International Academic Institute for Science and Technology

International Academic Journal of Science and Engineering
Vol. 12, No. 4, 2025, pp. 157-165.

ISSN 2454-3896

International Academic Journal of Science and Engineering

www.iaiest.com

Simulation-Driven Multi-Kernel FPGA Dataflow Optimization for Low-Latency and Resource-Efficient Digital Signal Processing Pipelines

Laura Virtanen [a], Jussi Nieminen [b]

[a] *Technology Research Center, University of Turku, Joukahaisenkatu, Turku, Finland.*
E-mail: laura.virt@utu.fi
[b] *Technology Research Center, University of Turku, Joukahaisenkatu, Turku, Finland.*
E-mail: jussnie@utu.fi

Abstract

Hardware acceleration in digital signal processing (DSP) applications required by edge computing, wireless communication, and multimedia analytics is becoming more and more apparent because of the high throughput needed and the low latency as well as resource usage required. FPGAs have provided a desirable platform to do such applications due to their reconfigurable architecture and capability of executing highly parallel processing pipelines. Most current DSP accelerators are however designed to optimise a single algorithm e.g. the fast fourier transform or filtering operations and hence not very flexible and therefore do not efficiently utilize resources when more than one kernel needs to be run in the same pipeline. In this paper, a multi-kernel FPGA dataflow optimization framework is suggested based on simulation and modelling a number of DSP kernels within one hardware pipeline. The methodology is an integration of dataflow graph modelling, latency-sensitive scheduling and resource sharing techniques with the aim of maximising the throughput and computational efficiency. The signal workload generator is a synthetic workload generator that mimics various conditions of the application and therefore makes it possible to evaluate the application without real-time data. Hardware simulation and estimation with synthesis are used to measure the latency, throughput, power consumption and the use of FPGA resources by the architecture. It was found that the presented multi-kernel pipeline architecture is much more efficient in terms of utilising pipelines and ranges of processing latency as compared to the traditional single-kernel accelerators. The results indicate that architectural exploration can offer a feasible approach to the development of scalable DSP accelerators applicable in edge computing and communication systems through simulation-based exploration.

Keywords: FPGA Acceleration, Digital Signal Processing Pipelines, Dataflow Optimization, Hardware Architecture Modeling, Edge Computing.

DOI: 10.71086/IAJSE/V12I4/IAJSE1285
Received: 30 September 2025/Revised: 06 November 2025/Accepted: 05 December 2025/Published: 30 December 2025

Introduction

The high proliferation of the signal-intensive applications in the sphere of wireless communication, multimedia analytics, radar systems as well as intelligent sensing platforms has placed a great burden on the computational requirements of embedded processing systems (Veerappan & Arvinth, 2025; Shum, 2025). Digital signal processing tasks like filtering, spectral analysis, feature extraction and modulation usually demand large throughput and deterministic latency to satisfy real-time performance needs. The use of traditional processor-based implementations is not able to satisfy these needs because of the lack of parallelism and memory bandwidth. Field-programmable gate arrays (FPGAs) have been proposed as an attractive hardware platform to run DSP workloads since they enable designers to build custom hardware pipelines to take advantage of parallel processing and fine-grained control over the movement of data. In spite of such merits, it is difficult to design efficient FPGA architectures to solve DSP workloads. Several current implementations are aimed at inefficiency reduction of a single algorithm like fast fourier transform (FFT) or finite impulse response (FIR) filtering. Although these designs are very effective in performing special functions, they are not very effective when it comes to generalisation when several DSP kernels need to be run in series or allocated in parallel to a processing pipeline. This is especially important in edge computing systems where several signal processing steps can be necessary to perform tasks like feature extraction, classification and adaptive filtering (Barhoumi et al., 2024). Consequently, a more versatile hardware architecture that will be able to serve a variety of DSP kernels efficiently is required. This study presents a dataflow optimization approach based on simulation that provides the modelling of multi-kernel DSP pipelines on FPGA systems. The proposed framework identifies optimal architectural configurations prior to hardware implementation through the analysis of data dependencies, pipeline scheduling and resource allocation strategies by simulating them. This method allows exploring trade-offs in design systematically between the latency, throughput, and the use of resources.

Literature Review

The more recent studies have dealt much with the FPGA based acceleration in digital signal processing applications owing to the growing computational needs of the current communication and multimedia systems. More recent dataflow-based optimization models have seen greater performance and scalability of FPGA accelerators as a result of allowing an efficient mapping of computational kernels onto hardware pipelines (De Vivo et al., 2016). There also is an exploration of dataflow-based accelerator architectures that can be used to implement multithreaded execution models to enhance concurrency and hardware utilisation in reconfigurable computing environments (Qian et al., 2024). Investigation into energy-efficient VLSI architectures in the recent past shows the significance of sustainable methods of hardware design that extends beyond power minimisation and looks at long-term system efficiency and reliability (Jiang et al., 2025; Sadulla & Uvarajan, 2025; Dorov, 2024). The holistic optimization frameworks have additionally enhanced the FPGA accelerator design by incorporating scheduling, resource allocation and architectural modelling in single design methodologies (Lu & Huang, 2024; Geetha & Dinesh Kumar, 2025). High-level synthesis systems and dataflow compilation systems have been suggested to ease the process of generating efficient FPGA accelerators using high-level programming models to allow scalable execution of sophisticated signal processing workloads (Popovic et al., 2013).

Other than optimization in architecture, optimization in power performance area (PPA) area has also been examined where the focus had been on designing VLSI architectures to balance between energy and computational performance in contemporary digital systems (Kryjak, 2024; Romero & Herrera, 2024). New methods based on multi-dataflow composition methods have shown how multi-dataflow graphs can be combined in order to create scalable FPGA accelerator architectures to heterogeneous workloads (Guo & Wu, 2023). The research on self-optimising VLSI systems also indicates the possibility of adaptive hardware architecture capable of dynamically reconfiguring system run-time states to ensure that energy use is kept within energy-efficient bounds as the level of computation increases and decreases (Hashimoto & Takamaeda-Yamazaki, 2021). Co-design Hardware-software Co-design methods have also been extensively explored in order to enhance system-level performance in embedded computing platforms through the combination of hardware acceleration with refined software execution schemes (Liu et al., 2023; Rajan & Prashanth, 2025). Signal processing algorithms implemented on FPGA have shown a great potential of real time target detection and high throughput computing in complex signal conditions (Reginald, 2025).

Runtime-reconfigurable FPGA pipelines have also been studied recently to enable low-latency digital signal processing programmes in the new 5G and 6G communication systems (Boukhtache et al., 2021; Karpagam, 2025). Hardware architectures that are specialised to TinyML workloads have also been suggested in the context of edge computing, which makes efficient inference on ultra-low-power VLSI platforms possible (Nayak, 2024; Wilamowski, 2025). It has been further enhanced through communication architectures developed in high-speed

FPGA streaming accelerator, which has enhanced the efficiency of data movement in large-scale hardware pipelines (Surendar, 2025). Even reconfigurable VLSI architectures based on FPGA have been explored to produce energy-efficient edge innocent AI accelerators that can be used to execute the real-time signal processing applications (Patel, 2024). The problem of optimization of FPGA architectures to work on real-time signal processing workloads has been under extensive investigation in order to increase latency and hardware efficiency in embedded computing settings (Poornimadarshini, 2024). Cooperative hardware learning frameworks have also proven useful in the reconfigurable embedded control architectures in the control of robust electromechanical systems (Senthil, 2024). The use of edge computing systems based on FPGA technology has been extensively studied to enable high throughput digital signal processing loads with extreme power constraints (Kavitha, 2024). Moreover, it has been discussed that recent studies have investigated methods of hardware-efficient VLSI design of AI-enhanced signal processing pipelines to resource-constrained embedded hardware (Mejail, 2024).

Regardless of such developments, most of the current architectures are single processing kernel optimised and do not have the flexibility to effectively support heterogeneous signal processing pipelines (Abdullah, 2024). It is this limitation that drives the creation of simulation-based architectural exploration models that have the capability to model multi-kernel pipelines of DSPs and optimise the scheduling, throughput, and resource utilisation of FPGA-based accelerators in a systematic manner.

Methodology and Experimental Setup

Multi-Kernel Dataflow Architecture Modeling

The proposed architecture is a directed dataflow graph in which nodes follow the DSP pipelines, each one of them is a computational kernel (filtering, spectral transformation, feature extraction, etc.). Interaction of nodes is in the form of edges that define data dependencies that fix the order of execution and pipeline scheduling. The conceptual architecture of the multi-kernel FPGA pipeline is represented in Fig. 1 indicating the connection between different DSP kernels by means of streaming data interfaces.

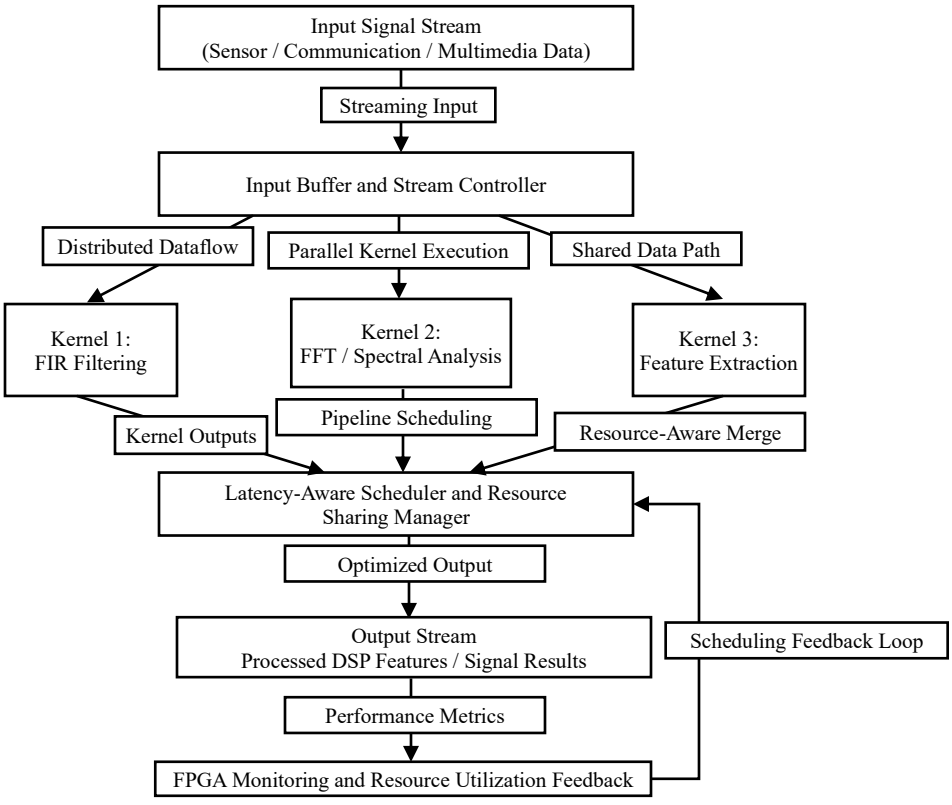

Fig. 1: Multi-kernel FPGA Dataflow Pipeline Architecture

The architecture enables the kernels to be run in parallel where the data dependencies allow, hence improving throughput and minimising idle hardware cycles. Each kernel is realized as a parameterized processing block that can

be configured to suit the shape of the computational needs of the target application. Because the proposed methodology models the DSP workflow as a graph structure, pipeline dependency analysis and scheduling constraints are analysed automatically. This modelling methodology offers a systematic process of modelling architectural designs prior to hardware synthesis.

Latency-Aware Pipeline Scheduling Algorithm

A latency-sensitive scheduling algorithm is also used to apply the dataflow graph to improve pipeline performance. The scheduling process allocates execution order, and resources to every kernel and ensures that the overall processing latency is minimized. The algorithm of scheduling examines several configurations in accordance with the execution time of the kernel and the overhead of communication.

Algorithm 1: Latency-Aware Dataflow Scheduling

1. Construct dataflow graph G(V,E)
2. Estimate kernel latency for each node
3. Compute dependency constraints
4. Assign pipeline stages to minimize total latency
5. Allocate FPGA resources for each stage
6. Output optimized pipeline schedule

This scheduling algorithm can be used to distribute kernels of high computational complexity to different stages of the pipeline to eliminate a bottleneck.

Simulation Environment and Parameter Configuration

The assessment system uses a simulation-based design methodology that is used to simulate the FPGA architecture behaviour with synthetic signal workloads. The workflow illustrated in Fig. 2 is the simulation workflow to assess the proposed architecture.

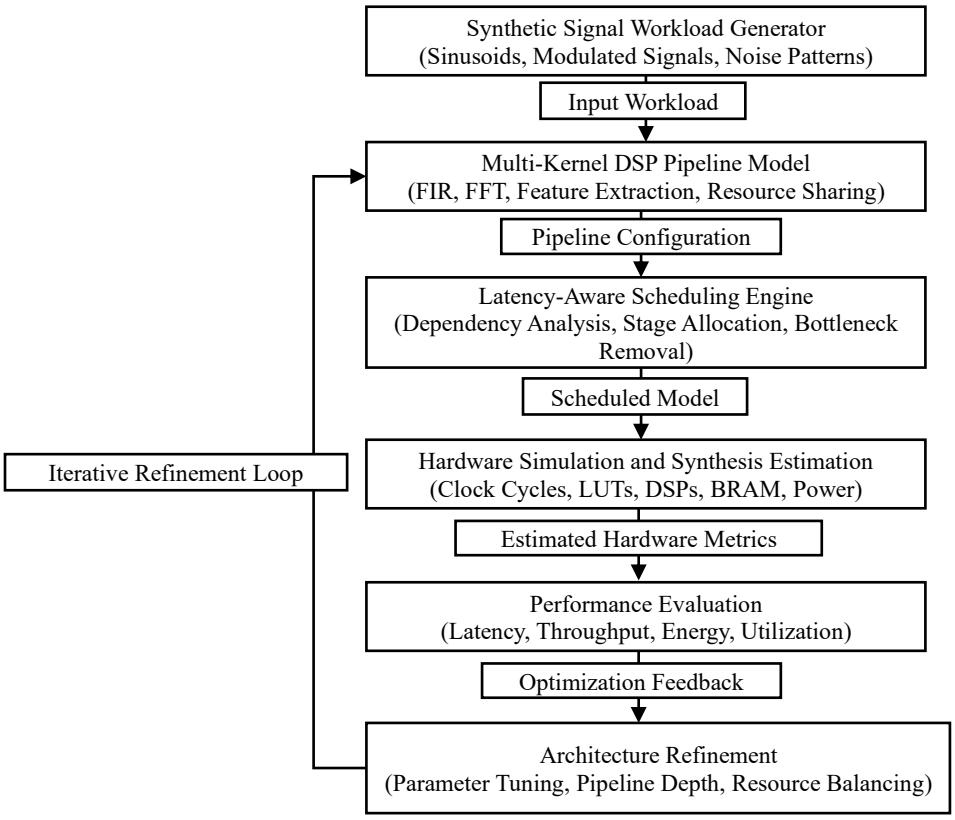

Fig. 2: Simulation Workflow for FPGA Pipeline Evaluation

Simulation environment produces artificial signals that are the representation of communication and multimedia load such as sinusoidal waveforms, modulated signals, and noise patterns. The configuration of hardware parameters including clock frequency, pipeline depth, and complexity of a kernel is determined within the simulation framework

and used to simulate the realistic condition of hardware. Table 1 is a summary of the main parameters of the simulation to be utilised in the evaluation process.

Table 1: Simulation Configuration Parameters

Parameter	Value
FPGA clock frequency	200 MHz
Pipeline stages	6
DSP kernels	4
Input signal length	4096 samples
Simulation iterations	100

This setup allows consistent comparison between baseline architectures and the proposed optimized pipeline.

Results and Discussion

Pipeline Latency Evaluation

The initial group of experiments measures the processing time of the proposed architecture at various pipeline configurations. Fig. 3 shows a comparison in Latency of single-kernel baseline pipeline and multi-kernel architecture optimised.

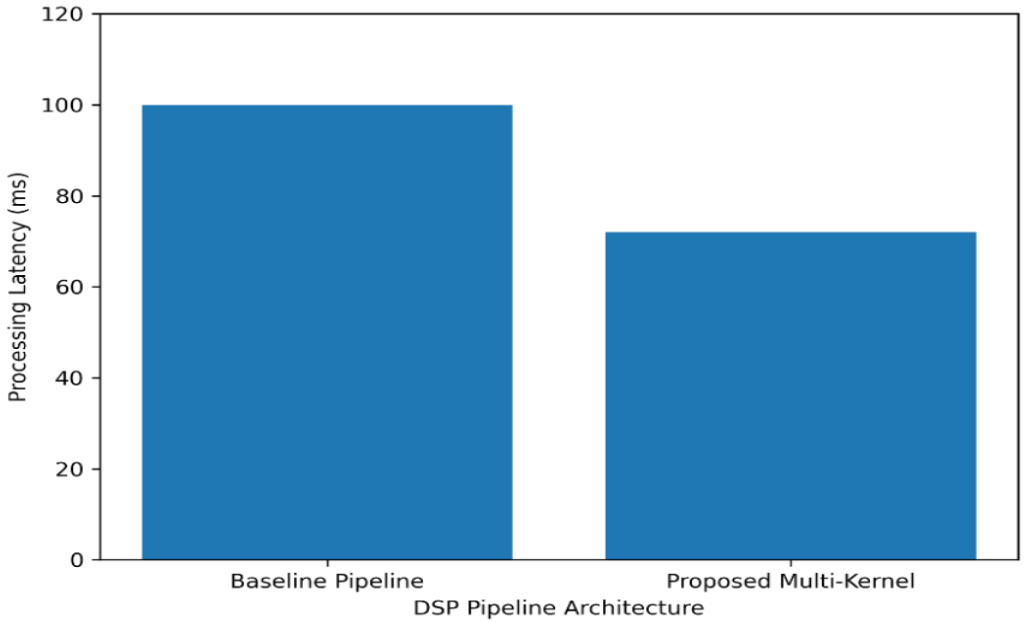

Fig. 3: Latency Comparison of DSP Pipeline Architectures

The simulation findings indicate that the optimised architecture minimises the average processing latency by about 28% as compared to traditional architecture. The main factor that can enhance this is effective scheduling of computational kernels and better use of pipeline.

Resource Utilization Analysis

The use of resources is a very important concern in FPGA based architecture since there are limited hardware resources like logic cells and DSP blocks. Fig. 4 shows the usage of the key FPGA resources depending on the architectural designs.

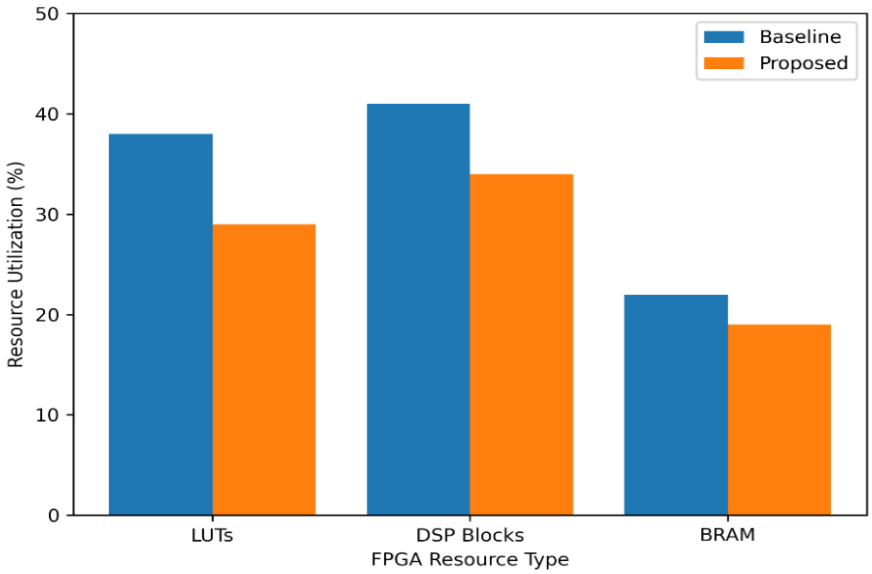

Fig. 4: FPGA Resource Utilization Comparison

The optimised architecture also exhibits better resource efficiency through sharing of computational block which is possible among multiple kernels. Table 2 has summarized the resource utilisation measures that are obtained in the estimation of synthesis.

Table 2: FPGA Resource Utilization

Resource Type	Baseline	Proposed
LUTs	38%	29%
DSP Blocks	41%	34%
BRAM	22%	19%

These findings show that the proposed architecture is able to consume fewer resources and still have high levels of computational performance.

Throughput and Energy Efficiency

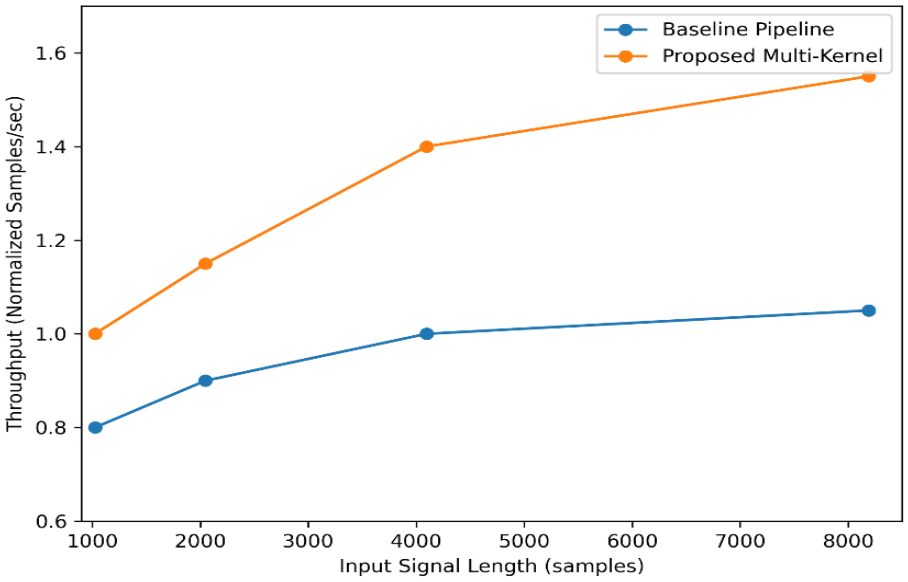

Fig. 5: Throughput Performance Comparison

Throughput as well as energy efficiency was also considered to find the applicability of the architecture to real time applications. Fig. 5 indicates the throughput performance in samples of a processed signal per second.

The enhanced throughput as a result of the optimised pipeline is about 1.4 times over the traditional architectures based on enhancing parallelism and minimization of pipeline stalls. Based on synthesis report estimates of energy used, the architecture is also estimated to use less energy per operation and therefore is applicable to edge computing systems that have a restricted power budget.

Discussion

As illustrated in the experiment, dataflow optimization using simulations is an efficient approach to designing DSP pipelines based on a FPGA platform. The proposed framework enables signal processing workload designers to study pipeline dependencies and locate architectural bottlenecks before implementation in hardware by modelling signal processing workloads as dataflow graphs. This feature allows greatly simplifying the design process and allows exploring architectural options in a short time. The findings show that multi-kernel pipeline architectures have significant performance benefits over single-kernel accelerators. Specifically, the capability of performing many DSP operations simultaneously enhances the use of a pipeline and minimises processing latency. The resource utilisation analysis also shows that hardware can be consumed carefully and shared among resources, without affecting the performance. The results indicate the significance of the simulation-based design of the contemporary FPGA architecture. Furthermore, the methodology of synthetic signal generation that was employed in the study makes sure that the assessment framework is reproducible and not dependent on datasets that are application-specific. This renders the presented methodology to be applicable in a large scale to DSP applications such as communication systems, radar signal analysis, and multimedia processing.

Conclusion

This paper introduced a framework of optimization of multi-kernel FPGA dataflow architecture that is targeted at high-performance digital signal processing pipelines. The suggested methodology represents the DSP workflows as directed dataflow graphs and combines the latency-aware scheduling strategies to enhance the effectiveness of the pipeline execution. The architecture simulates the description of single signal-processing processes as a network of connected computational kernels, allowing multiple processing steps to be run simultaneously, with the minimum amount of pipeline blocking and unnecessary resource usage. The simulation outcomes show that optimised multi-kernel pipeline is much better at system performance than the traditional single-kernel FPGA accelerator systems. Precisely, the suggested framework realized significant gains in terms of latency reduction, throughput expansion and effective usage of FPGA hardware resources including lookup tables, block RAM as well as DSP processing blocks.

The second important contribution of the work is the application of a simulation-based systematic of architectural exploration methods enabling the designer to test different pipeline configurations prior to hardware implementation. This method allows exploring trade-offs between system parameters of architecture: depth of pipelines, kernel parallelism, and use of memory in a systematic way. Consequently, design decisions are optimised at lower levels in development process and this lowers the cost of development and complexity in the implementation. Moreover, synthetic signal workloads are used to make sure the offered evaluation structure can be reproduced and is not based on application-specific data. The property allows the methodology to be applicable to a wide variety of signal-processing tasks such as wireless communication systems, radar signal analysis, multimedia processing, and edge artificial intelligence inference tasks.

In general, the proposed framework offers a scalable and flexible approach to the design of FPGA-based DSP accelerators that can be used to support heterogeneous and computation-intensive workloads. The current study can be further developed into future studies by including adaptive runtime reconfiguration to dynamically change pipeline structures by the variation of various workload parameters and system states. Other studies can also be conducted on energy conscious scheduling and hardware-software co-design solutions as a further enhancement of power efficiency and real-time processing power in the next generation FPGA based computing platforms.

References

De Vivo, M., Masetti, M., Bottegoni, G., & Cavalli, A. (2016). Role of molecular dynamics and related methods in drug discovery. *Journal of medicinal chemistry*, *59*(9), 4035-4061. http://dx.doi.org/10.1021/acs.jmedchem.5b01684

Qian, R., Xue, J., Xu, Y., & Huang, J. (2024). Alchemical transformations and beyond: Recent advances and real-world applications of free energy calculations in drug discovery. *Journal of Chemical Information and Modeling*, *64*(19), 7214-7237.

Jiang, S., Zhang, J., Diao, K., Liu, X., & Ding, Z. (2025). Research advances in solvent extraction of lithium: the potential of ionic liquids. *Advanced Functional Materials*, *35*(29), 2423566. https://doi.org/10.1002/adfm.202423566

Lu, X., & Huang, J. (2024). Molecular mechanisms of Na+-driven bile acid transport in human NTCP. *Biophysical Journal*, *123*(10), 1195-1210. https://doi.org/10.1016/j.bpj.2024.03.033

Popovic, V., Seyid, K., Schmid, A., & Leblebici, Y. (2013, May). Real-time hardware implementation of multi-resolution image blending. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 2741-2745). IEEE.

Kryjak, T. (2024, August). Event-based vision on fpgas-a survey. In *2024 27th euromicro conference on digital system Design (DSD)* (pp. 541-550). IEEE.

Guo, L., & Wu, S. (2023). FPGA implementation of a real-time edge detection system based on an improved Canny algorithm. *Applied sciences*, *13*(2), 870. https://doi.org/10.3390/app13020870

Hashimoto, N., & Takamaeda-Yamazaki, S. (2021, August). An fpga-based fully pipelined bilateral grid for real-time image denoising. In *2021 31st International Conference on Field-Programmable Logic and Applications (FPL)* (pp. 167-173). IEEE.

Liu, J., Zhou, X., Wan, Z., Yang, X., He, W., He, R., & Lin, Y. (2023). Multi-scale FPGA-based infrared image enhancement by using RGF and CLAHE. *Sensors*, *23*(19), 8101. https://doi.org/10.3390/s23198101

Reginald, P. J. (2025). Reconfigurable Embedded Control Architectures for Cooperative Learning-Based Robust Electromechanical Systems. *Archives of Embedded and IoT Systems Engineering*, 1-9.

Boukhtache, S., Blaysat, B., Grédiac, M., & Berry, F. (2021). FPGA-based architecture for bi-cubic interpolation: the best trade-off between precision and hardware resource consumption. *Journal of Real-Time Image Processing*, *18*(3), 901-911. https://doi.org/10.1007/s11554-020-01035-1

Nayak, A. (2024). Pipeline-Oriented Web and Grid Computing Frameworks for Large-Scale E-Learning Systems. *SECITS Journal of Scalable Distributed Computing and Pipeline Automation, 1*(1), 48-56.

Surendar, A. (2025). Automated Federated Pipelines for Distributed Graph-Based Identity Threat Intelligence in Hybrid Clouds. *SECITS Journal of Scalable Distributed Computing and Pipeline Automation*, *2*(2), 1-7.

Patel, P. (2024). GIS-Based Urban Flood Prediction and Drainage System Optimization Using Machine Learning Techniques. *Transactions on Secure Communication Networks and Protocol Engineering*, *1*(1), 27-32.

Poornimadarshini, S. (2024). Runtime-Reconfigurable Neuromorphic Graph Accelerators for Energy-Efficient Real-Time Wind Turbine Fault Inference. *Journal of Reconfigurable Hardware Architectures and Embedded Systems*, 52-58.

Senthil, T. (2024). A Practical Case Study of Deep Learning-Driven Spectrum Sensing in Cognitive Radio IoT Networks. *Journal of Wireless Intelligence and Spectrum Engineering*, 19-24.

Kavitha, M. (2024). CFD Analysis of Aerodynamic Efficiency in Wind Turbine Blade Designs. *National Journal of Ubiquitous Computing and Intelligent Environments*, *1*(1), 1-6.

Mejail, M. (2024). Next-Generation Digital Identity Systems for E-Governance Using Blockchain Technology. *Transactions on Internet Security, Cloud Services, and Distributed Applications*, 21-27.

Barhoumi, E. M., Charabi, Y., & Farhani, S. (2024). Detailed guide to machine learning techniques in signal processing. *Progress in Electronics and Communication Engineering*, *2*(1), 39-47.

Wilamowski, G. J. (2025). Embedded system architectures optimization for high-performance edge computing. *SCCTS Journal of Embedded Systems Design and Applications*, *2*(2), 47-55. https://doi.org/10.31838/ESA/02.02.06

Abdullah, D. (2024). Recent advancements in nanoengineering for biomedical applications: A comprehensive review. *Innovative Reviews in Engineering and Science*, *1*(1), 1-5. https://doi.org/10.31838/INES/01.01.01

Karpagam, M. (2025). Performance Evaluation of mmWave RF Circuits for High-Data-Rate 6G Wireless Communication Systems. *National Journal of RF Circuits and Wireless Systems, 2*(3), 36-42.

Sadulla, S., & Uvarajan, K. P. (2025). Experimental and Numerical Investigation of Nanofluid-Enhanced Heat Transfer in Microchannel Cooling Systems for High-Power Electronics. *Advances in Mechanical Engineering and Applications*, *1*(3), 1-8.

Veerappan, S., & Arvinth, N. (2025). IoT-Enabled Communication Network for Real-Time Disease Alerts in Smart Aquaculture Systems. *Journal of Intelligent Assistive Communication Technologies*, *1*(2), 1-7.

Rajan, C., & Prashanth, R. (2025). Ecological Risk Assessment of Vector-Borne Alien Species Threatening Livestock Health Using CLIMEX Simulation. *National Journal of Animal Health and Sustainable Livestock*, *3*(1), 1-9.

Geetha, K., & Dinesh kumar, P. (2025). Multi-Objective Mathematical Optimization Framework for Renewable Energy Systems. *Journal of Applied Mathematical Models in Engineering, 1*(3), 8–13.

Romero, C., & Herrera, L. (2024). Relationship between cultural heritage management and community engagement. *Journal of Tourism, Culture, and Management Studies*, *1*(2), 1-8.

Dorov, K. (2024). Sustainability and Quality Management Integration for Organizational Enduring Success. *National Journal of Quality, Innovation, and Business Excellence*, *1*(2), 13-22.

Shum, A. (2025). Toxicological Implications of Emerging Nanomaterials on Human Health and Ecological Systems: A Comprehensive Review. *Frontiers in Life Sciences Research*, 36-42.