2020 29th Wireless and Optical Communications Conference (WOCC)

Efficient Methods and Architectures for Mean and Variance Estimations of QAM Symbols

Guosen Yue and Xiao-Feng Qi
Radio Algorithms Research, Futurewei Technologies, Inc., New Jersey Research Center, Bridgewater, NJ 08807

***Abstract*— In this paper, we design efficient methods for the mean and variance estimations of QAM symbols with applications to iterative receivers. The proposed methods for optimal estimations enable scalable hardware implementations for any Gray mapped PAM or QAM with less circuitries. For variance estimations, the proposed method reduces the complexity from $\mathcal{O}\big((\log_2 N)^2\big)$ in the existing method to $\mathcal{O}\big(\log_2 N\big)$ for an N-QAM. Two suboptimal methods are also proposed to avoid the multiplications in the hardware implementations. The presented approximation approaches provide similar or better performance than the existing methods but with simpler implementation and less logical circuitries. In addition, based on the proposed architecture, we present novel unit module designs with disassembled estimation components and the schematics to virtualize the estimation hardware. With efficient design of unit module and control unit, maximized parallelization can be achieved.**

I. Introduction

The iterative receiver with soft interference cancellation (SIC) offers near optimal performance for joint demodulation and decoding. Such iterative receiver, or so-called turbo receiver, has been applied to the equalization [1], multiuser detection [2], and the multiple-input and multiple-output (MIMO) receivers [3]. At the base station side, the iterative receiver with SIC can be applied to the uplink single-carrier frequency division multiple access (SC-FDMA) [4], uplink multiuser MIMO [5], and the uplink coordinate multipoint (CoMP) reception with SIC (CoMP SIC) [6]. For 5G and future generation cellular systems, other than the mobile broadband (MBB) service, massive internet-of-things (IoT) and low latency communications are two important use cases where non-orthogonal multiple accesses for massive and fast connections become inevitable which incur strong interference among users [7]. Iterative SIC receiver is then a promising solution to combat the interference and significantly improve the system performance.

One of the key components for SIC is the estimation of the interfering quadrature amplitude modulation (QAM) symbols and residual interference after SIC, which is essentially the estimation of mean and variance of QAM given the log-likelihood ratio (LLR) of the mapped coded bits from the decoding outputs. Therefore, to reduce the complexity of the SIC receivers, efficient approaches for the QAM mean and variance estimations are critical. This is particularly important for high order QAM's, e.g., 256-QAM in LTE and 5G cellular systems and 4096-QAM in microwave transmissions. For example in [8] an iterative receiver is performed to estimate the phase noise in microwave backhaul receiver where the high-order QAM is adopted. Some approaches have been introduced in [9], [10] for a Gray mapped QAM symbol. The computational complexity can be reduced to the order of the number of the bits in a QAM symbol for the mean estimation and square of that for the variance estimation. Further complexity reduction is achieved in [10] via a suboptimal approximation by avoiding the multiplications.

In this paper, we first propose efficient methods and implementation architectures for optimal estimations that facilitate less circuitries in the hardware implementation. For the variance estimation, the proposed method not only enables efficient architecture but further reduce the complexity from $\mathcal{O}\big((\log_2 N)^2\big)$ in the existing method to $\mathcal{O}\big(\log_2 N\big)$ for an N-QAM. We also propose two alternative suboptimal methods that avoid the multiplications in the product of $\tanh$ outputs. The proposed methods provide further simplifications on the hardware implementation than the approach in [10]. The proposed recursive and scalable architectures make the implementations of adaptive modulations more efficient. Moreover, based on the proposed architectures, we introduce novel unit module designs with disassembled estimation components and the schematics to virtualize the estimation hardware. With efficient design of unit modules and control units, maximized parallelization can be achieved.

II. System Descriptions

A. Signal Model

We consider the same system setting as in [10] as an example, i.e., the iterative SIC receiver with minimum mean square error (MMSE) filtering for a MIMO system with M_T transmit and M_R receive antennas, where $M_R \geq M_T$. Denote $\boldsymbol{x} = [x_1, \cdots, x_{M_T}]^T$ as the QAM symbol vector. The received signal is given by

$$\boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{n} = \sum_{i=1}^{M_T} \boldsymbol{h}_i x_i + \boldsymbol{n}, \tag{1}$$

where $\boldsymbol{H} = [\boldsymbol{h}_1, \cdots, \boldsymbol{h}_{M_T}]$ is the $M_R \times M_T$ complex channel gain matrix and $\boldsymbol{h}_i$ is the channel vector from the ith transmit antenna to the receiver antenna array, and $\boldsymbol{n}$ denotes the i.i.d. zero-mean complex white Gaussian noise vector with unit variance for each entry, i.e., $\boldsymbol{n} \sim \mathcal{C}_{\mathcal{N}}(\boldsymbol{0}, \boldsymbol{I})$ and $\boldsymbol{I}$ is $M_R \times M_R$ identity matrix. We consider the QAM symbols from all transmit antennas are jointly coded with one channel code. We also consider the block fading where $\boldsymbol{H}$ remains constant for the entire code block.

978-1-7281-6124-2/20/$31.00 ©2020 IEEE

B. Iterative MMSE-SIC Receiver

Given the extrinsic LLRs of the coded bits from the soft channel decoder in the previous iteration, we first obtain the soft (mean) estimation of the QAM symbol $\tilde{x}_i = E\{x_i\}$, $i = 1, \cdots, M_T$. After SIC, the residual signal is then given by

$$\tilde{\boldsymbol{y}}_i = \boldsymbol{h}_i x_i + \sum_{j \neq i} \boldsymbol{h}_j (x_j - \tilde{x}_j) + \boldsymbol{n}. \tag{2}$$

With linear MMSE filtering after SIC, we obtain

$$\breve{x}_i = \tilde{\boldsymbol{w}}_i^{\dagger} \tilde{\boldsymbol{y}}_i, \tag{3}$$

where the linear MMSE filter $\tilde{\boldsymbol{w}}_i$ is given by

$$\tilde{\boldsymbol{w}}_i = (\boldsymbol{h}_i \boldsymbol{h}_i^{\dagger} + \sum_{j \neq i} \tilde{\sigma}_j^2 \boldsymbol{h}_j \boldsymbol{h}_j^{\dagger} + \boldsymbol{I})^{-1} \boldsymbol{h}_i, \tag{4}$$

and † denotes matrix Hermitian, $\sum_{j \neq i} \tilde{\sigma}_j^2 \boldsymbol{h}_j \boldsymbol{h}_j^{\dagger}$ is covariance of the residual interference after SIC, and

$$\tilde{\sigma}_j^2 = \text{var}\{x_j - \tilde{x}_j\} = E\{|x_j|^2\} - |\tilde{x}_j|^2. \tag{5}$$

Assuming that $\breve{x}_i$ is Gaussian distributed, we then obtain the LLR for the each bit in the QAM symbol x_i and send the extrinsic information to the soft channel decoder. The output extrinsic LLRs from the soft decoder are then fed back as the prior LLRs for the next iteration of MMSE-SIC. Initially, we set the soft estimate $\tilde{x}_i = 0$. The details of iterative MMSE-SIC receiver can be found in [11].

It is seen that in the SIC iterative receiver, it is essential to obtain the soft QAM estimation $\tilde{x}_i$ for SIC and the variance of the QAM symbol $\tilde{\sigma}_i^2$ for the estimation of the residual interference. Also from (5), when the soft estimation $\tilde{x}_i$ is obtained, the problem left is estimating the second moment of the QAM symbol, $E\{|x_j|^2\}$.

III. Soft Mean and Variance Estimations

A. Definitions

We consider an N-QAM constellation set $\mathcal{S}_{\text{QAM}} = \{s_1, \cdots, s_N\}$, with each symbol mapped from a length-J binary sequence, $b_1, \cdots, b_J$, where $J = \log_2 N$ and $b_i \in \{0, 1\}$. We first assume that the QAM symbols are integer values on $\mathcal{I}$ and $\mathcal{Q}$ components, i.e., $2z + 1$, $z = 0, \pm 1, \cdots$. Then we normalize the QAM signal for a unit average power with the scaling factor of $\frac{1}{\sqrt{E_N}}$, where E_N is the variance of the QAM inputs. Assuming equiprobable inputs, we then have $E_N = \frac{1}{N} \sum_{n=1}^{N} |s_n|^2 = \frac{2}{3}(N-1)$.

1) Mean Estimation: Given the LLR λ_i for b_i, $i = 1, \cdots, J$, mapped to the QAM symbol $x_{\text{qam}} \in \mathcal{S}_{\text{QAM}}$, we then form the soft symbol or the mean estimation of x_{qam} as

$$\tilde{x}_{\text{qam}} = \sum_{s \in \mathcal{S}_{\text{QAM}}} s \Pr(x_{\text{qam}} = s) = \sum_{n=1}^{N} s_n \prod_{i=1}^{J} P_{b_i^{(s_{n,i})}}, \tag{6}$$

where $s_{n,i}$ denotes the ith bit in the length-J bit sequence that is mapped to the QAM symbol s_n, $P_{b_i^{(\imath)}} \triangleq \Pr(b_i = \imath)$, $\imath \in \{0, 1\}$, and $P_{b_i^{(0)}} = (e^{-\lambda_i} + 1)^{-1}$, $P_{b_i^{(1)}} = (e^{\lambda_i} + 1)^{-1}$. We can see that the overall complexity from (6) is $\mathcal{O}(N \log N)$.

For squared QAM with orthogonal mapping of $\mathcal{I}$ and $\mathcal{Q}$ components, the QAM symbol estimation can be decoupled to two pulse-amplitude modulation (PAM) estimations, given as

$$\tilde{x}_{\text{pam}} = \sum_{s \in \mathcal{S}_{\text{PAM}}} s P_{x^{(s)}} = \sum_{n=1}^{N'} s_n \prod_{i=1}^{Q} P_{b_i^{(s_{n,i})}}, \tag{7}$$

where $N' = \sqrt{N}$, $Q = \frac{J}{2}$, and $s_{n,i}$ is now the ith bit mapped to the PAM symbol s_n. We can see that the PAM estimation in (7) requires QN' multiplications and $N' - 1$ additions. Thus overall the squared QAM estimation needs $2Q\sqrt{N}$ multiplications and $2\sqrt{N} - 2$ additions. The complexity order is then $\mathcal{O}(\sqrt{N} \log N)$, which is lower than the original approach.

2) Variance Estimation: As described in Section II, after obtaining the soft symbol estimate, the other term in the variance estimation of the residual interference after SIC is the second moment estimation. The general definition of the second moment estimation is given as

$$\begin{aligned} \tilde{\nu}_{\text{qam}}^2 &\triangleq E\{|x_{\text{qam}}|^2\} = \sum_{s \in \mathcal{S}_{\text{QAM}}} |s|^2 \Pr(x_{\text{qam}} = s) \\ &= \tilde{\nu}_{\mathcal{I}}^2 + \tilde{\nu}_{\mathcal{Q}}^2, \end{aligned} \tag{8}$$

where $\tilde{\nu}_{\mathcal{I}}^2 \triangleq \sum_{s \in \mathcal{S}_{\text{QAM}}} s_{\mathcal{I}}^2 \Pr(x_{\text{qam}} = s)$, $\tilde{\nu}_{\mathcal{Q}}^2 \triangleq \sum_{s \in \mathcal{S}_{\text{QAM}}} s_{\mathcal{Q}}^2 \Pr(x_{\text{qam}} = s)$, $s_{\mathcal{I}}$ and $s_{\mathcal{Q}}$ denote the $\mathcal{I}$ and $\mathcal{Q}$ values of the QAM symbol s, respectively. Obviously, the complexity of the estimation using above expression is $\mathcal{O}(N \log N)$.

Similarly as in mean estimation, we consider the squared QAM which can be decoupled to two orthogonal PAMs. Assuming an N'-PAM constellation set $\mathcal{S}_{\text{PAM}} = \{s_1, \cdots, s_{N'}\}$ with $N' = 2^Q$, we then have the second moment estimation for PAM symbols given as

$$\tilde{\nu}_{pam}^2 \triangleq \sum_{n=1}^{2^Q} s_n^2 \Pr(x = s_n) = \sum_{n=1}^{2^Q} s_n^2 \prod_{i=1}^{Q} P_{b_i^{(s_{n,i})}}. \tag{9}$$

The above equation can be applied to obtain $\tilde{\nu}_{\mathcal{I}}^2$ and $\tilde{\nu}_{\mathcal{Q}}^2$ in (8) separately. The complexity is then reduced to $\mathcal{O}(\sqrt{N} \log N)$

B. Existing Approaches For Gray Mapped QAM Symbols

The expressions for the efficient mean and variance estimations are obtained for both squared QAM and non-squared QAM in [9]. For squared QAM, the soft mean and variance (or specifically the second moment) estimations can be decoupled into two PAM estimations. Also we consider the binary reflect Gray code (BRGC) based mapping which is commonly used in current wireless systems.

1) Mean Estimation: For a 2^Q-PAM symbol with Gray mapping shown in Fig. 1, the soft mean estimation is given by

$$\tilde{x} = -\sum_{i=1}^{Q} 2^{Q-i} \prod_{j=1}^{i} \tanh\big(-\frac{\lambda_j}{2}\big). \tag{10}$$

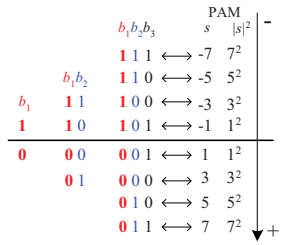

Fig. 1. BRGC mapped PAM constellations before power normalization. Left to right: 2-PAM(BPSK), 4-PAM, and 8-PAM.

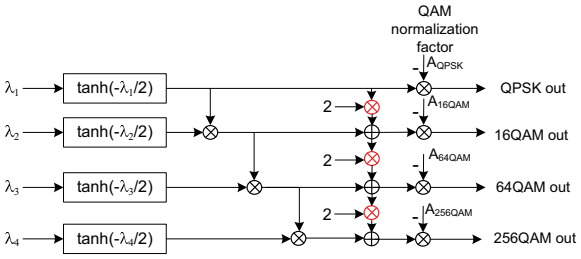

Fig. 2. The implementation schematics for the proposed procedures for various PAM (corresponding to $\mathcal{I}$ or $\mathcal{Q}$ component of a squared QAM) mean estimation.

Then a scaling factor $A_{N-QAM} = \frac{1}{\sqrt{E_N}}$ is applied to the result in (10) for both $\mathcal{I}$ and $\mathcal{Q}$ components for QAM normalization to obtain the final results. Based on (10), an algorithm for soft PAM estimation was then formed as [9]

- Initially set $\eta = 0$ and $\xi = 1$.
- For $i = 1 \cdots, Q$, update ξ and η sequentially as
$$\xi \Leftarrow \xi \cdot \tanh\left(-\tfrac{\lambda_i}{2}\right), \quad \text{then} \quad \eta \Leftarrow \eta + 2^{Q-i} \cdot \xi.$$
- The PAM soft estimate is then obtained as $\tilde{x} = -\eta$ using the last update of η.

The hyperbolic tangent function $tanh$ can be implemented by a look-up table (LUT). The multiplications with 2^i can be implemented with bit shifting. Therefore, the overall complexity for above soft PAM estimation is very low and in the order of $\log_2 N$.

2) Variance Estimation: As aforementioned, for variance estimation, we need to calculate the second moment estimation $\tilde{\nu}^2$ for a PAM symbol. An efficient expression of the second moment estimation derived in [10] for a 2^Q PAM is given as

$$\tilde{\nu}^2 = C_Q + \sum_{i=2}^{Q} 4^{Q-i+1} \sum_{j=2}^{i} 2^{i-j} \prod_{k=j}^{i} \tanh\left(-\frac{\lambda_k}{2}\right), \quad (11)$$

where C_Q can be obtained iteratively by

$$C_q = 4C_{q-1} + 1, \quad q = 1, \cdots, Q, \text{ with } C_0 = 0. \quad (12)$$

For N-QAM normalization, a scaling factor of $\frac{1}{E_N}$ is applied to $\tilde{\nu}^2$ in (11). The algorithm with computation order $\mathcal{O}\big((\log_2 N)^2\big)$ is formed in [10] and recapitulated as follows.

- Set $\eta = 0$. Obtain $\tanh(-\frac{\lambda_i}{2})$, $i = 2, \cdots, Q$, by LUT.
- For $i = Q, Q-1, \cdots, 2$,
  - Set $\zeta = 0$ and $\xi = 1$.
  - For $j = i, \cdots, 2$, update ξ and ζ sequentially by
$$\xi \Leftarrow \xi \cdot \tanh\left(-\tfrac{\lambda_j}{2}\right), \quad \text{then } \zeta \Leftarrow \zeta + 2^{i-j}\xi.$$
  - Update η by $\eta \Leftarrow \eta + 4^{Q-i+1}\zeta$.
- Obtain $\tilde{\nu}^2 = \eta + C_Q$ using the latest update of η.

C. Proposed Efficient and Hardware-Friendly Approaches

1) Mean Estimation: Instead of the procedures described in Sec. III-B.1, we make a change on the update of η as follows

- Initially set $\eta = 0$ and $\xi = 1$.
- For $i = 1 \cdots, Q$, update ξ and η sequentially as
$$\xi \Leftarrow \xi \cdot \tanh\left(-\tfrac{\lambda_i}{2}\right), \quad \text{then} \quad \eta \Leftarrow 2\eta + \xi.$$
- The PAM soft estimate is then obtained as $\tilde{x} = -\eta$ using the last update of η.

As we can see that with the above change, the update of η does not have the modulation dependent parameter 2^{Q-i}. Such change makes the circuit implementation for one iteration reusable for different modulations or makes the implementation scalable for higher order QAM modulations.

The implementation for QAM mean estimation according to the new method is illustrated in Fig. 2 where the scaling factor $A_N = \frac{1}{\sqrt{E_N}}$ for an N-QAM is applied in the end for the normalization. It is seen from Fig. 2 that for the mean estimations of PAM symbols, corresponding to the $\mathcal{I}$ or $\mathcal{Q}$ component of a squared QAM, one circuitry implementation can be applied to any Gray mapped PAM or square QAM without any parameter change. This is desirable for wireless communication systems as adaptive modulation is commonly employed to improve the throughput efficiency for channel fluctuations. The proposed implementation is scalable to any higher order QAM formats, which facilitates the implementation and verification procedures for the application-specific integrated circuit (ASIC) when new higher order PAM or QAM modulation is introduced to the system specification.

2) Variance Estimation: After carefully examining the expression in (11), we now propose a new algorithm for the second moment estimation as follows.

- Obtain $\tanh(-\frac{\lambda_i}{2})$, $i = 2, \cdots, Q$, by LUT. Initially set $\eta = 0$, $\zeta = \tanh\left(-\frac{\lambda_2}{2}\right)$.
- For $i = 3, \cdots, Q$,
  - Update ζ as
$$\zeta \Leftarrow (2\zeta + 1) \cdot \tanh\left(-\tfrac{\lambda_i}{2}\right).$$
  - Update η by
$$\eta \Leftarrow 4\eta + \zeta.$$
- Obtain $\tilde{\nu}^2 = 4\eta + C_Q$ using the latest update of η.

We can see that the inner iteration from the existing method has been eliminated in the new method. Therefore, the new method further reduces the computation complexity to the order $\mathcal{O}\big(\log N\big)$ from $\mathcal{O}\big((\log N)^2\big)$ for the method in [10]. The additional complexity introduced is the addition $+1$ when updating ζ in each iteration. The implementation based on the proposed method is illustrated in Fig. 3. From Fig. 3, the scalability of the new method for different QAM modulations becomes obvious. With another iteration of updating ζ and η with the input of λ_{Q+1}, we can obtain the second moment estimation for 2^{Q+1}-PAM from the results of 2^Q-PAM.

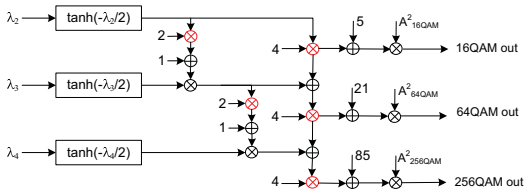

Fig. 3. The implementation schematics for the proposed procedures for various PAM (corresponding to $\mathcal{I}$ or $\mathcal{Q}$ component of a squared QAM) second moment estimation.

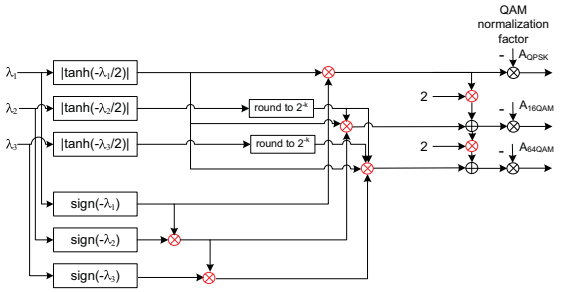

Fig. 4. The implementation schematics for 8-PAM estimation based on (10). The multiplication (cross symbol) in red means that the operation can be efficiently implemented, e.g., with bit shift operations.

IV. Suboptimal Approaches for Hardware Implementations

We can see that in (10) and (11), the computation complexity is mainly the product of $tanh$ functions. To further reduce the complexity, we now consider the suboptimal approach with approximations to avoid these multiplications.

A. Existing Approximation Approach

The basic idea is to find an approximation for the product of $\tanh$ functions, e.g., $\Theta_{j_1,j_2} \triangleq \prod_{i=j_1}^{j_2} \tanh(\lambda_i/2)$. A method is provided in [10] briefly recapitulated as follows.

- Let $\phi_i = \tanh(-\lambda_i/2)$ and $\psi_i = |\phi_i|$. Obtain the maximum $\psi_{i^*} = \max_{i=j_1,\cdots,j_2} |\phi_i|$.
- For other i, since $|\phi_i| \leq 1$, it is represented or quantized in the fixed point representation by $\sum_k a_{i,k} 2^{-k}$, $a_{i,k} \in \{0,1\}$. Denote k_i' as the smallest index where $a_{i,k'} = 1$. Then for any $|\phi_i|$, $i \neq i^*$ in the production, we round it to the nearest power of 2, i.e., $|\phi_i| \approx 2^{-(k_i' - a_{i,k_i'+1})}$. Then mathematically, we approximate Θ_{j_1,j_2} as

$$\Theta_{j_1,j_2} \approx \psi_{i^*} 2^{-\sum_{i=j_1,\cdots,j_2; i\neq i^*} (k_i' - a_{i,k_i'+1})} \prod_{i=j_1}^{j_2} \text{sgn}(\phi_i). \quad (13)$$

With this method, the production Θ_{j_1,j_2} can be approximated as the bit-wise shift of the largest $|\phi_i|$. However, finding the maximum ψ_{i^*} does not facilitate efficient and scalable architecture as presented in Sec. III-C. We now consider some new designs.

B. Proposed Suboptimal Approaches

The existing method does not consider the different reliabilities among the bits in a QAM or a PAM symbol. As shown in Fig. 1, it is known that among the bits mapped to a PAM symbol, the most significant bit b_1 is the most reliable after demodulation due to the largest average Euclidean distance between the PAM symbols with the mapped bit $b_1 = 0$ and that with $b_1 = 1$, while the least significant bit, b_Q, has least reliability. Considering the bit reliability, we now propose two approximation methods as follows.

1) Method I: Given the decreasing order of reliability for the bit sequence $b_1, b_2, \cdots, b_Q$, we first propose the approximation of the production Θ_{j_1,j_2} by keeping $|\phi_i|$ value for the bit with the highest reliability, i.e., $\psi_{i^*} = |\phi_{j_1}|$ and approximating other $|\phi_i|$ to the closet power of 2. We have

$$\Theta_{j_1,j_2} \approx |\phi_{j_1}| 2^{-\sum_{i=j_1+1,\cdots,j_2} (k_i' - a_{i,k_i'+1})} \prod_{i=j_1}^{j_2} \text{sgn}(\phi_i). \quad (14)$$

Since the selection of ψ_{i^*} is deterministic, the implementation of mean estimation can be also scalable to any QAM modulations, as shown in Fig. 4.

Saturation Protection:

Since $\tanh(x)$ is in the range of $[-1, +1]$, when $|\lambda_i|$ is large, the value of $\tanh(-\lambda_i/2)$ is easily saturated, particularly for the higher reliable labeling bit. It is then inefficient to retain the saturated value and approximate other $tanh$ outputs. To overcome this inefficiency, we then add a protection to the implementation as follows. First set a threshold δ_{th} which can be one or close to one. Then search from $i = j_1, ..., j_2$ and find the first $\psi_i = |\tanh(-\lambda_i/2)| \leq \delta_{th}$ as ψ_{i*}. For other $i \neq i^*$, round ψ_i to the closest power of 2, i.e., $2^{-(k_i' - a_{i,k_i'+1})}$. For fixed point implementation with quantized LLR, the threshold δ_{th} can be set to 1.

2) Method II: The second method is based on the thought that the least significant bit creates the most residual interference in the last several iterations. So instead of retaining the value of ψ_i for the most reliable bit, we keep the one for the least reliable bit, i.e., least significant bit in the bit sequence in a production $tanh$. Therefore, we have $\psi_{i^*} = |\phi_{j_2}|$ and round other ψ_i to the closest 2^{-k}. We then have

$$\Theta_{j_1,j_2} \approx \psi_{j_2} 2^{-\sum_{i=j_1,\cdots,j_2-1} (k_i' - a_{i,k_i'+1})} \prod_{i=j_1}^{j_2} \text{sgn}(\phi_i). \quad (15)$$

The resulting implementation is illustrated in Fig. 5. Again, the selection of a $tanh$ term in a production to keep or for approximation is deterministic. The implementation is scalable for any QAM formats.

V. Virtualization for Maximum Parallelization

As seen from Fig. 2 and Fig. 3, one circuit architecture can be used to any modulations. However, when it is used for a low order QAM estimations, some branches of circuits are not utilized. To make the hardware most efficient, we next propose the hardware virtualization for the estimators so that maximum parallelization can be achieved.

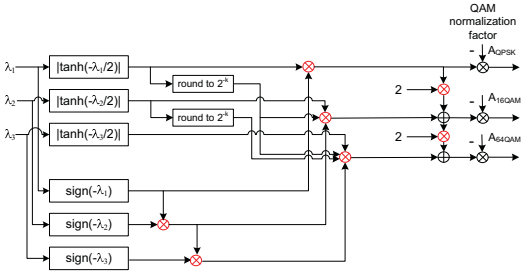

Fig. 5. The implementation schematics for 8-PAM estimation based on (10). The multiplication (cross symbol) in red means that the operation can be efficiently implemented, e.g., with bit shift operations.

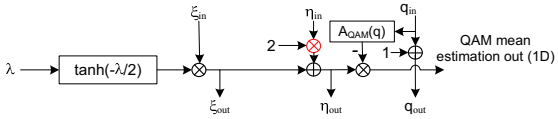

Fig. 6. Unit module $\mathcal{M}$ of mean estimation.

A. Disassembled Unit Modules for Virtualization

1) Mean Estimation: ***Unit Module and Virtualization***

Based on the scalable architecture in Fig. 2, we now present a unit module for QAM mean estimation as shown in Fig. 6. The module has λ, ξ_{in}, η_{in}, and q_{in} as the inputs. The outputs are ξ_{out}, η_{out}, q_{out}, and the estimation output $\tilde{x}_{out}$. From Fig. 6, the follwing functions are implemented in the module

$$\begin{aligned} \xi_{out} &= \xi_{in} \cdot \tanh\big(-\lambda/2\big), \\ \eta_{out} &= 2\eta_{in} + \xi_{out}, \\ q_{out} &= q_{in} + 1, \\ \tilde{x}_{out} &= A_{\mathrm{QAM}}(q_{in}) \cdot \eta_{out}, \end{aligned} \tag{16}$$

where $A_{\mathrm{QAM}}(q)$ is the QAM normalization factor given by $A_{\mathrm{QAM}}(q) = \frac{1}{\sqrt{\frac{2}{3}(2^{2q}-1)}}$ for the N-QAM ($N = 2^{2q}$). The value of $A_{\mathrm{QAM}}(q)$ can be pre-stored in the circuit. We can see from Fig. 6 and (16) that q_{in}/q_{out} not only serves as a counter for processing times with the module but also as the index of the normalization factor $A_{\mathrm{QAM}}(q)$.

With unit module $\mathcal{M}$ shown in Fig. 6, we can implement the estimation for PAM, $\mathcal{I}$ or $\mathcal{Q}$ of QAM, via either serial processing with one module as in Fig. 7(a) or via parallel processing with multiple modules as in Fig. 7(b). The initial values of ξ_{in}, η_{in}, and q_{in} are set to be 1, 0, and 1, respectively. In serial processing, with a clock control, the LLR's are sent to the module sequentially. The outputs ξ_{out}, η_{out}, and q_{out} generated in each time unit are sent back as the inputs ξ_{in}, η_{in}, and q_{in}, respectively. After Q times, the output x is generated for a PAM estimation. The inputs are reset to initialization values every Q processing times. In parallel processing, each LLR is dispatched to one module. The outputs from one module are sent to another module as the inputs. The estimation of a PAM is output from the Qth module.

2) Second Moment Estimation:: ***Unit Module and Virtualization***

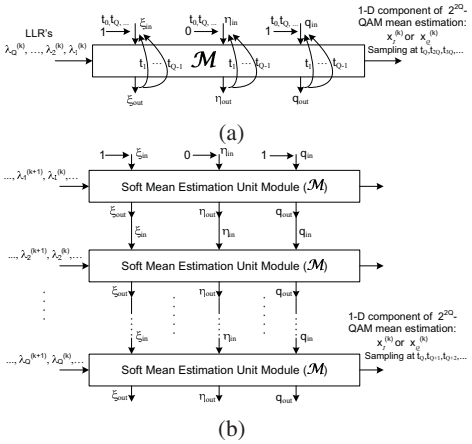

Fig. 7. Mean estimation with virtualized unit module: (a) sequential processing; (b) parallel processing..

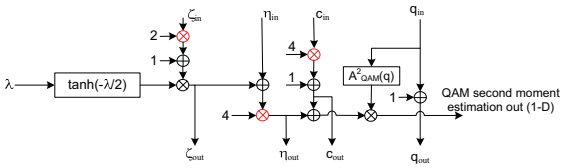

Fig. 8. Unit module $\mathcal{V}$ for second moment estimation.

The unit module architecture for QAM second moment estimation is shown in Fig. 8. As shown in Fig. 8, we have the λ, ξ_{in}, η_{in}, c_{in}, and q_{in} as the inputs. The outputs are ξ_{out}, η_{out}, c_{out}, q_{out}, and the estimation output $\tilde{x}_{out}$. The functions implemented in the module are given by

$$\begin{aligned} \zeta_{out} &= (2\zeta_{in} + 1) \cdot \tanh\big(-\lambda/2\big), \\ \eta_{out} &= 4(\eta_{in} + \xi_{out}), \\ c_{out} &= 4c_{in} + 1 \\ q_{out} &= q_{in} + 1, \\ \tilde{x}_{out} &= A^2_{\mathrm{QAM}}(q_{in}) \cdot \big(\eta_{out} + c_{out}\big). \end{aligned} \tag{17}$$

Similarly, with the module $\mathcal{V}$ processing functions in (17), we can obtain the second moment estimation via either serial processing with one module or parallel processing with multiple modules. The initial values of ξ_{in}, η_{in}, c_{in}, and q_{in} are set to 0, 0, 0, and 1, respectively.

B. Virtualization of Unit Modules and Estimation Functions

We now present a novel hardware function virtualization for QAM estimations. As shown in Fig. 9, we form a pool of unit modules $\mathcal{M}$ and $\mathcal{V}$. The virtualized function block includes a control unit that operates the modules for connecting the LLR inputs, connecting the modules (if parallel processing is performed), sampling the output at the appropriate location and the appropriate clock time for a certain QAM format. The maximum capacity (processing estimates per unit processing time) can be easily obtained and is known to the outer task scheduler. The outer scheduler then just sends one or multiple data streams at a certain rate. For example, with

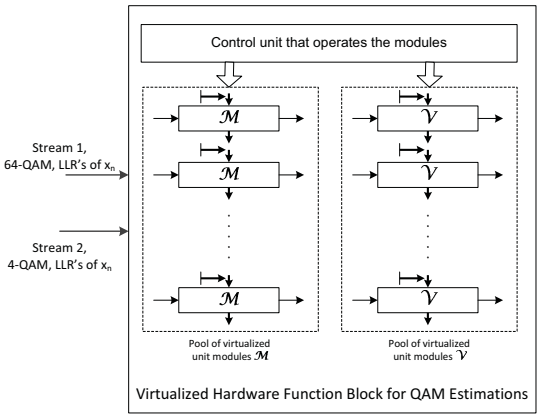

Fig. 9. Hardware virtualization with pools of virtualized unit modules.

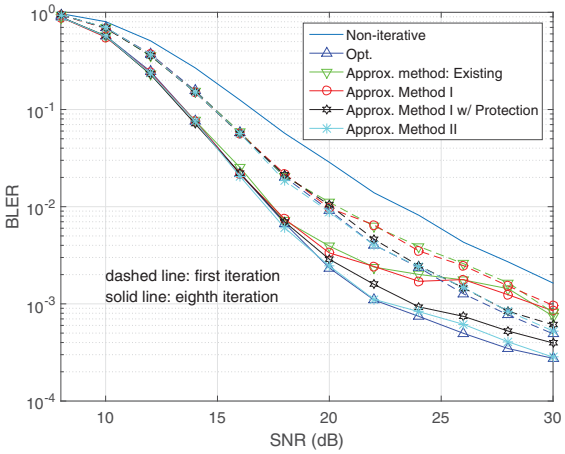

Fig. 10. BLER performance of iterative MMSE-SIC receiver with suboptimal approximation methods for soft symbol and variance estimations of 16-QAM in a 4×4 MIMO.

two data streams, i.e., one 4-QAM (QPSK) and one 64-QAM, maximum sets of 8 $\mathcal{M}$'s and 4 $\mathcal{V}$'s are allocated for parallel processing of mean and second moment estimations if they are available. If not, serial processing is scheduled with less modules. Then minimizing processing delay is considered in the control and scheduling unit. Therefore, with virtualized unit modules and pooling, maximum parallelism can be achieved with minimum processing delay, which is more efficient for massive IoT and massive non-orthogonal multi-access for 5G and beyond.

VI. Numerical Results

We now evaluate the performance of iterative MMSE-SIC receiver for 4×4 MIMO block fading channels with proposed suboptimal approximation approaches described in Section IV. The information block length is 1024. The fading channels are i.i.d. The results of the block error rate (BLER) performance as a function of SNR for 16-QAM are presented in Fig. 10. For comparisons, Fig. 10 also includes the performance of non-iterative MMSE receiver and the approximation method in [10]. A rate-0.5 turbo code in 3G is employed. For 16-QAM, we only have the approximation on the mean estimation. As we can see, the proposed Method I has the similar performance as the existing method but with less logics in the implementation. Method I with saturation protection and proposed method II perform better in the region where BLER is below 10^{-2}. Both perform close to the optimal solution without any approximation. Comparing these two approaches, Method II shows slightly better performance than Method I with protection in the low BLER region.

VII. Conclusion

In this paper, we have proposed enhanced methods and scalable architectures for mean and variance estimations of QAM symbols. The proposed approach for variance estimation reduces the complexity order from $\mathcal{O}\big((\log_2 N)^2\big)$ in the existing method to $\mathcal{O}\big(\log_2 N\big)$. We have also proposed two suboptimal approximation approaches which provide similar or better performance than the existing method but with simpler and scalable implementations. Based on the proposed architectures, we have presented novel unit module designs and the schematics to virtualize the estimation hardware so that maximized parallelization can be achieved with such hardware virtualization.

References

[1] M. Tüchler, R. Koetter, and A. C. Singer, "Turbo equalization: Principles and new results," *IEEE Trans. Commun.*, vol. 50, no. 5, pp. 754–767, May 2002.

[2] X. Wang and H. V. Poor, "Iterative (turbo) soft interference cancellation and decoding for coded CDMA," *IEEE Trans. Commun.*, vol. 47, no. 7, pp. 1046–1061, July 1999.

[3] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple-antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[4] T. Li, W. Wang, and X. Gao, "Turbo equalization for LTE uplink under imperfect channel estimation," in *Proc. Personal, Indoor, and Mobile Radio Commun. (PIMRC)*, Tokyo, Japan, Sept. 2009.

[5] M. Jiang, G. Yue, N. Prasad, and S. Rangarajan, "Link adaptation in LTE-A uplink with turbo SIC receivers and imperfect channel estimation," in *Proc. Conf. Info. Sci. Syst. (CISS)*, Baltimore, MD, Mar. 2011.

[6] T. Mayer, H. Jenkac, and J. Hagenauer, "turbo base-station cooperation for intercell interference cancellation," in *Proc. of Int. Conf. Commun. (ICC)*, Istanbul, Turkey, June 2006, pp. 4977–4982.

[7] L. Dai, B. Wang, Y. Yuan, S. Han, C. I, and Z. Wang, "Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 74–81, Sept. 2015.

[8] N. Kamiya and E. Sasaki, "Pilot-symbol-assisted phase noise compensation with forward-backward wiener smoothing filters," *IEEE Trans. Signal Processing*, vol. 65, no. 17, pp. 4443–4453, Sept. 2017.

[9] G. Yue and S. Rangarajan, "Low-complexity methods for soft estimation of QAM symbols," in *Proc. Conf. Info. Sci. Syst. (CISS)*, Baltimore, MD, Mar. 2013.

[10] G. Yue, N. Prasad, and S. Rangarajan, "Iterative MMSE-SIC receiver with low-complexity soft symbol and residual interference estimations," in *Proc. 39th Asilomar Conf. Signals, Systems and Computers*, Pacific Grove, CA, Oct. 2013.

[11] G. Yue and X. Wang, "Optimization of irregular repeat accumulate codes for MIMO systems with iterative receivers," *IEEE Trans. Wireless Commun.*, vol. 4, no. 6, pp. 2843–2855, Nov. 2005.