Waveform Level Computational Energy Management in Software Defined Radios

Dinesh Datla, Thomas Tsou, Tim Newman, Jeff Reed, and Tamal Bose
Wireless@VirginiaTech
Virginia Polytechnic Institute and State University
Blacksburg, Virginia 24061
Email: ddatla,ttsou,trnewman,reedjh,tbose@vt.edu
Telephone: 540-231-2958
Fax: 540-231-2968

***Abstract*—Unlike traditional digital radios, where the waveforms are implemented in low power hardware such as ASICs, software defined radio (SDR) designs have to be concerned with the waveform computational complexity. Consequently, in SDRs, where the system power consumption is influenced by both the computational as well as communication hardware power consumption, it is important to understand the tradeoff between the two power consumption components. In this paper, we exhibit the tradeoff in the context of channel coding where the decoder complexity can impact the transmit power required in order to meet a given bit-error-rate (BER). Specifically, we consider viterbi decoding and soft-in-soft-out (SISO) BCJR iterative decoding of minimum shift keying (MSK) symbols where the traceback length and the number of iterations of the decoder can be tweaked in order to vary the computational execution time.**

I. Introduction

In conventional radio system design, the communication waveform's computational complexity and its processing energy consumption are typically secondary to that of the RF components because the waveforms are implemented on low power devices such as ASICs. However, with the advent of software defined radios (SDRs), where the waveforms are implemented on flexible computing platforms, the computational complexity can significantly impact the overall system power consumption. SDR network operation involves a close interaction between the communication and computation processes that deliver the required services under stringent power and/or energy constraints. Consequently, in SDRs, where the system power consumption is influenced by both the computational as well as communication hardware power consumption, the tradeoff between the two power consumption components is of interest while designing and operating low power SDRs. Unlike traditional radio design, the design will now take into consideration the power consumption in the communication hardware such as the power amplifier (PA) as well as the power consumption in executing waveforms implemented in software. An example scenario is a SDR system consisting of a transmitter node characterized by its transmit power and a receiver node that processes a channel decoding algorithm of a certain complexity. For an equivalent bit error rate (BER), the transmit power consumption can be reduced by using robust channel codes, but there is an additional cost of receiver complexity.

In this paper, we show how the computational energy can be managed in a SDR at the waveform level in the context of channel coding. The tradeoff between channel decoding computational complexity and transmission energy is exhibited by tweaking the parameters in the channel decoder that impact the computational complexity as well as the transmission power required in order to meet the quality-of-service (QoS).

The tradeoff study presented in this paper is expected to aid in the design of wireless distributed computing services in SDR networks. In scenarios where the individual nodes in the network may not meet the high power and energy demands set by complex computational tasks, the computational workload is allocated among collaborative nodes. In a heterogeneous network, the nodes differ in their computational capability as well as power supply availability. In order to make optimal usage of the resources that are dynamically available, it is essential to understand the power and energy consumption profile of the communication and computation processes. Channel decoding can be considered as a distributed computational task where the total energy consumption load can be balanced between the transmitter and receiver by tweaking the decoder complexity.

The remaining portion of the paper is organized as follows. Section II presents the power consumption models for the computation and communication aspects of an SDR. The simulation and code execution time measurement setup is described in Section III. The simulation results are presented and discussed in Section IV. The conclusions are presented in Section V.

II. Communication and Computation Power Consumption Models

In a SDR network, each node primarily comprises of a communication and computation subsystem. In this section, abstract energy consumption models for the communication and computation subsystems are presented.

A. Computation Subsystem

The computational QoS is characterized by the computation's energy consumption E_{cp}, power consumption P_{cp}, and latency T_{cp}. The power consumption model of a CMOS-based processor has been proposed in [1], [2], as given by

$$\begin{aligned} P_{cp} &= C_L\, V_{dd}^2\, f_{cp} \,+\, V_{dd}\, I_{leak} && (1)\\ \text{where } V_{dd} &= C_1\, f_{cp} + C_2, && (2)\\ \text{and } I_{leak} &= I_o \,\times\, \exp\left(\frac{V_{dd}}{C_3}\right). \end{aligned}$$

The mean total load capacitance switched per clock cycle C_L abstracts the processor activity factor (or utilization) and switching activity. V_{dd} is the processor core supply voltage. C_1 (expressed in volts/Hz), C_2 (volts), C_3 (volts), and I_o (amperes) are constants that depend on the computing platform. I_{leak} is the leakage current. Equation 2 expresses the linear relationship between the processor clock frequency f_{cp} and the minimum core supply voltage required in order to operate at that frequency.

For our analytical convenience, we model a computational task in terms of abstract computational units (CUs). A computational task constitutes N_{CU} discrete CUs. Each CU consumes P_{cp} watts of power, N_{cycles} processor clock cycles, and E_{CU} joules of energy. The total time to process all the CUs in the computational task, is denoted by T_{cp} where the processing time per CU is denoted by T_{CU}. In the case of channel coding, which is the example computational task being considered in this paper, each data symbol can be considered as a CU.

A CU is represented by N_{bits}^{in} bits before processing and N_{bits}^{out} bits after processing, which are relates as

$$N_{bits}^{out} \;=\; \gamma\, N_{bits}^{in}, \qquad (3)$$

where γ is a positive scaling factor. In other words, when a computation subsystem processes one CU it accepts N_{bits}^{in} bits at the input and generates N_{bits}^{out} bits at the output. For example, one FFT operation can be considered as one CU and the number of bits at the input and output of the FFT processor are the same. In the case of a data compression task $\gamma < 1$ while in the case of binary channel encoding it is equal to the coding rate.

The computational energy consumed in processing a computational task is given by

$$E_{cp} = P_{cp}\, T_{CU}\, N_{CU}. \qquad (4)$$

B. Communication Subsystem

The communication subsystem is characterized by the communication power consumption P_{cm}, latency T_{cm}, and energy consumption E_{cm}. The total transmission time T_{cm} is a function of the number of bits transmitted and the radio transmission time per bit T_{bit} (which is a function of modulation constellation size and network delays). P_{cm} can constitute either the transmitter power consumption P_{tx}, receiver power consumption P_{rx}, or both depending on the radio's functionality.

1) Transmitter Power and Energy Consumption: The average power consumption of the transmitter is given by

$$\begin{aligned} P_{tx} &= P_{rft} + P_{DAC}, && (5)\\ \text{where } P_{rft} &= P_{trs} + P_{txelec} + P_{amp},\\ \text{and } P_{amp} &= \eta\, P_t + \beta. \end{aligned}$$

P_{rft} and P_{DAC} are the power consumed by the transmitter RF circuits and the DAC respectively. P_{amp} is the power consumed by the PA in order to produce an output power of P_t with a linear PA efficiency η and β is a constant amplifier inefficiency term [3]. P_{txelec} is the total power consumed by the active radio hardware components such as the mixer, transmit filter and local oscillator. P_{trs} is the transient power consumption of the radio just after it is switched-on and before it is operational [3].

The PA output power P_t (i.e. the antenna input power), which is determined using the Frii's free space path loss model and log-normal shadowing model [4] , is given by

$$\begin{aligned} P_{t,dBm} &= P_{rmin,dBm} - 10\, log_{10}\left[\frac{G_t G_r \lambda^2}{(4\pi d_o)^2}\right] + LM_{dB}\\ &\quad -Q^{-1}(p) \times \sigma_{s,dB} + 10\, n\, log_{10}\left(\frac{D}{d_o}\right), \qquad (6) \end{aligned}$$

where the minimum required received signal power (receiver sensitivity) $P_{rmin} = SNR_{min} \times N$. In Equation 6, $\sigma_{s,dB}$ represents the shadow fading standard deviation, n is the path loss exponent, $1-p$ is the channel outage probability, G_t and G_r are the transmitter and receiver antenna gains, λ is the signal wavelength, d_o is the near field reference distance, and D is the distance between the transmitter and receiver. The link margin LM_{dB} accounts for the miscellaneous losses in the system that are not explicitly modeled in Equation 6, such as small scale fading. The minimum required receive signal-to-noise ratio SNR_{min} is a function of the required bit-error-rate (BER), transmit and receiver waveform model (modulation, channel coding and signal processing), and the channel conditions. The mean receiver noise power N is computed as $N = k_B \times T_o \times B \times NF$, where $k_B = 1.3806 \times 10^{-23}$ J/K is the boltzman's constant, $T_o = 300$ K is the ambient temperature, B is the receiver bandwidth in Hz, and NF is the receiver noise figure. Note that the subscript dB indicates that the parameter is expressed in decibels.

By manipulating eqs.(5) and (6), the transmitter power consumption for a given radio system can be expressed as a function of the communication range D, bandwidth B, and the minimum required receive signal-to-noise ratio SNR_{min}, as given by

$$\begin{aligned} P_{tx} &= (G_1\, SNR_{min}\, D^n\, B) + G_2, && (7)\\ \text{where } G_1 &= \frac{\eta\, k_B\, T_o\, NF\, (\sigma_s)^{-Q^{-1}(p)}\, (4\pi)^2}{G_t\, G_r\, \lambda^2\, d_o^{n-2}}\, LM,\\ \text{and } G_2 &= P_{DAC} + P_{trs} + \beta + P_{txelec} \end{aligned}$$

A similar high level communication subsystem power model has been presented in [5].

The transmitter energy consumption is given by

$$E_{tx} = P_{trs}\,T_{trs} + T_{tx}\left[P_{amp} + P_{txelec} + P_{DAC}\right]$$
$$\text{where } T_{tx} = \frac{N_{txout}}{R_{txout}} = \frac{(N_B/R_c)}{k\,R_s}. \quad (8)$$

T_{trs} and T_{tx} are the transient time and the total transmission time (i.e. time to transmit all the encoded data bits) respectively. The total number of bits transmitted is given by $N_{txout} = N_B/R_c$, where N_B is the total number of data bits from the source and R_c represents the channel coding rate. The net radio transmission bit rate is given by $R_{txout} = k\,R_s$, where $k = log_2 M$ is the number of bits mapped into a symbol at the modulator, M is the modulation index, and R_s is the radio transmission symbol rate expressed in symbols per second. In our model, the transmission data rate has been fixed in relation to the transmission bandwidth. In order to comply with Nyquist's theoretical minimum bandwidth for baseband transmission, as well as account for bandwidth expansion due to the constraints of real filtering, we set $R_s = 1.4 \times B$ assuming a baseband digital communication throughput of 1.4 symbols/sec/Hz [6].

2) Receiver Power and Energy Consumption: The total receiver power and energy consumption is given by

$$P_{rx} = P_{trs} + P_{rxelec} + P_{ADC}, \quad (9)$$
$$E_{rx} = P_{trs}\,T_{trs} + T_{rx}\left[P_{rxelec} + P_{ADC}\right]. \quad (10)$$

$P_{rxelect}$ is the total power consumed by the active receiver radio hardware components such as the low noise amplifier, mixer, receive filter, automatic gain control and local oscillator. P_{ADC} is the power consumed by the ADC. Note that we have assumed the reception time $T_{rx} \approx T_{tx}$.

3) SDR System Model: In a SDR network, each node consumes power for computation purposes in addition to the communication power consumption. The computation processes such as algorithm execution and memory access and communication processes such as transmission, reception, and channel estimation can occur concurrently in order to meet certain net latency requirements or to avoid data buffering between the computation and communication subsystems. In this case, the power consumption in the i^{th} node of the network is given by

$$P_i = P^i_{cp}(f^i_{cp}, A) + P^i_{cm}(D^i, SNR^i_{min}, B), \quad (11)$$

where P^i_{cp} and P^i_{cm} indicate the power consumed by the computation and the communication processes that occur concurrently. SNR^i_{min} is a function of the node index since the channel conditions can vary spatially.

The i^{th} node of the network consumes energy in order to computationally process N^i_{CU1} CUs and transmit or receive the data bits corresponding to N^i_{CU1} CUs. The total network energy consumption is given by

$$E_{nw} = \sum_{i=1}^{N_{nodes}} P^i_{cp}(f^i_{cp}, A)\,T^i_{CU}\,N^i_{CU1} \quad (12)$$
$$+ \; P_{trs}\,T_{trs} + P^i_{cm2}\,T^i_{bit}\,N^i_{bits}\,N^i_{CU1}$$

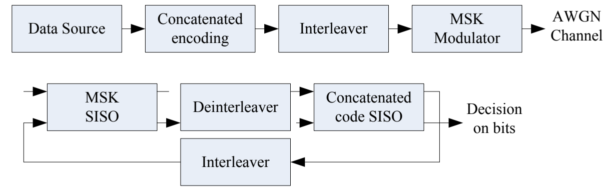

Fig. 1. Simulation block diagram with channel encoding and decoding.

TABLE I
SIMULATION PARAMETERS.

Parameter	Value	Parameter	Value
η	0.2	P_{txelec}	82.8 mW
NF	10 dB	β	174 mW
p	99 %	P_{trs}	58.7 mW
G_t, G_r	2 dBi	P_{DAC}	15.4 mW
f	450 MHz	n	3
d_o	10m	σ_s	8 dB
LM	10 dB	B	30 KHz
P_{rxelec}	102.8 mW	P_{cp}	0.3 W

where, $T^i_{CU} = N_{cycles}/f^i_{cp}$ and N_{nodes} is the number of nodes in the network. P_{cm2} is the power consumption during transmission or reception of channel bits. The number of bits transmitted or received per CU is denoted by N_{bits}.

III. Computer Simulation Setup

In our simulation, we consider a SDR based wireless communication system comprising of a transmitter and a receiver node. In simulation scenario 1, the viterbi algorithm is used to decode concatenated convolutional codes with the traceback length as the design parameter. In simulation scenario 2, we iteratively decode serially concatenated minimum-shift keying (SCMSK) symbols and the number of iterations is the design parameter. The block diagram for scenario 2 is shown in Figure 1 which is based on the theory presented in [7]. The block diagram is the same for viterbi decoding except that the bits are BPSK modulated and the decoding loop is replaced by a viterbi decoder. At the transmitter end, the data symbols are mapped to rate 1/2 convolutional codes followed by MSK modulation. At the receiver, the soft-input soft-output (SISO) modules in the decoder implement max-log versions of the BCJR algorithm. The channel is additive white Gaussian noise (AWGN). The generator matrix employed in simulation scenarios 1 and 2 are $[1 + D, 1 + D^2, 1 + D + D^2]$ and $[1 + D^2, 1 + D + D^2]$ respectively. All the decoder modules have been developed in MATLAB. In our simulation, a data block comprises of the number of bits equal to the decoder traceback length.

The simulation results presented in the remaining portion of the paper are based on the system parameters tabulated in

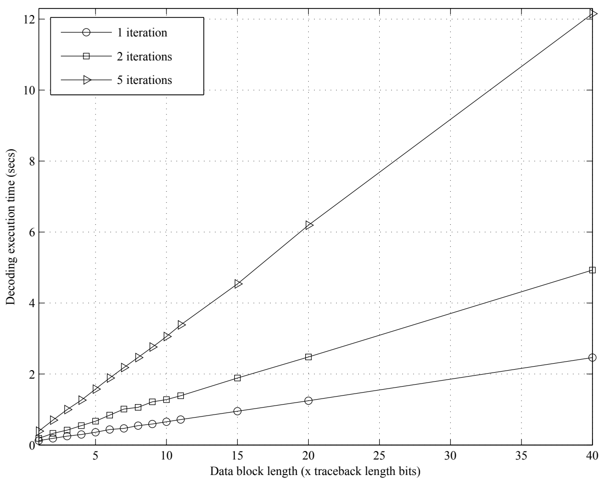

Fig. 2. Relationship between decoding execution time and transmitted data block length for various number of iterations of decoder with traceback length = 14 bits.

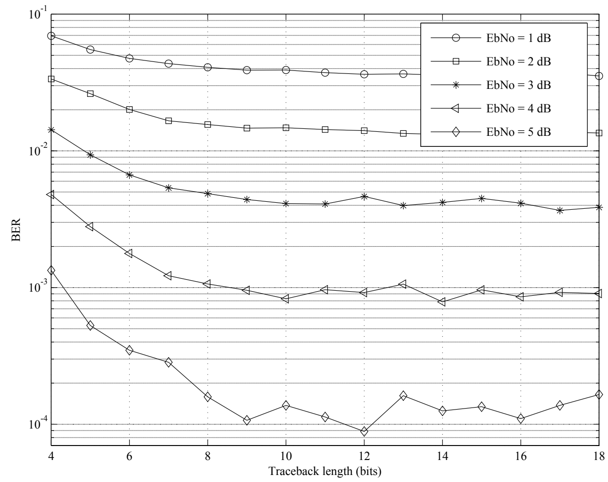

Fig. 3. BER performance by viterbi decoding scheme for various traceback lengths and SNR.

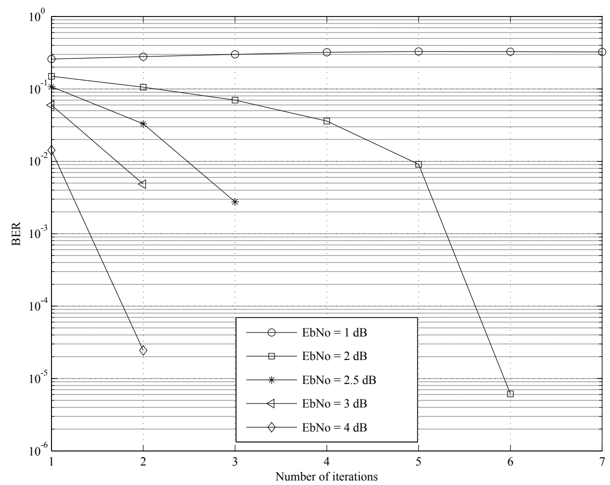

Fig. 4. BER performance by SCMSK coding scheme for various number of iterations and $SNRs$ using a traceback length of 14 bits.

Table I. The value of $G_1\ B$ has been determined as $G_1\ B = 8.0727 \times 10^{-12}\ W/m^3$ for $D = 500$ m with the help of empirical power consumption values for the ADC, DAC, and the various components in the RF front end which have been provided in the literature [1]–[3], [8], [9]. Since the measurement of computational power consumption is beyond the scope of this paper, we assume $P_{cp} = 300$ mW which is reasonable based on the channel decoding computational model presented in [3].

A. MATLAB Code Profiling

A laptop equipped with 4 GB of RAM, an Intel Core2 Duo T8300 CPU that can operate at a maximum clock frequency of 2.4 GHz, and that runs Microsoft Windows Vista was used for profiling the decoder routines. In order to obtain consistent and accurate results, we set the MATLAB process affinity in the Windows Task Manager to a single CPU so that the active number of CPUs is restricted to 1. The built-in MATLAB profiler was used to determine the execution time of the decoder routines that implement Viterbi decoding and SISO BCJR decoding. The measurements were collected when the laptop was operated at 2.4 GHz.

In order to gain insight into the scalability of execution time with the length of the transmitted data block (i.e. uncoded data bits), we took measurements of the execution time for iteratively decoding data blocks of different lengths, as plotted in Figure 2. From Figure 2, it is observed that there is a linear relationship between execution time and the computational workload for our MATLAB iterative channel decoding routine. The execution time for larger block lengths of about 14000 bits and 7 iterations of the decoding algorithm, which is not shown in Figure 2, was found to obey the linear relationship. The linear relationship allows us to display the execution time results for one data block ignoring the impact of the actual data block length.

IV. Simulation Results

The BER measured in the case of viterbi decoding for various traceback lengths and SNRs is plotted in Figure 3. The BER curves flatten for larger traceback lengths thereby implying that it is energy efficient to operate the viterbi decoder at the minimum traceback length for which the slope of the BER curve is nearly zero. The measured execution time did not vary significantly with the increase in the traceback length although the complexity of the trellis increases linearly with the traceback length. This indicates that the execution time is not a good measure of the complexity of the viterbi decoding algorithm and hence will not be used for the tradeoff analysis in the case of viterbi decoding.

The BER curve for the SCMSK coding scheme for various number of iterations and SNRs is plotted in Figure 4. At higher SNRs, the BER drops to zero for less than 3 iterations of decoding, as shown in Figure 4.

TABLE II
TIME TO EXECUTE ONE DATA BLOCK AND REQUIRED TRANSMIT SNR FOR DIFFERENT NUMBER OF ITERATIONS OF SCMSK ITERATIVE DECODING.

No. of iterations	Required SNR	Execution time (secs)
1	>4 dB	0.0718
2	4 dB	0.1342
3	3 dB	0.1934
4	2.5 dB	0.2558
5	2.5 dB	0.3244
6	1 dB	0.3712
7	1 dB	0.4368

The execution time of the iterative decoding routine was measured for different number of iterations and a fixed traceback length of 14. The SNRs required to maintain a BER of 10^{-3} and the measured execution times for various number of iterations are tabulated in Table II. Using the SNR and execution time values from Table II, the transmitter communication energy consumption, the total receiver energy consumption (which includes both its communication hardware and computational energy consumption) and the total energy consumption of the two node SDR system have been computed and plotted as shown in Figure 5. It is observed that the receiver energy consumption increases linearly with the number of iterations while the reduction in required SNR results in a decrease in transmitter communication energy. Since the total energy consumption is dominated by the receiver computational energy consumption, the reduction in transmit power does not compensate for the increase in receiver energy consumption. This indicates that, in the case of a complex channel decoding algorithm, it is energy efficient for the network to operate the decoder with a small number of iterations.

Similar calculations were made for a more harsh channel with $n = 5$ and $LM = 30$ dB. While a similar tradeoff trend exists between the transmitter and receiver energy consumptions, as plotted in Figure 6, the total energy consumption is not affected by the number of iterations of decoding. This represents an ideal scenario where the total energy consumption can be balanced between the transmitter and receiver by varying the number of iterations of decoding.

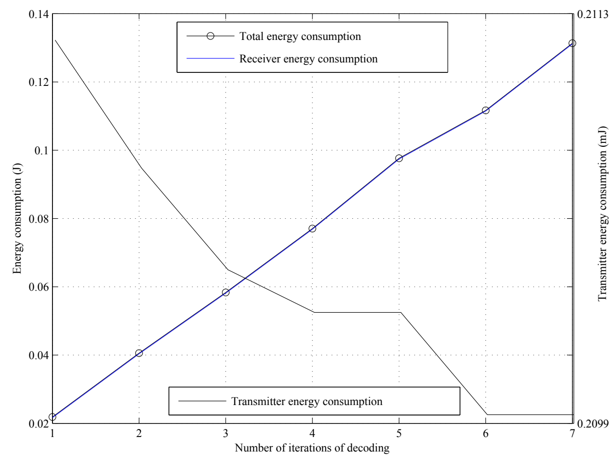

Fig. 5. Tradeoff between transmitter and receiver energy consumptions when employing SCMSK coding scheme for various number of iterations using a traceback length of 14 bits.

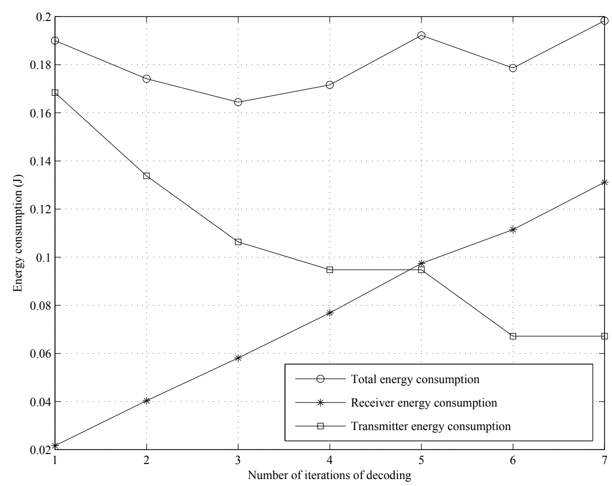

Fig. 6. Tradeoff between transmitter and receiver energy consumptions when employing SCMSK coding scheme for various number of iterations using a traceback length of 14 bits under harsh channel conditions.

V. CONCLUSIONS

In this paper, we have emphasized on the fact that the communication waveform's computational energy consumption plays a dominant role in the total system wide energy consumption of SDR networks unlike in traditional hardware defined radios. In this regard, the tradeoff between transmission energy and receiver decoding complexity was discussed. It was also shown that the approach to design SDR waveforms that meet network wide goals such as network energy efficiency differs from traditional design approaches where individual radio energy consumption matters most. This difference in the design approach was discussed in the context of channel coding where simple decoding schemes are preferred over complex iterative schemes under certain channel conditions in order to achieve network wide energy efficient communication.

It was also pointed out that the tradeoff relationship can vary according to the computing platform and the channel conditions. In this paper, we have discussed the tradeoffs in the scenarios where the overall system energy consumption is dominated by either the transmission energy consumption or the receiver computational energy consumption.

It is to be noted that the execution times measured in this paper are not for the purpose of quantifying the absolute complexity of the decoding process, but to demonstrate the tradeoff relationship between decoding complexity and transmission energy.

Acknowledgment

The authors would like to acknowledge the support of the Office of Naval Research, NSF, and VT-ICTAS.

References

[1] A. Sinha and A. P. Chandrakasan, "Energy aware software," in *Thirteenth International Conference on VLSI Design*, (Calcutta, India), pp. 50–55, 2000.

[2] A. Sinha and A. P. Chandrakasan, "Jouletrack-a web based tool for software energy profiling," in *Proceedings of Design Automation Conference*, pp. 220– 225, 2001.

[3] R. Min and A. Chandrakasan, "A framework for energy-scalable communication in high-density wireless networks," in *Proc. International Symposium on Low Power Electronics and Design ISLPED '02*, pp. 36–41, 2002.

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

[5] D. Maniezzo, K. Yao, and G. Mazzini, "Energetic trade-off between computing and communication resource in multimedia surveillance sensor network," in *4th International Workshop on Mobile and Wireless Communications Network*, (Stockholm, Sweden), pp. 373– 376, November 2002.

[6] B. Sklar, *Digital Communications: Fundamentals and Applications*. Pearson Education, Inc., second ed., 2002.

[7] P. Moqvist and T. M. Aulin, "Serially concatenated continuous phase modulation with iterative decoding," *IEEE Transactions on Communications*, vol. 49, pp. 1901–1915, November 2001.

[8] S. R. Cui, *Cross-Layer Optimization in Energy Constrained Networks*. PhD thesis, Stanford University, June 2005.

[9] S. Cui, A. J. Goldsmith, and A. Bahai, "Energy-efficiency of mimo and cooperative mimo techniques in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, pp. 1089–1098, AUGUST 2004.