FPGA-Based Communications Receivers for Smart Antenna Array Embedded Systems

Constantin Siriteanu,[1, 2] Steven D. Blostein,[1] and James Millar[3]

[1] *Department of Electrical and Computer Engineering, Queen's University, Kingston, ON, Canada K7L 3N6*

[2] *Communications Signal Processing Laboratory, Department of Electrical and Computer Engineering, Hanyang University, Seoul, Korea*

[3] *CMC Microsystems, Kingston, ON, Canada K7L 3N6*

Field-programmable gate arrays (FPGAs) are drawing ever increasing interest from designers of embedded wireless communications systems. They outpace digital signal processors (DSPs), through hardware execution of a wide range of parallelizable communications transceiver algorithms, at a fraction of the design and implementation effort and cost required for application-specific integrated circuits (ASICs). In our study, we employ an Altera Stratix FPGA development board, along with the DSP Builder software tool which acts as a high-level interface to the powerful Quartus II environment. We compare single- and multibranch FPGA-based receiver designs in terms of error rate performance and power consumption. We exploit FPGA operational flexibility and algorithm parallelism to design eigenmode-monitoring receivers that can adapt to variations in wireless channel statistics, for high-performing, inexpensive, smart antenna array embedded systems.

1. INTRODUCTION

Conventional wireless communications systems employ a single receiving antenna. Enhanced, antenna array receivers employing beamforming (BF) and maximal-ratio combining (MRC) can generate antenna and diversity gain, that is, increased average and instantaneous (with respect to channel fading) receiver signal-to-noise ratio (SNR) [1–4]. Although beneficial in terms of performance, these enhanced, multibranch algorithms can require much larger computational volumes than the conventional, single-branch receiver. Recent analytical and simulation studies [1–4] of a hybrid algorithm entitled maximal-ratio eigencombining (MREC) claimed efficient performance-complexity tradeoffs for smart antenna arrays.

Receiver algorithms have traditionally been deployed on general-purpose, sequential, digital signal processors (DSPs), or on application-specific integrated circuits (ASICs). Enhanced receiver algorithms, which are generally highly parallelizable, and higher data transmission rates can burden DSPs beyond their capacity for real-time processing. Time-critical, highly parallelizable applications are common in areas ranging from modern communications [5–7] to image [6] and speech [8] processing, and even bioinformatics [9].

ASICs are hardwired for specific tasks. Although fast (sometimes several orders of magnitude faster than DSPs, through hardware parallelism) and power-efficient, implemented designs are inflexible [7]. More importantly, ASIC design and production are time-consuming and extremely expensive for chips produced in small numbers, due to very high nonrecurring engineering cost.

Unlike ASICs, field-programmable gated arrays (FPGAs) are reconfigurable, that is, their internal structure is only partially fixed at fabrication, leaving to the application designer the wiring of the internal logic for the intended task. This can significantly shorten design and production, and thus time to market, for FPGA-based embedded systems. Although FPGAs tend to be slower and to consume more power than ASICs [7], FPGA reconfigurability can benefit platform longevity (which is extremely important in an era of fast-changing wireless communications standards) by allowing design changes/upgrades even in systems already in operation. This flexibility can be effectively exploited for rapid prototyping of advanced communications signal processing, such as Bell Labs Layered Space-Time (BLAST) multi-input multi-output (MIMO) architecture for third-generation Universal Mobile Telecommunications System (UMTS) [5]. Furthermore, an FPGA can, for example,

implement MRC branches either sequentially, or in parallel, or anywhere in between, depending on required speed, available chip resources, and power constraints. FPGA-based implementations concurrently operating several hardware modules can outpace many times their processor-based counterparts [6, 9]. An insightful DSP, FPGA, and ASIC implementation comparison for a four-antenna orthogonal frequency-division multiplexing (OFDM) receiver can be found in [7].

FPGAs are especially well suited for embedded systems (e.g., cellular system base station line cards, or mobile stations) because, beside an area of reconfigurable logical elements, they can also incorporate large amounts of memory, high-speed DSP blocks, clock management circuitry, high-speed input/output (I/O), as well as support for external memory, and high-speed networking and communications bus standards. For a small share of the resources, processors can be included within the FPGA fabric as well [9].

Power consumed in embedded systems is, in general, strictly limited. Otherwise, line-powered designs would require special and/or expensive power sources and heat sinks or may not operate reliably, while portable devices would quickly deplete the battery [10, 11]. Although FPGA chips are judiciously manufactured for power efficiency, application designers also need to carefully consider this issue because a consistently underutilized design wastes static and dynamic powers [10–13].

The objective of this paper is to investigate FPGA suitability for efficient smart antenna array embedded receivers. In the process, we overview an Altera FPGA-based design environment, and implement conventional and enhanced (BF, MRC, MREC) receiver algorithms. It is demonstrated that FPGA implementations of eigenmode-based combining adapted to the slow variations in channel statistics can yield near-optimum bit error rate (BER) performance, for affordable power budgets.

The paper is organized as follows Section 2 presents the received signal model, and overviews BF, MRC, and MREC. Section 3 describes the Altera software and hardware employed to design, simulate, analyze, and implement these receiver algorithms. Comparative performance and cost results are provided in Section 4.

2.   SIGNAL MODEL AND COMBINING METHODS

2.1.   *Received-signal model*

Consider a source transmitting a BPSK signal through a frequency-flat Rayleigh fading channel, and an L-element receiving antenna array. After demodulation, matched filtering, and symbol-rate sampling, the complex-valued received signal vector is given by [4]

$$\widetilde{\mathbf{y}} = \sqrt{E_s}b\widetilde{\mathbf{h}} + \widetilde{\mathbf{n}}, \tag{1}$$

where dependence on the sampling time is not explicit, to simplify notation. The L elements $\widetilde{y}_i$, $i = 1 : L \triangleq 1, \ldots, L$, of the received signal vector $\widetilde{\mathbf{y}} = [\widetilde{y}_1 \widetilde{y}_2 \cdots \widetilde{y}_L]^{\mathrm{T}}$ are called *branches*, and the elements $\widetilde{h}_i$, $i = 1 : L$, of the channel vec-

tor $\widetilde{\mathbf{h}} = [\widetilde{h}_1 \widetilde{h}_2 \cdots \widetilde{h}_L]^{\mathrm{T}}$, are called *channel gains*. In (1), E_s is the energy transmitted per symbol, and b is the transmitted BPSK symbol, with $|b|^2 = 1$ ($b = 1$ for transmitted bit 0, $b = -1$ for transmitted bit 1). We assume that the channel vector $\widetilde{\mathbf{h}}$ and the noise vector $\widetilde{\mathbf{n}}$ are complex-valued, mutually independent, zero-mean Gaussian, with $\widetilde{\mathbf{h}} \sim \mathcal{CN}(\mathbf{0}, \mathbf{R}_{\widetilde{\mathbf{h}}})$ and $\widetilde{\mathbf{n}} \sim \mathcal{CN}(\mathbf{0}, N_0\mathbf{I}_L)$, respectively. Further assumptions are that channel fading [14] is frequency-flat with unit variance on each branch, the noise is temporally white, and the received signal is interference-free. This signal model is simple, yet sufficient for basic performance evaluations [15]. Current-standard wireless communications signaling is beyond the scope of this work.

2.2.   *Azimuth angle spread model*

Due to radio-wave scattering, transmitted signals are received with azimuthal dispersion [14, 16]. Without loss of generality, numerical results presented herein assume truncated Laplacian power azimuth spectrum (p.a.s.) [4] because it accurately models empirical results [16]. The p.a.s. root second central moment is denoted as *azimuth spread* (AS) [16]. Analytical expressions for the elements of $\mathbf{R}_{\widetilde{\mathbf{h}}}$, obtained through straightforward calculations in [4] for a uniform linear array (ULA), indicate that antenna correlation (and thus receiver BER performance [1, 2]) is a function of p.a.s. type, azimuth spread, average angle of arrival (which is assumed to be zero with respect to the broadside, for all the results shown later), and normalized interelement distance d_n (i.e., the ratio between the physical interelement distance and half of the carrier wavelength).

The azimuth spread depends on the environment and antenna array location/height, and is variable [16]. Radio channel measurements for sub/urban scenarios [16] showed that base station azimuth spread is well modeled as a log-normal random variable [16, equation (9)]. For typical urban scenarios [16, Table I], these measurements found that base-station azimuth spread correlation decreases exponentially with the distance traveled by the mobile [16, equation (14)]. The azimuth spread *decorrelation distance*, that is, the distance over which the azimuth spread correlation decreases by a factor of two, was determined as $d_{\mathrm{AS}} = 50$ m [16]. Comparing d_{AS} with the fading coherence distance [17, equation (4.40.b)] d_c computed for the typical system parameter values from Table 1, we conclude that the azimuth spread variation is much slower (by about 3 orders of magnitude) than the fading. Furthermore, for this typical urban scenario, it was found in [16] that $\mathrm{Pr}(1° < \mathrm{AS} < 20°) \approx 0.8$, that is, azimuth spread is small to moderate, producing significant (greater than 0.5) correlations between adjacent elements of a compact ULA, for example, $d_n = 1$ [1, 3].

2.3.   *MRC and BF*

For perfectly known channel (p.k.c.), the optimum (maximum-likelihood) receiver linearly combines the received signal vector with the channel vector, that is, it computes

TABLE 1: Mobile, channel, and receiver (channel estimation) parameters.

Parameter	Value
Mobile speed	$v = 60 \, \text{km/h}$
Transmitted BPSK symbol rate	$f_s = 10 \, \text{ksps}$
Carrier frequency	$f_c = 1.8 \, \text{GHz}$
Pilot symbol period [18, Section III.C]	$M_s = 7$
Maximum Doppler frequency	$f_D = 100 \, \text{Hz}$
Normalized maximum Doppler frequency	$f_m = f_D/f_s = 0.01$
Channel coherence time [17, equation (4.40.b)]	$T_c \approx 1.8 \, \text{ms}$
Channel coherence distance	$d_c = v,$ $T_c \approx 30 \, \text{mm}$
Interpolator size [18, Section III.D]	$T = 11$

$\widetilde{\mathbf{h}}^H \widetilde{\mathbf{y}}$, and then detects the BPSK symbol as

$$\hat{b} = \text{sign} \left[\Re \left(\widetilde{\mathbf{h}}^H \widetilde{\mathbf{y}} \right) \right]. \tag{2}$$

This approach is also known as maximal-ratio combining (MRC) [19] because it maximizes the SNR (instantaneous, i.e., conditioned on the channel gains) at the combiner's output. MRC with $L = 1$ reduces to the conventional, single-branch, receiver.

In actual systems, with imperfectly known channel (i.k.c.), knowledge of the channel gains is acquired through estimation [1, 18]. The received symbol can then be detected as $\hat{b} = \text{sign}\{\Re[\widetilde{\mathbf{g}}^H \widetilde{\mathbf{y}}]\}$, where $\widetilde{\mathbf{g}} = [\widetilde{g}_1 \widetilde{g}_2 \cdots \widetilde{g}_L]^T$, and $\widetilde{g}_i$, $i = 1 : L$, are the channel gain estimates. This combining approach has often been employed and studied [1, 3, 15, 19], although it is suboptimal (when the channel gains are not independent and identically distributed—non-i.i.d.)[3].

MRC is known to provide full diversity gain [19]—that is, the greatest performance improvement, averaging over fading and noise, compared to a single-branch system—for i.i.d. branches. This requires either widely spaced elements, which are unfeasible for pocketsize mobile stations, or rich scattering, which is unlikely at base stations [16].

For narrow azimuth spread, received signals are highly correlated [1, 2] and the received signal energy, proportional to $\text{tr}(\mathbf{R}_{\widetilde{\mathbf{h}}}) \triangleq \sum_{i=1}^{L} (\mathbf{R}_{\widetilde{\mathbf{h}}})_{i,i} = \sum_{i=1}^{L} \lambda_i$, where λ_i, $i = 1 : L$, are the eigenvalues of $\mathbf{R}_{\widetilde{\mathbf{h}}}$, is concentrated within the first few eigenmodes. Then, the channel is said to be spatially nonselective, and the available diversity gain is small [20–22]. Enhanced performance can then be obtained by taking advantage of antenna gain using maximum average SNR beamforming (BF), that is, by combining the received signal vector with the dominant eigenvector of $\mathbf{R}_{\widetilde{\mathbf{h}}}$ [1–4]. Increasing azimuth spread decreases antenna correlation, that is, the channel becomes spatially more selective and higher diversity gain becomes available [1–4]. In subsequent sections, we show how to exploit available antenna and diversity gains within complexity and power constraints.

2.4. Eigencombining method

BF has traditionally been applied in scenarios with very small azimuth spread. Otherwise, MRC has been employed. However, it was recently claimed that a unifying approach, called maximal-ratio eigencombining (MREC), and described below, can adapt to channel correlation (i.e., azimuth spread) variation [1–4, 20]. Our analytical and simulation results have shown that MREC may thus outperform MRC and BF in terms of BER performance and complexity [1–4].

The channel correlation matrix $\mathbf{R}_{\widetilde{\mathbf{h}}}$ has real nonnegative eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_L \geq 0$, orthonormal eigenvectors $\mathbf{e}_i$, $i = 1 : L$, and can be decomposed as $\mathbf{R}_{\widetilde{\mathbf{h}}} = \mathbf{E}_L \mathbf{\Lambda}_L \mathbf{E}_L^H$, where $\mathbf{\Lambda}_L \triangleq \text{diag}\{\lambda_i\}_{i=1}^{L}$ is a diagonal matrix, and $\mathbf{E}_L \triangleq [\mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_L]$ is a unitary matrix. Hereafter, $\mathbf{R}_{\widetilde{\mathbf{h}}}, \mathbf{\Lambda}_L$, and $\mathbf{E}_L$ are assumed perfectly known because, in practice, enough independent channel samples would be available for an accurate estimation. Actual MREC could employ computationally insignificant low-rate eigenstructure updating [20].

MREC of order N consists of the following steps [1–4]:

(i) Karhunen-Loève transformation (KLT) [22] of the received signal vector from (1) with the full-column rank matrix $\mathbf{E}_N \triangleq [\mathbf{e}_1 \mathbf{e}_2 \cdots \mathbf{e}_N]$; the elements of the transformed signal vector, $\mathbf{y} = \mathbf{E}_N^H \widetilde{\mathbf{y}} = \sqrt{E_s} b \mathbf{E}_N^H \widetilde{\mathbf{h}} + \mathbf{E}_N^H \widetilde{\mathbf{n}} = \sqrt{E_s} b \mathbf{h} + \mathbf{n}$, are denoted as *eigenbranches*;

(ii) MRC of the N eigenbranches.

The components of the transformed channel gain vector $\mathbf{h} = \mathbf{E}_N^H \widetilde{\mathbf{h}}$ are further referred to as channel *eigengains*. They are mutually uncorrelated, with zero mean, and variances $\sigma_{h_i}^2 \triangleq E\{|h_i|^2\} = \lambda_i$, that is, $\mathbf{R_h} \triangleq E\{\mathbf{hh}^H\} = \mathbf{\Lambda}_N = \text{diag}\{\lambda_i\}_{i=1}^{N}$, for any channel gain distribution [21]. From the initial assumptions on fading and noise we obtain $\mathbf{h} \sim \mathcal{CN}(\mathbf{0}, \mathbf{\Lambda}_N)$, and $\mathbf{n} = \mathbf{E}_N^H \widetilde{\mathbf{n}} \sim \mathcal{CN}(\mathbf{0}, N_0 \mathbf{I}_N)$, so that the eigengains are independent, which supports straightforward MREC analysis [1–4].

Of all possible transforms, the KLT packs the largest amount of energy from the original, L-dimensional signal vector $\widetilde{\mathbf{y}}$ into the transformed, N-dimensional signal vector $\mathbf{y}$ [22], which is desirable for dimension (i.e., complexity) reduction. Note also that MREC of order $N = 1$ represents in fact BF, while it can be shown that full-MREC, that is, MREC of order $N = L$, is equivalent to MRC [1–4].

2.5. Order selection for MREC

A simple criterion for optimal MREC order selection is [21]

$$\min_{N=1:L} \left[E_s \cdot \sum_{i=N+1}^{L} \lambda_i + N_0 \cdot N \right], \tag{3}$$

better known as the *bias-variance tradeoff criterion* [3, 4] (BVTC) because (3) balances the loss incurred by removing the weakest $(L - N)$ intended-signal contributions (the first term) against the residual-noise contribution (the second term). Computer evaluations found the BVTC effective for MREC adaptation to channel conditions [3, 4]. Note
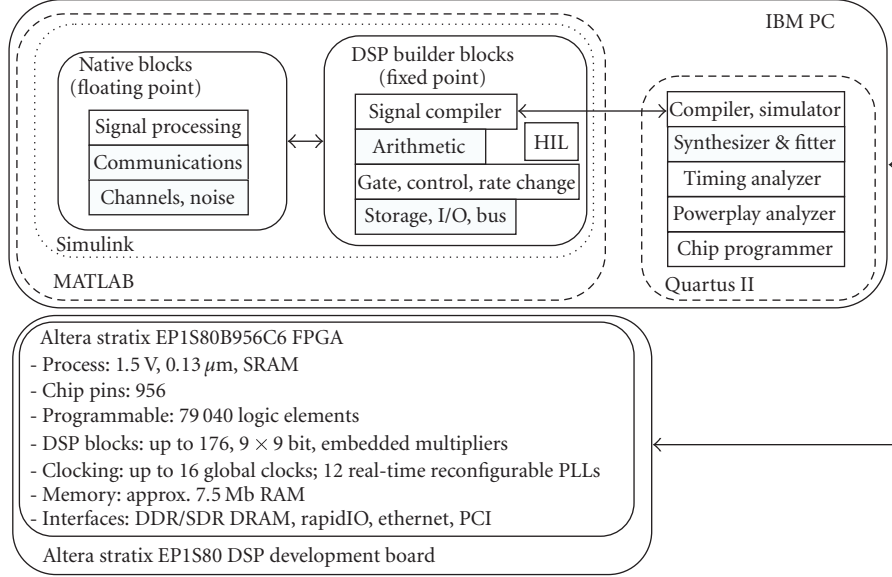
FIGURE 1: FPGA development system hardware/software diagram.

however that since BVTC disregards the MREC complexity, it can overload limited resources.

A different MREC adaptation criterion is described next. Assume that signals received (independently) from N_u mobile stations require processing at a base station with only $N_e \ll N_u L$ available eigenbranch processing modules. Then, a control algorithm determines the largest (dominant) N_e eigenmodes among all transmitting mobiles, and allocates available resources accordingly. For instance, if a receiving antenna array system with $L = 4$ elements has only $N_e = 3$ available eigenbranch processing modules while $N_u = 2$, the available resources are allocated as follows: if the 3 largest eigenvalues (out of $N_u L = 8$) are such that two correspond to User 1, and one to User 2, then two eigenbranch processing modules are allocated to process the received signal vector from User 1, and the other available eigenbranch is allocated to User 2. This approach to selecting eigenbranches for MREC is hereafter denoted as the *eigenvalue-based trade-off criterion* (EVTC), while MREC adapted based on EVTC is referred to as EVTC MREC.

2.6. Channel estimation using pilot-symbol-aided modulation (PSAM)

In PSAM, the transmitter periodically inserts known pilot symbols b_p of energy E_p ($= E_s$ for results shown herein), into the information-encoding symbol stream, and the receiver interpolates the pilot samples acquired across several slots to estimate the channel during data symbols [1–4, 18]. The notation (t, m) is used below to denote temporal indexing, where $t = -T_1 : T_2$ is the time slot index, and $m = 0 : M_s - 1$ is the symbol index within the slot of length M_s. Here $t = 0$ refers to the slot in which estimation takes place, $m = 0$ corresponds to pilot symbols, and $m = 1 : M_s - 1$ corresponds

to data-encoding symbols; $T = T_1 + T_2 + 1$ slots (in general, $T_1 = T_2$) are used for interpolation.

The estimate of the ith eigengain at the mth data symbol position in the current slot can be written as

$$g_i(0, m) = \mathbf{v}_i^{\mathrm{H}}(m)\mathbf{r}_i, \tag{4}$$

where $\mathbf{v}_i(m)$ is the interpolation filter and

$$\mathbf{r}_i \triangleq \frac{1}{\sqrt{E_p}\, b_p}\big[y_i(-T_1, 0), \ldots, y_i(T_2, 0)\big]^{\mathrm{T}} \tag{5}$$

contains the samples taken during pilot symbols.

The interpolation filter chosen for the numerical results shown later is the filter with brick-wall-type frequency response, which is optimum in the absence of noise; we will refer to this filter, with impulse-response tapered by a raised-cosine window [1, 2], as the SINC filter, and the corresponding estimation approach as SINC PSAM. The interpolator coefficients, given by

$$\big[\mathbf{v}(m)\big]_{t+T_1+1} = \mathrm{sinc}\left(\frac{m}{M} - t\right)\frac{\cos[\pi\beta(m/M - t)]}{1 - [2\beta(m/M - t)]^2}, \tag{6}$$

enter the FPGA-based receiver designs from Section 4. Note that channel estimation is among the most demanding receiver functions resource-wise [5].

3. FPGA HARDWARE AND SOFTWARE

3.1. FPGA system description

CMC Microsystems provided the system shown in Figure 1. The Altera DSP Development Kit Stratix Professional Edition, which comprises the Stratix EP1S80 DSP development board, is built around the Stratix EP1S80B956C6 FPGA

chip, and comes with the DSP Builder interface to the Quartus II design flow.

Quartus II provides a comprehensive design, synthesis, and analysis environment for system-on-a-programmable-chip (SoPC) applications. DSP Builder helps, create the hardware representation of the required digital signal processing functions using the MATLAB and Simulink user-friendly algorithm-development environments, for shorter design and implementation cycles. MATLAB functions and native Simulink blocks can be combined with Altera DSP Builder library blocks (see Figure 1) to create FPGA designs which can be simulated under Simulink. For automated design flow, the "signal compiler" block, which is at the core of DSP Builder, can generate hardware description language (HDL) code, and scripts for Quartus II-based synthesis and fitting from within Simulink. Furthermore, the DSP Builder "hardware in the loop" (HIL) block enables chip programming for hardware-software cosimulation.

3.2. Power usage considerations

Power loss in FPGA devices can be categorized as static and dynamic [10–13]. Static (standby) power is consumed by the chip when no input signals are exercised [10]. This loss occurs due to transistor leakage, which is frequency-independent, but highly dependent on junction temperature and transistor size. Static power has been increasing (exponentially, at processes below $0.25\,\mu$m [11]) with each finer semiconductor technology, to become the dominant loss component in current chips. This is a concern for designers of portable embedded systems which spend long intervals in standby mode [10]. Dynamic power is consumed in normal operation, due to the charging and discharging of the internal capacitive loads, and is proportional to gate output load, square of the supply voltage, clock frequency, and gate switching activity [10–13]. Although the supply voltage has decreased significantly in newer process technologies, high operating frequencies can still yield significant dynamic power losses [10]. A tight power budget may thus limit clock speed.

Line-powered embedded systems are more competitive when they require less expensive power supplies and cooling devices [10]. Designs for portable products should aim for the longest possible battery life. Moreover, devices operating at high temperatures can become unreliable, emphasizing the importance of minimizing power consumption in embedded systems. FPGA structure is judiciously designed to minimize power losses [10–12, 23]. Nonetheless, power-aware application design can also increase efficiency, for example, by using gated clock signals, and thus virtually turning off unnecessary chip sections [10, 12, 23]. Gating as close as possible to the clock source is a good practice since clock signal trees are important dynamic power consumers [12]. On the other hand, static power consumption can be reduced by adaptive distribution of available FPGA resources, as shown in Section 4.3.

For the designs described further below, we relied on Quartus II reports on resource usage, for example, the number of logic elements (LEs), chip pins, and dedicated 9×9-bit DSP blocks. Static and dynamic power losses were estimated using the Quartus II Powerplay analyzer (dynamic power was estimated for default toggle rates of 12.5%).

4. FPGA-BASED WIRELESS COMMUNICATIONS RECEIVERS

For the system shown in Figure 1, we focus on FPGA-based receiver algorithm implementation, assuming availability of digitized received signals. The transmitted signal and channel/receiver impairments, that is, noise and temporally and spatially correlated fadings, are generated in MATLAB and Simulink. Various receiver algorithms were simulated and run from the FPGA, through DSP Builder HIL. Computer simulations and the corresponding hardware/software HIL co-simulations were found to perform identically. Computations done in MATLAB or with native Simulink blocks are very precise, due to floating-point number representation. On the other hand, DSP Builder relies on fixed-point representation, which can limit the dynamic range and can introduce quantization noise.

As mentioned earlier in Table 1, we consider a scenario with Doppler spread $f_D = 100$ Hz and transmission rate $f_s = 10$ ksps, that is, normalized Doppler spread $f_m = 0.01$ Hz. PSAM with slot length $M_S = 7$ (1 pilot symbol followed by 6 information-encoding symbols) is combined with SINC interpolation over $T = 11$ slots ($T_1 = T_2 = 5$), for channel estimation as in (4)–(6). ULA with $d_n = 1$ is assumed to provide the received signals for the enhanced receivers.

4.1. Conventional, single-branch versus enhanced, multibranch MRC receivers

In this section, a conventional, single-branch receiver, and an enhanced MRC receiver, with $L = 2$ i.i.d. branches, are considered. We employ the well-established Jakes' model [14] for temporal channel fading correlation, with parameters given in Table 1. For BPSK, receiver BERs were computed for perfectly known channel (p.k.c.), as well as imperfectly known channel (i.k.c.) for SINC PSAM. We verified that BER expressions derived in [1] and the corresponding MATLAB simulation results agree closely for p.k.c. as well as for i.k.c. Then, for i.k.c., FPGA-based designs were simulated as well as hardware-software (HIL) cosimulated. For HIL cosimulation, the receiver design is compiled and then downloaded into the FPGA chip. Afterwards, received signals emulated using MATLAB are processed online by the programmed FPGA. In terms of numerical representation precision within the FPGA for the computer-generated received signal $\tilde{\mathbf{y}}$, two cases are compared next: (1) 8 bits for the integer part and 8 bits for the fractional part (denoted further as 8.8); (2) the 4.4 case. Finally, the channel gain estimation root mean-square error (RMSE) is determined from theory [4], simulations, and HIL implementations.

The upper part of Figure 2 shows the Simulink/DSP Builder design involved in channel gain estimation for one branch, while the lower part details our "SINC interpolator"
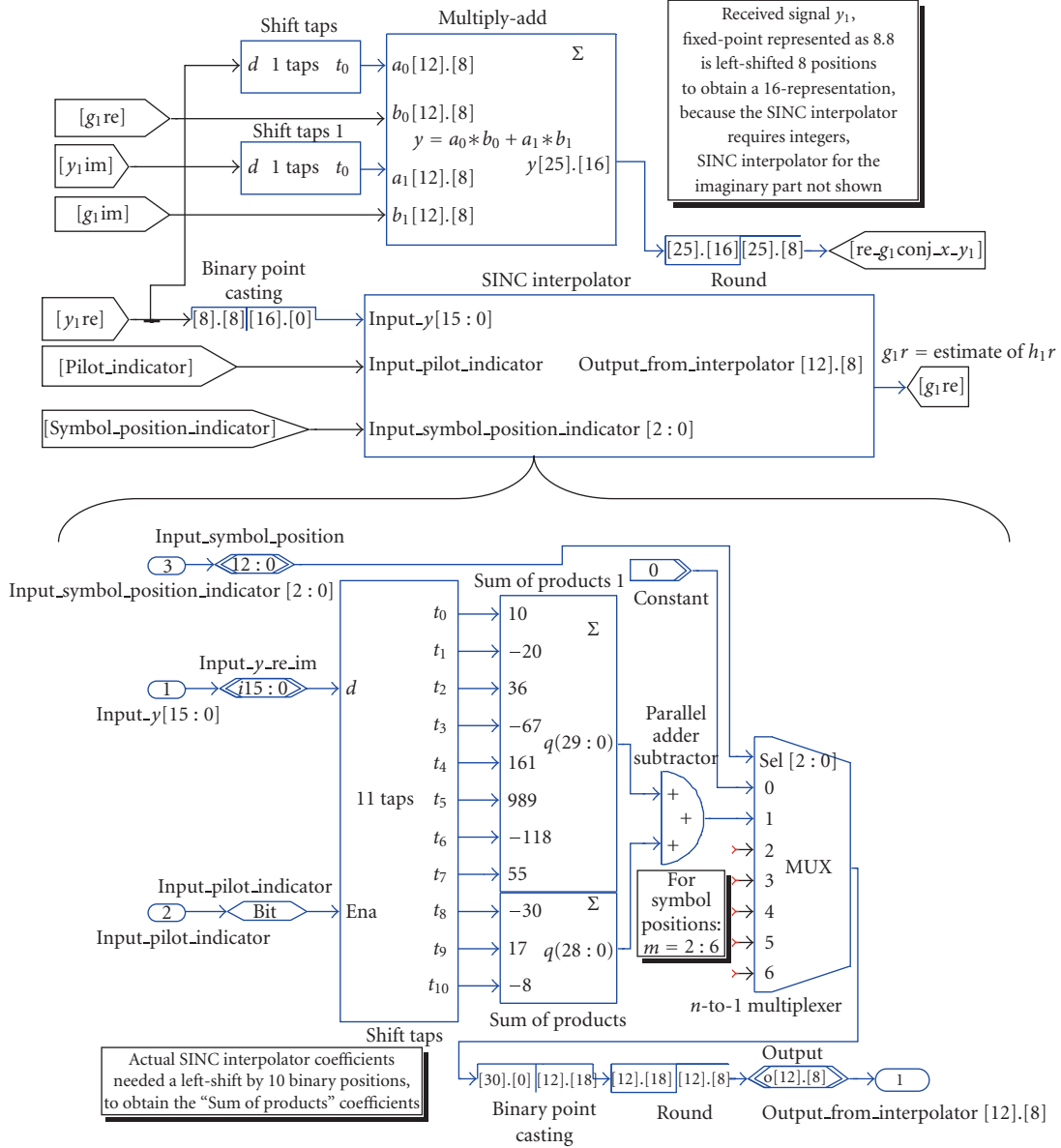
FIGURE 2: Simulink model detail with DSP Builder blocks implementing channel gain estimation (through SINC interpolation) for MRC.

design. (Symbols appear without the tilde due to Simulink editing limitations.) The upper "shift taps" DSP Builder blocks delay the received signal by $(T_1 + 1)M_s = 42$ samples, while the "multiply-add" block computes $\Re(\tilde{g}_1^* \ \tilde{y}_1)$, used as test variable for symbol detection. Since the DSP Builder blocks "sum of products" in the "SINC interpolator" design require integer input and coefficients, binary shifting of the received signal and interpolator coefficients (computed from [1, Table 1]) is required. The "SINC interpolator" "shift taps" block outputs $\Re(\tilde{\mathbf{r}}_1)$, see (5), while the "parallel Adder/Subtractor" outputs $\Re(\tilde{g}_1)$—see (4). The interpolator output is then used for combining. Notice that channel estimation can be very demanding resource-wise, especially for multibranch receivers.

The RMSE subplot in Figure 3 indicates that 4.4 and 8.8 fixed-points FPGA computation does not visibly degrade channel estimation accuracy compared to floating-point (computer) computation. Nevertheless, the lower subplots show that fixed-point computation with narrow word (i.e., poor precision, narrow dynamic range) can significantly degrade BER performance, an effect which cumulates with more branches.

Figure 3 also indicates that the performance degradation (i.e., about 3.4 dB) which occurs for a conventional receiver due to i.k.c. can be successfully compensated for an FPGA-based dual-branch MRC, due to its diversity gain. Confidence intervals for all these results are very tight, since 10 000 slots, that is, 60, 000 data symbols, were detected.
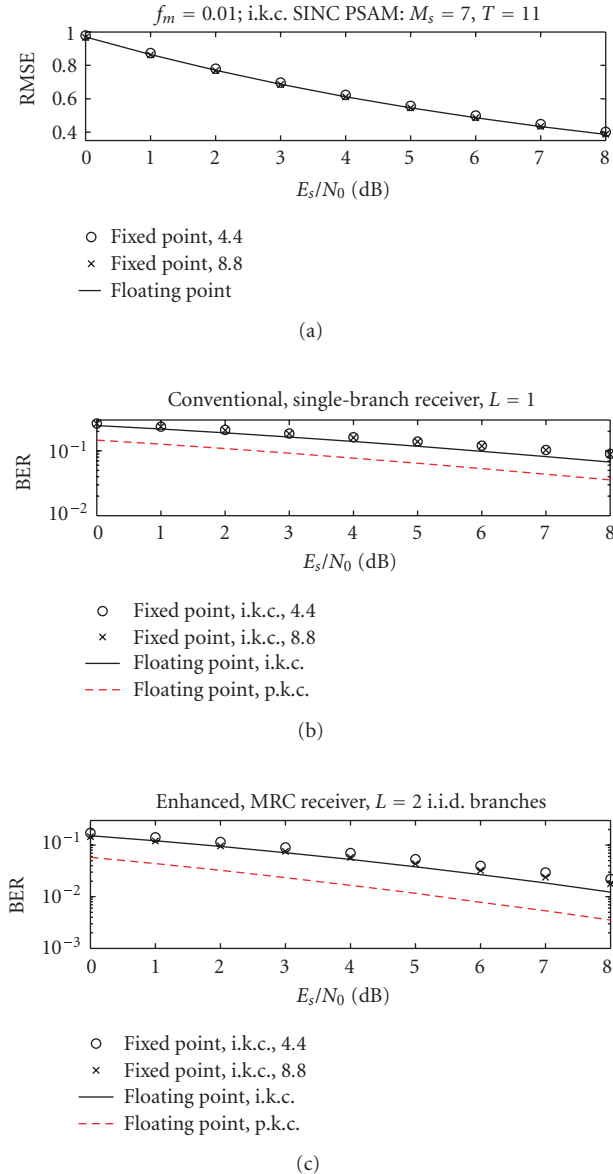
FIGURE 3: (a) RMSE for channel gain estimates. (b) and (c) Performance of the conventional, single-branch receiver, and of the dual-branch MRC receiver for various computer- and FPGA-based implementations. Fixed-point results correspond to both DSP Builder-based simulations and HIL implementations.

TABLE 2: Resource usage for 8.8 implementations of MRC, BF, and adaptive MREC, for up to $L = 4$ branches.

Method	LEs (79 040)	Pins (692)	DSP (176)	CF (MHz)	DP (mW)
MRC $L = 1$	13,227 16.73%	43 6.21%	16 9.09%	41.06	69.35
MRC $L = 2$	26,478 33.49%	83 11.99%	32 18.18%	38.56	119.67
MRC $L = 3$	39,731 50.27%	123 17.77%	48 27.27%	38.35	169.78
MRC $L = 4$	55,983 70.83%	167 24.13%	64 36.36%	36.74	221.62
BF $L = 4$	13,457 17.02%	259 37.43%	48 27.27%	40.57	74.95
BVTC MREC $L = 4, N = 1$	13,458 17.02%	262 37.86%	48 27.27%	41.15	74.95
BVTC MREC $L = 4, N = 2$	26,940 34.08%	358 51.73%	96 54.54%	39.73	130.89
BVTC MREC $L = 4, N = 3$	40,423 51.14%	454 65.60%	144 81.81%	39.09	186.64
BVTC MREC $L = 4, N = 4$	55,847 70.66%	550 79.48%	176 100%	38.82	244.64
EVTC MREC $L = 4, N = 1$	13,561 17.16%	424 61.27%	48 27.27%	41.09	75.67
EVTC MREC $L = 4, N = 2$	27,372 34.63%	524 75.72%	96 54.54%	39.14	132.95
EVTC MREC $L = 4, N = 3$	40,983 51.85%	624 90.17%	144 81.81%	35.43	189.23

For designs shown hereafter, we settled for an 8.8-representation, since it was found to offer a fair compromise between representation accuracy/dynamic range (i.e., receiver performance) and FPGA resource utilization. Furthermore, we instructed DSP Builder to allocate hard-wired DSP circuitry embedded into the reconfigurable FPGA fabric, which yields effective and efficient chip utilization [7]. Then, Quartus II reports on FPGA resource usage, maximum allowable clock frequency (CF), and dynamic power (DP) usage, as shown in Table 2. Estimated static power loss is 1.395 W. Note that for the BER advantage shown in Figure 3 over the conventional receiver, dual-branch MRC nearly doubles resource requirements and dynamic power loss. Since the MRC performance gradient diminishes with increasing number of branches [4], implementation/operational costs can be minimized either with tightly matched chips, or through clock gating of excess resources.

In the above MRC receiver design, channel gains on different branches were considered statistically independent, for simplicity. However, this is rarely the case in practice [16]. Although scattering is richer around the mobile than around the base station, mobile antenna array size limitations can still lead to large interbranch correlation, that is, scarce diversity gain availability. Then, adaptive MREC [3, 4] may provide more suitable tradeoffs between performance and resource/power utilization, as shown next.

4.2. Enhanced MREC receiver designs: the case of a single user processed per FPGA chip

We extended the previously discussed FPGA-based MRC receiver design to support $L = 4$ branches, and also designed the BF, and the BVTC adaptive MREC receivers. See Table 2
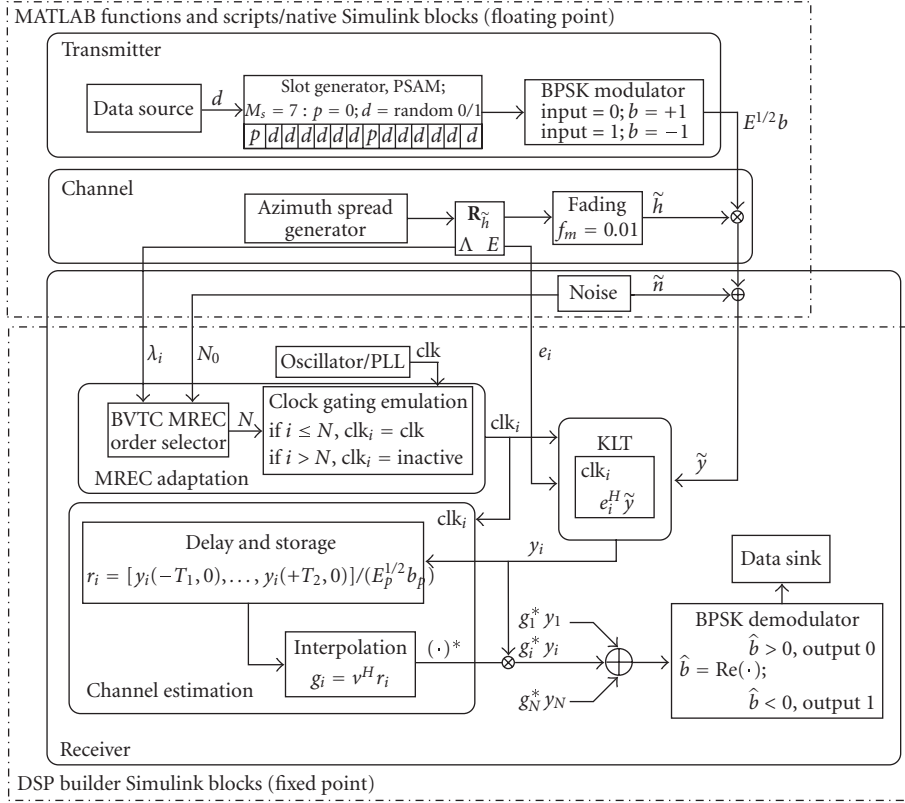
FIGURE 4: Transmitter, channel, and FPGA-based BVTC MREC receiver diagram.

for the resource and power usage report. Note that a stand-alone BF implementation takes about as many resources as order-1 MREC takes in the BVTC MREC implementation since these two designs are almost identical. Furthermore, MRC can be obtained from an MREC design by bypassing the KLT. Thus, an MREC design can easily be reconfigured (even during operation, on the fly) to implement BF or MRC instead. Implementation details are provided in Figure 4, for the case when the receiver implements BVTC adaptive MREC.

For resource/power usage and performance evaluation, we model a typical urban scenario for realistic channel conditions from the base station perspective [16], and apply the conventional and enhanced receiver combining algorithms (after estimating channel gains and eigengains as in Section 2.6) to detect the transmitted symbols. Using MATLAB and Simulink, the actual log-normal distributed, time-correlated azimuth spread is simulated and then employed to compute the spatial correlation matrix, for realistic Laplacian power azimuth spectrum (p.a.s.) [16]—see Figure 4. In an actual embedded receiver, the channel correlation matrix and its eigenvalue decomposition could be updated by a processor (e.g., Altera's soft-core FPGA-based Nios II). We selected a correlation update period of 0.14 second (denoted further as a *frame*, corresponding to a distance of roughly 2.3 m traveled by the mobile) since the azimuth spread remains relatively constant over this interval [16], providing

the processor with sufficient time and uncorrelated samples for eigenstructure updating [3, 4]. The computed correlation matrix $\mathbf{R}_{\tilde{h}}$ inputs a customized Simulink "multipath Rayleigh fading channel" block to simulate $L = 4$ correlated branches.

The top subplot in Figure 5 depicts an azimuth spread sequence generated using the model described in Section 2.2. The predominantly small-to-moderate azimuth spread values indicate that we should often expect significant spatial correlation [1, 3], that is, small available diversity gain. Performance enhancement can then arise from BF antenna gain. Occasionally however, the azimuth spread can also become fairly large, but then the available diversity gain cannot benefit BF performance. On the other hand, significant diversity gain may be available too infrequently to justify permanent use of an MRC receiver. As we will see, an FPGA-based MREC receiver can provide, for a channel with slowly varying statistics, flexibility that yields affordable performance.

The main benefit of an FPGA-based BVTC adaptive MREC receiver is that unnecessary eigenbranches can be virtually turned off using the clock gating technique [12] to reduce dynamic power loss, while necessary eigenbranches can be implemented to run in parallel, for high speed. Exempting weak eigenbranches can also benefit performance [1]. Furthermore, as mentioned earlier, an MREC implementation can easily be reduced to standalone BF or MRC implementations, if required, either at system setup or during operation.
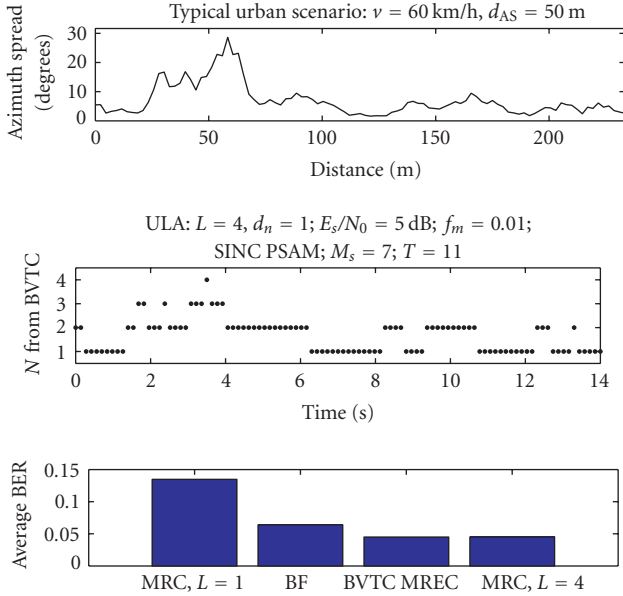
FIGURE 5: Azimuth spread, MREC order selected with the BVTC, and BER performance (averaging over trial) for BF, MRC, and BVTC MREC.

Altera documentation states that clock gating is available only through lower-level (Quartus II) design. Therefore, clock gating was only emulated in DSP Builder, for the BVTC MREC implementation shown in Figure 4. First, nonadaptive MREC designs with $N = 1 : 4$ eigenbranches were compiled to determine their resource usage (shown in Table 2). Then, after each eigenstructure update during the BVTC MREC simulation, we stored the selected MREC orders and disconnected unused eigenbranches from the active structure. Finally, average resource usage was computed. Figure 5 shows in the middle subplot the MREC order selected adaptively using the BVTC, and in the lower subplot the BER averaged over the trial. Notice that for $L = 4$, MRC and BVTC adaptive MREC slightly outperform BF, and greatly outperform the single-branch receiver.

For the same typical urban scenario and system parameters, Figure 6 shows resource usage, in percentage points of the total available, and dynamic power consumption, averaged over 8 trials. In each trial, the azimuth spread samples are correlated, as described in Section 2.2, but the azimuth spread sequences are independent between trials. Note that BF and BVTC MREC require a significantly smaller share of the FPGA programmable fabric, that is, LEs, compared to MRC (for $L = 4$), but more dedicated DSP blocks, due to KLT. The upper-right subplot appears to imply more chip pins demand for BF and MRC, because a MATLAB/Simulink-computed eigenvector matrix $\mathbf{E}_N$ inputs the FPGA. Nevertheless, eigenstructure updating is possible with a soft processor, from within the FPGA.

Figure 7 shows performance and total (dynamic + static) power used by a cellular operator's large network of base stations similar to the one described in [11]. The single-branch receiver consumes least but performs poorly. For performance similar to BF and BVTC MREC, MRC (with $L = 4$) doubles the dynamic power loss (see also Figure 6(d)). Thus, BF and BVTC MREC appear to provide a better tradeoff. Recall however that a compact ULA with $d_n = 1$ is considered. For larger interelement distances (feasible at base stations), MREC with more than one eigenbranch can significantly outperform BF [4].

Note that significant branch correlation can occur even at mobile stations, due to limited antenna spacing, so that an FPGA-based BVTC MREC implementation employing clock gating can efficiently achieve near-optimum performance.

Notice from Figure 5(b) that, frequently, only one or two (out of the four implemented) eigenbranches were actually employed for MREC for that particular azimuth spread sequence. Similar results were obtained in other trials for independent azimuth spread sequences. This suggests that adaptive FPGA chip resource allocation among several active users may significantly increase base station user processing capacity, or, equivalently, reduce the required number of FPGA chips per base station, lowering both hardware cost and static power losses. A possible path towards such implementations is described next.

4.3. Enhanced MREC receiver designs: the case of two users processed per FPGA chip

EVTC-based adaptive MREC, described in Section 2.5, can provide more consistent use of the FPGA chip, compared to BVTC MREC. We propose to efficiently exploit a total of 3 eigenbranch processing modules, which fit into our FPGA, to process concurrently the signals received with $L = 4$ branches from two mobiles (without interference). Rather than permanently allotting chip processing resources to a certain user (which may or may not need to use them, depending on channel conditions and required performance), herein we will adaptively deploy these resources to simultaneously detect the symbols transmitted from two mobiles.

Resource usage information for EVTC MREC when $N = 1 : 3$ eigenbranches are selected can be found in Table 2. Note that the BVTC and EVTC MREC implementations differ significantly only in the required number of chip pins. The larger number of pins required for EVTC MREC (to input the received signals from two mobiles) limits to 3 the possible number of implemented eigenbranches. Larger N_e leads to unsuccessful compilation. Mutually independent azimuth spread sequences for the signals arriving at the base station from the two mobile stations were simulated, as shown in the top subplots of Figure 8. The MREC orders selected with the EVTC for each of the users are shown in the middle subplots. The lower subplots indicate that EVTC MREC can perform remarkably close to the enhanced receivers discussed previously.

Figure 9(a) indicates that our FPGA would not fit concurrent four-branch MRC implementations for the two users. On the other hand, the successfully compiled two-user EVTC MREC implementation with $N_e = 3$ requires about half of the dynamic power consumed by MRC, for similar
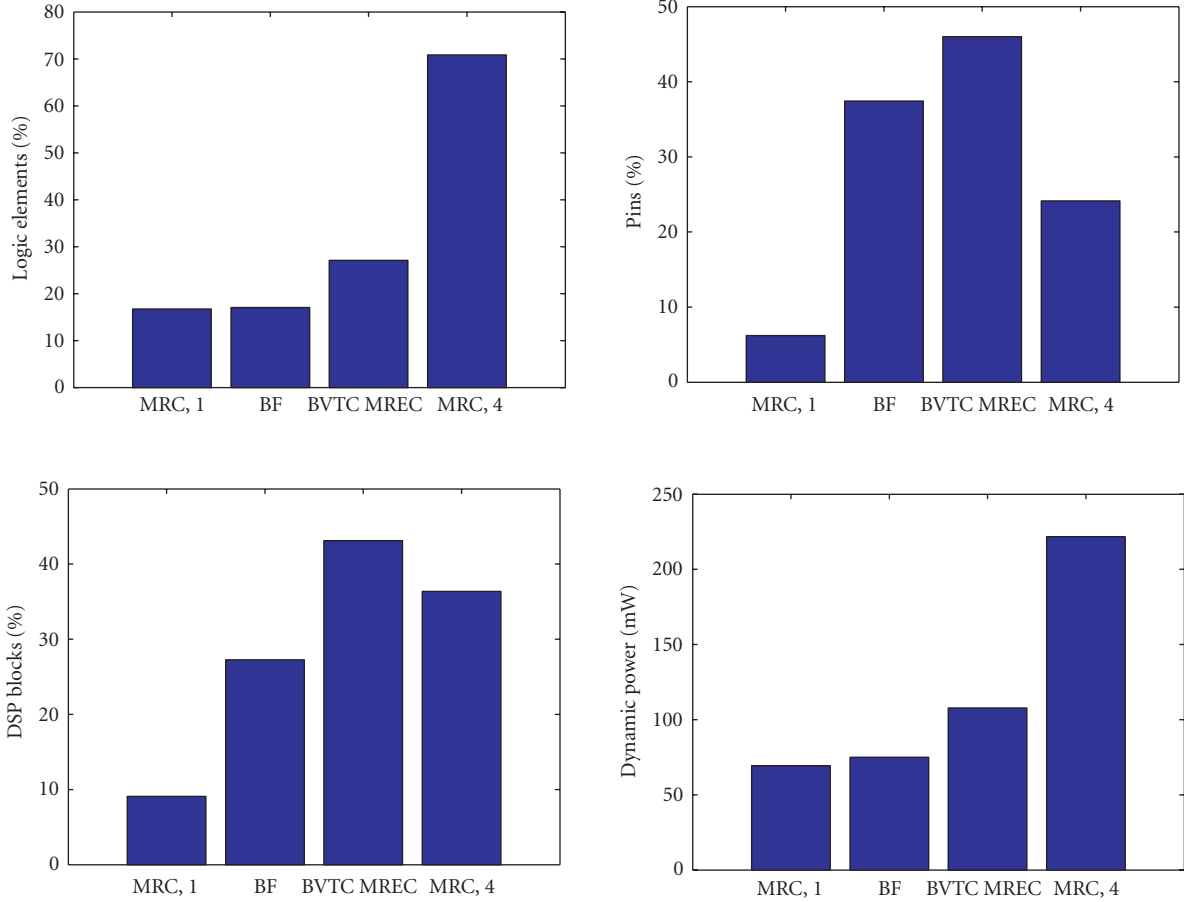
FIGURE 6: Average resource and dynamic power usage for BF, BVTC MREC, and MRC, over 8 trials with mutually independent azimuth spread sequences.

performance. Furthermore, since EVTC MREC allows for effective concurrent processing of two users on a single FPGA, it yields a twofold reduction in static power consumption or a doubling of the base station user processing capacity. Thus, both implementation and operational costs can be drastically reduced with EVTC MREC.

Ideally, an FPGA-based embedded base station receiver would comprise: (1) a number of FPGAs programmed for KLT, channel estimation, signal combining, and symbol detection; (2) an embedded processor monitoring each user's channel conditions (i.e., eigenmodes). At the beginning of each frame, the embedded processor browses a user hierarchy, and allocates the FPGA resources so as to achieve desired performance for minimum resource/power consumption [3, 4]. Thus, it is possible that for a certain period, several users whose respective received signals are highly correlated will share the resources of a single FPGA because none of them will demand a large number of eigenbranches. If the azimuth spread for one of these users later widens significantly (yielding more available diversity gain) or if its SNR degrades (while a certain steady performance level is imposed), a larger share of the FPGA resources can be al-

located accordingly. An FPGA-based embedded system for a performance- and a power-aware antenna array receivers can thus be flexibly implemented.

5. CONCLUSIONS

We have described and implemented adaptive techniques that enhance the performance and reduce the power consumption for Altera-FPGA-based embedded wireless receivers. We found that smart antenna array receiver algorithms, for example, beamforming (BF) and maximal-ratio combining (MRC), outperform the conventional, single-branch receiver, but the performance gain may not always justify the additional implementation and operational costs. Tracking the slowly varying dominant channel eigenmodes, and using maximal-ratio eigencombining (MREC) is found to benefit more than BF and MRC from the parallelism and flexibility of FPGA-based implementation. For similar performance, a twofold increase in user processing capacity or decrease in power consumption is found possible over MRC, for a typical urban scenario and 4 receiving antennas. Adaptive MREC outperforms BF, for slightly
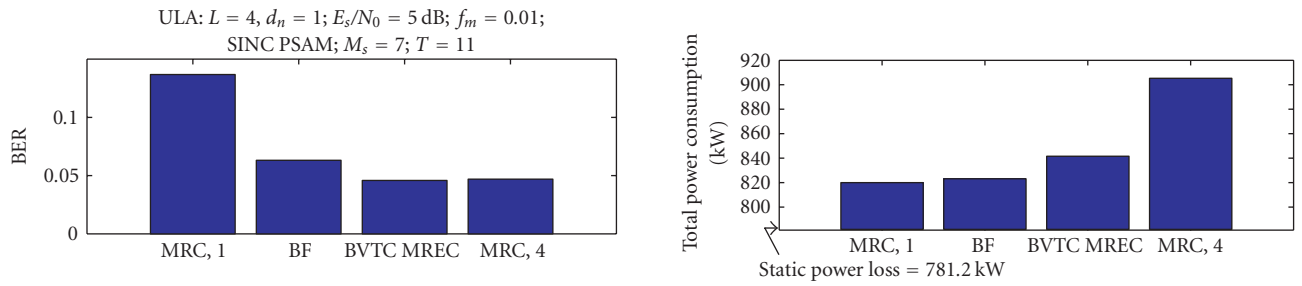
FIGURE 7: Average BER and total (static + dynamic) power consumption for BF, BVTC MREC, and MRC, over 8 independent azimuth spread trials.
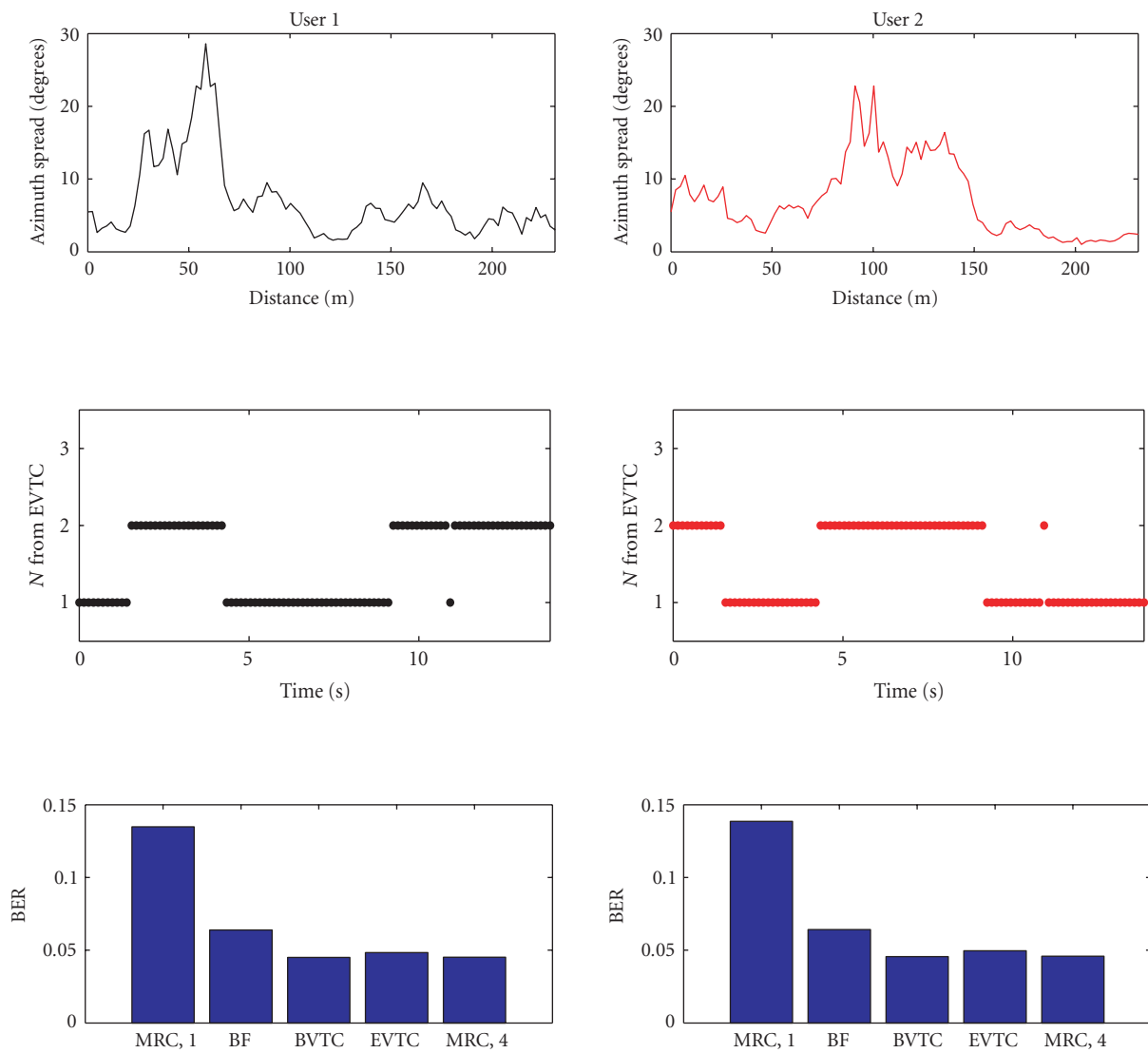

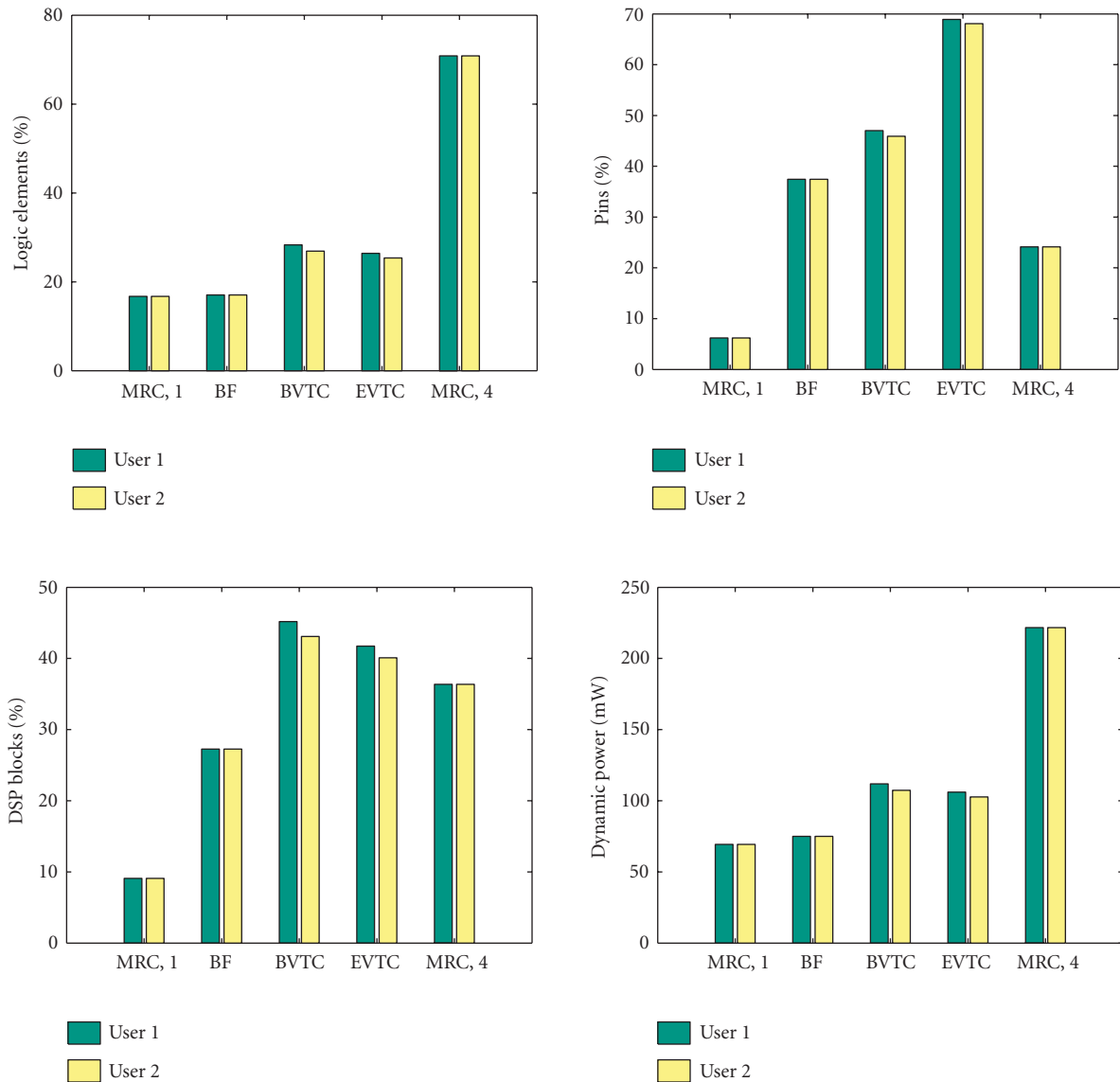FIGURE 8: Azimuth spread, EVTC MREC order, and average BER performance, for two users.

FIGURE 9: Resource usage (in percentage of total available) and dynamic power consumption for all discussed receiver algorithms, for two independent users.

higher resource consumption. FPGA flexibility and wide range of on-chip resources can thus yield very efficient embedded implementations of adaptive receivers for current and future generations of wireless communications systems.

ACKNOWLEDGMENT

REFERENCES

[1] C. Siriteanu and S. D. Blostein, "Maximal-ratio eigen-combining: a performance analysis," *Canadian Journal of Electrical and Computer Engineering*, vol. 29, no. 1, pp. 15–22, 2004.

[2] C. Siriteanu and S. D. Blostein, "Smart antenna arrays for correlated and imperfectly-estimated Rayleigh fading channels," in *Proceedings of IEEE International Conference on Communications (ICC '04)*, vol. 5, pp. 2757–2761, Paris, France, June 2004.

[3] C. Siriteanu and S. D. Blostein, "Maximal-ratio eigen-combining for smarter antenna arrays," to appear in *IEEE Transactions on Wireless Communication*.

[4] C. Siriteanu, "Maximal-ratio eigen-combining for smarter antenna arrays," Ph.D. dissertation, Queen's University, Kingston, ON, Canada, September 2006.

[5] M. Guillaud, A. Burg, M. Rupp, E. Beck, and S. Das, "Rapid prototyping design of a 4×4 BLAST-over-UMTS system," in *Proceedings of Conference Record of the 35th Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 1256–1260, Pacific Grove, Calif, USA, November 2001.

[6] B. L. Hutchings and B. E. Nelson, "Gigaop DSP on FPGA," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '01)*, vol. 2, pp. 885–888, Salt Lake City, Utah, USA, May 2001.

[7] C. Ebeling, C. Fisher, G. Xing, M. Shen, and H. Liu, "Implementing an OFDM receiver on the RaPiD reconfigurable architecture," *IEEE Transactions on Computers*, vol. 53, no. 11, pp. 1436–1448, 2004.

[8] F. L. Vargas, R. D. R. Fagundes, and D. Barros Jr., "A FPGA-based Viterbi algorithm implementation for speech recognition systems," in *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '01)*, vol. 2, pp. 1217–1220, Salt Lake City, Utah, USA, May 2001.

[9] T. S. T. Mak and K. P. Lam, "Embedded computation of maximum-likelihood phylogeny inference using platform FPGA," in *Proceedings of IEEE Computational Systems Bioinformatics Conference (CSB '04)*, pp. 512–514, Stanford, Calif, USA, August 2004.

[10] S. Sharp, "Conquering the three challenges of power consumption," *XCell Journal*, no. 53, 2005.

[11] A. Telikepalli, "Performance vs. power: getting the best of both worlds," *XCell Journal*, no. 54, 2005.

[12] L. Benini, G. De Micheli, and E. Macii, "Designing low-power circuits: practical recipes," *IEEE Circuits and Systems Magazine*, vol. 1, no. 1, pp. 6–25, 2001.

[13] A. Yang, "Design techniques to reduce power consumption," *XCell Journal*, no. 54, 2005.

[14] W. C. Jakes, Ed., *Microwave Mobile Communications*, John Wiley & Sons, New York, NY, USA, 1974.

[15] J. Proakis, *Digital Communications*, McGraw-Hill, Boston, Mass, USA, 4th edition, 2001.

[16] A. Algans, K. I. Pedersen, and P. E. Mogensen, "Experimental analysis of the joint statistical properties of azimuth spread, delay spread, and shadow fading," *IEEE Journal on Selected Areas in Communications*, vol. 20, no. 3, pp. 523–531, 2002.

[17] T. S. Rappaport, *Wireless Communications. Principles and Practice*, Prentice-Hall, Upper Saddle River, NJ, USA, 1996.

[18] J. K. Cavers, "An analysis of pilot symbol assisted modulation for Rayleigh fading channels," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 4, pp. 686–693, 1991.

[19] M. K. Simon and M.-S. Alouini, *Digital Communication over Fading Channels: A Unified Approach to Performance Analysis*, John Wiley & Sons, New York, NY, USA, 2000.

[20] C. Brunner, W. Utschick, and J. A. Nossek, "Exploiting the short-term and long-term channel properties in space and time: eigenbeamforming concepts for the BS in WCDMA," *European Transactions on Telecommunications*, vol. 12, no. 5, pp. 365–378, 2001, special issue on Smart Antennas, http://www.chrisbrunner.org.

[21] J. Jelitto and G. Fettweis, "Reduced dimension space-time processing for multi-antenna wireless systems," *IEEE Wireless Communications*, vol. 9, no. 6, pp. 18–25, 2002.

[22] F. Dietrich and W. Utschick, "On the effective spatio-temporal rank of wireless communication channels," in *Proceedings of the 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 5, pp. 1982–1986, Lisbon, Portugal, September 2002.

[23] T. Simunic, L. Benini, and G. De Micheli, "Energy-efficient design of battery-powered embedded systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 9, no. 1, pp. 15–28, 2001.

Constantin Siriteanu was born in Sibiu, Romania, in 1972. He received his B.S. and M.S. degrees in electrical engineering, from "Gheorghe Asachi" Technical University, Iasi, Romania, in 1995 and 1996, respectively. Between 1995 and 1997, he was a Part-Time Engineer with the Research Institute for Automation, Iasi, Romania, working on data transmission over power lines. Between 1996 and 1998, he was a Research Assistant with the Department of Automatic Control and Computer Science, "Gheorghe Asachi" Technical University, Iasi, Romania, working on digital control systems. Since 1998, he has been a Research Assistant with the Department of Electrical and Computer Engineering, Queen's University, Kingston, Canada. His Ph.D. research has been in adaptive signal processing for smart antenna array receivers, with a focus on performance-complexity tradeoffs based on channel statistics. Between 2004 and 2006, he has also been a Course Instructor for the 4th year undergraduate Electrical/Computer Engineering Project Course at Queen's University.

Steven D. Blostein received his B.S. degree in electrical engineering from Cornell University, Ithaca, NY, in 1983, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign, in 1985 and 1988, respectively. He has been on the faculty at Queen's University since 1988 and he currently holds the position of Professor and Head of the Department of Electrical and Computer Engineering. From 1999 to 2003, he was the Leader of the Multirate Wireless Data Access Major Project sponsored by the Canadian Institute for Telecommunications Research. He has also been a Consultant to industry and government in the areas of image compression and target tracking, and was a Visiting Associate Professor in the Department of Electrical Engineering at McGill University in 1995. His current interests lie in the application of signal processing to wireless communications systems, including smart antennas, MIMO systems, and space-time frequency processing for MIMO-OFDM systems. He served as Chair of IEEE Kingston Section in 1993–1994, Chair of the Biennial Symposium on Communications in 2000 and 2006, and as Associate Editor for IEEE Transactions on Image Processing from 1996 to 2000.

James Millar has B.S. degree in electrical engineering and computer science from the University of Western Ontario. He received his M.S. degree from the University of Manitoba. He previously worked with Nortel Networks providing carrier network design and planning services, and was an Instructor of electronics at St. Lawrence College in Kingston, Canada. Currently he is working with CMC Microsystems in Kingston, Canada, as a System-On-Chip Design Engineer with an interest in system-level design for microsystems integration.