A Novel Technique for Efficient Hardware Simulation of Spatiotemporally Correlated MIMO Fading Channels

Amirhossein Alimohammad, Saeed Fouladi Fard, Bruce F. Cockburn, Christian Schlegel
Department of Electrical and Computer Engineering, University of Alberta
Edmonton, Alberta, T6G 2V4, Canada
{amir,saeed,cockburn,schlegel}@ece.ualberta.ca

***Abstract*— We present a fading model with a compact and fast hardware implementation suitable for correlated Rayleigh fading channel simulators. The proposed scheme is based on the sum-of-sinusoids model because of its flexibility and efficient mapping onto hardware. Using numerical simulation, it is shown that the statistical properties of the generated fading variates match the theoretical reference model. Since the cross-correlations between sequences of generated fading variates are small, this model can also be used to implement a time-correlated multiple-input multiple-output (MIMO) fading channel simulator on a single field-programmable gate array (FPGA). The MIMO channel simulator can also be extended to support spatial correlation between generated fading samples. An implementation of a spatiotemporally correlated $(4, 4)$ MIMO channel simulator on a Xilinx Virtex-II Pro XC2VP100-6 FPGA uses 46% of the configurable slices, 30% of the dedicated multipliers, and 32% of the on-chip block memories while generating 4×201 million 2×16-bit complex-valued fading samples per second.**

I. Introduction

A fading channel simulator is a key component for the repeatable performance evaluation of communication systems under realistic propagation conditions. A fading channel is commonly modeled using a complex Gaussian wide-sense stationary uncorrelated scattering (WSSUS) process $c(t) = c_i(t) + jc_q(t)$ [1], where the envelope $|c(t)|$ follows the Rayleigh distribution. In a two-dimensional isotropic scattering environment with an omnidirectional receiving antenna at the receiver, the autocorrelation function (ACF) associated with either $c_i(t)$ or $c_q(t)$ is given by $\mathbf{R}_{c_i,c_i}(\tau) = \mathbf{R}_{c_q,c_q}(\tau) = \mathcal{J}_0(2\pi f_D \tau)$ [2] where f_D is the maximum Doppler frequency and $\mathcal{J}_0(\cdot)$ is the zeroth-order Bessel function of the first kind [3]. The cross-correlation function (CCF) between $c_i(t)$ and $c_q(t)$ is $\mathrm{R}_{c_i,c_q}(\tau) = \mathrm{R}_{c_q,c_i}(\tau) = 0$. The power spectral density functions (PSDs) associated with the $c_i(t)$ or $c_q(t)$ is the so-called Jakes PSD [4], which can be written as $\mathbf{S}_{c_i}(f) = \mathbf{S}_{c_q}(f) = \left(4\pi^2(f_D^2 - f^2)\right)^{-1/2}$ when $|f| < f_D$ and 0 elsewhere. For the Jakes PSD, the level crossing rate (LCR) is given by $N_R = \sqrt{2\pi} f_D \lambda e^{-\lambda^2}$ [5], where $\lambda = R_{th}/R_{rms}$ is the value of the specified threshold level R_{th}, normalized to the rms value of the fading envelope.

A well-known approach for generating fading variates with the required statistical properties is the filter-based method. This technique uses a lowpass filter to shape the flat spectrum of uncorrelated Gaussian random variables [6]. The U-shaped frequency response of the shaping filter is equal to the square root of the Jakes PSD. Different fading channel simulators using the filter-based method have been proposed [7]–[10]. The main drawback of the filter-based method is that it is very computationally-intensive as it requires Gaussian variate generators, shaping filters, and multi-stage interpolators [7], [11]. This complexity may become prohibitive when implementing all of the channels that would be required in a multiple-input multiple-output (MIMO) fading channel simulator.

Another well-established technique for modeling the behavior of a Rayleigh fading channel involves superimposing a sufficient number of sinusoidal waveforms [1], [12], [13]. The accuracy of the so-called sum-of-sinusoids (SOS) simulator depends on the amplitudes, frequencies and phases of the component sinusoids. Each of these parameters can be a random variable or a constant, producing stochastic or deterministic models, respectively. Due to the significantly reduced computation required in this approach compared to the filter-based method, the SOS-based models have been widely used to implement fading channel simulators [14], [15]. There are also several hardware-based implementations of SOS-based fading channel simulators based on both deterministic models [16], [17] and stochastic models [18]–[21].

The performance of wireless communication systems strongly depends on radio channel characteristics. Thus, it is crucial that the chosen channel model accurately reproduce the statistical properties of the real world conditions. Unfortunately, some of the proposed models and, consequently, the fading channel simulators based on those models, have weaknesses in their statistical properties such as their stationarity and ergodicity [22], [23]. The main contributions of this work are as follows:

- We propose an improved statistical SOS-based model. The time-averaged statistical properties of the new model are confirmed in simulation, showing good agreement with the theoretical reference functions. Moreover, the cross-correlations between sequences of generated fading variates are small, which makes this model attractive for simulating wideband (i.e., frequency-selective) and MIMO channels.
- While software-based MC simulation techniques can be extremely slow and commercial hardware-based simulators tend to be bulky and expensive [24], [25], the proposed hardware-based fading channel simulator can be implemented on a single field-programmable gate array (FPGA). To the best of our knowledge, there is no hardware-based implementation of a SOS-based model that requires such a small number of sinusoids (e.g., 8) for generating fading variates with similarly accurate time-averaged statistical properties. The new

978-1-4244-2075-9/08/$25.00 ©2008 IEEE

simulator employs time-multiplexed functional resources to reduce the hardware size, without degrading the fading variate generation rate.

• The new fading channel simulator is compact enough that a MIMO channel simulator with a moderate number of antennas (e.g., 4×4) can be implemented on a single FPGA. A scalable implementation of a time-correlated MIMO fading channel simulator on a FPGA is provided. Each channel can be parameterized independently to support various propagation conditions. The MIMO channel simulator can be readily extended to model spatial correlation between antennas.

The rest of this paper is organized as follows. Section II presents our proposed improved model. Section III describes an efficient fixed-point FPGA implementation of the proposed SOS-based channel simulator. The spatiotemporally correlated fading MIMO channel simulator is presented in Section IV. Finally, Section V makes some concluding remarks.

II. The Proposed SOS-based Fading Channel Model

Various fading channel simulators have been proposed over the last three decades [13], [16], [22], [26]–[30]. Unfortunately some of the proposed models, such as the deterministic Jakes' simulator [2] and its derivatives (either deterministic or stochastic models), have undesirable statistical properties. For example, the cross-correlation function can deviate from zero [22] or the model can require a large number of sinusoids to produce multiple uncorrelated Rayleigh fading processes [29], [31]). Comparative analyses of the SOS-based models can be found in [32]–[34]. It was concluded in [32] that the statistical model in [30] has superior properties compared to the other SOS-based models. This model (henceforth called *Model I*) requires a relatively small number of sinusoids (between 8 and 12) to generate statistically accurate fading variates and, hence, appears suitable for the efficient implementation of fading channel simulators [18], [19]. The complex envelope of the fading signal in *Model I* can be written as follows:

Model I:

$$c(t) = \sqrt{\frac{2}{N}}\Big[\sum_{n=1}^{N}\cos\left(2\pi f_D t\cos(\alpha_n)+\varphi_n\right) + j\sum_{n=1}^{N}\cos\left(2\pi f_D t\sin(\alpha_n)+\psi_n\right)\Big] \quad (1)$$

$$\alpha_n = \frac{2\pi n - \pi + \theta}{4N}, \qquad n = 1, \cdots, N \quad (2)$$

where θ, φ_n and ψ_n are mutually independent random variables, uniformly distributed over $[-\pi, \pi)$. The main drawback of *Model I* is that its statistical properties converge to the desired properties only over a relatively large number of simulation trials. However, the statistical properties of a single trial, no matter how long, deviate from the reference properties [23]. If the channel simulator is ergodic, then each simulation trial produces the same statistics, which drastically reduces the overall simulation time [23]. Fig. 1 plots the ACF of the channel generated using one trial with *Model I* for one block containing 2×10^6 samples, $f_D T_s = 0.01$, and $N = 8$. Clearly, the ACF deviates from the reference ACF, especially at the larger lags, while the CCF stays close to zero. To generate more accurate results one could frequently reset the random phases in *Model I*. However, this is not desirable in practical channel simulations since it would require multiple simulation restarts and introduce discontinuities between channel samples. At the receiver, the channel estimation or carrier recovery would need to be re-acquired at the start of each new trial.

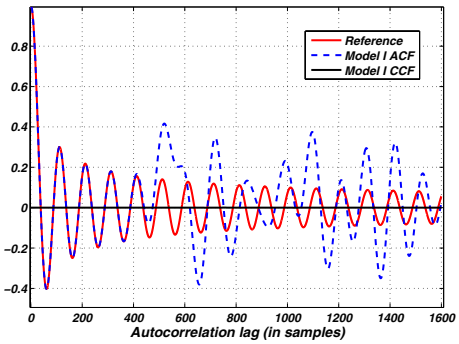

Fig. 1. ACF and CCF for one block containing 2×10^6 fading samples using the Zheng and Xiao's SOS model with $f_D T_s = 0.01$ for $N = 8$.

To overcome the limitations of *Model I*, we proposed in [33] to use a stochastic process [35] rather than a random variable for the angle of arrival. This change was motivated by the fact that in an isotropic scattering environment, the angle of arrival changes slowly and continuously. Thus $\theta[m]$, which is uniformly distributed over $[-\pi, \pi)$, must be highly correlated, and so the time samples behave like a random walk process [36]. It was shown in [33] that the time-averaged ACF of the fading components closely follows the theoretical reference ACF while the CCF between quadrature components is close to zero. This model is very attractive for simulating fading channels as single-pass time-averaging is usually much more desirable than ensemble averaging [37]. To further reduce the cross-correlation between different sequences of generated fading samples using the model in [33] (as required in wideband channels and multiple antenna systems), we propose to replace the random variables φ_n and ψ_n in *Model I* with random walk processes too. This new model (henceforth called *Model II*) can be written in discrete time as follows:

Model II:

$$c[m] = c_i[m] + jc_q[m]$$

$$c_i[m] = \sqrt{\frac{2}{N}}\sum_{n=1}^{N}\cos\left(2\pi f_{D_n} T_s m\cos(\alpha_n[m])+\varphi_n[m]\right) \quad (3)$$

$$c_q[m] = \sqrt{\frac{2}{N}}\sum_{n=1}^{N}\cos\left(2\pi f_{D_n} T_s m\sin(\alpha_n[m])+\psi_n[m]\right) \quad (4)$$

where m is the discrete-time index, $f_{D_n} T_s$ is the normalized maximum Doppler frequency of the n-th sinusoid, T_s is the symbol period, and $\alpha_n[m] = (2\pi n - \pi + \theta[m])/(4N)$ where

θ is a stationary stochastic process. In *Model II*, φ_n and ψ_n are also independent stationary random processes (RPs). Each random process θ, φ_n, and ψ_n is updated using Algorithm 1, where u_i is a random variable with independent uniformly distributed samples over $[0, 1)$ and "fadingLength" denotes the required number of generated fading samples. The update rate K of the random processes will be discussed in more detail in Section III. The variable ξ should be small enough

Algorithm 1 The proposed updating procedures for random processes θ, φ_n, and ψ_n.

Initialize $\xi \ll 1$, $\theta[0] = u_1 \in U(-\pi, \pi)$, $\varphi[0] = u_2 \in U(-\pi, \pi)$, $\psi[0] = u_3 \in U(-\pi, \pi)$;
$\delta_\theta = \delta_o$; $\delta_\varphi = \delta_o$; $\delta_\psi = \delta_o$;
$s_\theta = 1$; $s_{\varphi_n} = 1$; $s_{\psi_n} = 1$;
for $m = 1 :$ fadingLength **do**
  if $\mod(m, K) == 0$ **then**
    $\delta_\theta = s_\theta(\xi \times u_1[m])$;
    $\theta[m] = \theta[m-1] + \delta_\theta$;
    if $\theta[m] > +\pi$ **then**
      $\theta[m] = +\pi$; $s_\theta = -s_\theta$;
    end if
    if $\theta[m] < -\pi$ **then**
      $\theta[m] = -\pi$; $s_\theta = -s_\theta$;
    end if
    for $n = 1 : N$ **do**
      $\delta_{\varphi_n} = s_{\varphi_n}(\xi \times u_2[m])$;
      $\varphi_n[m] = \varphi_n[m-1] + \delta_{\varphi_n}$;
      if $\varphi_n[m] > +\pi$ **then**
        $\varphi_n[m] = +\pi$; $\delta_{\varphi_n} = -\delta_{\varphi_n}$;
      end if
      if $\varphi_n[m] < -\pi$ **then**
        $\varphi_n[m] = -\pi$; $s_{\varphi_n} = -s_{\varphi_n}$;
      end if
      $\delta_{\psi_n} = s_{\psi_n}(\xi \times u_3[m])$;
      $\psi_n[m] = \psi_n[m-1] + \delta_{\psi_n}$;
      if $\psi_n[m] > +\pi$ **then**
        $\psi_n[m] = +\pi$; $s_{\psi_n} = -s_{\psi_n}$;
      end if
      if $\psi_n[m] < -\pi$ **then**
        $\psi_n[m] = -\pi$; $s_{\psi_n} = -s_{\psi_n}$;
      end if
    end for
  end if
end for

to produce highly correlated (or slowly changing) angles of arrivals and phases for the Doppler rates of interest. Suitable values for ξ were suggested in [33]. The step size δ_χ for each random walk process is the product of ξ and a generated uniform sample $u_i[m]$.

Fig. 2 plots the ACF and CCF of the generated fading variates with *Model II* for one block containing 2×10^6 samples, $f_D T_s = 0.01$, and $N = 8$. Note that the approximate ACF matches the theoretical ACF, even at the larger lags, while the CCF of the quadrature components is close to zero. Figs. 3 and 4 show close agreement between the theoretical functions and approximate probability density functions (PDFs) and the LCRs of the channel, with the same parameters as in Fig. 2. Fig. 5 shows the small cross-correlation between two sequences of 2×10^6 generated fading variates.

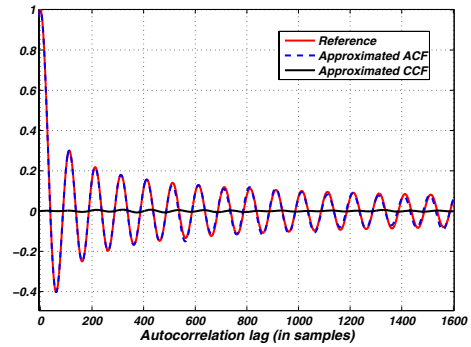

Fig. 2. ACF and CCF for one block containing 2×10^6 fading samples with $f_D T_s = 0.01$ and $N = 8$.

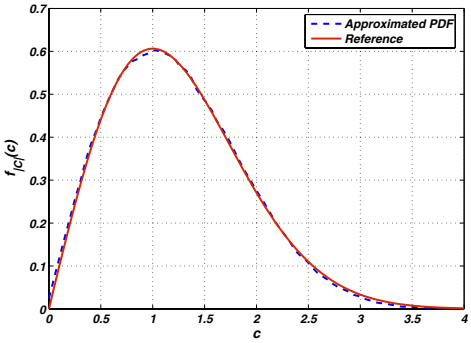

Fig. 3. PDF for one block containing 2×10^6 fading samples with $f_D T_s = 0.01$ and $N = 8$.

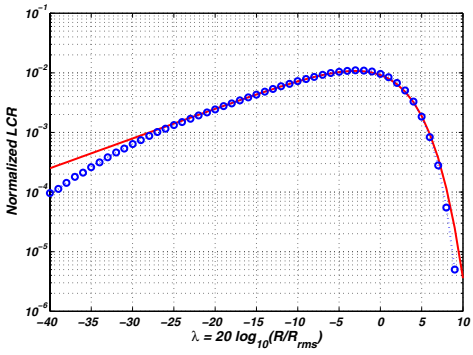

Fig. 4. LCR for one block containing 2×10^6 fading samples with $f_D T_s = 0.01$ and $N = 8$.

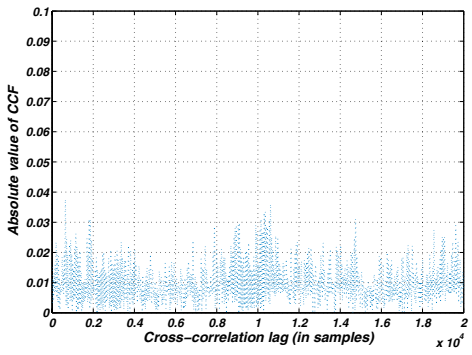

Fig. 5. Cross-correlation between two sequences of 2×10^6 generated fading samples with $f_DT_s = 0.01$ and $N = 8$.

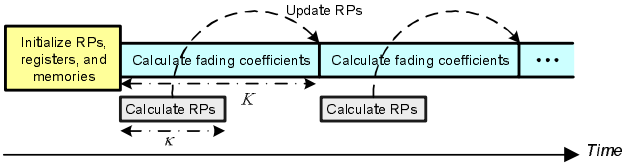

Fig. 6. Dataflow for updating random processes and calculating fading coefficients.

III. Implementation of the Fading Channel Simulator

In this section we propose a compact channel simulator that generates multiple streams of independent fading coefficients. We use a time-overlapped approach where one simulation trial is divided into shorter intervals, each of length K time units (e.g., clock cycles). To minimize the number of functional units and memories, instead of updating the RP values every clock cycle, their values are updated every κ clock cycles where $1 < \kappa \leq K$. Rather than computing the RPs in parallel, they can be calculated sequentially using time-shared arithmetic resources but distinct register sets. The RPs are calculated using Algorithm 1. The updated RP values are copied at the beginning of the next interval to another set of registers where they can be read during the next fading coefficient generation interval. This dataflow schedule is shown in Fig. 6.

We will now present the computation of the $c_i[m]$ component of $c[m]$. $c_q[m]$ can be evaluated similarly. First, we factor out the 2π in (3) and re-write $c_i[m]$ as follows:

$$c_i[m] = \sqrt{\frac{2}{N}} \sum_{n=1}^{N} \cos\Big(2\pi\big((f_{D_n}T_s \cos\,\alpha_n[m]).m + \tilde{\varphi}_n[m]\big)\Big), \tag{5}$$

where $\tilde{\varphi}_n[m] \in [-0.5, 0.5)$. We can write $\alpha_n[m]$ as follows:

$$\alpha_n[m] = \frac{2\pi(n - 0.5 + \tilde{\theta}[m])}{4N}, \tag{6}$$

where $\tilde{\theta}[m] \in [-0.5, 0.5)$. Note that $\theta[m] \in [-\pi, \pi)$ and $\varphi_n[m] \in [-\pi, \pi)$ in (3) are rescaled as $\tilde{\theta}[m]$ and $\tilde{\varphi}_n[m]$ in (5), respectively, both within $[-0.5, 0.5)$. To update the random walk process $\tilde{\theta}$ (and also the other RPs $\tilde{\varphi}_n$ and $\tilde{\psi}_n$), the random step size δ for each process is calculated by multiplying a generated uniform pseudo-random number (PN) in $(0, 1)$ by a positive value $\xi \ll 1$. Note that the fading variate generator must create long periods of nonrepeating propagation conditions to evaluate the error-rate performance accurately and ensure confidence in the test results. To ensure that test conditions do not repeat during test runs, we used a combined linear pseudo-random number generator (PNG) [38] that has a very long period and substantially better randomness and correlation properties compared to conventional linear PNGs. To eliminate the multiplication operation, the value of ξ is chosen to be 2^{-d}. Thus, the *"Bit Mask"* module masks the d most significant bits of the generated PNs and uses the result $\delta_{\tilde{\theta}}$ as a random updating step for the RP $\tilde{\theta}$, as shown in Fig. 7. The RPs $\tilde{\varphi}_1, \cdots, \tilde{\varphi}_n$ are updated similarly to $\tilde{\theta}$, as shown in Fig. 7.

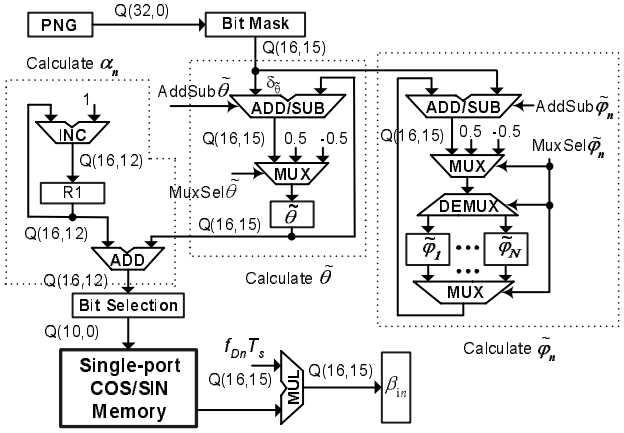

Fig. 7. Datapath for updating the random processes using shared hardware.

According to (6), for $n = 1, \cdots, 8$, the lower and upper bounds of α_n are $[0, \pi/16)$ and $[7/16\pi, \pi/2)$, respectively. Therefore, we only need to calculate the cosine values (for $c_i[m]$) and the sine values (for $c_q[m]$) of $\alpha_n[m]$ over the interval $[0, \pi/2)$. We uniformly quantize the cosine and sine values within $[0, \pi/2)$ over 512 sub-intervals and store their values in one of the Block RAM (BRAM) memories in the FPGA. This BRAM is used in the 1024×16 configuration with the first 512 words storing the cosine values and the second half storing the sine values, both in the 2's complement fixed-point format $\mathcal{Q}(16, 15)$ (i.e., 16-bit words with 15 least significant bits dedicated to the fractional part). The *"Bit Selection"* module selects the 10 least significant bits of the adder output (i.e., the value of α_n) and addresses the *"COS/SIN Memory"*. The N values of $\cos\,\alpha_n[m]$ are read from the *"COS/SIN Memory"* in N clock cycles and are multiplied by $f_{D_n}T_s$ sequentially to generate N values of $\beta_{i_n} = f_{D_n}T_s \cos\,\alpha_n$. The N values of $\beta_{q_n} = f_{D_n}T_s \sin\,\alpha_n$ are calculated similarly in the next N clock cycles. Note that if the Doppler frequencies of the N sinusoids are equal to a common value f_D, we can precompute

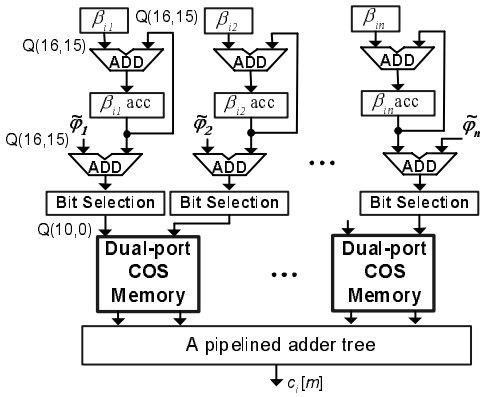

Fig. 8. Datapath for calculating $m(f_D T_s \cos\alpha_n[m])$ and generating $c_i[m]$.

and store $f_D T_s \cos\alpha_n$ and $f_D T_s \sin\alpha_n$ in the *"COS/SIN Memory"* to avoid the multiplication operation. However, to support variable Doppler frequencies, $f_{D_n} T_s$ are implemented as variables read from the primary inputs. Since $f_{D_n} T_s \ll 1$ and since the sine and cosine values are within $[-1, 1)$, β_{i_n} and β_{q_n} are within $[-1, 1)$ represented in $\mathcal{Q}(16, 15)$.

Fig. 8 shows the datapath for calculating $m(f_{D_n} T_s \cos\alpha_n[m])$. After calculating $\beta_{i_n}[m] = f_{D_n} T_s \cos\alpha_n[m]$ (as shown in Fig. 7), integer multiples of β_{i_n} can be obtained using an accumulator instead of a multiplier. Then the $m\,\beta_{i_n}[m]$ values are added to the $\tilde{\varphi}_n$ values and used to address $N/2$ dual-port cosine memories. Note that when calculating the $\cos(2\pi\gamma)$ functions, only the fractional part of the γ is required and the integer part can be ignored. Hence, the adders and registers used in Fig. 8 are only 16-bit modules. The outputs of the cosine memories are then passed to a pipelined adder tree to compute the in-phase component of $c[m]$. Computation of $c_q[m]$ can be performed simultaneously using a similar datapath to that in Fig. 8.

The registers and the operation of the functional modules for computing $c_i[m]$ and $c_q[m]$ are controlled using a K-state machine. The value of K depends on the number N of sinusoids, the dependency between operations, and also the number of pipeline stages in the longest path in the datapath that updates the RPs. For clarity let us ignore the number of pipeline stages. As shown in Fig. 7, since we use only one PNG and since updating $\tilde{\theta}$, $\tilde{\varphi}_n[m]$ and $\tilde{\psi}_n[m]$ requires independent uniform PNs, these RPs are updated sequentially in $2N + 1$ clock cycles. After updating $\tilde{\theta}[m]$, values $\beta_{i_n}[m]$ and $\beta_{q_n}[m]$ can be calculated sequentially for N sinusoids in $2N$ clock cycles. Note that register $\tilde{\theta}$ is disabled during the calculation of β_{i_n} and β_{q_n}. The process of updating $\beta_{i_n}[m]$ and $\beta_{q_n}[m]$ takes place in parallel with the updating of $\tilde{\varphi}_n[m]$ and $\tilde{\psi}_n[m]$. Therefore, K can be any integer number greater than or equal $2N + 1$. For $N = 8$ sinusoids, and including several pipeline stages in the design, we chose $K = 32$. Note that register R_1 is initialized to 0.5 at the start (i.e., $n = 1$) of the $\cos\alpha_n[m]$ and $\sin\alpha_n[m]$ computations.

An implementation of a fading channel simulator using $N = 8$ sinusoids on a Xilinx Virtex-II Pro XC2VP100-6 FPGA uses 1105 of the 44096 configurable slices (2%), one of the 444 dedicated multipliers (< 1%), and 9 of the 444 on-chip memory blocks (1%) while generating over 200 million 16-bit complex-valued fading samples per second. As shown in Table I, the fading channel simulator in [33] requires 10% of the dedicated resources in a Xilinx Virtex-II Pro XC2VP100-6 FPGA. The reason that the model in [33] uses such a large number of dedicated resources is because in order to compute $mf_D T_s$ in (3) and (4), since m, which is a discrete-time index, can be a large integer, in [33] we used a 48-bit register to accumulate successive multiples of $f_D T_s$. Then we used N 60-bit multipliers to multiply the 12-bit $\cos\alpha_n[m]$ with the 48-bit value of $mf_D T_s$. Thus this earlier scheme is certainly not compact enough for multiple antenna systems with more than 10 channels given present FPGAs. However, the proposed scheme uses substantially fewer dedicated multipliers, fewer BRAMs, and also fewer configurable slices than the proposed implementation scheme in [33], while the slightly lower fading generation rate is negligible. For comparison, key implementation results of

TABLE I

CHARACTERISTICS OF SOS-BASED FADING CHANNEL SIMULATORS BASED ON THE MODEL IN [33] AND ON THE PROPOSED MODEL

Design	A [a]	B	C
Model	[33]	*Model II*	*Model II*
Clock freq. (MHz)	204.75	201.1	177.99
Output rate (MSamps/sec)	204	201	177
Configurable slices	2444 (5%)	1105 (2%)	1100 (4%)
Dedicated resources	48 (10%)	1 (< 1%)	1 (< 1%)
On-chip memory blocks	12 (2%)	9 (2%)	9 (7%)

[a]Designs A and B are implemented on a Xilinx Virtex-II Pro XC2VP100-6 FPGA. Design C is synthesized on a Xilinx Virtex-II XC2V4000-6 FPGA.

stochastic block-based SOS-based models are presented in Table II. As mentioned in Section II, the SOS-based fading channel simulators in Table II are suitable only for block-based simulations.

TABLE II

IMPLEMENTATION RESULTS OF SOS-BASED FADING CHANNEL SIMULATORS USING DIFFERENT MODELS ON A XILINX VIRTEX-II PRO XC2VP100-6 FPGA.

Design	D [18][a]	E [19]	F [17] [b]
Model	[30]	[30]	[39]
Clock freq. (MHz)	201.69	210.3	50
Output rate (MSamps/sec)	201	210	4×1.5
Configurable slices	542 (1%)	8814 (19%)	22576 (58%)
Dedicated resources	8 (1%)	256(57%)	-
On-chip memory blocks	8	–	17%

[a]Designs D and E are implemented on a Xilinx Virtex-II Pro XC2VP100-6 FPGA. Design F is synthesized on a Altera APEX EP20K1000EBC652-3.
[b]Designs D and E are single antenna systems. Design F is a 4×4 MIMO system.

The proposed fading channel model can be easily scaled, with the parameters of each channel configured separately,

to increase the number of simultaneous channels to simulate n_T-input, n_R-output (n_T, n_R) MIMO channels and other diversity scenarios. Even though the complexity of a multiple antenna fading channel simulator grows quadratically with the number of antennas involved, the compact implementation results, as given in Table I, permit the implementation of a MIMO channel with a moderate number of antennas on a single FPGA. For example, an implementation of a 16-path flat-fading $(4, 4)$ MIMO system on a Xilinx Virtex-II Pro XC2VP100-6 FPGA uses 17678 (40%) of the configurable slices, 16 (3%) of the dedicated multipliers, and 144 (32%) of the BRAMs while generating 4×201 million 2×16-bit complex-valued fading samples per second.

Ideally, a simulator should be scalable to support any number of transmit/receive antenna pairs. However, the total number of channels that can be realized is limited by the available configurable resources on the given FPGA. For a communication system with a greater number of channels, one could consider the time-multiplexed execution of different channels using the same set of hardware. For example, one instance of the channel simulator could be used for $p > 1$ channels. Therefore, each element of the (n_T, n_R) channel matrix would be a p-fold down-sampled version of the actual fading waveform. Thus the proposed scheme offers a trade-off between the number of realizable channels on a single FPGA and the fading generation rate.

IV. Implementation of Spatial Correlated MIMO Rayleigh Channel Model

While the above model assumes that different fading sequences are correlated in time but uncorrelated in space, in a real world scenario, the fades usually exhibit spatial correlations. The spatial correlation depends on the physical parameters of the multi-element antenna (e.g., antenna spacing and orientation) and on the scattering characteristics of the environment (e.g., the presence of independent propagation paths). For simulating multiple antenna systems, these spatial correlations between channels needs to be user-controllable. To obtain the space-time correlation characteristics, a temporally correlated random process can be followed by a linear transformation to become also spatially correlated. Assume that $\mathbf{\Psi}_T$ and $\mathbf{\Psi}_R$ are the long-term stable transmitter and receiver correlation matrices, respectively. If $\mathbf{H}_w$ denotes a $n_R \times n_T$ matrix of temporally correlated i.i.d Gaussian variables of unity variance, then the spatially/temporally correlated MIMO channel model can be written as $\mathbf{H} = \mathbf{A}^H \mathbf{H}_w \mathbf{B}$ [40], where $\mathbf{A}$ and $\mathbf{B}$ can be obtained using the Cholesky decomposition of $\mathbf{\Psi}_T = \mathbf{B}\mathbf{B}^H$ and $\mathbf{\Psi}_R = \mathbf{A}\mathbf{A}^H$, respectively. To normalize the channel gain, we set $\mathrm{Tr}(\mathbf{\Psi}_T) = n_T$ and $\mathrm{Tr}(\mathbf{\Psi}_R) = n_R$, where $\mathrm{Tr}(\mathbf{X})$ denotes the sum of the diagonal elements of $\mathbf{X}$. The Cholesky decomposition of correlation matrices can be calculated in software in advance and then loaded as constant parameters into the hardware simulator to reduce the computational complexity. For example, the transmitter is often assumed to be elevated and unobstructed while the receiver is taken to be surrounded by a scattering ring. Thus the receive correlations are much smaller than the transmit ones.

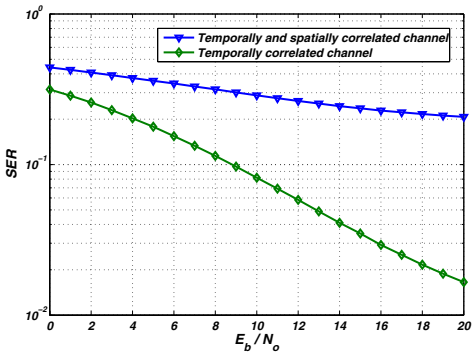

Fig. 9. SER versus E_b/N_o for a $(4, 4)$ MIMO system with 4-QAM modulation and MMSE decoding.

The computationally intensive matrix operations that provide spatial correlation to a MIMO channel are also implemented on the FPGA. An implementation of a spatially and temporally correlated $(4, 4)$ MIMO system using the SOS-based Rayleigh fading channel simulator presented in Section III on a Xilinx Virtex-II Pro XC2VP100-6 FPGA uses 20603 (46%) configurable slices, 136 (30%) dedicated multipliers, and 144 (32%) on-chip BRAMs while generating 4×201 million 2×16-bit complex-valued fading samples per second. Fig. 9 plots the SER versus SNR for a $(4, 4)$ MIMO system correlated only in time and also correlated in time as well as space using a 4-QAM modulation and $f_D T_S = 0.01$. The element-wise absolute value of the transmitter correlation matrix used in the simulation is:

$$\mathrm{abs}(\mathbf{\Psi}_T) = \begin{pmatrix} 1.000 & 0.3727 & 0.3689 & 0.3576 \\ 0.3727 & 1.000 & 0.3727 & 0.3689 \\ 0.3689 & 0.3727 & 1.000 & 0.3727 \\ 0.3576 & 0.3689 & 0.3727 & 1.000 \end{pmatrix}$$

This multi-channel real-time emulator can be used for various scenarios such as IEEE $802.11n$ MIMO system characterization, debugging, validation, performance testing, and optimization.

V. Conclusions

Using a hardware-based channel simulator is a cost-effective way of avoiding the inconvenience and cost of field trial measurements while still permitting the accurate and repeatable characterization of wireless systems. We proposed an improved model based on the sum-of-sinusoids technique that permits efficient implementations of correlated Rayleigh fading channel simulators. Using numerical simulation it was shown that the time-averaged statistical properties of the generated fading samples do indeed match the desired theoretical properties. Multiple independent and/or correlated fading channels can be generated efficiently to support the modeling and verification of different realistic scenarios such as frequency-selective channels and multiple antenna systems.

For example, implementation of a spatially and temporally correlated $(4, 4)$ multiple antenna channel simulator on a Xilinx Virtex-II Pro XC2VP100-6 FPGA uses less than half of the configurable slices, 30% of the dedicated multipliers, and 32% of the on-chip memory blocks, while generating over 4×200 million 2×16-bit fading samples per second.

REFERENCES

[1] P. Bello, "Characterization of randomly time-variant linear channels," *IEEE Trans. Commun.*, vol. 11, no. 4, pp. 360–393, 1963.

[2] W. C. Jakes, *Microwave Mobile Communications*. Piscataway, NJ: Wiley-IEEE Press, 1994.

[3] I. S. Gradshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products*. San Diego: Academic Press, 2000.

[4] R. H. Clarke and W. L. Khoo, "3-D mobile radio channel statistics," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 798 – 799, 1997.

[5] T. S. Rappaport, *Wireless Communications: Principles and Practice*. Prentice Hall, 2002.

[6] G. L. Stüber, *Principles of Mobile Communication*. New York: Kluwer Academic Publishers, 2001.

[7] M. A. Wickert and J. Papenfuss, "Implementation of a real-time frequency-selective RF channel simulator using a hybrid DSP-FPGA architecture," *IEEE Trans. Microw. Theory Tech.*, vol. 49, no. 8, pp. 1390 – 1397, 2001.

[8] A. K. Salkintzis, "Implementation of a digital wide-band mobile channel simulator," *IEEE Trans. Broadcast.*, vol. 45, no. 1, pp. 122 – 128, 1999.

[9] A. Alimohammad, S. F. Fard, B. F. Cockburn, and C. Schlegel, "A compact fading channel simulator using timing-driven resource sharing," in *IEEE ASAP*, 2007.

[10] ——, "A compact single-FPGA fading channel simulator," *to appear in IEEE Trans. Circuits Syst. II*.

[11] C. Komninakis, "A fast and accurate Rayleigh fading simulator," in *Proceedings of the IEEE Global Telecommunications Conference*, 2003, pp. 3306–3310.

[12] E. N. Gilbert, "Energy reception for mobile radio," *Bell System Technical Journal*, pp. 1779 – 1803, October 1965.

[13] R. H. Clarke, "A statistical theory of mobile-radio reception," *Bell System Technical Journal*, vol. 47, pp. 957–1000, 1968.

[14] W. Cheng-Xiang and M. Pätzold, "Efficient simulation of multiple cross-correlated Rayleigh fading channels," in *IEEE PIMRC*, 2003, pp. 1526 – 1530.

[15] R. Brown, R. Beaudin, and H. Hamel, "Frequency selective RF channel simulator," in *IEEE MILCOM*, 2002, pp. 617–621.

[16] M. Pätzold, R. Garcia, and F. Laue, "Design of high-speed simulation models for mobile fading channels by using table look-up techniques," *IEEE Trans. Veh. Technol.*, vol. 49, no. 4, pp. 1178–1190, July 2000.

[17] M. Cui, H. Murata, and K. Araki, "FPGA implementation of 4×4 MIMO test-bed for spatial multiplexing systems," in *IEEE PIMRC*, 2004, pp. 3045–3048.

[18] A. Alimohammad and B. F. Cockburn, "Compact implementation of a sum-of-sinusoids Rayleigh fading channel simulator," in *Proceedings of the IEEE ISSPIT*, 2006, pp. 253–257.

[19] ——, "A reconfigurable SOS-based Rayleigh fading channel simulator," in *IEEE Intl. Workshop on Signal Processing Systems, Design and Implementation*, 2006, pp. 39–44.

[20] T.-P. Wang, C.-H. Liao, and T.-D. Chiueh, "A real-time digital baseband MIMO channel emulation system," in *IEEE ISCAS*, 2007, pp. 2606 – 2609.

[21] C.-H. Liao, T.-P. Wang, , and T.-D. Chiueh, "A novel low-complexity Rayleigh fader for real-time channel modeling," in *IEEE ISCAS*, 2007, pp. 2602 – 2605.

[22] M. F. Pop and N. C. Beaulieu, "Limitations of sum-of-sinusoids fading channel simulators," *IEEE Trans. Commun.*, vol. 49, pp. 699–708, 2001.

[23] M. Pätzold and B. O. Hogstad, "Classes of sum-of-sinusoids Rayleigh fading channel simulators and their stationary and ergodic properties Part I," *WSEAS Transactions on Mathematics*, vol. 5, no. 2, pp. 222 – 230, 2006.

[24] *ACE 400NB MIMO Channel Emulator*, Azimuth, 2006.

[25] *MIMO Channel Simulator*, ARC SmartSim, 2007.

[26] W. C. Jakes, *Microwave Mobile Communications*. Piscataway, NJ: Wiley-IEEE Press, 1974.

[27] M. Pätzold, *Mobile Fading Channels*. West Sussex, U.K.:Wiley, 2002.

[28] P. Hoeher, "A statistical discrete-time model for the WSSUS multipath channel," *IEEE Trans. Veh. Technol.*, vol. 41, pp. 461–468, 1992.

[29] Y. Li and X. Huang, "The simulation of independent Rayleigh faders," *IEEE Trans. Commun.*, vol. 50, no. 9, pp. 1503–1514, 2002.

[30] Y. R. Zheng and C. Xiao, "Improved models for the generation of multiple uncorrelated Rayleigh fading waveforms," *IEEE Commun. Lett.*, vol. 6, pp. 256–258, 2002.

[31] C.-X. Wang and M. Pätzold, "Methods of generating multiple uncorrelated Rayleigh fading processes," in *Proceedings of the IEEE Semiannual Vehicular Tech. Conf.*, 2003, pp. 510–514.

[32] C. S. Patel, G. L. Stüber, and T. G. Pratt, "Comparative analysis of statistical models for the simulation of Rayleigh faded cellular channels," *IEEE Trans. Commun.*, vol. 53, pp. 1017–1026, 2005.

[33] A. Alimohammad, S. F. Fard, B. F. Cockburn, and C. Schlegel, "An improved SOS-based fading channel emulator," in *IEEE Fall Veh. Tech. Conf.*, 2007, pp. 931–935.

[34] A. M. M. Donald and J. C. Olivier, "A comparative study of deterministic and stochastic sum-of-sinusoids models of Rayleigh-fading wireless channels," in *IEEE WCNC*, 2007, pp. 2027 – 2031.

[35] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, 2002.

[36] P. Pampaloni and S. Paloscia, *Microwave Radiometry and Remote Sensing of the Earth's Surface and Atmosphere*. VNU Science Press, 2000.

[37] C. Xiao, Y. R. Zheng, and N. C. Beaulieu, "Novel sum-of-sinusoids simulation models for Rayleigh and Rician fading channels," *IEEE Trans. Wireless Commun.*, vol. 5, no. 12, pp. 3667 – 3679, 2006.

[38] P. L'Ecuyer, "Tables of maximally equidistributed combined LFSR generators," *Math. of Comp.*, vol. 68, no. 225, pp. 261–269, 1999.

[39] P. Dent, G. E. Bottomley, and T. Croft, "Jakes fading model revisited," *Electronics Letters*, vol. 29, no. 13, pp. 1162–1163, June 1993.

[40] M. Kiessling and J. Speidel, "Statistical transmit processing for enhanced MIMO channel estimation in presence of correlation," in *IEEE Global Telecommunications Conference*, 2003, pp. 2411 – 2415.