

基于 LSTM 的逻辑综合阶段延时预测方法

王庆康¹, 周冉冉^{1*}, 王永^{1,2}

(1.山东大学集成电路学院, 山东 济南 250101; 2.泉城实验室, 山东 济南 250103)

摘要:为了提高数字集成电路逻辑综合阶段的延时评估精度与效率,提出一种基于长短期记忆网络(long short-term memory, LSTM)的逻辑综合阶段延时预测方法。将时序路径表示为由标准单元构成的有序序列,提取并构建包含单元类型、扇出、负载电容和输入转换时间等关键特征参数的结构化序列数据;通过 LSTM 时序建模中的上下文记忆能力,捕捉路径中各级单元之间复杂的时序依赖关系,实现对路径延时的高精度预测。试验结果表明,对比现有对单元延时和线延时进行累加的机器学习估算方法,在预测精度上,基于 LSTM 的预测方法在保证高准确率的前提下,对不同类型的案例具有更好的适应性;在运行速度上,在多数测试案例上实现 2.8~3.2 倍加速。在无工艺信息的通用门级网表上验证本研究方法的预测能力,其表现优于传统静态时序分析方法,验证了该方法在早期设计阶段的有效性和应用前景。

关键词:长短期记忆网络;逻辑综合;静态时序分析;路径延时预测;机器学习

中图分类号:TP319; TN47 **文献标志码:**A

引用格式:王庆康,周冉冉,王永. 基于 LSTM 的逻辑综合阶段延时预测方法[J]. 山东大学学报(工学版), 2026, 56(2): 43-51.

WANG Qingkang, ZHOU Ranran, WANG Yong. A logic synthesis delay predicting method based on LSTM[J]. Journal of Shandong University (Engineering Science), 2026, 56(2): 43-51.

A logic synthesis delay predicting method based on LSTM

WANG Qingkang¹, ZHOU Ranran^{1*}, WANG Yong^{1,2}

(1. School of Integrated Circuits, Shandong University, Jinan 250101, Shandong, China; 2. Quan Cheng Laboratory, Jinan 250103, Shandong, China)

Abstract: To improve the prediction accuracy and efficiency for logic synthesis in digital integrated circuit design process, a logic synthesis delay predicting method based on long short-term memory (LSTM) was proposed. The timing path was treated as an ordered sequence of standard cells, and key feature parameters such as cell type, fanout, load capacitance, and input transition time were extracted and organized into structured sequence data. With the context memory capability of LSTM-based timing modeling, the complex timing dependencies between cells at different levels in the path were captured, achieving high-precision prediction of path delay. Experimental results showed that, compared to existing machine learning-based estimation methods that accumulate cell delays and wire delays, the LSTM-based prediction method demonstrated better adaptability to different types of cases while maintaining accuracy. In terms of running speed, a speedup of 2.8 to 3.2 times was achieved in most test cases. The prediction method was also validated on generic netlists without technology information and the performance was superior to traditional static timing analysis methods, demonstrating its effectiveness and potential for early-stage design applications.

Keywords: long short-term memory; logic synthesis; static timing analysis; path delay predicting; machine learning

收稿日期:2025-04-25

基金项目:国家重点研发计划资助项目(2021YFA1003604);国家自然科学基金资助项目(U23A20348)

第一作者简介:王庆康(1999—),男,安徽滁州人,硕士研究生,主要研究方向为电子设计自动化。E-mail:qingkangw@mail.sdu.edu.cn

* 通信作者简介:周冉冉(1990—),女,山东泰安人,高级实验师,硕士生导师,博士,主要研究方向为集成电路设计与自动化。

E-mail:rzhou@sdu.edu.cn

0 引言

在数字集成电路设计中,时序是决定电路性能的关键因素之一。在整个设计流程中,各阶段都需要进行静态时序分析(static timing analysis, STA),确保设计满足当前阶段的时序约束,才能进入下一阶段的设计^[1-2]。STA是一种无需激励矢量、通过静态遍历电路路径分析时序的手段,其核心包括延时建模、时序路径分析、裕量计算和关键路径识别等多个方面。分析结果不仅决定设计是否满足时序约束,也为后续时序优化提供指导依据。一旦某阶段发生时序违例,设计者需进行时序优化修复该违例。如果该时序违例无法通过当前阶段的时序优化解决,则必须退回至上一阶段重新进行设计,造成设计周期延长和设计成本上升。

在现代超大规模集成电路(very large scale integrated circuits, VLSI)设计中,逻辑综合起着至关重要的作用,其核心任务是将寄存器传输级(register-transfer level, RTL)描述转换为优化后的门级网表。在逻辑综合阶段,布局布线尚未完成,互连线的实际走线路径和物理长度未知,延时评估通常依赖简化的统计模型。然而,该统计模型通常基于历史布线数据和统计规律建立,其预测值与实际物理实现可能存在显著偏差。随着半导体制造技术进入几十纳米至几纳米节点,线延时在路径总延时中占据主导地位,逻辑结构对物理实现的敏感性增强,传统估算方式的误差进一步放大。这不仅降低关键路径识别的准确率,也影响时序优化的有效性,容易导致后续阶段出现难以解决的时序违例,增加设计迭代次数。

机器学习(machine learning, ML)方法广泛应用于数字VLSI设计不同阶段,以提升时序分析和性能预测的精度^[3-8]。当前已有大量研究致力于应用ML模型对布局^[9]、布线^[10-14]、时钟树综合^[15-17]、签核^[18-19]等后端阶段的延时指标进行建模与预测。这些方法多依赖于完整的物理信息,因此主要用于设计流程后期。相比之下,针对逻辑综合阶段的延时建模研究相对较少。现有相关工作多聚焦于关键路径预测^[20-21]、线网长度估计^[22-28]及线延时建模^[29]等方面。文献[20-21]使用词嵌入与卷积神经网络对逻辑路径进行建模,以判断布线后是否为关键路径;文献[22-27]从线网层面出发,通过建立多项式模型、人工神经网络(artificial neural network, ANN)、细粒度聚类算法或启发式算法估算线网长

度;文献[28]集成多个ML模型,将多个模型加权估计整条路径的线长;文献[29]采用随机森林方法分别预测线延时和负载;文献[30]首次提出在逻辑综合阶段预测路径延时的方法,分别训练两个模型估算门延时和线延时,但未能充分考虑路径结构的全局上下文特性,难以捕捉门与门之间的时序依赖关系。

当前基于ML的逻辑综合阶段时序预测方法存在以下不足:首先,现有线网长度估计和延时建模方法虽然能够预测单一路径延时或线延时,但缺乏对时序路径全局信息(如路径延时、裕量等)的充分利用,无法为后续设计阶段提供有效的时序优化指导;其次,现有方法大多在单个单元或单一线网层面进行时序预测,忽视了时序路径各部分之间的内在关联性。因此,这些方法不仅无法全面捕捉路径的全局时序特性,往往还需要较长的计算时间,限制它们在大规模设计中的实际应用。

为了克服上述方法的局限,本研究提出一种基于长短期记忆网络(long short-term memory, LSTM)的逻辑综合阶段路径延时预测方法,提取逻辑综合阶段的时序信息作为LSTM网络的序列特征(feature),以提高时序预测的准确性。所选特征基于与布线后延时密切相关的单元延时和线延时属性。由于LSTM网络擅长处理序列数据,本研究借鉴自然语言处理中的方法,将时序路径表示为由多个标准单元组成的特征序列。选取单元扇出、单元负载电容等关键参数作为序列每一级的特征,构建特征序列输入。以布线后路径延时为标签,训练LSTM模型进行路径延时预测。将该方法应用于数字集成电路逻辑综合阶段,可在尚未完成布局布线的早期阶段预判设计的时序风险,包括潜在的时序违例及严重程度,实现前置化修复,减少设计迭代次数。本研究方法不仅可以提升设计效率,也具备较强的推广应用潜力,可集成至现有逻辑综合工具中,辅助早期设计决策并优化时序收敛策略。

1 相关理论与技术

1.1 逻辑综合阶段延时估算方法

逻辑综合阶段的STA旨在评估门级网表在当前设计约束下的时序性能,是逻辑优化和路径修剪等流程的基础。然而,此阶段尚未完成布局布线,缺乏精确的物理信息,如单元位置、线网拓扑、耦合寄生等,因此,延时估算需依赖统计模型完成路径级分析。

门延时通常使用标准单元库中的查找表(lookup table, LUT)进行估算。LUT 以输入上升或下降时间和输出负载电容为二维输入,通过插值获得对应的延时。输出负载电容包括下一级门输入电容与连线的寄生电容,因此,线延时的建模精度直接影响路径级延时估算的准确性。

为弥补缺乏布线信息的不足,综合工具通常依赖线负载模型对线长进行估算。线负载模型如图 1 所示。模型根据线网扇出给出预估线长,进一步计算对应线网的电容和电阻。在该模型中, capacitance 和 resistance 分别表示单位长度互连线的电容与电阻, fanout_length(n, L) 指当线网扇出为 n 时,估计的总线长为 L 。综合工具根据当前网表中线网的扇出,从表中查得相应长度,再乘以单位电容或电阻估算线性寄生,从而估算线延时。尽管这种方式实现简便,但估算精度受两方面限制:一是线长的估计高度依赖扇出,未充分考虑实际物理分布、路径拓扑、设计拥塞等因素,导致误差随设计规模和复杂性增大而放大;二是该估算方法只关注局部节点,忽略时序路径整体的结构相关性,无法反映全局时序信息,在计算路径延时,误差容易在累加过程中放大。随着工艺的不断改进,线延时在路径总延时中的占比上升,线负载模型误差逐渐成为逻辑综合时序估算精度的主要瓶颈。因此,如何在缺乏精确布线信息的前提下提升路径级延时预测的准确性,成为综合阶段亟待解决的重要问题。

```
wire_load("1K_hvratio_1_4") {
  capacitance : 1.774000e-01;
  resistance : 3.571429e-03;
  slope : 5.000000;
  fanout_length (1, 1.3207);
  fanout_length (2, 2.9813);
  fanout_length (3, 5.1135);
  fanout_length (4, 7.6639);
  fanout_length (5, 10.0334);
  fanout_length (6, 12.2296);
  fanout_length (8, 19.3185);
}
```

图 1 线负载模型示例

Fig.1 Example of wire load model

1.2 词嵌入方法

为了将逻辑综合阶段的时序路径输入 LSTM 模型中进行学习与预测,需要将路径表示为 LSTM 网络可以处理的向量序列^[31]。类似于自然语言处理中的词嵌入方法,本研究将一条时序路径形式化为来自标准单元库的一系列电路单元的链,每个单

元相当于一个词元(token),整条路径则类似于一句话。词嵌入的目的是将单词表示为一个由实数组成的向量,且词向量的语义依赖词汇的上下文。类似地,一条时序路径的延时也依赖路径结构及其组成单元的电气属性。

为了适应 LSTM 对序列化输入的要求,本研究将时序路径中的每个单元编码为包含单元类型、扇出、负载电容和输入转换时间等关键特征的向量,形成可供模型处理的特征序列。这些特征向量构成路径的嵌入表示,能够有效捕捉路径结构及时序信息。这种方式不仅将路径中的每个单元表示为一个向量,同时保留每个单元在电路设计中的功能和交互关系,为 LSTM 模型学习路径延时提供丰富的输入数据。

1.3 LSTM 网络

LSTM 由传统的循环神经网络(recurrent neural network, RNN)改进而来^[32],能够有效缓解传统 RNN 在处理长时序数据时出现的梯度消失和梯度爆炸问题。LSTM 通过引入门控机制,在时间步之间有选择地传递和更新信息,有效捕捉序列数据中的长期依赖关系。LSTM 由输入门、遗忘门和输出门 3 部分组成,使 LSTM 能够灵活地、选择性地存储或遗忘信息,确保重要的时序信息得到有效保存。

LSTM 已在自然语言处理、语音识别等领域取得广泛应用,但尚未在逻辑综合阶段的路径延时预测问题中得到应用。对于时序路径延时预测问题, LSTM 网络的优势主要体现在能够处理和学习路径中各个单元之间的复杂依赖关系。逻辑综合阶段的时序路径通常由多个单元通过互连线连接而成,单元的延时和互连线的延时共同决定路径的总延时。由于这些单元之间存在时序依赖关系,传统基于物理模型的延时估算方法难以全面捕捉这些复杂的相互影响。LSTM 网络作为一种强大的时序模型,能够通过门控机制学习路径中各个单元和延时的长期依赖,从而实现高精度的延时预测。

传统方法通常依赖手工提取特征,通过累加单元延时与估算线延时获取路径总延时,难以全面刻画路径中单元之间复杂的时序依赖关系。LSTM 网络具备自动特征学习能力,能够从数据中挖掘影响延时的潜在因素与非线性模式,提升预测的准确性和适应性。同时, LSTM 网络能够处理变长的时序输入,适应不同路径结构和拓扑复杂度,在建模灵活性方面具有天然优势。试验结果表明, LSTM 模型在预测精度上优于传统方法,在推理速度上也实现数量级提升,为高效时序分析与快速设计迭代提

供有力支持。

2 特征选择与数据处理

2.1 特征选择

在逻辑综合阶段进行路径延时预测时,特征选取对模型性能至关重要。为全面刻画路径中各个单元对总延时的影响,本研究选取4个具有代表性的特征:单元门延时、输入转换时间、输出负载电容及扇出。这些特征能够有效反映单元的驱动能力及其对后续单元的影响,是构建路径延时模型的关键数据基础。单元门延时反映在特定输入与负载条件下的信号传播时间,是路径总延时的重要组成部分;输入转换时间表示信号电平翻转所需的时间,对门电路的延时具有显著影响,转换时间越长,门延时越长;输出负载电容反映单元电路驱动的电容负载,对线延时和门延时均有影响;扇出衡量一个单元输出连接的后续单元数量,扇出增加会导致传播延时上升。其中,输入转换时间与输出负载电容为逻辑综合阶段基于简化模型估算所得,虽不如布线后精确,但均能提供有价值的单元特性,在路径延时预测中具有较强的指示作用。所有特征均可在逻辑综合阶段的时序报告中获得,具有良好的可获取性。经过标准化处理后,将特征构建为符合LSTM网络要求的时序输入序列,为后续高精度延时预测奠定基础。

2.2 数据处理

为构建高质量的延时预测模型,本研究对数据进行预处理与归一化操作。在逻辑综合阶段,由STA工具生成时序报告,从中提取每条时序路径上各级单元的关键时序特征作为输入特征数据。对应的预测标签来自布局布线完成后的时序报告,提取该路径在布线完成后的实际总延时作为监督学习中的真实值。时序路径在结构上存在差异,导致对应的特征序列长度不尽相同。为满足LSTM网络输入维度一致的要求,本研究采用统一长度的补齐策略。将统计训练集中所有路径的最大序列长度 $l_{en,max}$ 作为所有序列的长度,对长度不足 $l_{en,max}$ 的路径,在末尾填充全0向量。

在定义路径结构时,本研究将每条路径定义为从输入端到输出端的信号传播过程,或从起始触发器输出引脚到接收触发器输入引脚的信号传播过程。为提高训练数据质量,筛选阶段剔除不含组合逻辑、级数不超过2级的路径样本。

为避免特征维度间数值差异对模型训练产生

不利影响,本研究对所有输入特征进行归一化处理,即每个维度的样本值减去该维度样本均值后除以标准差,使特征分布具备0均值和单位方差。这种归一化处理不仅有助于提高模型收敛速度,还能显著增强训练稳定性与预测性能。

3 逻辑综合路径延时预测方法

3.1 整体框架

本研究提出一种基于LSTM的逻辑综合阶段延时预测方法,整体流程如图2所示。对输入的RTL设计执行逻辑综合,完成布局布线,分别在逻辑综合阶段和布线后调用STA工具,提取时序路径的相关信息。逻辑综合阶段生成的时序报告包含每条路径的结构及特征,如各级单元的类型、扇出、输入转换时间与输出负载电容等。布线后的报告包含真实路径延时信息,作为监督学习的标签。通过起点与终点标识对路径进行匹配,建立逻辑综合路径与延时标签的一一对应关系。随后,将时序路径建模为由多个标准单元构成的有序序列,并转换为结构化的特征序列,作为神经网络的输入。将该特征序列输入LSTM网络,用以预测路径延时,从而在逻辑综合阶段实现对后续时序表现的准确预估。

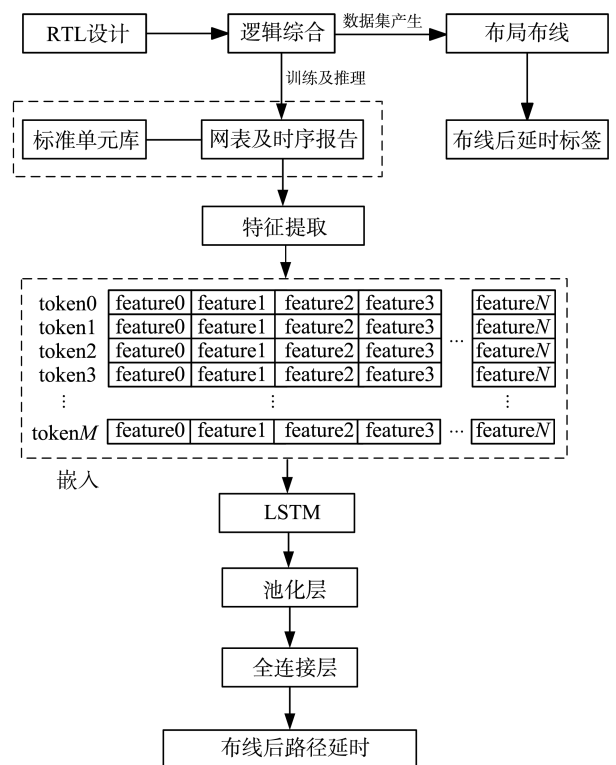


图2 逻辑综合阶段延时预测方法流程图
Fig.2 Flow diagram of timing prediction method for logic synthesis stage

3.2 时序路径的嵌入

电路路径由一系列离散、异质性的逻辑单元组成。这些单元具有不同的电气属性与功能。因此,如何有效地将这些离散单元转化为神经网络能够处理的连续数值输入,是构建路径延时预测模型的关键前提。为了解决这一问题,本研究采用词向量(vector)嵌入机制,将电气特征融入电路单元的向量化表示中。将离散的电路单元映射到低维的连续向量空间,向量编码由该单元的关键电气参数(如扇出、输入转换时间、输出负载电容等特征)组成。

时序路径嵌入过程如图 3 所示。将多个标准单元组成的时序路径表示为一个特征向量序列,每个序列元素对应路径中的一个逻辑单元,例如 D 触发器(D flip-flop, DFF)、反相器(inverter, INV)、2 输入与门(two-input AND gate, AND2)、2 输入或门(two-input OR gate, OR2)、2 输入与非门(two-input NAND gate, NAND2)、2 输入异或门(two-input XOR gate, XOR2)、2 输入或非门(two-input NOR gate, NOR2)等。通过嵌入机制,将这些离散的单元类型映射为连续空间中的向量,构造出可供神经网络处理的数值序列数据。

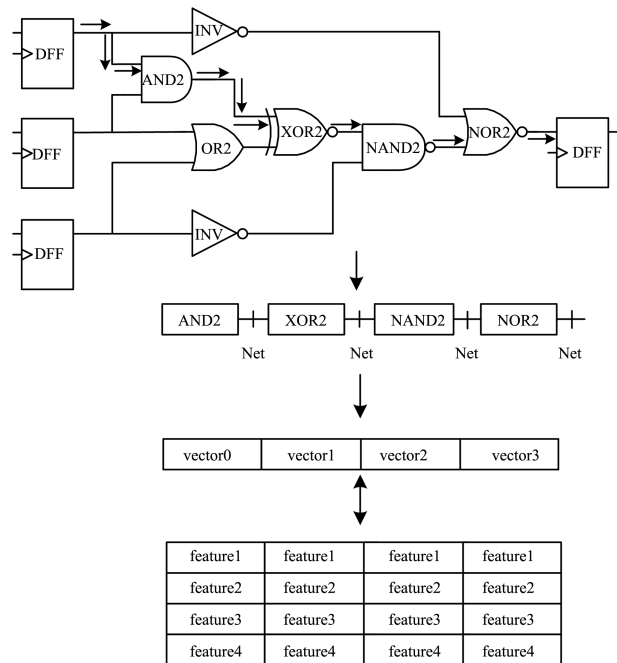


图 3 时序路径嵌入过程

Fig.3 Timing path embedding procedure

3.3 基于 LSTM 的路径延时预测

在构建结构化路径输入后,关键在于如何捕捉路径中各单元之间的复杂依赖关系。路径延时并非简单的门级求和过程,而是高度依赖前后级单元的逻辑结构和它们之间的电气连接特性。传统 STA 工具通常采用逐级传播的方式估算延时,缺

乏对路径上下文关系的建模能力。以图 3 所示的路径“DFF→AND2→XOR2→NAND2→NOR2→DFF”为例,在逻辑综合阶段,该路径的门级拓扑结构已知,但由于布局布线尚未完成,缺乏后端寄生参数。因此,在逻辑综合阶段,仅能基于估算的负载电容、标准单元特性和驱动能力对路径延时进行粗略预测。路径中 NAND2 的延时高度依赖前一级 XOR2 的输出负载变化,而该负载在布局布线完成后才能准确计算。此外,AND2 与 XOR2 之间的连接可能因布线拓扑不同而存在不同的走线长度,从而造成显著的延时差异。这种基于不完整信息所做的估计易导致误差累积,影响预测精度。

为缓解这一问题,本研究利用 LSTM 网络的时序建模能力捕捉路径中单元间的复杂依赖关系。LSTM 网络通过递归结构,在处理序列数据时能够记忆前一时刻的状态,在当前计算中融合上下文信息,特别适用于处理此类路径中存在强时序依赖的序列建模问题。在每一级的计算中,LSTM 网络会综合当前级的输入信息和前一时刻的隐藏状态,捕捉前后级之间的依赖关系。与传统 STA 方法不同,LSTM 在保留路径中每个单元局部特性的同时,还能学习它们在整条路径中所处的位置和不同单元之间的时序依赖,提升模型对路径延时的准确预测能力。

LSTM 网络结构如图 4 所示。将逻辑综合阶段提取的路径特征序列 $\{X_1, X_2, X_3, \dots, X_m\}$ 输入多层 LSTM 单元中,逐层处理以捕捉路径中单元之间复杂的时序依赖关系,其中 X_m 为第 m 个路径特征向量。隐藏层 h_m 存储从路径起点到当前位置的动态特征信息,隐藏层的输出经过池化操作压缩时间维度信息,传入全连接层完成特征融合与延时回归预测。该结构能够有效利用 LSTM 的记忆与顺序建模能力,实现对复杂逻辑路径延时的高精度预测。

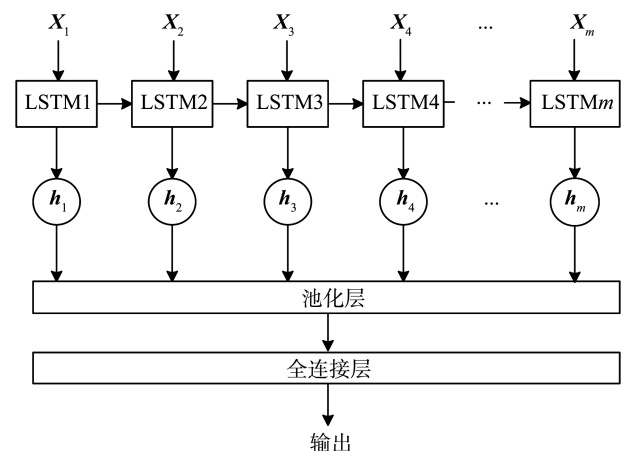


图 4 LSTM 网络结构

Fig.4 The network structure of LSTM

4 试验结果与分析

4.1 试验设置

本研究采用 Python 编程语言,基于 PyTorch 深度学习框架实现该逻辑综合阶段的路径延时预测方法。为评估该模型在不同电路设计间的泛化能力,构建一个覆盖多种电路结构与复杂度的多样化数据集。该数据集包含来自 ITC'99、ANUBIS、Faraday、Gaisler 和 OpenCores 共 5 个公开测试基准的 10 个代表性数字电路设计,具体信息如表 1 所示,其中, n_{net} 为设计网表包含的连线数, n_{cell} 为设计网表包含的门单元数。不同设计案例的路径长度分布(即不同路径长度样本的比例)如表 2 所示。所有设计均基于 45 nm FreePDK 工艺,通过 Cadence Genus v20.12 工具完成逻辑综合流程,并通过 Cadence Innovus v20.10 工具完成布局与布线。从逻辑综合阶段与布线后阶段的 STA 报告中分别提取时序路径特征与实际路径延时,构成用于训练与评估模型的输入与标签数据。该试验配置确保数据的全面性和代表性,为后续模型性能评估提供可靠基础。

表 1 数据集概况
Table 1 Overview of the dataset

基准库	设计案例	$n_{\text{net}}/10^5$ 条	$n_{\text{cell}}/10^5$ 个
ITC'99	B18	1.15	0.51
	B21	0.39	0.19
ANUBIS	Or1200	8.47	5.80
	Morlkx	1.78	0.72
	ALPHA	0.41	0.14
Faraday	RISC	0.98	0.40
	Leon2	8.35	5.84
Gaisler	Leon3mp	6.40	3.92
	Netcard	5.51	3.01
OpenCores	Ethernet	0.71	0.31

表 2 数据集中路径长度分布
Table 2 Distribution of path length of the dataset

设计案例	路径长度分布/%			
	3~7	8~11	12~15	≥ 16
B18	52	43	5	0
B21	39	57	4	0
Or1200	23	5	46	26
Morlkx	26	2	40	32
ALPHA	87	12	1	0
RISC	84	14	2	0
Leon2	3	4	9	84
Leon3mp	21	6	49	24
Netcard	30	21	28	21

注:3~7 表示路径长度包含 3~7 个门单元,8~11 表示路径长度包含 8~11 个门单元,12~15 表示路径长度包含 12~15 个门单元, ≥ 16 表示路径长度包含不少于 16 个门单元。

为了确保试验的科学性与合理性,同时全面验证所提方法的实用性与有效性,本试验采用交叉验证方式评估模型性能,即每次以其中一个设计作为测试集,其余 9 个设计作为训练集,循环进行 10 轮试验。该设置确保测试数据与训练数据完全隔离,从而全面衡量模型在处理未知新设计时的泛化性能。模型训练过程中使用均方误差作为损失函数,优化器选择自适应矩估计(adaptive moment estimation, Adam)方法,初始学习率设为 0.005。为量化模型预测性能,采用决定系数 R^2 作为主要评估指标,衡量预测结果与实际延时之间的拟合程度。 R^2 的计算式为

$$R^2 = 1 - \frac{\sum_{i=1}^p (y_i - \hat{y}_i)^2}{\sum_{i=1}^p (y_i - \bar{y})^2},$$

式中, y_i 为第 i 个样本的实际值, \hat{y}_i 为第 i 个样本的预测值, \bar{y} 为所有实际值的平均, p 为样本数。本研究中 y_i 为采用 Innovus 中的延时计算引擎,对布局布线后的电路进行精确计算得到的路径延时。 R^2 越接近 1,表示预测效果越好。通过 R^2 可以清晰比较不同方法在整体预测精度上的优劣,尤其能体现所提方法在保持高准确性的同时减少误差累积的能力。

4.2 预测准确度分析

本研究所提框架预测准确率及与其他方法的比较结果如表 3 所示。对比方法包括两类:一类是通过 STA 工具输出结果的方法,分别记为 report_timing^{arc}方法和 report_timing^{path}方法,前者利用 STA 工具输出的弧延时(包括线延时和单元延时)作为预测目标,后者基于完整路径延时进行预测;另一类是文献[30]中提出的两种基线模型,分别记为 TCAD22^{arc}方法和 TCAD22^{path}方法,前者以单个弧延时为预测对象,后者直接预测整条路径的延时。在传统方法中,路径延时的计算通常建立在弧延时准确性的基础上,误差容易在累加过程中放大。

由表 3 可知,与 report_timing^{path}和 TCAD22^{path}相比,本研究方法在准确率上表现较优,平均 R^2 分别提高 0.91 和 0.04,显示出更强的拟合能力和泛化性能。相较于弧级预测累加的方式,路径级直接建模不仅能有效捕捉路径内部复杂的时序依赖关系,还可显著降低误差传播,进一步验证路径级建模在延时预测任务中的合理性与必要性。本研究方法在保证预测精度的同时,对不同样本的检测结果比较平均,不会出现 TCAD22 方法对不同电路预测误差差异较大的现象,说明本研究方法具有较好的适应性。

表 3 工艺映射阶段预测准确度对比
Table 3 Comparison of prediction accuracy at technology mapping stage

设计案例	R^2				
	report_timing ^{arc}	report_timing ^{path}	TCAD22 ^{arc} [30]	TCAD22 ^{path} [30]	本研究方法
B18	0.88	0.92	0.95	0.99	0.98
B21	0.77	0.57	0.92	0.99	0.98
Or1200	0.10	-0.86	0.51	0.76	0.93
Morlkx	-0.20	—	0.27	—	0.95
ALPHA	0.74	0.80	0.91	0.98	0.98
RISC	0.84	0.79	0.93	0.96	0.97
Leon2	-0.25	—	0.65	—	0.94
Leon3mp	-0.09	-1.87	0.50	0.86	0.95
Netcard	-0.33	—	0.49	—	0.96
Ethernet	0.58	-0.03	0.83	0.91	0.94
中位数	0.34	0.57	0.74	0.96	0.96
平均	0.30	0.05	0.70	0.92	0.96

注:“—”表示数据无法获得。

4.3 运行时间分析

基于机器学习的路径延时预测方法整体运行时间主要由特征提取、数据预处理和模型推理 3 部分构成。为全面评估所提方法在运行效率上的表现,本研究将其与文献[30]中的预测时间进行对比,结果如表 4 所示。试验结果表明,本研究方法在推理速度上显著优于文献[30]中的方法,在多数设计案例上的整体运行时间为文献[30]方法的 2.8~3.2 倍,体现出明显的效率优势。这一差异主要源于两种方法在特征处理流程与路径延时建模策略上的不同。文献[30]的方法依赖图结构建模,需执行图划分、图特征提取等较为复杂的预处理步骤,耗时较长,采用局部弧级延时预测,通过遍历图结构以获得完整路径延时,进一步加重了计算负担。本研究方法利用 LSTM 网络实现端到端的路径级延时预测,避免了繁琐的图操作和后处理步骤,整体运行效率大幅提升。因此,本研究方法在保持较高预测精度的同时,具备更低的运行时间和更强的可扩展性,尤其适用于大规模、复杂电路设计场景下对快速、精准时序分析的需求。

表 4 工艺映射阶段运行时间对比

Table 4 Comparison of runtime at technology mapping stage

设计案例	运行时间/min	
	文献[30]方法	本研究方法
B21	0.25	0.01
Or1200	5.80	2.04
Morlkx	0.97	0.34
Leon2	5.00	1.76
Leon3mp	3.80	1.26
Netcard	2.90	0.90

4.4 工艺映射前预测性能

本研究进一步将所提路径延时预测框架应用于工艺无关的通用网表,以验证其在早期设计阶段的适用性。本研究方法与传统 STA 工具在工艺映射前的预测准确性对比如表 5 所示。由表 5 可知,本研究方法在延时预测精度上明显优于传统方法,说明即使缺乏详细工艺信息,LSTM 模型仍能有效捕捉路径结构与延时之间的潜在关联,实现更可靠的预测。与工艺映射后的网表相比,通用网表缺少具体电气特性和布线信息,拓扑结构与最终实现之间差异较大,导致准确预测更具挑战性。尽管如此,本研究方法在这一阶段展现出良好的泛化能力和实际应用潜力,进一步突显工艺无关延时建模的研究价值。

表 5 工艺映射前预测准确度分析

Table 5 Comparison of prediction accuracy before technology mapping stage

案例	R^2	
	STA 工具	本研究方法
Or1200	0.61	0.52
Morlkx	-0.26	0.84
Leon2	-0.59	0.70
Leon3mp	-2.89	0.12
Netcard	-0.11	0.66
平均	-0.65	0.57

5 结论

针对逻辑综合阶段与物理设计阶段时序计算结果不一致的问题,本研究提出一种基于 LSTM 的路径延时预测方法。通过将时序路径建模为标准

单元序列,提取关键结构与时序特征输入 LSTM 网络,实现逻辑综合阶段路径延时的精确预测。试验验证表明,本研究方法可以取得与 TCAD22 相比拟的精度,在不同路径类型上展现出良好的适应能力。在运行效率方面,本研究方法在多数案例上实现 2.8~3.2 倍的加速。在预测工艺无关的通用网表路径延时,相较于传统 STA 方法,本研究方法具有更高的预测精度。上述结果表明,本研究方法不仅能有效应对逻辑综合阶段的挑战,还具有较强的泛化能力,能够在缺乏工艺信息的情况下进行高质量的时序预测,具有较大的应用潜力。在后续研究工作中,将结合更多类型的网络结构和特征自动提取机制,进一步提升本研究方法在复杂电路设计中的适应性和准确性。

参考文献:

- [1] 贺旭,王耀,傅智勇,等.敏捷设计中基于机器学习的静态时序分析方法综述[J].计算机辅助设计与图形学学报,2023,35(4):640-652.
HE Xu, WANG Yao, FU Zhiyong, et al. A survey on machine learning-based technology for static timing analysis in agile design[J]. Journal of Computer-Aided Design & Computer Graphics, 2023, 35(4): 640-652.
- [2] ACHARYAL C, SHARMA A, MISHRA N, et al. Switching activity factor-based ECSM characterization (SAFE): a novel technique for aging-aware static timing analysis[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2024, 43(12): 4715-4725.
- [3] 郭静静,宗璟宜,查佩文,等.基于机器学习的多输入切换效应的统计静态时序分析方法[J].微电子学,2024,54(3):458-467.
GUO Jingjing, ZONG Jingyi, ZHA Peiwen, et al. Machine learning-based method for statistical static timing analysis of multiple input switching effects[J]. Microelectronics, 2024, 54(3): 458-467.
- [4] SHONIKER M, OLEJNIKOV O, COCKBURN B F, et al. Automatic selection of process corner simulations for faster design verification[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2018, 37(6): 1312-1316.
- [5] KAHNG A B, LIN B, NATH S. Enhanced metamodeling techniques for high-dimensional IC design estimation problems[C]//2013 Design, Automation & Test in Europe Conference & Exhibition (DATE). Grenoble, France: IEEE, 2013: 1861-1866.
- [6] CHENG H H, JIANG I H, OU O. Fast and accurate wire timing estimation on tree and non-tree net structures [C]//2020 57th ACM/IEEE Design Automation Conference (DAC). San Francisco, USA: IEEE, 2020: 9218712.
- [7] KAHNG A B, LUO M L, NATH S. SI for free: machine learning of interconnect coupling delay and transition effects[C]//2015 ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP). San Francisco, USA: IEEE, 2015: 7171706.
- [8] KAHNG A B, MALLAPPA U, SAUL L. Using machine learning to predict path-based slack from graph-based timing analysis[C]//2018 IEEE 36th International Conference on Computer Design (ICCD). Orlando, USA: IEEE, 2018: 603-612.
- [9] ZHAO X T, WANG T J, JIAO R, et al. Standard cells do matter: uncovering hidden connections for high-quality macro placement[C]//2024 Design, Automation & Test in Europe Conference & Exhibition (DATE). Valencia, Spain: IEEE, 2024: 25-27.
- [10] BARBOZA E C, SHUKLA N, CHEN Y R, et al. Machine learning-based pre-routing timing prediction with reduced pessimism[C]//Proceedings of the 56th Annual Design Automation Conference 2019. Las Vegas, USA: ACM, 2019: 106.
- [11] CAO P, HE G Q, YANG T. TF-predictor: transformer-based prerouting path delay prediction framework[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2023, 42(7): 2227-2237.
- [12] HE G Q, DING W J, YE Y Y, et al. An optimization-aware pre-routing timing prediction framework based on heterogeneous graph learning[C]//2024 29th Asia and South Pacific Design Automation Conference (ASP-DAC). Incheon: IEEE, 2024: 177-182.
- [13] GUO Z Z, LIU M J, GU J Q, et al. A timing engine inspired graph neural network model for pre-routing slack prediction[C]//Proceedings of the 59th ACM/IEEE Design Automation Conference. San Francisco, USA: ACM, 2022: 1207-1212.
- [14] KAHNG A B, MANTIK S. A system for automatic recording and prediction of design quality metrics[C]//Proceedings of the IEEE 2001. 2nd International Symposium on Quality Electronic Design. San Jose, USA: IEEE, 2001: 81-86.
- [15] KAHNG A B, LIN B, NATH S. High-dimensional metamodeling for prediction of clock tree synthesis outcomes[C]//2013 ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP). Austin, USA: IEEE, 2013: 6681685.
- [16] LU Y C, LEE J, AGNESINA A, et al. GAN-CTS: a generative adversarial framework for clock tree

- prediction and optimization[C]//2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). Westminister, USA: IEEE, 2019: 8942063.
- [17] 张雷,刘伟景,赵启林.一种高效实现时序优化的时钟树综合方案[J].上海电力大学学报,2025,41(1): 85-89.
- ZHANG Lei, LIU Weijing, ZHAO Qilin. A clock tree synthesis method for efficient timing optimization[J]. Journal of Shanghai University of Electric Power, 2025, 41(1): 85-89.
- [18] KAHNG A B, KANG S, LEE H, et al. Learning-based approximation of interconnect delay and slew in signoff timing tools[C]//2013 ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP). Austin, USA: IEEE, 2013: 6681682.
- [19] HAN S S, KAHNG A B, NATH S, et al. A deep learning methodology to proliferate golden signoff timing [C]//2014 Design, Automation & Test in Europe Conference & Exhibition (DATE). Dresden, Germany: IEEE, 2014: 273.
- [20] SONG Q Q, CHENG X, CAO P. Critical paths prediction under multiple corners based on BiLSTM network [C]//2023 60th ACM/IEEE Proceedings of Design Automation Conference (DAC). San Francisco, USA: IEEE, 2023: 10247984.
- [21] NETO W L, MOREIRA M T, AMARU L, et al. Read your circuit: leveraging word embedding to guide logic optimization [C]//Proceedings of the 26th Asia and South Pacific Design Automation Conference. Tokyo, Japan: ACM, 2021: 530-535.
- [22] BODAPATI S, NAJM F N. Prelayout estimation of individual wire lengths[J]. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2001, 9(6): 943-958.
- [23] FATHI B, BEHJAT L, RAKAI L M. Apre-placement net length estimation technique for mixed-size circuits [C]//Proceedings of International the 11th Workshop on System Level Interconnect Prediction. San Francisco, USA: ACM, 2009: 45-52.
- [24] LIU Q, MA J G, ZHANG Q J. Neural network based pre-placement wirelength estimation [C]//2012 International Conference on Field-Programmable Technology. Seoul: IEEE, 2012: 16-22.
- [25] HU B, MAREK-SADOWSKA M. Wire length prediction based clustering and its application in placement [C]//Proceedings of Design Automation Conference. Anaheim, USA: IEEE, 2003: 800-805.
- [26] KAHNG A B, REDA S. Intrinsic shortest path length: a new, accurate a priori wirelength estimator[C]//ICCAD-2005. IEEE/ACM International Conference on Computer-Aided Design. San Jose, USA: IEEE, 2005: 173-180.
- [27] LIU Q H, MAREK-SADOWSKA M. Pre-layout wire length and congestion estimation [C]//Proceedings of the 41st annual Design Automation Conference. San Diego, USA: ACM, 2004: 582-587.
- [28] HYUN D, FAN Y P, SHIN Y. Accurate wirelength prediction for placement-aware synthesis through machine learning[C]//2019 Design, Automation & Test in Europe Conference & Exhibition (DATE). Florence, Italy: IEEE, 2019: 324-327.
- [29] 蒋政涛,贺旭,李琼,等.基于随机森林的网表级时序预测模型[J].微电子学与计算机,2022,39(12): 107-114.
- JIANG Zhengtao, HE Xu, LI Qiong, et al. Random forest-based netlist-level timing prediction [J]. Microelectronics & Computer, 2022, 39(12): 107-114.
- [30] XIE Z Y, LIANG R J, XU X Q, et al. Preplacement net length and timing estimation by customized graph neural network [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022, 41(11): 4667-4680.
- [31] ZHU L Y, GUO X F. Delay-driven physically-aware logic synthesis with informed search [C]//2023 IEEE 41st International Conference on Computer Design (ICCD). Washington, DC, USA: IEEE, 2023: 327-335.
- [32] HOCHREITER S, SCHMIDHUBER J. Long short-term memory[J]. Neural Computation, 1997, 9(8): 1735-1780.

(编辑:孙亚彤)