A 4×4 FPGA-based wireless testbed for LTE applications

Dale Bates, Soren Henriksen, Brett Ninness, Steven R. Weller
School of Electrical Engineering and Computer Science
The University of Newcastle
NSW, Australia, 2308
Email: {dale.bates, soren.henriksen, brett.ninness, steven.weller}@newcastle.edu.au
Web: http://sigpromu.org

***Abstract*—We present early results from a 4×4 2.4GHz ISM band multiple-input multiple-output (MIMO) testbed developed at the University of Newcastle for over-the-air evaluation of MIMO algorithms. To provide maximum flexibility for development of 3GPP Long Term Evolution (LTE) algorithms, the testbed is designed around a field-programmable gate array (FPGA). The testbed features a simple GUI interface for testbed control as well as a MATLAB interface allowing real channel measurements to be used in algorithm development and evaluation. These features are illustrated here with experimental results from the implementation of Alamouti Space-Time Coding with 64-QAM.**

I. Introduction

There has recently been strong research activity and interest in the area of testbeds for evaluating and developing MIMO wireless systems. Depending on the research requirements, MIMO testbeds can generally be characterised into three types, software-defined, high performance realtime based, and FPGA or digital signal processor (DSP) based. The first type are used to simply transmit data that has been generated off-line, then process the results off-line as well. For example, the Hydra testbed from UT Austin [1] makes use of low cost/complexity hardware such as Ettus kits [2]. Here the focus is on developing software to implement different algorithms of interest to be tested across the air and can be considered as a Software Defined Radio (SDR).

A common feature of this type of testbed is the ability to utilise MATLAB or a similar environment to generate the signal to be sent, transmit across the air via the testbed which returns received samples to be processed off-line. Often the testbeds are synchronised with a common clock to simplify the frame and sampling timing between testbeds. This type of testbed provides a valuable tool for simplifying the use of real channels in research.

However, when the intended testbed use includes developing hardware algorithm implementations, then inclusion of high performance FPGA or DSP cores is required to facilitate real-time processing. There are many examples of such systems [3]–[7] which often utilise multiple FPGA's and DSP's. For example, the Vienna MIMO Testbed [8], [9] supports 4×4 MIMO systems using the 2.45GHz band, and is made using Commercial off-the-shelf (COTS) components. This testbed is controlled using MATLAB scripts to send and receive data.

Finally, there are testbeds that offer modest real-time hardware capabilities, being in the middle of the two groups already discussed. Rice University [10] utilise COTS boards to build an FPGA based testbed capable of either transmit to receive combinations of 6×1 and 3×2 through the use of a channel emulator. The use of a channel emulator does offer an advantage over a real channel in that repeatable testing over known channel parameters is possible. A similar COTS testbed has also been developed at the University of Alberta [11]. This testbed also had the added requirement of being portable to allow channel measurement experiments to be carried out with relative ease. Finally, the University of Queensland testbed [12], [13] employs FPGA development kits to implement 2×2 MIMO systems.

At both ends of this spectrum of testbed designs, a shared problem is that MIMO processing requires significant amounts of data to be transferred between the successive receiver blocks.

In this paper we profile the University of Newcastle testbed, designed as a low cost platform for experimenting with real physical channels and real-time hardware implementation of algorithms for MIMO wireless systems. The testbed is aimed at the physical (PHY) and medium access control (MAC) layers of a typical system and, is designed for flexibility. In particular, the testbed is designed to support application specific integrated circuit (ASIC) development and testing.

II. System Architecture

The MIMO testbed that has been designed and built is shown in Fig. 1 and is based on a main baseband board capable of controlling up to 4 RF boards. Combining two of these complete systems provides capability for any MIMO antenna configuration scenario up to 4×4. The digital board provides an interface to a host PC via USB or Ethernet connection, and an FPGA to process baseband signals to ADC's and from DACs on the radio modules. The baseband design is shown in Fig. 2 and is centered around an Altera Stratix II FPGA [14].

One of the main concerns with a MIMO testbed design is the amount of data that must be transferred between the baseband and radios. For example, using 4 receivers with 14-bit ADCs at 40MHz sampling rates requires a transfer rate of 4.48Gbps. By using the high I/O capability of an FPGA these

978-1-4244-2644-7/08/$25.00 ©2008 IEEE

parallel I/O requirements necessary to support the required data transfer to each radio module are achieved. To allow the capture of long frame lengths, additional storage is provided to the FPGA using an external static random access memory (SRAM). This memory can store 144Mbits of I/Q data, which is equivalent to a transmission block length of 15.7ms using 12 bit I/Q data. This frame length is adequate for testing short burst transmissions. A common support bus provides host interface connections via USB 2 or Ethernet as well as Flash memory and program memory for the Nios II [15] processor core running on the FPGA.

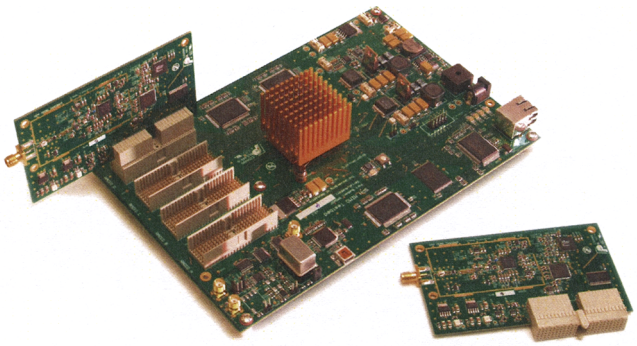

Fig. 1. Photograph of testbed with a single radio module fitted.

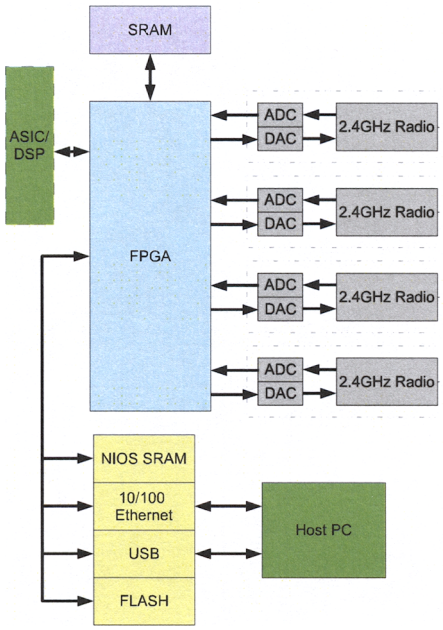

Fig. 2. Testbed block diagram

The FPGA is the main processing component, under the control of a Nios II core. This core runs custom software that provides access to and from the host PC to control the testbed. The core interprets host commands and configures the radio modules via control registers. Once the testbed is configured, the host can then begin a test by starting the signal processing block on the FPGA. In the simplest configuration, this block will simply send I/Q data from the host to the transmitters and capture received signals for offline processing by the host. In more complicated scenarios the user can implement algorithms to process baseband signals in a realtime environment on the FPGA.

Additional computational power can be added via a high speed 32-bit bus. The testbed has been designed to enable the direct connection of a Texas Instruments DSP development kit via the Universal Host-Port Interface (UHPI). This bus can also be used to interface to custom ASICs to provide a versatile testing environment. The expansion bus is provided to address the finite resources available in the FPGA, as complex MIMO receiver algorithms require more resources than provided on the baseband board.

Multiple baseband boards may also be combined to generate MIMO test systems with higher than 4×4 capability through common clocking schemes and a small number of control signals.

The radio frequency (RF) modules developed use the 2.4GHz to 2.5GHz ISM band, and provide a transmission bandwidth up to 40MHz via the Maxim MAX2829 transceiver chip. These implement a direct conversion process, removing the need for IF stage design. Moreover, the MAX2829 chips are designed for use in MIMO applications, with a mode to enable clock synchronisation between multiple transceiver chips. Additionally, a power amplifier and T/R switch are incorporated. Each RF board contains I/Q ADC and DAC's capable of 12 or 14 bit operation and connect directly to the baseband board FPGA. The ADC can be operated at up to 80Msps whilst the DACs are rated at 500MHz, allowing more flexibility for pulse shaping.

All RF boards use a common 40MHz clock generated on the baseband board that can be adjusted using firmware to provide frequency offset corrections. The ADC/DAC can run off the same clock as the transceivers, or a user defined clock by configuring the FPGA PLL's. An external clock source may also be used via a SMA connector if desired. Most configurable options for the RF module components are carried out through serial interfaces, with some direct I/O connections from the FPGA.

The testbed also allows for a variety of clock source schemes. A typical scenario could be to emulate a Base Station that has very accurate Oven Controlled Crystal Oscillators (OXCO - typically $\leq$ 50ppb) in one testbed and low cost crystal oscillators emulating User Equipment to provide typical frequency stability characteristics. Carrier frequency offset (CFO) estimation should also be performed in this scenario, which may be implemented in the FPGA. A transmit and receive testbed may alternatively be synchronised to the same clock using the external SMA connections provided.

The testbed offers both USB2 and 100Mbps Ethernet connectivity options. USB is the primary interface as it offers a

TABLE I
SUMMARY OF BASEBAND BOARD FEATURES.

FPGA	Altera Stratix II EP2S60 2.5Mb Memory 60,440 Logic Elements 9-bit DSPs
Baseband Memory	144Mb (2M x 72-bit)
Interfaces	USB2, 10/100 Ethernet, 32-bit Expansion Bus
Oscillator Stability	0.1ppm @ 40Mhz

TABLE II
SUMMARY OF RADIO MODULE FEATURES.

Frequency Band	2.4-2.5GHz
Bandwidth	40MHz
Transmit Power	22dBm
DAC	14 bit, 500Msps
ADC	14 bit, 80Msps

high transfer speed with relatively low resource demand for implementing the stack. The PLX NET2272 device is used, and the USB stack is implemented on the NIOS II processor core on the FPGA.

With USB, a transfer rate of approximately 20MBytes/s may be reached. While this is insufficient for real-time streaming of multiple 40Msps data streams direct from the ADCs or to the DACs, it is suitable for downsampled values or higher layer symbol streams.

Due to the computational limitations of the processor core, an ethernet stack implemented in this way will only reach about 2MBytes/s, but this is adequate for many testing scenarios.

While these interfaces are designed to support data transfers to/from the testbed when delivering realtime hardware implementations of MIMO algorithms, the board can also support offline processing. In this case, I/Q data to be transmitted, that has been computed on a separate host computer, is stored in the testbed RAM prior to the transmission. At the receiver frame synchronisation is used to detect the transmitted packet, and the received data from the DACs is again stored into testbed RAM for later retrieval by the host computer.

The testbed design was completed by 3 engineers responsible for development of:

- Baseband and radio modules.
- PCB layout.
- VHDL firmware to provide full control of testbed.
- NIOS core software.
- MATLAB and PC interface.

III. INITIAL RESULTS

Initial results from the testbed are presented here demonstrating both SISO and 2×2 MIMO wireless systems. The test setup used is shown in Fig. 3. A custom frame format incorporates a synchronisation signal, followed by length N samples of random symbols. Frame synchronisation is performed using the ZC sequence

$$S_M(n) = \begin{cases} \exp\left(-j\frac{\pi Mn(n+1)}{63}\right) & n = 0, 1, ..., 30 \\ \exp\left(-j\frac{\pi M(n+1)(n+2)}{63}\right) & n = 31, 32, ..., 61 \end{cases} \quad (1)$$

with root index $M = 25$ as defined in 3GPP LTE standards [16]. Detection of the synchronisation sequence is achieved using a cross-correlator implemented in the FPGA.

In the results shown here, both testbeds are running the same VHDL firmware and NIOS software, and are setup for transmit or receive functionality through configuration options. All tests use a 40MHz sampling rate for both the transmitter DAC and receiver ADC. Apart from the synchronisation the data received is currently being processed on the host PC of the receiver testbed.

A custom interface application has been developed to allow testing and configuration of all aspects of the baseband board and each radio module. Options to configure radio, DAC and ADC registers are available such as centre frequency, I/Q balance adjustments, bandwidth and power amplifier linearity. Overall control of the testbed is performed through a top level control panel which defines such aspects as whether a transmit or receive operation is to be carried out, what frame length is to be sent or captured, etc.

The transmit testbed may be configured using the GUI to generate data to be transmitted and add the synchronisation signal to the beginning of each frame. The data is then downloaded and stored in the SRAM ready for transmission. On the receiver it is configured as required and waits until a frame synchronisation signal is detected. Once synchronisation is detected the I/Q samples from the receiver radio modules are stored in SRAM for retrieval by the host PC once the frame has ended. This mode of operation represents the off-line operation typical of software-defined testbeds.

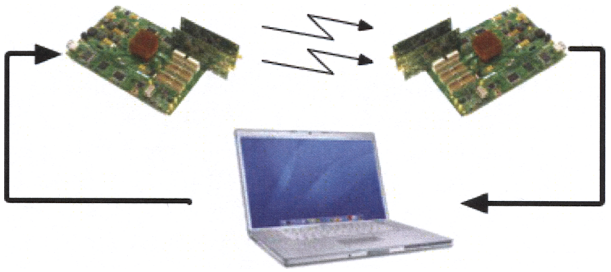

Fig. 3. Testbed setup used for results.

The first test result demonstrates a basic SISO system transmitting random 16QAM data at a symbol rate of 13.33Msps. The received constellation is shown in Fig. 4 where no symbol phase correction has been carried out.

Next the results using a 2×2 Alamouti [17] scheme with a symbol rate of 10Msps and 16-QAM are shown in Fig. 5. The

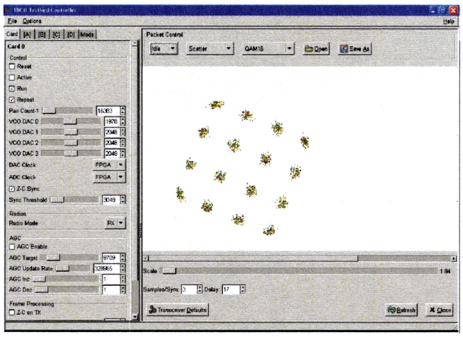

Fig. 4. Received 16-QAM constellation

TABLE III
RESOURCE USE FOR NIOS II CORE, CONTROL LOGIC AND SYNCHRONISATION LOGIC.

Resource type	Synchronisation	NIOS II core	Total Available
ALM	4653 (17%)	3184 (12%)	27104
DSP Blocks	124 (97%)	0	128
Memory bits	15672 (1.1%)	46848 (3.4%)	1369728

constellation on the left is derived using data from only the first receiver. The middle constellation is from the second receiver, while the third plot is the combined Alamouti constellation. Below the constellations are BER for each receiver (blue = Rx1, red = Rx2) and the overall BER (purple bar) with the scale from 10^{-4} to 10^{0}. Results using 64-QAM with the 2×2 Alamouti scheme are also shown in Fig. 6.

The custom application is one of several ways to access the testbed. An alternate interface to the testbed is available as a MATLAB function. With this facility a simulated channel can easily be replaced with a physical channel via the testbed. The MATLAB function has the following format, `rxdata = tb_txrx(txdata, opts)` and is implemented via MATLAB's MEX interface. The data to be transmitted over the air is stored in an I/Q vector `txdata`, and the testbed configuration options are provided with the `opts` argument. The testbed utilises frame detection to capture the received signal vector from the radio modules and store the result in `rxdata`. Processing of the received data is then continues according to the MATLAB model. This provides a valuable tool for assessing algorithms using real hardware impairments such as I/Q mismatch, power amplifier non-linearities, timing errors and carrier frequency offsets.

A. FPGA Resource Usage

The basic resource in Altera Stratix II devices is an Adaptive Look-Up Table (ALUT). Table III summarises the main resource usage in the FPGA. Note that these figures refer to the EP2S30 FPGA used in the test results. The baseband board design is pin compatible with the EP2S60 device which doubles the resources available.

B. Future Work

Whilst demonstrating the basic functionality of the testbed here, it only represents the first stages of development and usage. The testbed will be used for a hardware implementation of orthogonal frequency division multiplexing (OFDM) MIMO and this requires more stringent firmware development. CFO estimation must be implemented in real-time, as OFDM is sensitive to transmitter and receiver mismatch in oscillator frequency. Voltage controlled oscillators (VCOs) are provided with simple control to deliberately misalign the CFO for testing purposes. Typical hand-held device oscillator characteristics can be evaluated as well with stability in the order of $\pm$20ppm.

Algorithms currently used in off-line processing will be implemented in the FPGA. This will allow implementation aspects of MIMO algorithms to be evaluated in terms of complexity and performance.

The expansion bus will be utilised to expand the processing capabilities of the testbed. Initially this will be a TI development kit, with later work to include additional FPGA support such as Altera Stratix IV or ASIC devices.

Experiments involving 4×4 antenna configurations are also planned for which the baseband board is hardware capable. Finally to support research across different frequency bands, additional radio modules may be developed.

IV. CONCLUSIONS

This paper has presented first results of a wireless testbed that while inexpensive, in terms of bill of material cost, has been designed to be capable of high performance. Initial results presented here have demonstrated its capabilities on 2×2 systems with up to 64-QAM constellations, although it has been successfully tested up to 64-QAM and with OFDM modulation. Future work involves implementing a 3GPP-LTE compliant wireless link.

ACKNOWLEDGMENT

The authors would like to thank (in alphabetical order) Dr's Mark Bickerstaff, Dave Garrett, Adriel Kind, Geoff Knagge, Chris Nicol, and Graeme Woodward and Mr. Charles Thomas for their help and advice in the development of the testbed. The work was funded by the Australian Research Council via LP0211210.

REFERENCES

[1] K. Mandke, S. Choi, G. Kim, R. Grant, R. Daniels, W. Kim, R. Heath Jr, and S. Nettles, "Early Results on Hydra: A Flexible MAC/PHY Multihop Testbed," *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, pp. 1896–1900, 2007.

[2] M. Ettus, "Ettus Research, LLC," *Online information on USRP board). http://www. ettus. com.*

[3] T. Haustein, A. Forck, H. Gabler, V. Jungnickel, and S. Schiffermuller, "Real-Time Signal Processing for Multiantenna Systems: Algorithms, Optimization, and Implementation on an Experimental Test-Bed," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–21, 2006.

[4] D. Borkowski, L. Brühl, C. Degen, W. Keusgen, G. Alirezaei, F. Geschewski, C. Oikonomopoulos, and B. Rembold, "SABA: A Testbed for a Real-Time MIMO System," *EURASIP Journal on Applied Signal Processing*, vol. 2006, no. 1, pp. 143–143, 2006.

[5] W. Xiang, P. Richardson, B. Walkenhorst, X. Wang, and T. Pratt, "A High-Speed Four-Transmitter Four-Receiver MIMO OFDM Testbed: Experimental Results and Analyses," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–10, 2006.

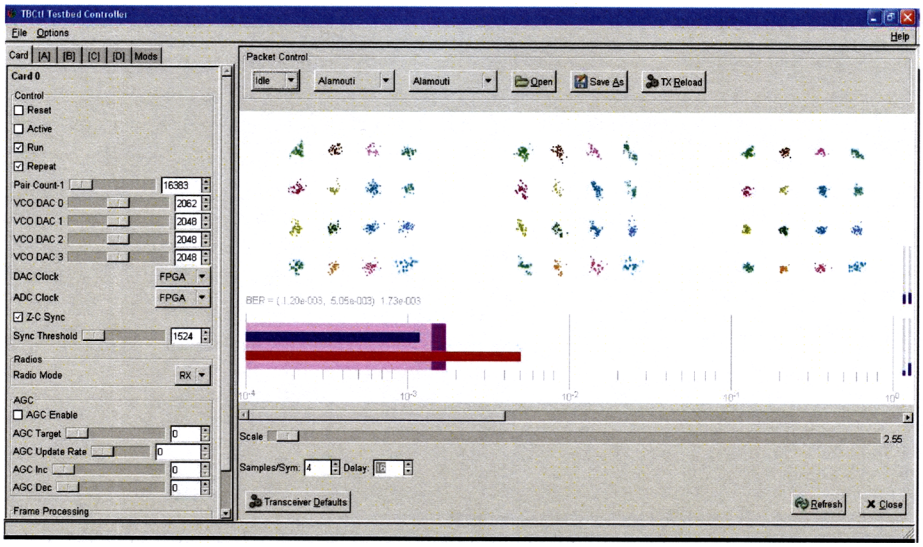

Fig. 5. 2×2 Alamouti using 16-QAM

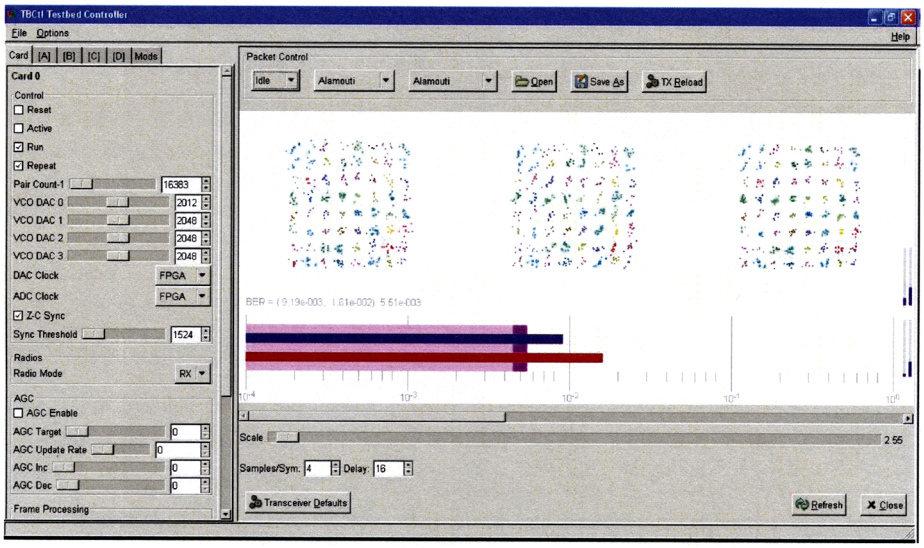

Fig. 6. 2×2 Alamouti using 64-QAM

[6] W. Zhu, D. Browne, and M. Fitz, "An open access wideband multiantenna wireless testbed with remote control capability," *Testbeds and Research Infrastructures for the Development of Networks and Communities, 2005. Tridentcom 2005. First International Conference on*, pp. 72–81, 23-25 Feb. 2005.

[7] J.-W. Weijers, V. Derudder, S. Janssens, F. Petr, and A. Bourdoux, "From MIMO-OFDM Algorithms to a Real-Time Wireless Prototype: A Systematic Matlab-to-Hardware Design Flow," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–12, 2006.

[8] S. Caban, C. Mehlfhrer, R. Langwieser, A. L. Scholtz, and M. Rupp, "Vienna MIMO testbed," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1451–1458, 2006.

[9] E. Aschbacher, S. Caban, C. Mehlfuhrer, G. Maier, and M. Rupp, "Design of a Flexible and Scalable 4× 4 MIMO Testbed," *Proceedings of 11th IEEE Digital Signal Processing Workshop (DSP04)*, pp. 178–181.

[10] P. Murphy, F. Lou, A. Sabharwal, and J. Frantz, "An FPGA based rapid prototyping platform for MIMO systems," *Signals, Systems and Computers, 2003. Conference Record of the Thirty-Seventh Asilomar Conference on*, vol. 1, pp. 900–904 Vol.1, 9-12 Nov. 2003.

[11] P. Goud Jr, R. Hang, D. Truhachev, and C. Schlegel, "A Portable MIMO Testbed and Selected Channel Measurements," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–10, 2006.

[12] K. Bialkowski, P. Uthansakul, M. Bialkowski, and A. Postula, "Design of MIMO Testbed with an FPGA Board for Fast Signal Processing," *Auswireless 2006 Conference*, 2006.

[13] P. Uthansakul, K. Bialkowski, M. Bialkowski, and A. Postula, "Assessing an FPGA Implemented MIMO Testbed with the Use of Channel Emulator," *Microwaves, Radar & Wireless Communications, 2006. MIKON 2006. International Conference on*, pp. 410–413, 2006.

[14] Altera, *Stratix II Device Handbook*, August 2006.

[15] Altera, *Nios II Processor Reference Handbook*, March 2007.

[16] 3GPP TS 36.211, *Evolved Universal Terrestrial Radio Access (E-UTRA): Physical Channels and Modulation*, 2008.

[17] S. Alamouti, "A simple transmit diversity technique for wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 8, pp. 1451–1458, Oct 1998.