DGNN-Booster：一种用于动态图神经网络推理的通用现场可编程门阵列加速器框架

Hanqiu Chen佐治亚理工学院电气与计算机工程系美国亚特兰大hchen799@gatech.edu

Cong Hao 电气与计算机工程佐治亚理工学院 亚特兰大，美国 callie.hao@ece.gatech.edu

摘要——**动态图神经网络（DGNN）凭借其在分析和预测复杂互连图系统演变方面的有效性，正日益受到欢迎。然而，动态图神经网络的硬件部署仍面临挑战。首先，由于时间数据依赖导致硬件并行性较低，动态图神经网络无法充分利用硬件资源。此外，目前缺乏通用的动态图神经网络硬件加速器框架，而现有的图神经网络加速器框架处理拓扑结构和节点特征不断变化的动态图的能力有限。为应对上述挑战，本文提出 DGNN-Booster，这是一种基于高级综合（HLS）、用于实时动态图神经网络推理的新型现场可编程门阵列（FPGA）加速器框架。它包含两种采用不同数据流的现场可编程门阵列加速器设计，可支持应用最广泛的动态图神经网络。我们在 ZCU102 开发板上实现并评估了两种具有代表性的动态图神经网络模型，并测量其端到端性能，以展示所提设计的有效性。实验结果表明，与中央处理器基线（6226R）相比，DGNN-Booster 的加速比最高可达 6× ；与图形处理器基线（A6000）相比，最高可达 4× ；与未采用本文所提优化的现场可编程门阵列基线相比，最高可达 1× 。此外，与中央处理器和图形处理器基线相比，DGNN-Booster 的运行时能效分别可提升超过 100× 和超过 1000×。我们的实现代码和板上测量结果已公开，网址为 https://github.com/sharc-lab/DGNN-Booster。**

I. 引言

图神经网络（GNNs）是用于捕获图结构数据中关系的强大工具，可应用于多个领域，包括推荐系统 [1], 药物发现 [2], 欺诈检测 [3] 和交通流预测 [4]。在实际应用中，动态图神经网络（DGNNs）相比传统图神经网络具有多项优势。它们能够捕获图中节点与边之间的时间依赖关系，从而在交通模式预测 [5] 和股价预测 [6]等时间相关任务中取得更好的性能。此外，动态图神经网络（DGNNs）具有很高的灵活性，可以将不同类型的图神经网络用于空间编码，并结合循环神经网络（RNNs）进行时间编码，从而提升性能。

尽管不同类型的动态图神经网络在软件 [7]–[9],中取得了成功，但其硬件部署仍面临挑战：(1) 并行性低。由于不同时间的图之间存在时间数据依赖，因此难以在硬件上并行执行计算。(2) 内存消耗大且内存访问频繁。时变图嵌入会导致内存消耗大，并在片上和片外存储器之间频繁传输数据。(3) 能耗高：动态图神经网络由于计算密集型矩阵乘法和复杂的数学运算而能耗较高。

为应对上述挑战，本文提出了**DGNN-Booster**，这是一种通用的现场可编程门阵列加速器框架，适用于动态图神经网络，可实现高速、低能耗的板上推理，并可应用于多种流行的动态图神经网络。我们的贡献总结如下：

1) **通用且开源**。DGNN-Booster 是一个模型通用框架，采用高级综合（HLS）开发，易于使用。它为图神经网络和循环神经网络设计了模块化的处理单元（PE），并支持多种类型的图神经网络和循环神经网络。该框架已公开发布，其板载测量和端到端功能均通过与 PyTorch 代码交叉验证。2) **硬件高效**。DGNN-Booster 具备多级并行性，并采用硬件架构优化，旨在以低于中央处理器和图形处理器的能耗实现实时性能。不同时间步的不同图可以连续流入并即时处理。3) **采用不同数据流的两种加速器设计**。DGNN-Booster 提供两种设计，支持图神经网络与循环神经网络之间的不同数据流。DGNN-Booster V1 在相邻时间步中使二者重叠，而 DGNN-Booster V2 则在单个时间步内通过数据流将二者连接起来。两种设计都在循环神经网络内部采用数据流，以提高并行性并降低片上存储器消耗。

4) **板载评估**。我们使用两个具有代表性的 DGNN 模型，基于两个流行的时序图数据集，在不同时间步下针对不同图规模，利用 Xilinx ZCU102 FPGA 对 DGNN-Booster 进行验证。此外，我们还进行了消融研究，以证明所设计的多级并行方案的有效性。

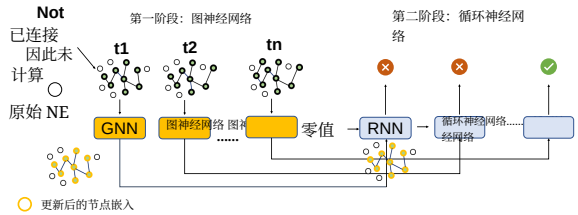


图1：堆叠式动态图神经网络在中央处理器和图形处理器上的实现数据流概览。图神经网络在不同时间步的输出将按顺序馈入循环神经网络。只有循环神经网络在最后一个时间步的输出会用于后续计算。（NE：节点嵌入）

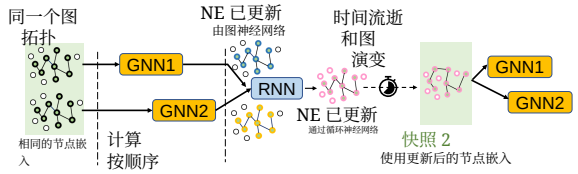


图2：集成式动态图神经网络在中央处理器和图形处理器上的实现数据流概览。最后一个时间步中循环神经网络的输出将用作下一个时间步中图神经网络的输入。两个图神经网络按顺序执行。图神经网络和循环神经网络也按顺序计算。（节点嵌入：节点嵌入）

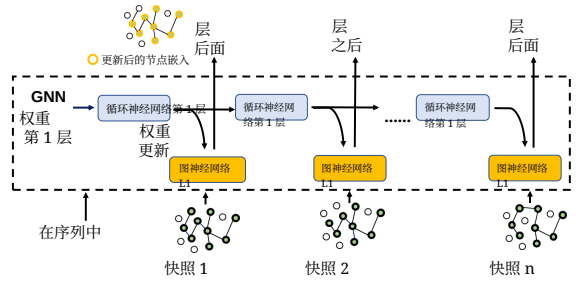


图3：权重演化式动态图神经网络在中央处理器和图形处理器上的实现数据流概览。权重由循环神经网络演化，并按顺序供图神经网络使用。（节点嵌入：节点嵌入）

II. 动态图神经网络背景

动态图神经网络（DGNNs）是专为动态图结构和特征设计的图神经网络。根据一项最新综述 [10], , 动态图神经网络可分为两种不同的类别：离散时间动态图神经网络和连续时间动态图神经网络。DGNN-Booster 支持离散时间动态图神经网络，它们使用一组有序图（快照）来表示动态图。

$$DG = \{G^1, G^2, ..., G^T\} \tag{1}$$

其中，T 表示快照的数量。这种动态图的离散时间表示方式使得能够使用传统静态图神经网络进行空间信息编码，并使用循环神经网络进行时间信息编码。

我们根据图神经网络与循环神经网络之间的数据流关系，将离散时间动态图神经网络分为三种类型，具体总结见表I。

- **堆叠式动态图神经网络**。这是对离散时间动态图进行建模最直接的方法。图神经网络将其编码为时间序列信息，并输入循环神经网络。此过程可表示为：

$$\begin{aligned} X^1, X^2, ..., X^t &= \text{GNN}(G^1, G^2, ..., G^t) \\ O^{t+1} &= \text{RNN}(X^1, X^2, ..., X^t) \end{aligned} \tag{2}$$

其中，G^t 是时间 t的快照的节点嵌入，X^t 是图神经网络在时间 t更新后的节点嵌入，O^{t+1}是循环神经网络在时间 $t+1$的输出。这类动态图神经网络的高级数据流图如图1所示。

- **集成式动态图神经网络**。这种离散时间图编码方式在一个时间步内将图神经网络与循环神经网络结合起来，具体方法是用图相关操作（如图卷积）替换循环神经网络中的矩阵乘法。它可以表示为以下方程：

$$\begin{aligned} X_1^t &= \text{GNN1}(G^t) \\ X_2^t &= \text{GNN2}(G^t) \\ G^{t+1} &= \text{RNN}(X_1^t, X_2^t) \end{aligned} \tag{3}$$

其中，G^t 表示时间为 t 的快照的节点嵌入。X_1^t 和 X_2^t 是由 GNN1 和 GNN2 使用不同权重更新得到的两种不同的节点嵌入。这类动态图神经网络的高层数据流图如图2所示。

- **权重演化式动态图神经网络**。这类动态图神经网络中图神经网络与循环神经网络之间的数据流与堆叠式动态图神经网络相似。区别在于循环神经网络演化的对象。与堆叠式动态图神经网络不同，后者由循环神经网络演化图神经网络更新的节点嵌入，而这里由循环神经网络演化图神经网络的权重。可用以下方程表示：

$$\begin{aligned} W^t &= \text{RNN}(W^{t-1}) \\ O^t &= \text{GNN}(W^t, G^t) \end{aligned} \tag{4}$$

其中，W^t 表示时刻 t 的图神经网络权重，G^t 表示时刻 t 的节点嵌入，O^t 表示时刻 t 的图神经网络输出。这类动态图神经网络的高层数据流图如图 3 所示。

III. 研究动机与创新

A. 相关工作与动机

动态图神经网络硬件加速器领域近期取得了一些进展。Zhou等人 [19] 针对基于存储器的时序图神经网络开展了模型架构协同设计。Cambricon-G [20] 是首个旨在利用多维多级分块技术挖掘更多数据复用机会的硬件加速器。Chakaravarthy等人 [21] 通过设计多GPU动态图神经网络训练系统，完成了首个针对动态图神经网络的扩展性研究。DynaGraph [22] 和TGL [23] 则是另外两个基于图形处理器的高性能动态图神经网络训练框架，分别侧重于空间与时间知识的统一，以及简化用户配置。

表I：不同类型的离散时间动态图神经网络及其数据流类型汇总。我们还展示了 DGNN-Booster 中可应用的加速器设计。

动态图神经网络类型	相关工作	数据流类型	DGNN Booster V1	DGNN Booster V2
堆叠的 DGNN	GCRN-M1 [11],RgCNN [12] WD-GCN [13], DyGGNN [14]	• 一个时间步内图神经网络与循环神经网络之间的数据依赖。 • 不同时间步的独立图神经网络。	✔	✔
集成 DGNN	GCRN-M2 [11], GC-LSTM [15] LRGCN [16], RE-Net [17]	• 相邻时间步中图神经网络与循环神经网络之间的数据依赖。 • 不同时间步之间存在依赖关系的图神经网络。	✗	✔
权重演化 DGNN	EvolveGCN [18]	• GNN 的权重由 RNN 演化。 • 不同时间步的独立 GNN。	✔	✗

然而，动态图神经网络硬件部署仍面临一些挑战。**能耗高、计算资源利用率低**。以往的工作主要关注在图形处理器上部署动态图神经网络。然而，由于时间数据依赖，这些设计存在能耗高、计算资源利用率低的问题。 **图神经网络与循环神经网络之间缺乏并行性**。以往的研究将图神经网络和循环神经网络视为彼此独立的部分，从而限制了并行性。 **缺乏将图神经网络和循环神经网络的优化整合到单一系统中**。以往的研究通常分别优化图神经网络和循环神经网络，因此难以实现最佳硬件效率。

B. 创新之处

受这些挑战的启发，我们提出了 DGNN-Booster，以实现在现场可编程门阵列（FPGA）上的高速、低能耗 DGNN 推理。我们的设计能够覆盖表 I 中展示的大多数 DGNN 类型。与以往的 DGNN 加速器设计相比，我们的设计具有以下几项优势：

• **具备高度灵活性的更优硬件性能**。与 CPU 和 GPU 相比，DGNN-Booster 具有更低的延迟和能耗。其采用高层次综合（HLS）实现的 GNN 和 RNN 模块化设计，能够轻松集成不同的 GNN 和 RNN。此外，GNN 采用消息传递机制实现。我们特别强调 DGNN-Booster 对边嵌入的支持，而现有的 DGNN 加速器尚未考虑边嵌入，但边嵌入已广泛应用于大多数 GNN 模型中。

• **多级并行**。 我们的设计中存在两个层次的并行性。在高层并行中，我们在 DGNN-Booster V1 中对相邻时间步的图神经网络和循环神经网络进行并行处理，而在 V2 中则对同一时间步内的图神经网络和循环神经网络进行并行处理。所支持的动态图库神经网络类型如表I所示。此外，在 V1 中，我们将图加载与图神经网络推理进行重叠处理。在底层并行中，我们基于 GenGNN [24] 的消息传递机制实现图神经网络。在 DGNN-Booster V2 中，消息传递和节点变换以数据流方式进行。此外，在这两种加速器设计中，我们还针对循环神经网络内部的不同阶段实现了数据流。

• **硬件高效架构设计**。我们提出了一种任务调度方案，为中央处理器和现场可编程门阵列分配最合适的任务。此外，我们实现了图重新编号和格式转换，使我们的设计更适合硬件。此外，我们利用片上不同类型的随机存取存储器来提高存储效率。

IV. 硬件架构设计

DGNN-Booster 基于中央处理器-现场可编程门阵列异构平台开发，其中主机程序将权重和节点特征加载到动态随机存取存储器中，并执行图预处理。图预处理完成后，现场可编程门阵列通过 PCIe 将准备好的数据加载到片上缓冲区中。现场可编程门阵列加速器针对各种动态图神经网络进行了优化，其定制IP核采用 HLS 设计，包含用于并行处理的处理单元（处理单元），可并发执行图神经网络和循环神经网络的推理，从而实现最佳硬件性能。

A. 图预处理与数据通信

DGNN-Booster 的输入图采用坐标（COO格式）表示，这是动态图数据集中最广泛使用的格式。在 COO格式中，边以任意顺序存储在列表中，列表中的每个条目由与该边相关联的源节点、目标节点、数据和时间组成。主机程序负责根据所选的时间划分器，按照时间顺序将大型输入图切分为较小的快照。应适当设置时间划分器，以确保每个快照的大小既不过大也不过小。在生成快照的过程中，中央处理器还将计算每个快照中的节点数和边数。

权重和快照信息的数据通信模式不同。权重在不同时间步之间共享，因此，在现场可编程门阵列上开始计算之前，权重加载只需付出一次开销。每个快照的边列表、节点嵌入、边嵌入以及节点数和边数，都会按时间顺序从动态随机存取存储器发送到片上缓冲区，等待在现场可编程门阵列上进行计算。由于现场可编程门阵列上的片上存储器资源非常有限，因此无法一次性将不同时间步中具有不同节点和边特征的快照信息全部传输到片上缓冲区。因此，只会将下一时间步待处理快照的信息发送到片上缓冲区。

B. 图重新编号与格式转换

在 FPGA 运行时，片上缓冲区中仅存储一个快照。为确保对片上缓冲区的数据访问正确，

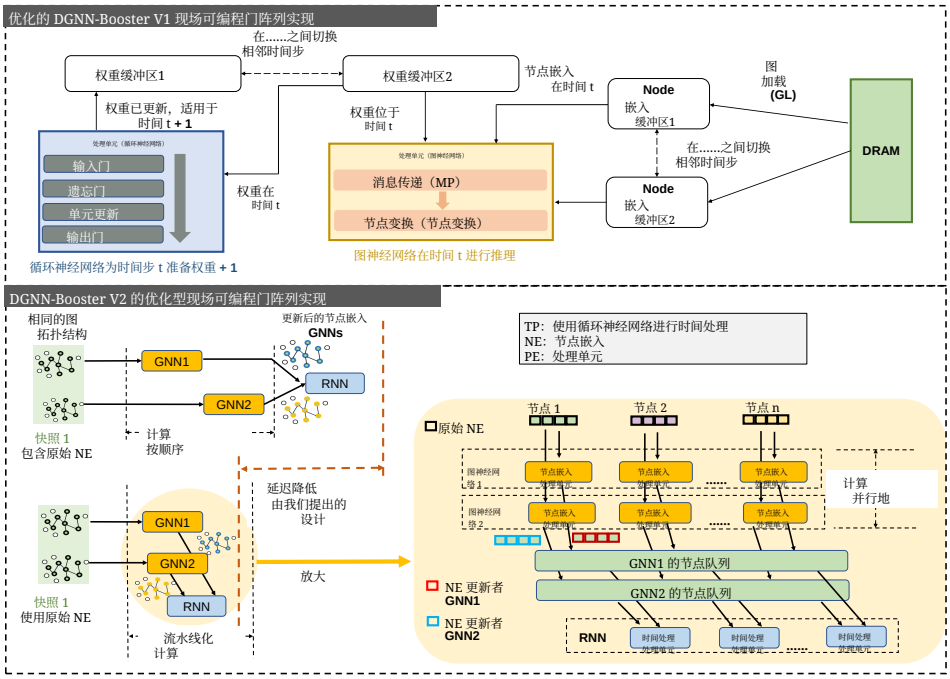


图 4：基于 EvolveGCN 的 DGNN-Booster V1 和基于 GCRN-M2 的 V2 的优化型现场可编程门阵列实现。DGNN-Booster V1 使用乒乓缓冲区存储权重，从而重叠相邻时间步中图神经网络和循环神经网络的计算。图加载也以相同方式与图神经网络推理重叠。DGNN-Booster V2 利用由先进先出队列实现的节点队列，重叠单个时间步内图神经网络和循环神经网络的计算。

我们需要知道大规模原始图中每个节点的原始索引，以及其在块随机存取存储器（BRAM）中的对应地址。主机程序将为每个快照生成一张重新编号表，用于记录节点索引的重新编号信息。在推理过程中，处理单元（PEs）将首先查询该重新编号表，以执行节点索引转换，然后从片上缓冲区中正确获取数据。重新编号表还将指导现场可编程门阵列（FPGA）从动态随机存取存储器（DRAM）中正确读取数据并将数据写回。

尽管 COO格式便于图数据生产者使用，并且通常是实际应用中的原始数据格式，但它并不适合硬件，因为查找邻居节点涉及不规则的计算模式，通常会在现场可编程门阵列上带来较大的开销。我们没有使用 COO格式，而是通过在现场可编程门阵列上设计用于格式转换的转换器，采用压缩稀疏行（CSR）格式或压缩稀疏列（CSC）格式进行图神经网络推理。

图重新编号和格式转换相结合，可确保快照数据在片上缓冲区中连续存储，从而避免不规则的片上存储器访问。

C. 多级并行

DGNN-Booster V1 和 V2 中用于实现多级并行的技术有所不同。

1）DGNN-Booster V1：DGNN-Booster V1 旨在减少堆叠式动态图神经网络和权重演化动态图神经网络中循环神经网络与图神经网络顺序计算所造成的硬件开销，其中央处理器和图形处理器的数据流分别如图1和图3所示。该设计使相邻时间步中的图神经网络和循环神经网络能够并行执行。DGNN-Booster V1 的详细现场可编程门阵列实现如图4所示。

- **RNN 中的乒乓缓冲区和数据流先进先出队列**。我们通过为权重和节点嵌入使用两对乒乓缓冲区来避免数据冲突。如图 4 所示，GNN 可以从缓冲区2读取权重，同时 RNN 可以更新下一时间步的权重，并将结果存储在缓冲区1中。同样，节点嵌入的乒乓缓冲区支持数据加载与 GNN 推理并行进行。我们还利用先进先出队列（先进先出）连接 RNN 内部不同的计算阶段，使这些阶段能够在节点级进行流水线化，从而进一步降低 RNN 的延迟。

- **执行流程**。一个时间步内的推理过程可以分为四个独立部分：图加载（GL）、消息传递（MP）、节点变换（NT）和 RNN。其中，MP 必须等待 GL 的结果，而 NT 必须等待 MP 和 RNN。为实现最佳性能，我们安排 RNN 在 $t+1$ 与 MP 在 t 并行执行，并安排 GL 在 $t+1$ 与 NT 在 t 并行执行。这是因为 MP 和 RNN 是计算量相对大于 GL 和 NT 的两个模块，采用这一调度方案可以避免工作负载不均衡。

2）DGNN-Booster V2：DGNN-Booster V2 旨在降低堆叠式动态图神经网络和

集成式动态图神经网络中由循环神经网络和图神经网络计算顺序执行所造成的硬件开销，其中央处理器和图形处理器的数据流分别如图1和图2所示。该设计使同一时间步内的图神经网络和循环神经网络能够并行执行。DGNN-Booster V2 的高级数据流和现场可编程门阵列架构如图4所示。

• **节点队列**。 节点队列使用先进先出队列实现，以便让图神经网络和循环神经网络计算重叠进行。当图神经网络完成一个节点嵌入的更新后，会将更新后的节点嵌入发送到节点队列。执行循环神经网络的处理单元会按相同顺序从节点队列中获取节点嵌入。同时，图神经网络将被触发以获取新的节点嵌入。因此，图神经网络和循环神经网络可以并行处理不同的节点。

• **执行流程**。 在本设计中，图神经网络中的消息传递和节点变换采用数据流方式。此外，循环神经网络的不同阶段也采用数据流方式。将这些优化与连接图神经网络和循环神经网络的节点队列相结合，我们实现了端到端的节点级流水线。

*D. CPU-FPGA*异构平台上的任务调度

为了更好地发挥中央处理器和现场可编程门阵列各自的优势，我们提出了一种任务调度方案。中央处理器能够灵活处理各种类型的任务，而现场可编程门阵列则能够在计算模式简单的任务上提供较高的峰值性能。为了实现最佳硬件性能，我们将图预处理和重新编号任务分配给中央处理器执行。图格式转换、图神经网络和循环神经网络推理则分配给现场可编程门阵列执行。这是因为，图预处理和重新编号表生成需要复杂的控制流，以及不规则且频繁的内存访问，但计算强度较低。图神经网络和循环神经网络推理涉及大量矩阵乘法，虽然计算密集度高，但计算模式较为简单。

E. 片上缓冲区设计

FPGA 上有两种类型的随机存取存储器。查找表 RAM 由查找表构成，对于小容量随机存取存储器而言，其效率更高，因为它可以根据需求灵活配置为多种不同的容量。对于块随机存取存储器（BRAM），其最小存储容量为 18KB。如果一个块随机存取存储器中的部分存储空间未被使用，这部分空间就无法用于其他用途，从而造成浪费。DGNN-Booster 在图神经网络和循环神经网络内部采用了细粒度流水线设计。在这种情况下，权重缓冲区将被划分为大量小容量随机存取存储器，如果使用块随机存取存储器存储权重，就会造成片上存储资源的巨大浪费。因此，我们将权重分配到查找表 RAM 中。相反，由于节点嵌入和边嵌入的数据量大于权重，且需要存储在片上连续的存储空间中，因此我们将它们分配到块随机存取存储器中。我们的片上缓冲区设计展现了较高的存储效率，并支持在片上存储相对较大的快照，从而避免在消息传递阶段主机与 FPGA 之间频繁进行数据交换。

V. 实验

A. 实现细节

我们使用 Vitis HLS 和 Vivado 的高级综合（HLS），在 Xilinx ZCU102 FPGA开发板上部署 DGNN-Booster，如图5所示；其可用资源如表II所示，目标时钟频率为 100MHz。我们选择 EvolveGCN [18] 作为 DGNN-Booster V1 的基础模型，并选择 GCRN-M2 [11] 作为 V2 的基础模型。两者均使用 GCN [25] 作为图神经网络。EvolveGCN 使用 GRU [26] 作为循环神经网络，而 GCRN-M2 使用 LSTM [27] 。权重和图嵌入均采用 32 位浮点数。

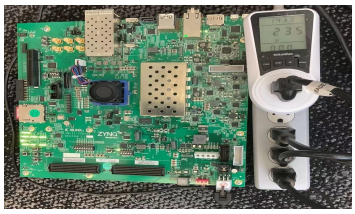


图5：带功耗测量的板载实现。

表II：Xilinx ZCU102 FPGA 上的资源利用率。时钟频率为 100MHz。结果由 Vivado 实现后报告给出。

模型	LUT	查找表 RAM	FF	BRAM	DSP
可用的	274,080	144,000	548,160	912	2520
EvolveGCN	142,488 **52%**	31,210 **22%**	88,930 **16%**	496.50 **54%**	1952 **77%**
GCRN-M2	151,302 **55%**	27,482 **19%**	121,088 **22%**	382.50 **42%**	2242 **89%**

表I　　II：实验所用数据集的信息。

数据集	Avg 节点	Avg 边	Max 节点	Max 边	Time 分割器	快照 计数
BC-Alpha	107	232	578	1686	3周	137
UCI	118	269	501	1534	1天	192

B. 数据集信息

实验中使用了两个公开可用的基准动态图数据集。Bitcoin Alpha（BC-Alpha） [28]包含来自 Bitcoin Alpha 交易平台的用户间信任/不信任网络相关信息。一个节点代表一名交易者，一条边代表交易者之间的信任程度。UC Irvine 消息（UCI） [29]包含有关加利福尼亚大学欧文分校学生在线社区的信息。一个节点代表一名用户，一条边代表一条已发送的消息。有关这些数据集的更多详细信息如表III所示。

C. 端到端评估

我们在完成布局布线后，对 EvolveGCN 和 GCRN-M2 进行了全面评估，并与中央处理器和图形处理器基线进行对比。我们测量板载端到端延迟，包括权重加载和图加载，并对每个数据集中的快照取平均值。

结果如表IV所示。对于 EvolveGCN 和 GCRN-M2，DGNN-Booster 相较于中央处理器和图形处理器均实现了显著加速。由于消息传递机制并不适合在图形处理器上实现 [30] ，而且时序数据依赖和频繁的数据交换会导致

表IV：EvolveGCN 和 GCRN-M2 每个快照的板载延迟（单位：毫秒）。

模型（数据集）	CPU	GPU	FPGA（我们的方法）	FPGA（相较于中央处理器）	FPGA（相较于图形处理器）
EvolveGCN (BC-Alpha)	3.18	4.01	0.76	**4.16**×	**5.25**×
EvolveGCN (UCI)	3.68	4.19	0.86	**4.26**×	**4.84**×
GCRN-M2 (BC-Alpha)	7.39	11.35	1.35	**5.48**×	**8.42**×
GCRN-M2 (UCI)	8.50	9.74	1.51	**5.63**×	**6.45**×

表 V：DGNN-Booster 的能效（包括空闲和运行时，单位为焦耳/100 个快照）。

模型（数据集）	CPU	GPU	FPGA（我们的方法）	FPGA（对比中央处理器）	FPGA（对比图形处理器）
EvolveGCN（BC-Alpha）	5.84	32.16	1.92	**3.04**×	**16.75**×
EvolveGCN（UCI）	6.64	32.97	2.13	**3.12**×	**15.48**×
GCRN-M2（BC-Alpha）	15.29	73.03	3.17	**4.82**×	**23.04**×
GCRN-M2（UCI）	17.59	85.14	3.54	**4.97**×	**24.05**×

表VI：DGNN-Booster 的运行时能效（单位：J/100 个快照）。

模型（数据集）	CPU	GPU	FPGA（本研究）	FPGA（相比中央处理器）	FPGA（相比图形处理器）
EvolveGCN (BC-Alpha)	1.83	21.01	0.02	**91.5**×	**1050.5**×
EvolveGCN (UCI)	2.08	21.54	0.03	**69.3**×	**718**×
GCRN-M2 (BC-Alpha)	6.57	47.71	0.05	**131.4**×	**954.2**×
GCRN-M2 (UCI)	7.56	55.63	0.06	**126.0**×	**927.2**×

图形处理器资源利用率低，且中央处理器与图形处理器之间的通信开销较大 [31], ，因此图形处理器基线报告的延迟略高于中央处理器。

此外，我们使用连接到电路板的功率计评估了 DGNN-Booster 在 BC-Alpha 和 UCI 数据集上的能效，如图5所示。表 V 和表 VI 分别展示了 DGNN-Booster 与中央处理器和图形处理器基线相比的总能效和运行时能效。显著的加速比和能效提升证明了 DGNN-Booster 的有效性。

D. 设计空间探索与消融研究

我们进行了一项简单的设计空间探索，以选择最优的加速器配置，同时考虑 GNN 和 RNN 计算之间的权衡。在 DGNN-Booster V1 中，由于 RNN 的计算开销高于 GNN，因此我们为 RNN 分配了更多的数字信号处理器。相反，在 DGNN-Booster V2 中，由于 GNN 的计算开销更高，因此我们为 GNN 分配了更多的数字信号处理器。表VII 详细列出了各个版本的数字信号处理器分配情况。我们进一步通过消融研究评估 DGNN-Booster，并将其与图形处理器进行比较。

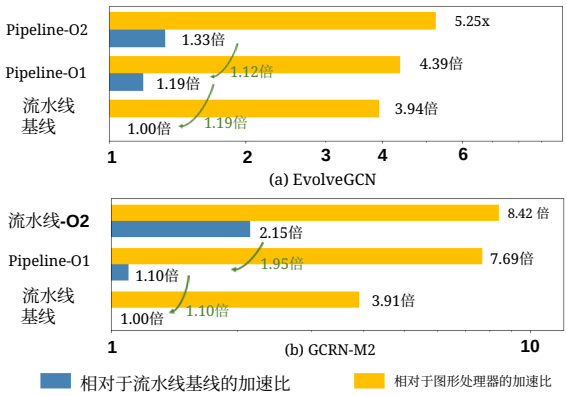


图6：DGNN-Booster 两种设计的有效性。加速比以对数刻度绘制。**基线：** 未采用本文提出的优化。**Pipeline-O1：** 在循环神经网络内部对不同阶段进行流水线处理。**Pipeline-O2：** 将图神经网络与循环神经网络重叠执行。

以及一个未经优化的 FPGA 基线。我们测试了两个级别的流水线优化（Pipeline-O1 和 Pipeline-O2）。Pipeline-O1 通过对循环神经网络（RNN）内部的不同阶段进行流水线化处理来实现优化；Pipeline-O2 则通过在模块级别重叠图神经网络（GNN）和循环神经网络（RNN）的执行，进一步引入更多优化。图6展示了相较于两个基线，推理速度逐步提升的情况。这表明，从多级层面对 GNN 和 RNN 进行流水线化处理，能够非常有效地降低延迟。

表VII：设计空间探索结果。延迟取自 BC-Alpha 和 UCI 数据集中的所有快照，并计算平均值。DSP 利用率结果由 Vivado 报告。

框架	模块	延迟（毫秒）	DSP
DGNN-Booster V1（EvolveGCN）	GNN	0.36（43%）	288（15%）
	RNN	0.47（57%）	1658（85%）
DGNN-Booster V2（GCRN-M2）	GNN	0.82（49%）	2171（96%）
	RNN	0.85（51%）	78（4%）

VI. 结论与未来工作

本文提出了 **DGNN-Booster**，这是一个 FPGA 加速器框架，包含两种具有不同数据流模式和多级并行性的设计。DGNN-Booster 已开源，能够以较低能耗实现实时推理速度。此外，它是一个通用框架，支持多种类型的图神经网络和循环神经网络，并集成了图神经网络与循环神经网络的硬件优化，以实现最佳性能提升。未来的工作包括在板载实现，以避免因相邻时间步的快照相似而产生冗余的数据通信和计算。我们还计划进行设计空间探索，以平衡空间和时间编码部分的计算资源。

致谢

本研究部分得到美国国家科学基金会（NSF）资助，项目编号为 ECCS-2202329。

参考文献

[1] C. Gao、X. Wang、X. He 和 Y. Li，“用于推荐系统的图神经网络”，载于第十五届 ACM 网络搜索与数据挖掘国际会议论文集，2022年，第 1623–1625 页。[2] J. Gilmer、S. S. Schoenholz、P. F. Riley、O. Vinyals 和 G. E. Dahl，“用于量子化学的神经消息传递”，载于机器学习国际会议。PMLR，2017年，第 1263–1272 页。[3] L. Akoglu、H. Tong 和 D. Koutra，“基于图的异常检测与描述：综述”，《数据挖掘与知识发现》，第 29 卷，第 3 期，第 626–688 页，2015年。[4] Z. Cui、K. Henrickson、R. Ke 和 Y. Wang，“交通图卷积循环神经网络：用于网络规模交通流学习与预测的深度学习框架”，《IEEE智能交通系统汇刊》第 21 卷，第 11 期，第 4883–4894 页，2020年。[5] L. Zhao、Y. Song、C. Zhang、Y. Liu、P. Wang、T. Lin、M. Deng 和 H. Li，“T-GCN：用于交通流预测的时序图卷积网络”，《IEEE智能交通系统汇刊》第 21 卷，第 9 期，第 3848–3858 页，2019年。[6] J. Rios、K. Zhao、N. Street、H. Tian 和 X. Zheng，“一种基于供应链网络的动态股票走势预测混合深度学习模型”，载于信息技术与系统研讨会（WITS），2020年。[7] S. Kumar、X. Zhang 和 J. Leskovec，“预测时序交互网络中的动态嵌入轨迹”，载于第 25 届 ACM SIGKDD知识发现与数据挖掘国际会议论文集，2019年，第 1269–1278 页。[8] D. Xu、C. Ruan、E. Korpeoglu、S. Kumar 和 K. Achan，“时序图上的归纳式表示学习”，arXiv预印本 arXiv:2002 07962，2020年。[9] S. Guo、Y. Lin、H. Wan、X. Li 和 G. Cong，“学习时空图数据的动态特性与异质性以进行交通流预测”，《IEEE知识与数据工程汇刊》，2021年。[10] J. Skarding、B. Gabrys 和 K. Musial，“使用动态图神经网络对动态网络进行基础研究与建模：综述”，《IEEE Access》，第 9 卷，第 79 143–79 168 页，2021年。[11] Y. Seo、M. Defferrard、P. Vandergheynst 和 X. Bresson，“利用图卷积循环网络进行结构化序列建模”，载于神经信息处理国际会议。Springer，2018年，第 362–373 页。[12] A. Narayan 和 P. H. Roe，“使用深度神经网络学习图动力学”，《IFAC-PapersOnLine》，第 51 卷，第 2 期，第 433–438 页，2018年。[13] F. Manessi、A. Rozza 和 M. Manzo，“动态图卷积网络”，《模式识别》，第 97 卷，第 107000 页，2020年。[14] A. Taheri、K. Gimpel 和 T. Berger-Wolf，“利用循环模型学习表示动态图的演化”，载于2019年万维网大会附属会议论文集，2019年，第 301–307 页。[15] J. Chen、X. Wang 和 X. Xu，“GC-LSTM：用于动态链接预测的图卷积嵌入式长短期记忆网络”，arXiv预印本 arXiv:1812 04206，2018年。[16] J. Li、Z. Han、H. Cheng、J. Su、P. Wang、J. Zhang 和 L. Pan，“预测时变图中的路径失效”，载于第 25 届 ACM SIGKDD知识发现与数据挖掘国际会议论文集，2019年，第 1279–1289 页。[17] W. Jin、M. Qu、X. Jin 和 X. Ren，“循环事件网络：时序知识图谱上的自回归结构推断”，arXiv预印本 arXiv:1904.05530，2019年。[18] A. Pareja、G. Domeniconi、J. Chen、T. Ma、T. Suzumura、H. Kanezashi、T. Kaler、T. Schardl 和 C. Leiserson，“EvolveGCN：用于动态图的演化图卷积网络”，载于AAAI人工智能大会论文集，第 34 卷，第 04 期，2020年，第 5363–5370 页。[19] H. Zhou、B. Zhang、R. Kannan、V. Prasanna 和 C. Busart，“面向现场可编程门阵列上高性能时序图神经网络推理的模型—架构协同设计”，arXiv预印本 arXiv:2203.05095，2022年。[20] X. Song、T. Zhi、Z. Fan、Z. Zhang、X. Zeng、W. Li、X. Hu、Z. Du、Q. Guo 和 Y. Chen，“Cambricon-G：用于动态图神经网络的多用途高能效加速器”，《IEEE集成电路与系统计算机辅助设计汇刊》，第 41 卷，128，2021 1，116第。–。[21] V. T. Chakaravarthy、S. S. Pandian、S. Raje、Y. Sabharwal、T. Suzumura 和 S. Ubaru，“动态图神经网络的高效扩展”，载于高性能计算、网络、存储与分析国际会议论文集，2021年，第 1–15 页。

[22] M. Guan、A. P. Iyer 和 T. Kim，“Dynagraph：大规模动态图神经网络”，载于第 *5* 届 *ACM SIGMOD* 图数据管理实践与系统联合国际研讨会（*GRADES*）和网络数据分析（*NDA*）论文集，2022年，第 1–10 页。[23] H. Zhou、D. Zheng、I. Nisa、V. Ioannidis、X. Song 和 G. Karypis，“TGL：面向十亿规模图的时序图神经网络训练通用框架”，*arXiv* 预印本 *arXiv:2203.14883*，2022年。[24] S. Abi-Karam、Y. He、R. Sarkar、L. Sathidevi、Z. Qiao 和 C. Hao，“GenGNN：一种用于图神经网络加速的通用现场可编程门阵列框架”，*arXiv* 预印本 *arXiv:2201.08475*，2022年。[25] T. N. Kipf 和 M. Welling，“基于图卷积网络的半监督分类”，*arXiv* 预印本 *arXiv:1609.02907*，2016年。[26] K. Cho、B. Van Merri¨enboer、,D. Bahdanau 和 Y. Bengio，“神经机器翻译的特性：编码器-解码器方法”，*arXiv* 预印本 *arXiv:1409.1259*，2014年。[27] F. A. Gers、J. Schmidhuber 和 F. Cummins，“学会遗忘：基于长短期记忆网络的持续预测”，*Neural Computation*，第 12 卷，第 10 期，第 2451–2471 页，2000年。[28] S. Kumar、B. Hooi、D. Makhija、M. Kumar、C. Faloutsos 和 V. Sub-rahmanian，“Rev2：评分平台上的欺诈用户预测”，载于第十一届 *ACM* 网络搜索与数据挖掘国际会议论文集。ACM，2018年，第 333–341 页。[29] J. Kunegis，“KONECT：科布伦茨网络集合”，载于第 *22* 届万维网国际会议论文集，2013年，第 1343–1350 页。[30] R. Sarkar、S. Abi-Karam、Y. He、L. Sathidevi 和 C. Hao，“Flowgnn：一种通过多队列流式处理实现通用图神经网络推理的数据流架构”，*arXiv* 预印本 *arXiv:2204.13103*，2022年。[31] H. Chen、Y. Alhinai、Y. Jiang、E. Na 和 C. Hao，“中央处理器和图形处理器上动态图神经网络推理的瓶颈分析”，载于*2022*年 *IEEE* 国际工作负载特征分析研讨会（*IISWC*）。IEEE，2022年，第 130–145 页。