算时间为约束,以多任务完成时间最小化为目标,构建多计算任务调度优化模型,获得最优定制计算加速单元部署方式和多任务调度策略;最后,在 FPGA 中部署定制计算加速单元和调度多个监测任务,完成机载智能状态监测单元的构建。解决计算资源和实时性多约束条件下的机载多个智能监测任务的计算难题,保障无人机的飞行安全。

1 状态监测原理与多计算任务问题定义

首先,简要介绍基于堆叠 LSTM 的智能状态监测原理,然后,给出机载多个智能状态监测任务实时推理面临的现实问题。

1.1 基于堆叠 LSTM 的智能状态监测原理

LSTM 是循环神经网络(recurrent neural networks, RNN)的一种应用最广泛的变体,其有效缓解了 RNN 的梯度爆炸问题^[20]。堆叠 LSTM 模型则由多个 LSTM 层级联而成,每个 LSTM 层包含多个 LSTM cell,每个 LSTM cell 是由输入门、遗忘门和输出门构成,其详细结构参见文献[14]。

基于堆叠 LSTM 的智能状态监测模型主要包括堆叠 LSTM 预测模型和预测残差检测器两部分。预测模型原理如图 1 所示,其输入为多维飞行数据;隐藏层为两层 LSTM 层;输出层为全连接层,该层对最后一层 LSTM 输出的信息进行加权,输出一个被监测参数的预测值。

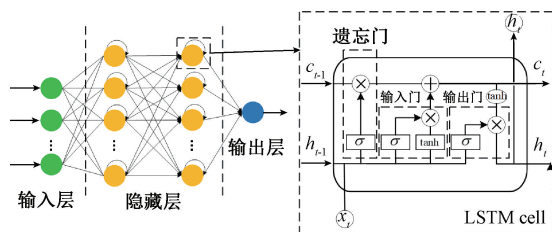


图1 堆叠 LSTM 预测模型原理框图

Fig. 1 Schematic diagram of the stacked LSTM prediction model

堆叠 LSTM 预测模型将 t 时刻的 d 维输入飞行数据与前 $L-1$ 个时刻历史飞行数据组合,构成长度为 L 的输入样本 X_t ,以自动提取被监测参数与其他飞行参数间的

时序和空间关联特征,获得 $t+1$ 时刻被监测参数的预测值。记预测模型的输入样本 X_t 为:

$$X_t = \{x_t, x_{t-1}, \dots, x_{t-L-1}\} \quad (1)$$

其中, x_i 为 i 时刻的飞行参数, $x_i \in R^d, i = t - L - 1,$

$\dots, t_0$ 。记 $t+1$ 时刻预测模型输出为 $\tilde{Y}_t$ 为:

$$\tilde{Y}_{t+1} = f_{LSTM}(X_R) \quad (2)$$

其中, f_{LSTM} 为堆叠 LSTM 预测模型。

在此基础上,将 $t+1$ 时刻的预测值 $\tilde{Y}_{t+1}$ 和实际测量值 Y_{t+1} 送入预测残差检测器,计算预测残差:

$$e = \tilde{Y}_{t+1} - Y_{t+1} \quad (3)$$

若预测残差 e 超过设置的监测阈值 Thr , 则认为被监测参数偏离其正常运行状态,即该参数对应的设备出现异常/故障。基于堆叠 LSTM 的智能状态监测模型详细建模过程及监测阈值 Thr 选取如文献[21]所示。

1.2 机载多计算任务问题定义

在机载状态监测中,利用 FPGA 的低功耗、低延迟、高并行硬件定制计算加速优势,构建智能状态监测模型在 FPGA 中的定制计算加速单元,实现模型机载实时推理,是保障飞行安全的关键。但是,无人机通常需要对多个关键部件的飞行参数进行监测,形成了不同尺度的堆叠 LSTM 预测模型,产生了对应尺度的多计算任务。而在有限的 FPGA 计算资源下,无法将不同尺度的模型所对应的定制加速计算单元全部部署在 FPGA 中,此时,多计算任务只能对 FPGA 计算资源进行分时复用,完成所有模型的推理计算。

因此,如何合理选择相应模型尺度的定制加速计算单元在 FPGA 中部署,并在部署后对各尺度监测模型的计算任务进行优化调度,成为决定机载智能状态监测多计算任务运行时间的关键问题。

本文采用两个实例分别介绍多任务调度方式和定制计算加速单元部署对多任务执行时间的影响。假设待执行的计算任务尺度分别为 20、30、40、50 和 60,对应的计算任务数量分别为 2、3、2、2 和 1,共 10 个任务。通过将各尺度监测模型对应的定制计算加速单元在 FPGA 中部署运行,测得各尺度模型实时推理一次所消耗的时间分别为 0.381、0.561、0.765、0.994 和 1.247 ms。不同加速单元消耗的查找表 (look-up-table, LUT)、触发器 (flip-flop, FF)、硬件乘累加 (digital signal processing, DSP) 和块随机存储器 (block random-access memory, BRAM) 等 FPGA 计算资源百分比如图 2 所示。假设将所有尺度的定制计算加速单元全部部署于 FPGA 中,则 LUT、DSP、BRAM 等资源消耗占比均超过 100%。表明实际应用中无法全部部署,只能选择部分尺度的定制加速计算单元进行部署。此时,小尺度的监测模型可以复用大尺度监测模型的定制计算加速单元完成模型实时推理,但消耗的时间为大尺度监测模型的计算时间。

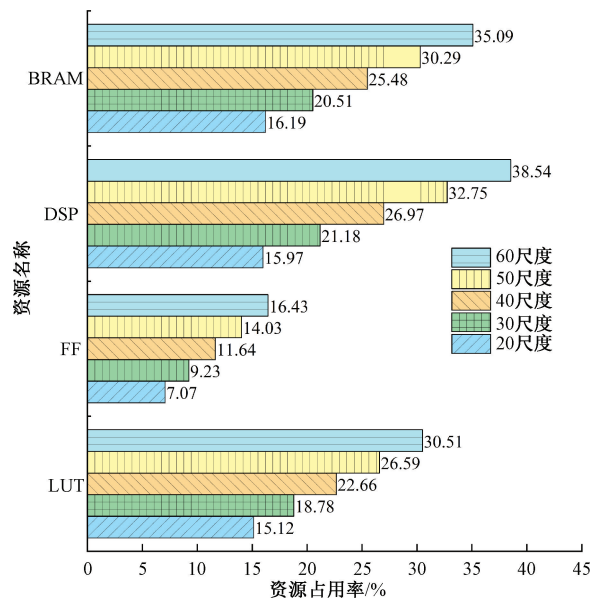


图2 不同尺度加速计算单元的 FPGA 资源消耗占比
Fig.2 Proportion of FPGA resources consumed by accelerated computing units with different scales

针对上述有限 FPGA 资源下多计算任务执行场景,多任务调度方式和定制计算加速单元部署对多计算任务执行时间的影响分析如下:

1) 多任务调度方式对执行时间的影响

在 FPGA 中同时部署尺度为 40、50 和 60 的 3 个定制计算加速单元,并由 10 个计算任务共享上述单元完成计算。如图 3 所示,图 3(a) 采用轮询的调度方式执行 10 个计算任务,时间耗时为 3.741 ms。图 3(b) 采用以最小任务完成时间为优化目标的多任务调度方式,执行 10 个计算任务耗时为 3.060 ms。表明在部署相同的定制计算加速单元情况下,不同的多任务调度方式,对任务总执行时间有着明显影响。

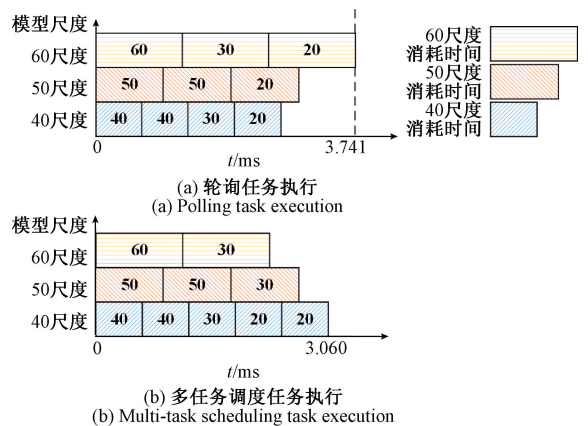
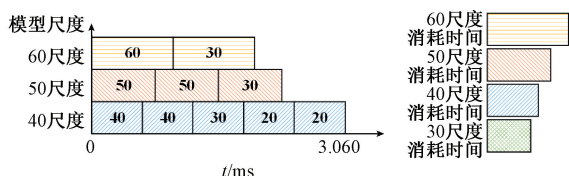


图3 部署相同的加速单元时不同任务调度方式的执行时间
Fig.3 Execution time of different task scheduling methods when deploying the same acceleration units

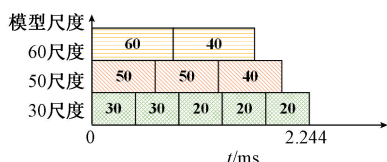
2) 定制计算加速单元部署对执行时间的影响

采用与图 3(b) 相同的多任务调度方式, 分析不同加速单元部署方案对多任务执行效率的影响。如图 4 所示, 图 4(a) 为 FPGA 中部署 3 个尺度分别为 40、50 和 60 尺度的加速单元, 总耗时为 3.060 ms; 图 4(b) 为 FPGA 中部署 3 个尺度分别为 30、50 和 60 尺度的加速单元, 总耗时为 2.244 ms。表明在相同的多任务调度方式下, 不同的定制计算加速单元部署方案, 对任务总执行时间有着显著影响。



(a) 部署60、50和40尺度加速计算单元任务执行

(a) Deploy 60, 50 and 40 scale accelerated computing unit task execution

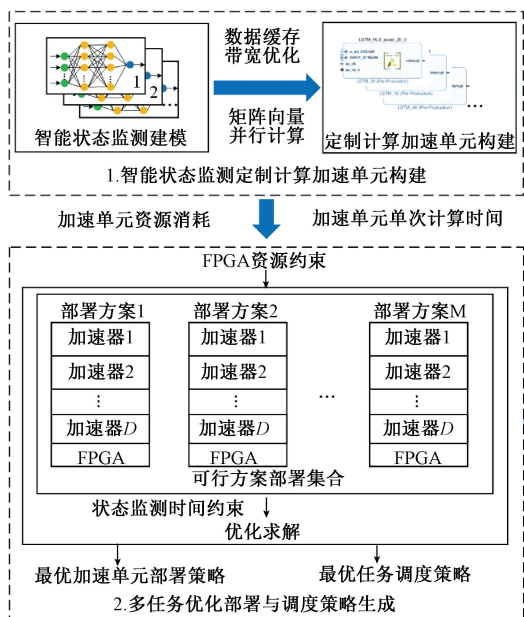


(b) 部署60、50和30尺度加速计算单元任务执行

(b) Deploy 60, 50 and 30 scale accelerated computing unit task execution

图4 相同任务调度方式下不同加速单元部署的执行时间

Fig.4 Execution time of different acceleration unit deployment schemes using the same task scheduling method



综上所述, 不同的多任务调度方式和不同的定制计算加速单元部署方案均会对多计算任务的总完成时间产生明显影响。因此, 在有限的 FPGA 资源下, 获取最优的加速单元部署方案和多计算任务调度策略, 提升多任务完成效率, 成为本研究解决的关键问题。

2 机载智能状态监测单元

针对机载多计算任务高效运行问题, 提出有限 FPGA 资源下多计算任务调度优化的机载智能状态监测单元, 如图 5 所示, 其包括智能状态监测定制计算加速单元构建、多任务优化部署与调度策略生成、以及机载智能状态监测单元构建 3 部分。其中, 智能状态监测定制计算加速单元构建部分为不同飞行参数构建合适尺度的智能状态监测模型, 并基于 FPGA 建立对应尺度的定制加速计算单元, 提升单一模型的计算效率; 多任务优化部署和调度策略生成部分则在 FPGA 计算资源约束下, 以最短任务完成时间为寻优目标, 获得最优加速单元的部署方案和多计算任务调度策略; 机载智能状态监测单元构建则根据加速单元的最优部署方案, 在 FPGA 中完成对应加速单元部署, 并根据多任务调度策略实现多个计算任务与有限计算资源的匹配, 保证机载状态监测模型高效推理。

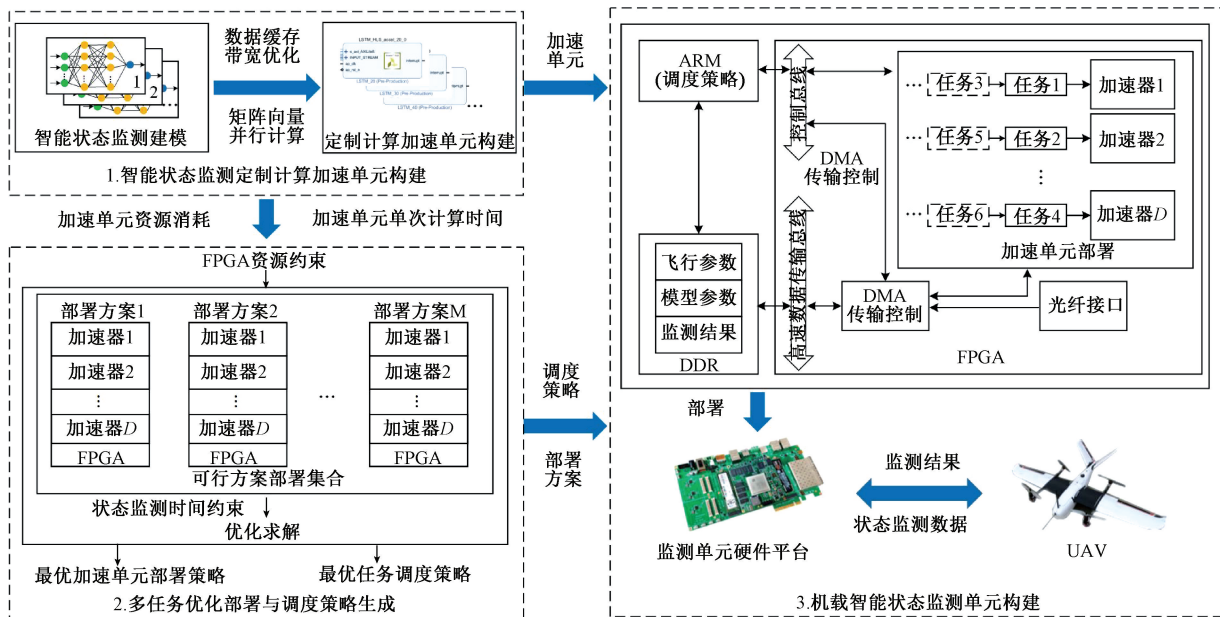


图5 有限 FPGA 资源下多计算任务调度优化的机载智能状态监测单元原理框图

Fig.5 Schematic diagram of airborne intelligent condition monitoring unit based on multiple computing tasks scheduling optimization under limited FPGA resources

2.1 智能状态监测定制计算加速单元构建

首先, 针对无人机的状态监测需求, 采用 1.1 节基于

堆叠 LSTM 的智能状态监测方法, 建立多个被监测参数的监测模型。由文献[14]可知, 智能状态监测模型中最

主要的计算负载为矩阵向量乘法,为在机载端提升单个智能监测模型的计算效率,并满足低延迟处理需求,采用FPGA 高层次综合 (high-level synthesis, HLS) 构建监测模型的定制计算加速单元,分别从模型数据缓存的带宽优化和矩阵向量乘法的并行计算两方面进行硬件计算优化^[22],如图6所示。

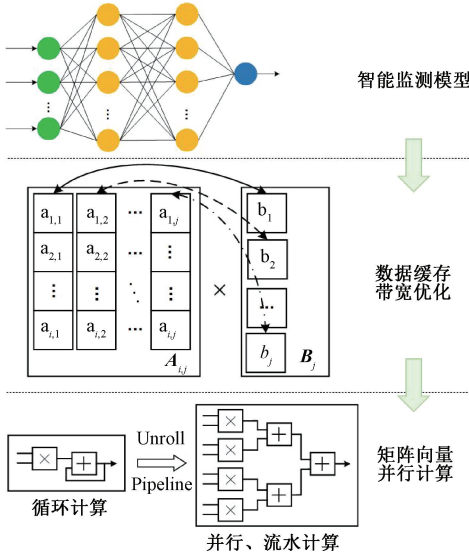


图6 基于HLS的状态监测模型定制计算加速^[23]

Fig. 6 HLS-based customized computing acceleration for the condition monitoring model

在模型数据缓存带宽优化方面,状态监测模型的数据主要缓存在FPGA的BRAM中,然而,BRAM在同一个时刻内只能获取两个数据,若将整个矩阵中的数据存储至一个BRAM中,数据读取带宽便会受到影响,进而影响矩阵向量计算的并行度。为此,本文采用HLS的矩阵分割指令将LSTM中的矩阵 $A_{i,j}$ 和向量 B_j 分割为 j 个数据块,每个数据块使用单独的BRAM进行存储。并采用矩阵重构指令对上述数据块进行重新排序,以满足矩阵和向量并行、流水的数据访问需求,完成模型数据缓存带宽优化。

在矩阵向量乘法的并行计算方面,重点对智能状态监测模型内的乘累加循环操作进行优化。首先,采用HLS流水指令对具有数据依赖关系的循环操作进行流水计算优化。然后,采用HLS展开指令将无数据依赖关系的循环操作全部展开或部分展开,实现完全并行计算。

在FPGA中完成上述不同尺度的状态监测模型HLS优化后,获得各模型对应的定制计算加速单元,用于机载智能状态监测单元的模型部署和推理计算。各加速单元的计算架构相同,模型参数可配置,但尺度不同,小尺度的模型可以在大尺度模型对应的定制计算加速单元中完成推理计算。

2.2 多任务优化部署和调度策略生成

通过智能状态监测定制计算加速单元构建,得到无人机各被监测参数的最佳状态监测模型,及其对应尺度的定制计算加速单元所消耗的FPGA资源和单次推理的时间。在有限的FPGA资源下,为获得多计算任务的最佳部署方案 p 与任务调度策略 s ,最小化多计算任务的执行时间。首先,需明确部署 D 种尺度的定制计算加速单元时的资源约束条件,如下式所示。

$$\begin{cases} \sum_{d=1}^D n_d \times IP_d^{DSP} \leq Q_{DSP} \\ \sum_{d=1}^D n_d \times IP_d^{BRAM} \leq Q_{BRAM} \\ \sum_{d=1}^D n_d \times IP_d^{FF} \leq Q_{FF} \\ \sum_{d=1}^D n_d \times IP_d^{LUT} \leq Q_{LUT} \end{cases} \quad (4)$$

其中, n_d 表示第 d 种尺度的定制计算加速单元在FPGA中部署的数量, IP_d^{DSP} 、 IP_d^{BRAM} 、 IP_d^{FF} 和 IP_d^{LUT} 分别表示部署一个第 d 种尺度的加速单元所消耗的DSP、BRAM、FF和LUT资源, Q_{DSP} 、 Q_{BRAM} 、 Q_{FF} 和 Q_{LUT} 分别为FPGA中上述4种资源的总量。

其次,多计算任务完成的时间 t_o 需要小于等于可接受的时间 T ,即状态监测的实时性约束,记为:

$$t_o \leq T \quad (5)$$

再次,假设有 N 种多任务调度策略,其中第 i 种调度策略中, t_d 表示第 d 种尺度的加速单元单次计算所消耗的时间, e_d 表示第 d 种尺度的加速单元被调度占用的次数,则在同一个加速单元部署方案中,第 i 种调度策略的总任务完成时间 t_i 为 $t_d \times e_d$ 的最大值,记为:

$$t_i = \max_{1 \leq d \leq D} (t_d \times e_d) \quad (6)$$

最后,以完成时间最小化为多任务优化部署和调度策略生成的最终目标,记为:

$$t = \min_{1 \leq i \leq N} (t_i) \quad (7)$$

其中, t_i 为第 i 种部署方案下多任务的完成时间。

在构建上述多任务优化部署和调度策略生成目标后,最优加速单元部署和任务调度策略求解步骤为:

1) 根据FPGA的计算资源、各尺度定制计算加速单元的资源消耗,采用穷举法获取满足式(4)的部署方案集合:

$$P = \{p_1, p_2, \dots, p_M\} \quad (8)$$

其中, M 表示可行的部署方案数量。

2) 从集合 P 中选取一种可行的部署方案 p_m ,构建式(6)和(7)所示的优化目标。采用粒子群优化算法求解当前部署方案下的最优调度策略。

3) 判断部署方案 p_m 下, 最优调度策略的多计算任务完成时间是否小于当前最优完成时间 t 。若是, 则更新加速单元部署方案 p 和多任务调度策略 s 。

4) 判断集合 P 中可行部署方案是否遍历完成, 若否, 则更新部署策略 p_m , 执行步骤 2), 否则执行步骤 5)。

5) 判断 t 是否满足式 (5), 若是, 输出当前的 p 和 s 作为最优加速单元部署方案和多任务调度策略; 否则, 当前配置下无法满足实时状态监测要求。

上述求解流程如图 7 所示。

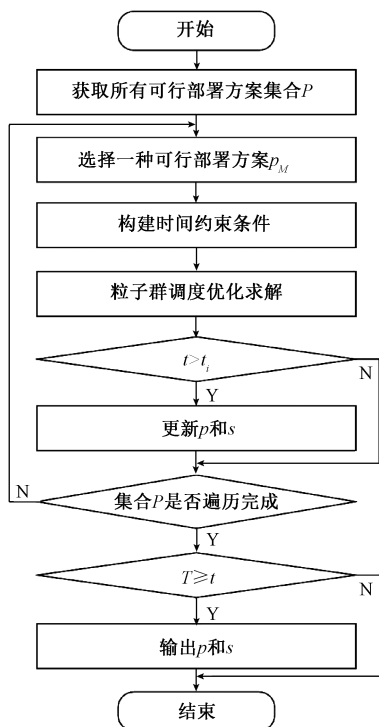


图7 多任务优化部署和调度策略生成流程

Fig. 7 Flowchart of multi-task optimal deployment and scheduling strategy generation

2.3 机载智能状态监测单元构建

在获取最优加速单元部署方案和多任务调度策略后, 构建如图 8 所示的机载智能状态监测单元, 利用光纤接口采集机载状态监测数据, 采用 ARM+FPGA 的异构处理器 Zynq UltraScale+ MPSoC MZU07A-EV 作为状态监测单元的监测数据处理核心, 并使用动态随机存储器 (double data rate, DDR) 实现状态监测模型参数和状态监测数据的大容量缓存。其中, FPGA 根据寻优后获得的定制计算加速单元部署方案, 完成选定的加速单元在 FPGA 中的配置; ARM 则根据寻优后获得多任务调度策略, 完成多个计算任务在部署加速单元上的执行顺序控制。具体控制过程如下:

1) 采用 FPGA 中的直接存储器访问 (direct memory access, DMA) 将光纤接口接收到的状态监测数据传输至 DDR 中进行缓存。

2) ARM 按照多任务调度策略控制 DMA, 逐一将 DDR 中缓存的不同监测模型参数和接收的状态监测数据配置到 FPGA 中对应尺度的加速单元。

3) FPGA 中部署的加速单元在调度策略的控制下完成多计算任务, 并将计算结果反馈至 DDR。

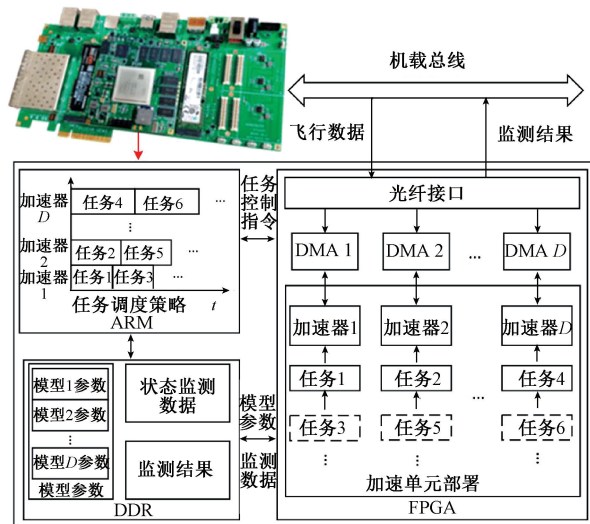


图8 机载智能状态监测单元

Fig. 8 Airborne intelligent condition monitoring unit

3 实验验证与结果分析

本文采用无人机的实际状态监测数据对有限 FPGA 资源下多计算任务调度优化的机载智能状态监测单元的有效性进行验证。

3.1 实验数据

为验证机载智能状态监测单元的性能, 选取俯仰角、滚转角、偏航角、俯仰角速率、滚转角速率、偏航角速率、副翼舵、升降舵、偏航舵和油门等 10 个飞行参数进行监测。使用表 1 中除所需监测的参数外的其余 16 个参数作为状态监测模型的输入, 通过式 (1) ~ (3), 构建被监测飞行参数的监测模型。

通常情况下, 无人机在整个飞行过程中处于巡航的时间最长。因此, 本文选取巡航阶段的飞行数据进行验证, 如图 9 所示, 为归一化后的飞行轨迹片段, 共 30 000 个样本点。采用式 (1) 对数据重构后, 选取前 80% 的数据对各状态监测模型进行训练, 剩余 20% 的数据注入最大值的 10% 作为故障幅值, 用于对模型的监测异常性能进行验证。

3.2 实验设置

本文根据文献 [24] 共设计 4 组实验, 实验 4 为本文提出的对定制计算加速单元部署和任务调度进行联合

表1 无人机状态监测建模采用的飞行参数

Table 1 Flight parameters used in UAV condition monitoring modeling

标号	飞行参数	单位	标号	飞行参数	单位
1	海拔高度	m	10	油门	%
2	纬度	(°)	11	空速	m/s
3	经度	(°)	12	俯仰角	(°)
4	东北天 X 轴速度	m/s	13	滚转角	(°)
5	东北天 Y 轴速度	m/s	14	偏航角	(°)
6	东北天 Z 轴速度	m/s	15	俯仰角速率	(°)/s
7	副翼舵	(°)	16	滚转角速率	(°)/s
8	升降舵	(°)	17	偏航角速率	(°)/s
9	偏航舵	(°)			

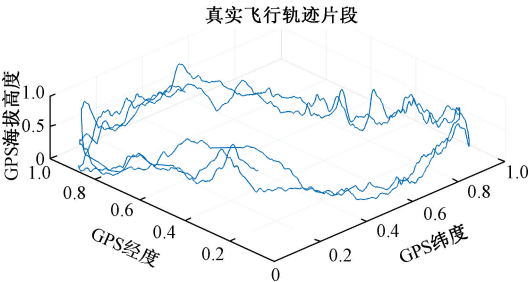


图9 巡航阶段归一化飞行轨迹

Fig.9 Normalized flight trajectory during cruise

优化方法的验证。实验1中任务采用串行执行,以验证实验4的任务调度和定制计算加速单元部署优势。实验2和4采用相同的定制计算加速单元部署方案,实验2中任务执行采用轮询调度策略,以验证实验4的任务调度策略性能。实验3和4采用相同的任务调度优化,不同的定制计算加速单元部署方案,以验证实验4联合优化的性能。各实验具体介绍如下:

1)实验1:本实验中计算任务的总数为10。按照无人机飞行控制周期要求,多计算任务的完成时间要小于20 ms。在FPGA中部署尺度为60、50和30的定制计算加速器各一个,任务调度策略采用顺序执行的方式,在部署加速计算单元上逐一串行执行,即任务到来后,选择对应尺度的加速单元,待该计算任务结束后,执行下一个计算任务,直至完成所有计算任务,以验证多计算任务串行执行所消耗的时间。

2)实验2:采用与实验1相同的多计算任务、时间约束、以及加速单元部署方案,但多计算任务调度采用轮询策略,即计算任务到来后,选择当前空闲的加速单元,且其尺度需要大于等于计算任务的尺度,如不满足上述条件,则判断下一个待执行任务是否满足上述条件,直至所

有任务完成计算,以验证多计算任务轮询执行所消耗的时间。

3)实验3:采用与实验1相同的多任务和时间约束,但加速单元部署方案采用尺度为60、50和40的加速单元各一个。以最小化多任务完成时间为目标,采用粒子群优化算法求取该加速单元部署条件下的多任务调度策略。其中,优化算法参数设置如下:粒子数为100,惯性常数为0.7,加速常数为2,迭代次数100。以验证在采用多任务调度优化的情况下,本文同时优化加速单元部署方案的必要性。

4)实验4:采用和实验1相同的多任务和时间约束。并使用本文多任务优化部署和调度策略生成方法获取优化的加速单元部署方案和多任务调度策略,其中,优化目标求解算法的参数与实验3保持一致。以验证同时对加速单元部署方案和多任务调度策略进行优化,对提高多计算任务完成效率的必要性。

3.3 评价指标

本文对机载智能状态监测单元的评价主要包括两个方面:1)对状态监测模型的性能评价,采用测试数据集评估不同状态监测模型的监测效果,评价指标为受试者工作特性曲线(receiver operating curve, ROC)下的面积(area under the ROC curve, AUC),AUC取值范围为[0,1],AUC值越大表明监测方法对故障的监测能力越强。2)对多计算任务调度优化性能进行评价,采用多计算任务完成时间作为调度性能的评判指标^[25]。多计算任务完成时间是开始运行到最后一个任务输出结果的时间,完成时间越短,则调度策略服务质量越高。

3.4 实验结果与分析

实验验证中首先通过均衡AUC指标和FPGA资源消耗,完成不同参数状态监测模型的尺度选择;在此基础上,开展上述4组实验,对比本文提方法和其他3组实验完成10个状态监测任务的任务完成时间,来对机载智能状态监测单元的性能进行评估。

1)不同飞行参数状态监测模型的尺度选择

对不同的飞行参数搜索不同的监测模型尺度,并计算其AUC,如图10所示。随着模型尺度增大,监测模型的AUC并不是呈递增趋势。而是当模型尺度达到一定大小时,监测性能不再大幅提高,部分存在下降情况。此时,若选用大尺度模型进行状态监测,不仅带来FPGA资源浪费和计算时间的延长,还影响监测精度。

不同飞行参数监测模型尺度选择方法:首先,测试不同尺度监测模型在FPGA中部署所消耗的计算资源,结果如图2所示;然后,对同一飞行参数设置不同的监测模型尺度,计算各尺度下的AUC值,如图10所示;最后,当AUC随模型尺度变化较大时,选择大AUC值中小尺度的

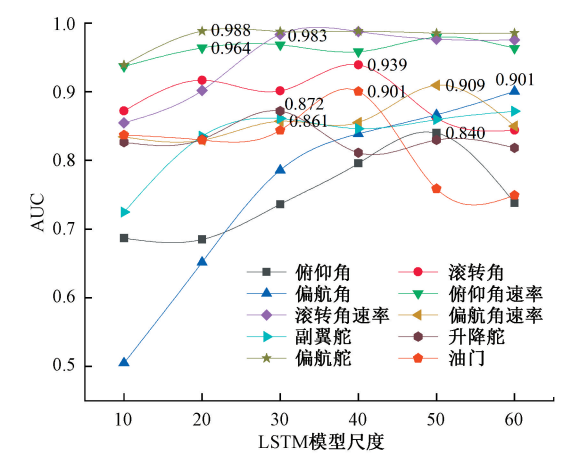


图 10 不同参数使用不同尺度模型监测 AUC 指标图
Fig. 10 The AUC index graph for Different parameters monitored by models with different scales

模型作为最终监测模型的尺度,当 AUC 值随模型尺度波动平缓时,选择 AUC 值趋于相对平稳后的第一个模型尺度,作为最终监测模型的尺度。10 个飞行参数的监测任务选择的模型尺度和对应的 AUC 如表 2 所示,共使用了 5 种不同尺度的状态监测模型,模型尺度分别为 20、30、40、50 和 60。通过实际部署测量上述尺度模型对应加速单元单次计算所消耗时间分别为 0.380、0.561、0.765、0.994 和 1.247 ms。10 个飞行参数的状态监测平均 AUC 为 0.916。

表 2 不同飞行参数的状态监测模型尺度及 AUC
Table 2 Condition monitoring model scale and AUC for different flight parameters

参数名称	状态监测模型尺度	AUC	任务标识符
俯仰角	50	0.839	T1
滚转角	40	0.938	T2
偏航角	60	0.901	T3
俯仰角速率	20	0.964	T4
滚转角速率	30	0.983	T5
偏航角速率	50	0.909	T6
副翼舵	30	0.861	T7
升降舵	30	0.872	T8
偏航舵	20	0.988	T9
油门	40	0.901	T10

2) 机载智能状态监测单元多任务调度性能

机载智能状态监测单元在有限 FPGA 资源下的多计算任务调度性能验证实验设置如 3.2 节所述,4 组实验所消耗的 FPGA 计算资源如图 11 所示。

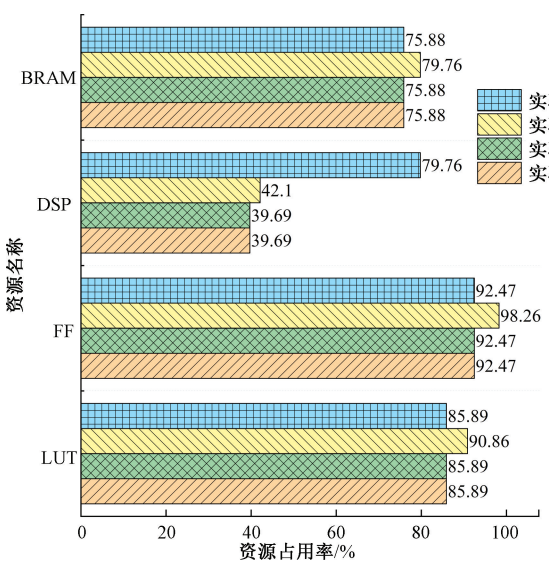


图 11 4 组实验对应 FPGA 资源的消耗占比
Fig. 11 Proportion of FPGA resource consumption corresponding to the four groups of experiments

实验 1~4 中加速单元部署与多任务调度执行的示意图分别如图 12~15 所示。

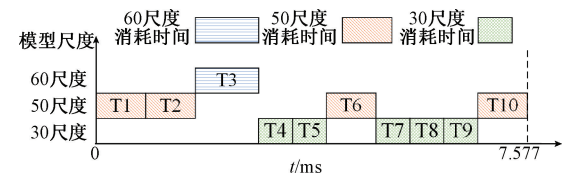


图 12 实验 1 加速单元部署与多任务调度执行结果
Fig. 12 Results of acceleration unit deployment and multi-task scheduling execution in experiment 1

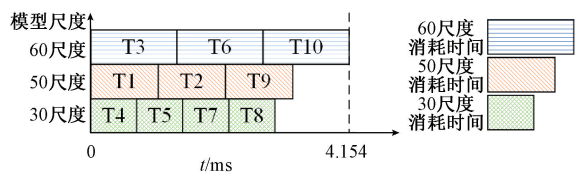


图 13 实验 2 加速单元部署与多任务调度执行结果
Fig. 13 Results of acceleration unit deployment and multi-task scheduling execution in experiment 2

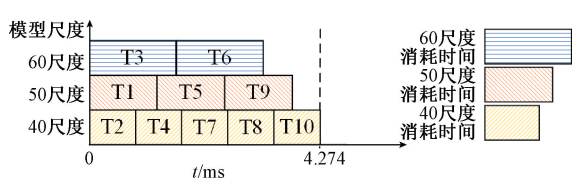


图 14 实验 3 加速单元部署与多任务调度执行结果
Fig. 14 Results of acceleration unit deployment and multi-task scheduling execution in experiment 3

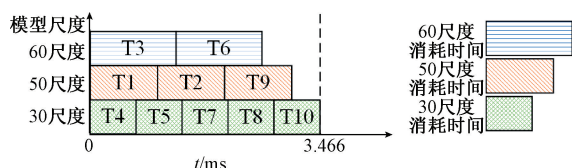


图15 实验4加速单元部署与多任务调度执行结果

Fig. 15 Results of acceleration unit deployment and multi-task scheduling execution in experiment 4

从图12~15中可以看出,实验1~4执行完相同的10个状态监测任务,多计算任务完成时间分别为7.577、4.154、4.274和3.466 ms。

对比实验1与2、3、4可知,多计算任务并行执行的效率明显优于串行执行的效率。

对比实验2和4可知,在均进行相同的加速单元优化部署的条件下,实验4以最小化多任务完成时间为目标进行多任务调度优化,对任务在所部署加速单元中执行的顺序进行了更加合理安排,使得实验4所消耗时间小于实验2,表明本文提出的多任务调度优化方法优于轮询调度方法。

对比实验3和4可知,在均以最小化多任务完成时间为目标进行多任务调度优化的条件下,实验4对加速单元的部署方案同步进行了优化。由于不同部署方案下加速单元单次执行所消耗的时间不同,使得不同部署方案每次运行都会产生时间差,直接影响多任务的完成时间。结果表明本文提出多任务优化的加速单元部署方案可以缩短多任务的完成时间。

综上所述,本文的机载智能状态监测单元在FPGA资源有限的情况下对定制计算加速单元优化部署和多任务执行优化调度,在执行相同的10个计算任务时,完成时间比串行执行、仅优化加速单元部署方案和仅优化任务调度策略,分别缩短了4.111、0.668、0.808 ms,即计算效率分别提升了54.26%、16.08%和18.91%。

4 结 论

针对无人机机载FPGA资源限制条件下,需要在规定时间内完成多个智能状态监测任务计算的问题。本文提出了一种有限FPGA资源下多计算任务调度优化的机载智能状态监测单元。首先,针对不同监测任务需求构建了不同尺度的智能状态监测模型,并基于FPGA建立了各模型的定制计算加速单元,提升了单一模型的计算

效率;然后以FPGA资源和完成时间为约束,建立了多任务优化部署和调度策略生成模型,实现了加速单元的优化部署和多任务的优化调度;最后,按照部署方案和调度策略构建了机载智能状态监测单元。采用实际飞行数据进行验证,与串行执行调度、仅优化加速单元部署方案和仅优化任务调度策略的方法相比,本文提出方法的计算效率分提升了54.26%、16.08%和18.91%。表明有限FPGA资源下,本文构建的状态监测单元同时优化加速单元部署方案和多任务调度策略,可显著优化状态监测任务的执行效率,保障机载监测的实时性。

未来将采用其他嵌入式计算平台进行验证,并研究在线动态调度策略获取方法,实现对调度策略的进一步优化。此外,该机载状态监测单元也将扩展应用于其他状态监测模型,对其功能进行拓展。

参考文献

- [1] 彭宇,何永福,王少军,等.飞行数据异常检测技术综述[J].仪器仪表学报,2019,40(3):1-13.
PENG Y, HE Y F, WANG SH J, et al. Flight data anomaly detection: A survey [J]. Chinese Journal of Scientific Instrument, 2019, 40(3): 1-13.
- [2] ZHAI Q Q, YE Z S. How reliable should military UAVs be? [J]. Quality & Reliability Engineering, 2020, 52(11): 1234-1245.
- [3] ZHANG Q, XIANG J, ZHANG Y. An active fault-tolerant control method for a low-cost and fixed-wing UAV[C]. IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC), 2016: 173-178.
- [4] 贺思捷,刘大同,彭宇.遥测数据驱动的无人机飞行状态识别方法[J].仪器仪表学报,2016,37(9):2004-2013.
HE S J, LIU D T, PENG Y. Flight mode recognition method of the unmanned aerial vehicle based on telemetric data [J]. Chinese Journal of Scientific Instrument, 2016, 37(9): 2004-2013.
- [5] WANG B K, LIU D T, PENG Y, et al. Multivariate regression-based fault detection and recovery of UAV flight data[J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69(6): 3527-3537.
- [6] WANG Y S, JIANG K, XU L. Sensor-data-driven fusion prognostic framework for complex engineered systems[J]. IEEE Sensors Journal, 2021, 21(18): 20421-20430.

- [7] 张玉杰, 彭宇, 刘大同. 飞机机电系统部件数据驱动健康状态在线估计方法综述[J]. 仪器仪表学报, 2022, 43(6): 118-130.
- ZHANG Y J, PENG Y, LIU D T. Review on data-driven health state on-line estimation methods for aircraft electro-mechanical system components[J]. Chinese Journal of Scientific Instrument, 2022, 43(6): 118-130.
- [8] ZHU H L, ZHU Y X, WU D, et al. Correlation coefficient based cluster data preprocessing and LSTM prediction model for time series data in large aircraft test flights[J]. Smart Computing and Communication, 2018, 11344: 376-385.
- [9] XU C, JIANG S, LUO G J, et al. The case for FPGA-based edge computing[J]. IEEE Transactions on Mobile Computing, 2022, 21(7): 2610-2619.
- [10] LIU X, YANG J F, ZOU C M, et al. Collaborative edge computing with FPGA-based CNN accelerators for energy-efficient and time-aware face tracking system[J]. IEEE Transactions on Computational Social Systems, 2022, 9(1): 252-266.
- [11] WANG B K, CHEN Y F, LIU D, et al. An embedded intelligent system for on-line anomaly detection of unmanned aerial vehicle[J]. Journal of Intelligent & Fuzzy Systems, 2018, 34(6): 3535-3545.
- [12] MOOSBRUGGER P, ROZIER K Y, SCHUMANN J. R2U2: Monitoring and diagnosis of security threats for unmanned aerial systems[J]. Lecture Notes in Computer Science, 2017, 51: 31-61.
- [13] WANG B K, PENG X Y, JIANG M, et al. Real-time fault detection for UAV based on model acceleration engine[J]. IEEE Transactions on Instrumentation and Measurement, 2020, 69(12): 9505-9516.
- [14] HARIHARAN I, KANNAN M. Algorithms for reducing reconfiguration overheads using prefetch, reuse, and optimal mapping of tasks [J]. Concurrency and Computation: Practice and Experience, 2021, 33(7): 1.
- [15] REZA R. A prefetch-aware scheduling for FPGA-based multi-task graph systems [J]. The Journal of Supercomputing, 2020, 76: 7140-7160.
- [16] ZHU Z W, ZHANG J N, ZHAO J J, et al. A hardware and software task-scheduling framework based on CPU+ FPGA heterogeneous architecture in edge computing[J]. IEEE Access, 2019, 7: 148975-148988.
- [17] 戴紫彬, 曲彤洲. 基于预配置和配置重用的粗粒度动态可重构系统任务调度技术[J]. 电子与信息学报, 2019, 41(6): 1458-1465.
- DAI Z B, QU T ZH. Task scheduling technology for coarse-grained dynamic reconfigurable system based on configuration prefetching and reuse [J]. Journal of Electronics & Information Technology, 2019, 41(6): 1458-1465.
- [18] DU P, SUN Z C, ZHANG H T, et al. Feature-aware task scheduling on CPU-FPGA heterogeneous platforms[C]. 2019 IEEE 21st International Conference on High Performance Computing and Communications, 2019: 534-541.
- [19] CHANG L B, ZHANG S B, DU H M, et al. A reconfigurable neural network processor with tile-grained multicore pipeline for object detection on FPGA [J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2021, 29(11): 1967-1980.
- [20] 陶镛泽, 胡佳成, 施玉书, 等. 基于 LSTM 的矩形纳米光栅 AFM 图像复原方法[J]. 仪器仪表学报, 2021, 42(7): 50-57.
- TAO Y Z, HU J CH, SHI Y SH, et al. AFM image restoration method of rectangular nano grating based on LSTM [J]. Chinese Journal of Scientific Instrument, 2021, 42(7): 50-57.
- [21] WANG B K, CHEN B, LIU D T, et al. An embedded instrument for online condition monitoring of unmanned and autonomous systems [J]. IEEE Instrumentation & Measurement Magazine, 2022, 25(5): 23-29.
- [22] GEREON F, SEYIT H H. Automatic energy-minimized HW/SW partitioning for FPGA-accelerated MPSoCs [J]. IEEE Embedded Systems Letters, 2019, 11(3): 93-96.
- [23] 高琛, 张帆, 高彦钊. 利用数据稀疏性的 LSTM 加速器设计[J]. 电子学报, 2021, 49(2): 209-215.
- GAO CH, ZHANG F, GAO Y ZH. Design of LSTM accelerator by utilizing data sparseness [J]. Acta Electronica Sinica, 2021, 49(2): 209-215.
- [24] GEORGE C, LOSIF K, KYPRIANOS P, et al. Hardware task scheduling for partially reconfigurable FPGAs [C]. International Symposium on Applied Reconfigurable Computing, Bochum, Germany: 2015: 487-498.
- [25] BANSAL S, KUMAR P, SINGH K. An improved duplication strategy for scheduling precedence constrained

graphs in multiprocessor systems[J]. IEEE Transactions on Parallel and Distributed Systems, 2003, 14(6): 533-544.

作者简介



康守强, 2004 年于哈尔滨理工大学获得学士学位, 2007 年于哈尔滨理工大学获得硕士学位, 2011 年于白俄罗斯国立大学获得博士学位, 现为哈尔滨理工大学教授、博士生导师, 主要研究方向为系统状态感知与评估、测试信息智能处理和系统健康管理。

E-mail: kangshouqiang@163.com

Kang Shouqiang received his B.Sc. degree from Harbin University of Science and Technology in 2004, received his M.Sc. degree from Harbin University of Science and Technology in 2007, and received his Ph.D. degree from Belarusian State University in 2011. He is currently a professor and a Ph.D. advisor at Harbin University of Science and Technology. His main research interests include system state awareness and assessment, intelligent processing of test information, and system health management.



李坤, 2020 年于哈尔滨理工大学获得学士学位, 现为哈尔滨理工大学硕士研究生, 主要研究方向边缘智能计算和系统健康管理。

E-mail: Li_Kun97@163.com

Li Kun received his B.Sc. degree from Harbin University of Science and Technology in 2020. He is currently a master student at Harbin University of Science and Technology. His main research interests include edge intelligent computing and system health management.



王本宽(通信作者), 分别在 2013 年, 2015 年和 2022 年于哈尔滨工业大学获得学士学位、硕士学位和博士学位, 现为哈尔滨工业大学助理教授, 主要研究方向为边缘计算、系统状态感知与评估和测试信息智能处理。

E-mail: wangbenkuan@hit.edu.cn

Wang Benkuan (Corresponding author) received his B.Sc., M.Sc., and Ph.D. degrees all from Harbin Institute of Technology in 2013, 2015, and 2022, respectively. He is currently an assistant professor at Harbin Institute of Technology. His main research interests include edge computing, system state awareness and assessment, and intelligent processing of test information.



刘大同, 分别在 2003 年, 2005 年和 2010 年于哈尔滨工业大学获得学士学位、硕士学位和博士学位, 现为哈尔滨工业大学教授、博士生导师, 主要研究方向为复杂系统状态监测与测试、数据驱动的工业智能、无人系统健康管理、电子测量仪器与系统等。

E-mail: liudatong@hit.edu.cn

Liu Datong received his B.Sc., M.Sc., and Ph.D. degrees all from Harbin Institute of Technology in 2003, 2005, and 2010, respectively. He is currently a professor and a Ph.D. advisor at Harbin Institute of Technology. His main research interests include complex system condition monitoring and testing, data-driven industrial intelligence, health management for unmanned systems, electronic measuring instruments and systems, etc.