Decoherence-Optimized Circuits for Multidimensional and Multilevel-Decomposable Quantum Wavelet Transform

Naveed Mahmud, Andrew MacGillivray, Manu Chaudhary, and Esam El-Araby, *University of Kansas, Lawrence, KS, 66045, USA*

Algorithms such as quantum wavelet transform (QWT) can be utilized on quantum processors to gain significant speedup compared to their classical counterparts. Domains such as high-energy-physics and remote-sensing hyperspectral imagery require high compute capabilities and can benefit from applying QWT for dimension reduction of multidimensional data. However, quantum circuits for QWT include permutations that contribute to large overall circuit depth. Deep circuits pose a critical challenge for state-of-the-art quantum computers because of quantum decoherence. In this work, we propose QWT circuits, which are optimized in terms of circuit depth, to account for the effects of decoherence, and resulting in high fidelity and efficient implementation on quantum processors. We present the circuits in generalized forms and show that they can be used for multilevel decomposable, multidimensional wavelet operations. Experimental evaluation of the proposed circuits is performed through simulation using MATLAB and IBM-Q qasm, and implemented on a real 15-qubit quantum processor from IBM.

Quantum computing is establishing itself as a promising technology of the future. The current class of quantum hardware are termed as noisy-intermediate-scale-quantum (NISQ) devices[1] because of their sensitivity to external noise, and limited processing power (number of quantum bits). NISQ and other quantum computing devices rely on the coherence of quantum states to perform information processing operations. However, coherence is lost over time, a process called quantum decoherence, due to environmental influences that affect the fragile superposition states of quantum bits (qubits).[1] Quantum decoherence occurs within the range of microseconds on many state-of-the-art quantum devices.[1] Maintaining the runtime of nontrivial algorithms shorter than the decoherence time is critical for the correct operation of quantum computers. Consequently, optimizing the depth of quantum circuits minimizes decoherence effects and has lately become a common focus of research.

In this article, we focus our work on the optimization of quantum-wavelet-transform (QWT) circuits to minimize the effects of quantum decoherence. Classical wavelet-based techniques are popular for dimension reduction of multidimensional data and are effective in data preprocessing, reducing computation overhead, and improving classification accuracy.[2] QWT has been shown to be a more effective tool than the classical discrete wavelet transform (DWT) in performing the same tasks, and is particularly useful for reducing the dimensionality of multispectral data.[3–6] However, the circuits for QWT presented in previous and related works have high circuit depth, resulting in execution times that exceed the decoherence time of many modern quantum processors. Practical implementations of these circuits on real quantum computers require improvements in either qubit stability or the execution speed of the circuits. In this work, we present decoherence-optimized circuits for QWT, namely the quantum Haar transform (QHT).[5,6] The proposed QHT circuits

1089-7801 © 2021 IEEE
Digital Object Identifier 10.1109/MIC.2021.3133845
Date of publication 9 December 2021; date of current version 28 January 2022.

have been optimized for minimal depth to mitigate decoherence effects and facilitate efficient implementations on quantum processors. Generalized circuits are proposed for multidimensional, multilevel decomposable QHT in two variants: 1) sequential (d-stages) QHT, where one-dimensional QHT (1D-QHT) is applied sequentially to achieve d-dimensional QHT and 2) parallel (one-stage) QHT, where a d-dimensional QHT operation is applied in one parallel stage. For each variant, we propose optimized circuits, derive expressions for circuit depths, and use the three-dimensional QHT (3D-QHT) as an example. We also show that the proposed circuits are multilevel decomposable (packet-based and pyramidal) and can be used for multidimensional Haar operations. For experimental evaluation, we employ MATLAB for obtaining data visualization, and IBM quantum experience (IBM-Q)[7] framework for obtaining realistic runtimes and fidelity of the proposed circuits. RGB-colored and multispectral images are used as test-data and multilevel 3D-QHT is performed.

The rest of this article is structured as follows. The "Background and Related Work" section presents the background and related works. The "Proposed Quantum Circuits for Multilevel Multidimensional Quantum-Haar-Transform (QHT)" section contains the proposed circuits for QHT. The "Experimental Work" section contains the experimental work and results. Finally, the "Conclusion and Future Work" section concludes this article.

BACKGROUND AND RELATED WORK

Qubit Superposition

The qubit is the smallest unit of information in quantum computers.[8] A single qubit exists in superposition of two basis states, $|0\rangle$ and $|1\rangle$. The pure state $|\psi\rangle$ of the qubit is satisfied by the linear superposition (1), where α and β are complex numbers. Measurement of a qubit collapses its state to a basis state such that $|\alpha|^2$ and $|\beta|^2$ are the probabilities of finding the qubit in the basis states $|0\rangle$ and $|1\rangle$, respectively

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \equiv \begin{bmatrix} \alpha \\ \beta \end{bmatrix}. \tag{1}$$

Decoherence

Decoherence is the process by which the environment affects the state of qubits.[1] Interactions with the environment cause information to be lost and the qubit's state to become increasingly mixed.[1,8] The mixed state ρ can be written as a density matrix where off-diagonal elements decay over time[8]

$$\rho(t) = \begin{bmatrix} |\alpha|^2 & e^{-t/\tau}\alpha\beta^* \\ e^{-t/\tau}\alpha^*\beta & |\beta|^2 \end{bmatrix} \tag{2}$$

where * indicates the complex conjugate, t denotes the time, and τ denotes the decoherence time (constant). As time progresses, quantum interference is suppressed and the ability to perform additional operations is lost.[1,8]

The decoherence time of a quantum system is a real-time constraint for quantum circuits/algorithms. The desired output cannot be captured if the operations cannot be completed within decoherence time. Therefore, circuits must be optimized to meet the time constraint of their target quantum device/technology.

Quantum Gates

In a circuit model of quantum computing,[8] computation begins with the system in an unentangled quantum state $|\psi\rangle$, which can be expressed as a superposition of $N = 2^n$ basis states, where n is the number of qubits. Depending on the quantum algorithm, different unitary transformations or quantum gates can be applied to reach a final quantum state. In this work, we utilize the Hadamard H and SWAP gates[8] [see Figure 1(a)]. We also model the time delays of the H and the SWAP gates as τ_H and τ_{SWAP}, respectively.

Quantum Wavelet Transform (QWT)

The classical wavelet transform can be implemented as QWT[4,5] in the quantum-information-processing (QIP) domain. The general QWT can be expressed[6] by

$$\psi\rangle = \sum_{q=0}^{N-1} f(q \cdot \Delta t)|q\rangle, \text{ where } \sum_{q=0}^{N-1} |f(q \cdot \Delta t)|^2 = 1 \tag{3}$$

$$|\psi\rangle_{\text{QWT}} = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{q=0}^{N-1} f(q \cdot \Delta t) \Psi\left(\frac{q-j}{K}\right)|j\rangle \tag{4}$$

where Ψ is the mother wavelet function in complex conjugate form, Δt is the sampling period, K is the wavelet window size in samples, $N = 2^n$ is the number of data samples represented as the total number of quantum basis states, n is the number of qubits, $|\psi\rangle$ is the input state, and $|\psi\rangle_{\text{QWT}}$ is the output state.

Quantum Permutations

Permutations and perfect-shuffle permutations (PSP) are fundamental operations that are used in QWT and in many classical computations involving signal and image processing.[9] Quantum permutations can be described directly in terms of their effect on the ordering of qubits.[3–5] We, here, focus on two PSP operations

that will be used in building our proposed quantum circuits, i.e., rotate left (RoL) and rotate right (RoR). RoL $(n-1,0)$ (5) and RoR$(n-1,0)$ (6) operations are essentially circular (left/right) shifts of qubits. RoL/RoR can be implemented with networks of SWAP gates [see Figure 1(b)]. The number of levels of SWAP gates required for RoL$(n-1,0)$/RoR$(n-1,0)$ is simply $n-1$. The gate symbols we have used for RoL/RoR in our proposed circuits are shown in Figure 1(b)

$$\mathrm{RoL}(n-1,0):|q_{n-1}q_{n-2}\ldots q_1q_0\rangle \mapsto |q_{n-2}\ldots q_1q_0q_{n-1}\rangle \tag{5}$$

$$\mathrm{RoR}(n-1,0):|q_{n-1}q_{n-2}\ldots q_1q_0\rangle \mapsto |q_0q_{n-1}q_{n-2}\ldots q_1\rangle. \tag{6}$$

Related Work

Early work on QWT appeared in Fijany and Williams's work,[4] where they presented circuits for quantum Haar and Daubechies wavelets. They proposed techniques for implementation of permutation matrices for factoring the unitary operations of QWT. Ohnishi *et al.*[10] demonstrated quantum circuits for QWT based on the well-known packet and pyramidal algorithms of classical DWT. Sheng *et al.*[5] provided partial quantum circuit derivations for the Haar and Daubechies WT. The authors also proposed implementation circuits of multilevel and multidimensional packet QWT. Most of the previously reported works lacked detailed discussions, complete circuits, circuit depth optimizations, or actual hardware implementations.

The authors in a more recent work[6] proposed the application of QWT for reducing the dimensionality of high spatial-resolution data with improved processing time. They also implemented multipattern Grover's search for pattern recognition. Their proposed circuits were evaluated using an FPGA-based emulation architecture.

We focus our work on the optimization of QWT circuits to achieve minimal circuit depth, in addition to providing complete circuits with realistic experimental results and analyses. We present the circuits in a generalized, decomposable form, and use them for multilevel multidimensional Haar operations. To the best of our knowledge, our work is the first to perform the following:

1) propose QHT circuits optimized in terms of circuit depth, in order to mitigate decoherence effects;
2) provide circuits for both packet and pyramidal decompositions of QHT;
3) evaluate the proposed QHT circuits on a state-of-the-art quantum computing platform, providing theoretical estimations of run-times combined with realistic and high-fidelity simulation and hardware implementation results and analysis.

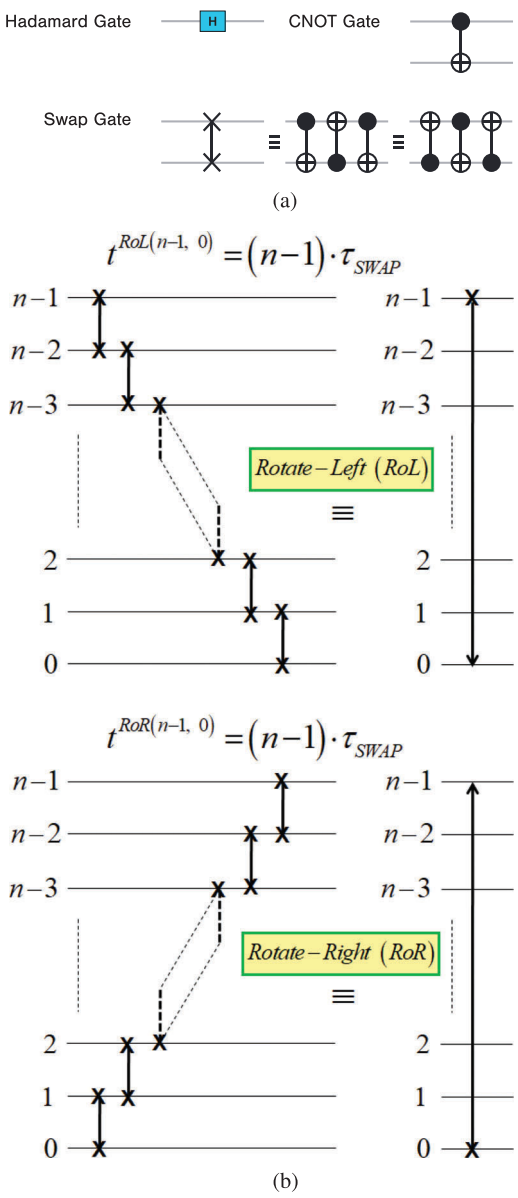

FIGURE 1. Quantum gates and operations used. (a) H, CNOT, and SWAP quantum-gates. (b) RoL and RoR operations.

PROPOSED QUANTUM CIRCUITS FOR MULTILEVEL MULTIDIMENSIONAL QHT

We denote a general d-dimensional QHT operation as $U^{d\text{-D-QHT}}$. The $U^{d\text{-D-QHT}}$ operation consists of three parts: 1) input permutations applied to the input state vector, 2) Haar-transform operations, and 3) output permutations applied to produce the output state

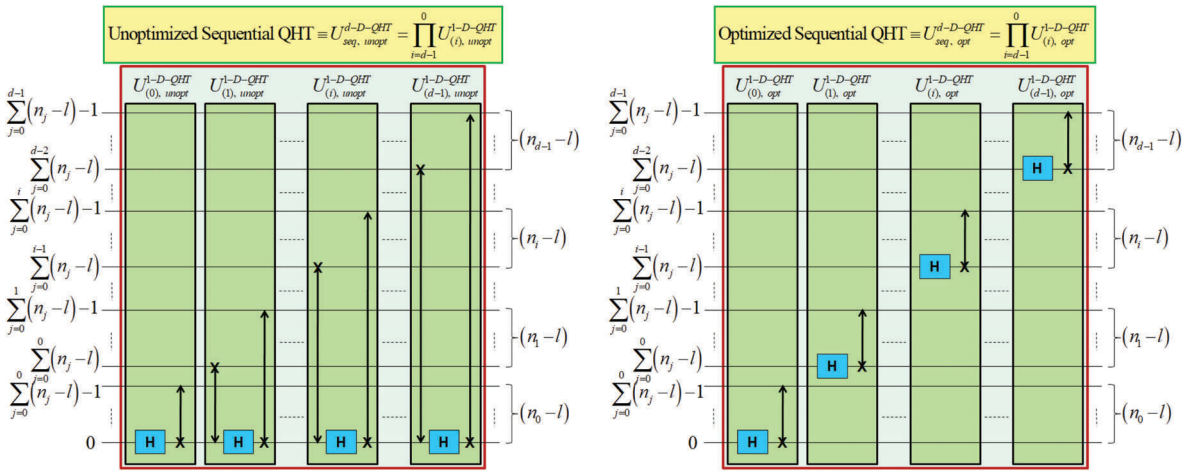

FIGURE 2. Pyramid-decomposable sequential (d-stage) d-dimensional QHT.

vector. The permutations are performed using RoL/RoR gates, while the Haar-transform operations are performed using Hadamard *H* gates (see Figure 1). Detailed examples of 2D-QHT ($U^{2\text{-D-QHT}}$) and 3D-QHT ($U^{3\text{-D-QHT}}$) can be found in Mahmud *et al.*'s paper.[6]

We present two circuit variants that perform the operation $U^{d\text{-D-QHT}}$: sequential (d-stage) d-dimensional QHT, and parallel (one-stage) d-dimensional QHT. The $U^{d\text{-D-QHT}}$ operation can be performed by any of the unoptimized and optimized circuits presented in Figures 2 and 3 for which expressions for circuit depth and time delay are derived. We also show how the sequential and parallel circuit variants use packet and pyramidal decomposition to perform *multilevel-decomposable* multidimensional QHT (see Figure 4). Terminology and notation used in the next sections are defined

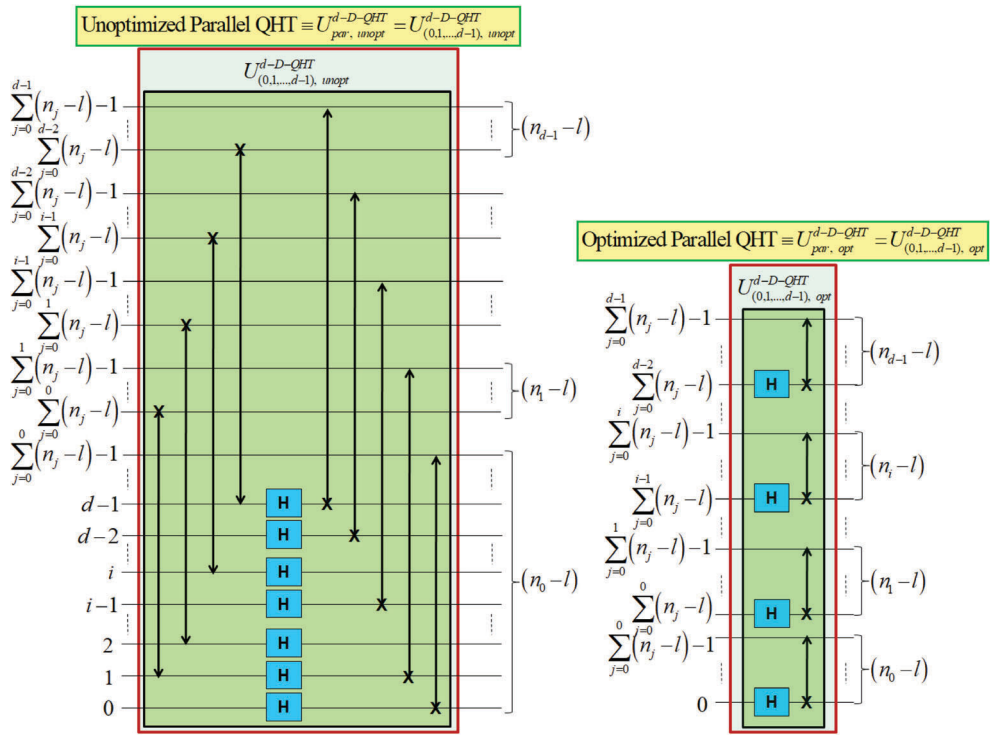

FIGURE 3. Pyramid-decomposable parallel (one-stage) d-dimensional QHT.

as follows:

$$
\begin{aligned}
&N \equiv \text{Total number of data samples}\\
&d \equiv \text{Total number of data dimensions}\\
&l \equiv \text{Number of decomposition levels}\\
&l_{\max} \equiv \text{Maximum number of decomposition levels}\\
&N_i \equiv \text{Total number of data samples in dimension } i\\
&n_i = \lceil \log_2(N_i) \rceil \equiv \begin{cases}\text{Total number of qubits}\\ \text{representing dimension } i\end{cases}\\
&n = \sum_{i=0}^{d-1} n_i \equiv \text{Total number of qubits}\\
&n_{\max} = \max_{i=0}^{d-1}(n_i) \equiv \begin{cases}\text{Maximum number of qubits}\\ \text{across all } d \text{ dimensions}\end{cases}\\
&n_{\min} = \min_{i=0}^{d-1}(n_i) \equiv \begin{cases}\text{Minimum number of qubits}\\ \text{across all } d \text{ dimensions}\end{cases}\\
&l_{lossless} = n_{\min} \equiv \begin{cases}\text{Maximum number of levels for}\\ \text{lossless decomposition}\end{cases}\\
&l_{\max}^{pkt} = \left\lfloor \frac{n}{d} \right\rfloor \equiv \begin{cases}\text{Maximum number of levels for}\\ \text{packet decomposition}\end{cases}\\
&l_{\max}^{pyr} = \left\lfloor \min\left(\frac{n}{d}, 1+\frac{n-n_0}{(d-1)}\right)\right\rfloor \equiv \begin{cases}\text{Maximum}\\ \text{number of}\\ \text{levels for}\\ \text{pyramidal}\\ \text{decomposition}\end{cases}\\
&\text{Condition for lossless decomposition}\\
&0 \le l \le l_{lossless}\\
&\Rightarrow \text{No dimensions are lost during decomposition}\\
&\text{Condition for lossy decomposition}\\
&l_{lossless} < l \le l_{\max}\\
&\Rightarrow \text{Dimensions are lost during decomposition}\\
&\\
&\tau_H \equiv \text{Time delay of the Hadamard-gate}\\
&\tau_{SWAP} \equiv \text{Time delay of the Swap-gate}\\
&t_{total} \equiv \text{Total time delay}\\
&\\
&U^{d-D-QHT} \equiv \text{Generic } d\text{-dimensional QHT operation}\\
&U_{pkt,\,l}^{d-D-QHT} \equiv \begin{cases}d\text{-dimensional QHT at level } l \text{ of}\\ \text{packet decomposition}\end{cases}\\
&U_{pyr,\,l}^{d-D-QHT} \equiv \begin{cases}d\text{-dimensional QHT at level } l \text{ of}\\ \text{pyramidal decomposition}\end{cases}\\
&U_{packet}^{d-D-QHT} \equiv \begin{cases}\text{overall } d\text{-dimensional QHT of}\\ \text{packet decomposition}\end{cases}\\
&U_{pyramidal}^{d-D-QHT} \equiv \begin{cases}\text{overall } d\text{-dimensional QHT of}\\ \text{pyramidal decomposition}\end{cases}
\end{aligned}
\tag{7}
$$

Packet and Pyramidal Decompositions of d-Dimensional QHT

Classical multilevel-decomposable DWT is often used for image compression because it maintains data spatial-locality. We show that, similar to classical DWT, d-dimensional QHT (sequential and/or parallel) is packet-/pyramid-decomposable for l levels.

In packet decomposition [see Figure 4(a)], $U^{d\text{-D-QHT}}$ is repeatedly applied for every level on all the data (qubits) and all data qubits are required throughout the entire process. The maximum number of levels for packet decomposition $l_{\max}^{\text{p kt}}$ [see (7)] depends on the total number of qubits n and the number of data dimensions d. However, for lossless decomposition, i.e., no data dimensions are lost during decomposition, the maximum number of levels is equal to the minimum number of qubits $n_{\min}$ across all d dimensions.

In pyramidal decomposition [see Figure 4(b)], for each level of decomposition, the d-dimensional QHT operates on fewer data qubits. Specifically, d qubits (1 qubit per dimension) are discarded after every decomposition level [see Figure 4(b)]. Similar to packet decomposition, the maximum number of levels for lossless pyramidal decomposition is also $n_{\min}$. The expression for the maximum number of possible pyramidal decomposition levels, $l_{\max}^{\text{pyr}}$, is given in (7). Pyramidal decomposition has certain advantages over packets in that the size and depth of the QHT circuit are reduced after every decomposition level. However, one drawback is that more inter-level permutations are required [see Figure 4(c)]. The time delay for inter-level pyramidal permutations could be derived using Figure 4(c) and is given as follows:

$$
t^{\text{pyr-perm}} = \left(n - n_0 - (d-1)\left(1 + \frac{(l-2)}{2}\right)\right)(l-1)\cdot \tau_{\text{SWAP}}. \tag{8}
$$

Sequential (d-Stage) QHT

The d-dimensional QHT can be performed by sequentially cascading d 1D-QHT transforms (see Figure 2). Each 1D-QHT, $U^{\text{1-D-QHT}}$, consists of RoL, Hadamard H gates, and RoR gates (see Figure 2). The 1D-QHT is consecutively repeated for every dimension indexed from 0 to $d-1$. The (unoptimized) sequential QHT circuit could be decomposed in multilevel packet and pyramidal forms. The total time delays for these decompositions are provided in (9) and (10), which could be derived using Figures 2 and 4

$$
\begin{aligned}
&t_{\text{total}}^{\text{seq, unopt,pkt}}\\
&= \left(\left((2d-1)n - 2\sum_{i=0}^{d-1} i\cdot n_i - d\right)\cdot \tau_{\text{SWAP}} + d\cdot \tau_H\right)\cdot l
\end{aligned}
\tag{9}
$$

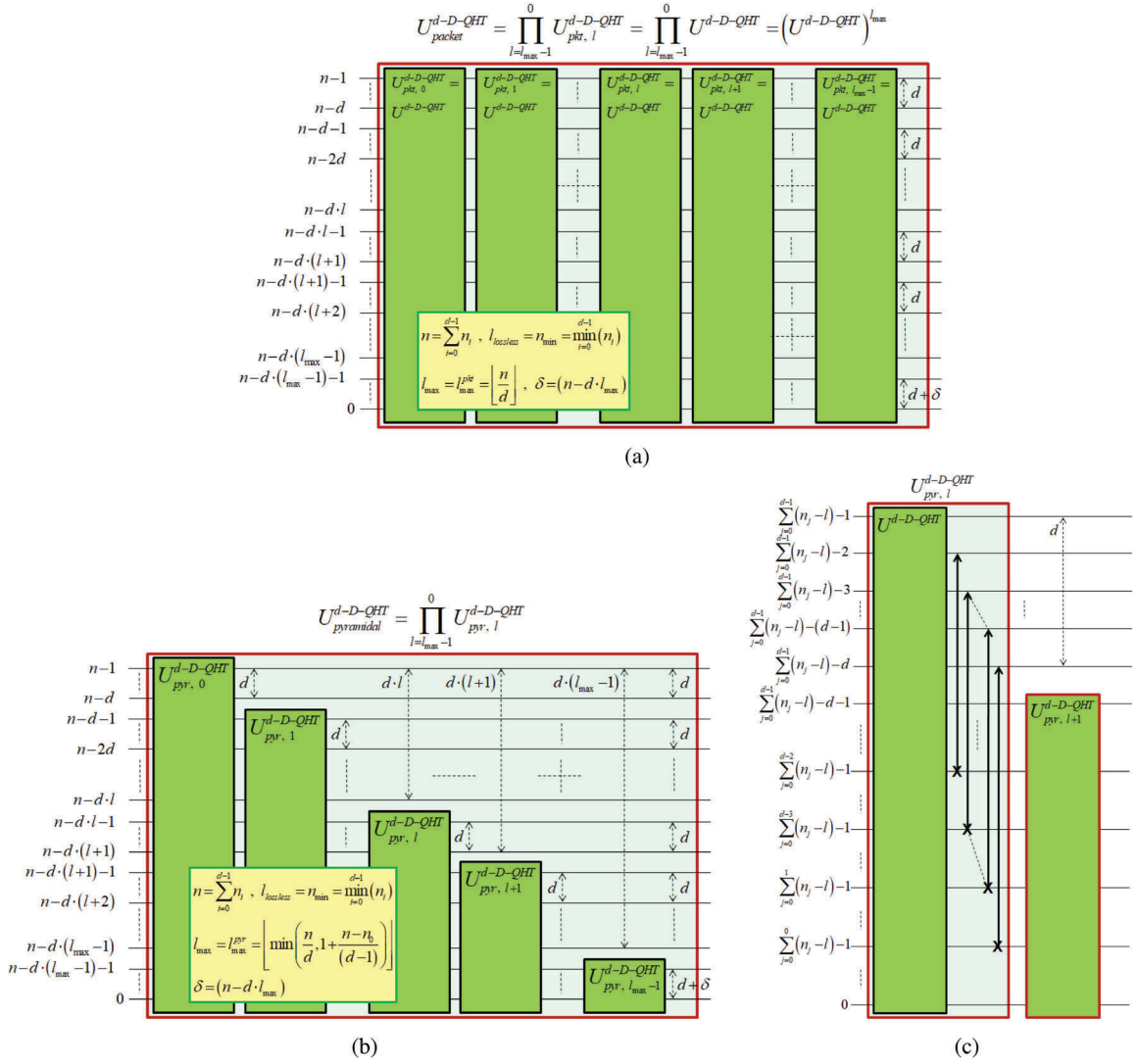

FIGURE 4. Multilevel decomposition of d-dimensional QHT. (a) Packet-decomposition. (b) Pyramidal-decomposition. (c) Inter-level pyramidal-permutations.

$$t_{\text{total}}^{\text{seq, unopt, pyr}} = \left(t_{\text{total}}^{\text{seq, unopt, pkt}} + t^{\text{pyr-perm}} - \frac{d^2 \cdot l(l-1)}{2} \cdot \tau_{\text{SWAP}}\right). \tag{10}$$

Optimized Sequential (d-Stage) QHT

The d-stage sequential QHT circuit can be optimized as shown in Figure 2. For each 1D-QHT operation, the RoL operation is eliminated and the kernel (H gate) is shifted up to its corresponding dimension i, where $i = 0, 1, \ldots, d-1$. This reduces the consecutive RoR operation (less depth) and thus reduces the overall circuit depth. The total time delays for the optimized sequential decomposable (packet/pyramidal) d-dimensional QHT circuits are given in (11) and (12), respectively, which could be derived using Figures 2 and 4

$$t_{total}^{\text{seq, opt, pkt}} = ((n-d) \cdot \tau_{\text{SWAP}} + d \cdot \tau_H) \cdot l \tag{11}$$

$$t_{\text{total}}^{\text{seq, opt, pyr}} = \left(t_{\text{total}}^{\text{seq, opt, pkt}} + t^{\text{pyr-perm}} - \frac{d \cdot l(l-1)}{2} \cdot \tau_{\text{SWAP}}\right). \tag{12}$$

Parallel (One-Stage) QHT

The Haar operation (H gates) can be applied in parallel (one-stage) instead of in sequence on each of the d

dimensions (see Figure 3). The RoR and RoL operations are grouped into sets of preceding and proceeding permutations, respectively (see Figure 3). This circuit variant can be used in packet or pyramidal decomposition. The total time delays of the decomposable circuits are given in (13) and (14), respectively, which could be derived using Figures 3 and 4

$$t_{\text{total}}^{\text{par, unopt,pkt}} = ((2n - n_{d-1} - (2d-1)) \cdot \tau_{\text{SWAP}} + \tau_H) \cdot l \tag{13}$$

$$t_{\text{total}}^{\text{par , unopt, pyr}} = \left(t_{\text{total}}^{\text{par,unopt, pkt}} + t^{\text{pyr-perm}} - \frac{(2d-1) \cdot l(l-1)}{2} \cdot \tau_{\text{SWAP}}\right). \tag{14}$$

Optimized Parallel (One-Stage) QHT

The parallel (one-stage) QHT is optimized further by positioning the H gates separated by n_i qubits, where $i = 0, 1, \ldots, d-1$ (see Figure 3). Due to this shift, no preceding permutations (RoL gates) are required. The proceeding RoR operations are also reduced in depth and can be applied in parallel as they are independent of each other. The total time delays for the optimized parallel decomposable (packet/pyramidal) d-dimensional QHT circuits are given in (15) and (16), respectively, which could be derived using Figures 3 and 4

$$t_{\text{total}}^{\text{par, opt,pkt}} = ((n_{\max} - 1) \cdot \tau_{\text{SWAP}} + \tau_H) \cdot l \tag{15}$$

$$t_{\text{total}}^{\text{par , opt, pyr}} = \left(t_{\text{total}}^{\text{par, opt,pkt}} + t^{\text{pyr-perm}} - \frac{l(l-1)}{2} \cdot \tau_{\text{SWAP}}\right). \tag{16}$$

EXPERIMENTAL WORK

Experiments Using MATLAB

Simulation models for each of the circuit variants were developed in MATLAB version R2020a. The test-data used were $64 \times 64 \times 3$ RGB images and high-resolution $1024 \times 1344 \times 33$ multispectral images. The multispectral images were captured in Minho, Portugal, using embedded neutral probe spheres.[11] Zero-padding was used to extend the data to power-of-2 datapoints in each dimension for the proper operation of the QHT kernel. Figure 5(a) and (b) shows a $64 \times 64 \times 3$ input image and the corresponding output image after one level of parallel (one-stage) 3D-QHT packet decomposition, respectively. After one level ($l = 1$) 3D-QHT, the dimensions were reduced by a factor of $\frac{1}{2^l} = \frac{1}{2}$, where l is the number of decomposition levels. Figure 5(d) and (e)

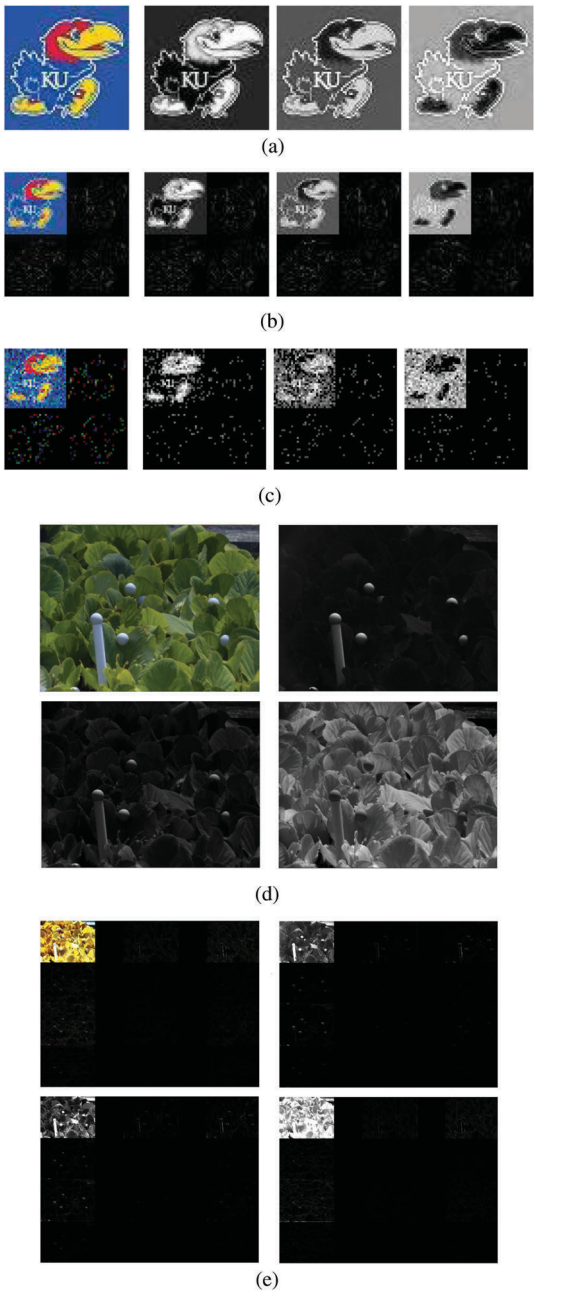

FIGURE 5. Dimension reduction of 64×64×3 RGB-image using one-level parallel (one-stage) 3D-QHT and of 1024×1344×33 multispectral image using two-level parallel (one-stage) 3D-QHT. (a) Input RGB-image and its 3 spectral-bands. (b) Decomposed RGB-image and its 3 spectral-bands after 1-level 3D-QHT in MATLAB. (c) Decomposed RGB-image and its 3 spectral-bands after 1-level 3D-QHT in the IBM-Q qasm simulator. (d) Input multi-spectral image in RGB representation, and its 1st, 10th, and 33rd spectral-bands. (e) Decomposed multi-spectral image in RGB representation, and its 1st, 3rd, and 8th spectral-bands after 2-level 3D-QHT.

TABLE 1. Theoretical expectations and experimental results for 14-Qubit 3D-QHT using IBM-Q.

Decoherence time = 62.1 usec		Theoretical Expectations			Simulation Results on the 32-qubit IBM QASM simulator (8,000 shots)			Hardware Results on the 15-qubit IBM quantum processor ibmq_16_melbourne (8,000 shots)	
		Run-time (sec)	Fidelity	Circuit Coherence Ratio	Run-time (sec)	Fidelity		Run-time (sec)	Fidelity
		QHT	QHT	QHT	State initialization + QHT	QHT	State initialization + QHT	QHT	QHT
Sequential QHT	Unopt.	1.0494E-04	100.00%	0.59	5.1529E-02	99.96%	72.21%	4.8823E+01	35.50%
	Opt.	2.4683E-05	100.00%	2.52	4.8907E-02	99.98%	72.35%	4.6082E+01	42.65%
	Fidelity improvement	NA	1.0000	NA	NA	1.0002	1.0020	NA	1.2013
	Speedup	4.2515	NA	NA	1.0536	NA	NA	1.0595	NA
Parallel QHT	Unopt.	4.6869E-05	100.00%	1.32	4.9562E-02	99.96%	72.58%	4.8298E+01	36.48%
	Opt.	1.1200E-05	100.00%	5.54	4.5523E-02	99.98%	72.64%	4.5726E+01	55.38%
	Fidelity improvement	NA	1.0000	NA	NA	1.0002	1.0009	NA	1.5182
	Speedup	4.1848	NA	NA	1.0887	NA	NA	1.0563	NA
Fidelity improvement (Parallel Opt. vs. Sequential Unopt.)		NA	1.0000	NA	NA	1.0002	1.0060	NA	1.5600
Speedup (Parallel Opt. vs. Sequential Unopt.)		9.3695	NA	NA	1.1319	NA	NA	1.0677	NA

shows a $1024 \times 1344 \times 33$ multispectral image and its corresponding decomposed image bands after two-level ($l = 2$) 3D-QHT packet decomposition, respectively, where each dimension is reduced by a factor of $\frac{1}{2^l} = \frac{1}{4}$. The experiments were repeated using MATLAB models for sequential (three-stage) 3D-QHT, producing consistent results.

Experiments Using IBM-Q

In our experiments, we utilized a real quantum processor from IBM,[7] i.e., *ibmq_16_melbourne*, which contains 15 qubits. The qubits on *ibmq_16_melbourne* have, on average, an operating frequency of 4.98 GHz, T1 (*amplitude damping*) time of 58.28 μs, and T2 (*decoherence constant*) time of 62.1 μs.[7]

Theoretical Expectations and Metrics

We estimated theoretical run-times (see Table 1) for our proposed circuit variants using real gate times of the *ibmq_16_melbourne* machine. The theoretical run-times in Table 1 refer to expected run-times for the proposed QHT circuits. The relative improvement between unoptimized and optimized circuits serves as a reference point to which we can compare the measured experimental run-times. We measured the gate times for SWAP and *H* gates on the IBM-Q systems to be $\tau_{\text{SWAP}} = 2229.33$ ns and $\tau_H = 53.333$ ns, respectively, and calculated realistic run-times for each circuit using the time-delay expressions from (8) to (16). The proposed optimizations provide theoretical speedups of 4.2515 and 4.1848 fold for sequential and

parallel QHT, respectively (see Table 1). Comparing the optimized parallel with the unoptimized sequential circuit shows a 9.3695 speedup.

We defined the *circuit coherence ratio (CCR)* (17) as a metric to evaluate how coherent a given circuit is by comparing its execution time to the system decoherence time T2. A CCR greater than unity corresponds to a coherent circuit, while a CCR less than unity corresponds to a decoherent circuit. The CCR is calculated for each of the proposed circuit variants (see Table 1). The CCR for the unoptimized sequential circuit is less than unity, which indicates that the circuit violates the decoherence time constraint. The CCR for the optimized sequential circuit is greater than unity, which indicates that the circuit execution time is within the decoherence time constraint. Thus, the proposed optimizations are favorable for improving coherence of the sequential QHT circuits. The optimizations for parallel QHT also significantly improve the respective CCRs from 1.32 to 5.54 (see Table 1)

$$\mathrm{CCR} = \frac{\mathrm{T2}}{t_{\mathrm{total}}^{\text{run-time}}}. \tag{17}$$

We also verified the correctness and evaluated the accuracy of each circuit by measuring the state fidelity. The state fidelity is a measure of the similarity of the measured output state $|\psi_{\mathrm{measured}}\rangle$ observed in simulation or implementation, to the theoretical or expected state $|\psi_{\mathrm{expected}}\rangle$. The Uhlmann–Jozsa fidelity for pure states,[12] given in (18), is used for our experiments. By comparing fidelities among the circuit variants, we determined how effective the optimizations were in reducing circuit depth and improving coherence and state-fidelity

$$F = |\langle \psi_{\mathrm{expected}} | \psi_{\mathrm{measured}} \rangle|^2. \tag{18}$$

Simulation Results

The circuits were implemented using IBM-Qiskit-SDK[7] and simulated using IBM-Q qasm (*qasm_simulator*).[7] Test-data was a $64 \times 64 \times 3$ RGB image [see Figure 5(a)], requiring a total of $6 + 6 + 2 = 14$ qubits. To encode the image data for the QHT circuits, Qiskit provides an *initialize()* function that generates the corresponding state-initialization circuit[13] from the image data. State initialization is a necessary step and an unavoidable overhead for any quantum algorithm and is not part of the main algorithm, in this case QHT. The circuit output measurements were obtained on a 14-bit classical register using 8000 shots (samples) to minimize the statistical noise of measurements. Table 1 shows the simulation run-times and fidelities obtained for each of the implemented circuit variants. Run-times were measured for QHT circuits with qubit state initialization (using image data), which resulted in very deep circuits. The additional time delay (overhead) of the state-initialization circuit is much larger than the actual QHT circuit execution time. This results in lower speedups for simulation compared to theoretical speedups that only take into account the QHT circuit execution time.

The state fidelities were measured from 8000-shot simulations. For QHT circuits without state initialization, the fidelities were above 99%. However, the circuit fidelities decreased because of the additional circuit required for state initialization with the image data, which introduced more noise to the measured results. Moving from sequential (unoptimized) to parallel (optimized), as the circuits were optimized in terms of depth, the fidelity improved from 72.21% to 72.35% for sequential QHT, and from 72.58% to 72.64% for parallel QHT. Figure 5(c) shows the noisy-output decomposed image and its bands when reconstructed from IBM-Q simulations of one-level, 14-qubit parallel (optimized) 3D-QHT.

Hardware Results

Hardware implementations were also performed on IBM's *ibmq_16_melbourne* quantum processor, and the obtained run-times and fidelities are shown in Table 1. Qubit state initialization with image data could not be implemented, as the resulting circuits were too large and run-times exceeded the device repetition and readout rate. The hardware run-times are in the range of seconds, compared to the simulation run-times, which were in milliseconds. This is due to the unavoidable configuration overhead of the quantum processor, i.e., the time taken to generate control pulses for the quantum gates, which is much larger than the actual circuit execution time.

The fidelities measured from hardware executions are also shown in Table 1. Due to high sampling noise of the actual quantum hardware, the fidelities are lower than 55%. However, the fidelities improve as the circuits become optimized (see Table 1). For further improving the fidelities, quantum error correction is required before sampling the quantum circuit and forming the probability distribution data.

Given the current status of the technology/tools, it is not possible to isolate the different types of run-time overheads, i.e., state-initialization overhead and hardware setup/configuration overhead, in experimental studies. The simulation and hardware run-times could consequently be incomparable. However, both experiments are useful to evaluate the effect of

optimizations on relative run-times for each experiment. Therefore, in our results, we have included the analysis of theoretical, simulation, and hardware experiments.

CONCLUSION AND FUTURE WORK

In this work, we proposed optimizations for QHT circuits that reduced circuit depths and ensured quantum coherence at high fidelity. We presented generalized sequential and parallel circuits that can be used to process data with high dimensionality. We also showed circuit variants for packet and pyramidal decompositions of QHT and derived expressions for circuit time delays. We used realistic gate delays from a real quantum processor in the expressions to accurately estimate circuit run-times. The estimated run-times were used as reference to validate experimental simulation and hardware results that were collected from the quantum processor. The experimental results showed that our proposed optimizations are effective in mitigating circuit decoherence and improving circuit fidelity.

Future work will include characterizing the run-time overheads of a quantum processing system and investigating methods to reduce hardware inefficiencies. In addition, other useful wavelet-transforms, such as Daubechies wavelet and their application in QIP, will be investigated.

REFERENCES

1. M. Schlosshauer, "Quantum decoherence," *Phys. Rep.*, vol. 831, pp. 1–57, 2019.
2. E. El-Araby, T. El-Ghazawi, J. Le Moigne, and K. Gaj, "Wavelet spectral dimension reduction of hyperspectral imagery on a reconfigurable computer," in *Proc. IEEE Int. Conf. Field-Programmable Technol.*, 2004, pp. 399–402.
3. P. Hoyer, "Efficient quantum transforms," *Quantum Physics (quant-ph)*, Feb. 1997, *arXiv:quant-Ph/9702028*.
4. A. Fijany and C. P. Williams, "Quantum wavelet transforms: Fast algorithms and complete circuits," in *Proc. NASA Int. Conf. Quantum Comput. Quantum Commun.*, 1998, pp. 10–33.
5. H.-S. Li, P. Fan, H.-Y. Xia, S. Song, and X. He, "The multi-level and multi-dimensional quantum wavelet packet transforms," *Sci. Rep.*, vol. 8, no. 1, pp. 1–23, 2018.
6. N. Mahmud, B. Haase-Divine, A. MacGillivray, and E. El-Araby, "Quantum dimension reduction for pattern recognition in high-resolution spatio-spectral data," *IEEE Trans. Comput.*, vol. 71, no. 1, pp. 1–12, Jan. 2022.
7. IBM, "Get started with IBM quantum experience," Accessed: Feb. 25, 2021. [Online]. Available: https://quantum-computing.ibm.com/docs/
8. C. P. Williams, *Explorations in Quantum Computing*. New York, NY, USA: Springer Sci. Bus. Media, 2010.
9. C. Van Loan, *Computational Frameworks for the Fast Fourier Transform*. Philadelphia, PA, USA: SIAM, 1992.
10. H. Ohnishi, H. Matsueda, and L. Zheng, "Quantum wavelet transform and matrix factorization," in *Proc. IEEE Int. Quantum Electron. Conf.*, 2005, pp. 1327–1328.
11. S. M. Nascimento, K. Amano, and D. H. Foster, "Spatial distributions of local illumination color in natural scenes," *Vis. Res.*, vol. 120, pp. 39–44, 2016.
12. R. Jozsa, "Fidelity for mixed quantum states," *J. Modern Opt.*, vol. 41, no. 12, pp. 2315–2323, 1994.
13. V. V. Shende, S. S. Bullock, and I. L. Markov, "Synthesis of quantum-logic circuits," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 25, no. 6, pp. 1000–1010, Jun. 2006.

NAVEED MAHMUD is currently in his 5th year working towards the Ph.D. degree in electrical engineering with the Department of Electrical Engineering and Computer Science, University of Kansas (KU), Lawrence, KS, USA. He is also a Graduate Teaching Assistant and Graduate Research Assistant with KU, where he teaches at Digital Systems Design and Digital Logic Design Labs. Prior to joining KU, he was a design verification engineer with Ulka Semi, developing automated self-checking testbenches for ASICs. He has authored or coauthored seven journal articles and 13 conference papers. His research interests include quantum computing, emulation of quantum algorithms on reconfigurable hardware, and high-level synthesis techniques for FPGAs. He is a graduate student member of the IEEE. He is the corresponding author of this article. Contact him at naveed_923@ku.edu.

ANDREW MACGILLIVRAY is currently an undergraduate student in computer science with the Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA. He has acquired industry experience through his ongoing employment as a programmer in the aerospace sector, where he focuses on the development of business-administration tools. He has authored or coauthored five journal articles and one conference paper. His research focusses on the emulation of quantum algorithms using FPGAs. Contact him at amacgillivray@ku.edu.

MANU CHAUDHARY is currently a Ph.D. candidate and a Graduate Teaching Assistant with the Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA. He has authored or coauthored one journal article and two conference papers. His research focuses on quantum computing and emulation of quantum algorithms on FPGAs. He received the M.S. degree in electrical engineering from The University of North Carolina, Charlotte, NC, USA, in 2019. Contact him at manu.chaudhary@ku.edu.

ESAM EL-ARABY is currently an Assistant Professor with the Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, USA. He was with the High-Performance Computing Laboratory (HPCL), George Washington University (GWU), Washington, DC, USA, and the National Science Foundation (NSF) Center for High-Performance Reconfigurable Computing (NSF-CHREC), GWU. His research work was funded by organizations such as DoD, DARPA, NSF, and NASA. His research interests include computer architecture, reconfigurable computing, quantum computing, quantum communications, reversible computing, heterogeneous computing, biologically-inspired and neuromorphic architectures, and evolvable hardware. He received the M.Sc. and Ph.D. degrees in computer engineering from GWU, USA, in 2005, and 2010, respectively. He was the recipient and the principal investigator of a number of significant awards from the National Science Foundation (NSF) such as the most prestigious Award of Faculty Early Career Development Program (NSF-CAREER) and the Major Research Instrumentation (NSF-MRI) Award. He is a senior member of the IEEE. Contact him at esam@ku.edu.