TURBO-BLAST for Wireless Communications: Theory and Experiments

Mathini Sellathurai, *Member, IEEE,* and Simon Haykin, *Fellow, IEEE*

***Abstract*—TURBO-BLAST is a novel multitransmit multireceive (MTMR) antenna scheme for high-throughput wireless communications. It exploits the following ideas: the Bell Labs layered space time (BLAST) architecture; random layered space-time (RLST) coding scheme by using independent block codes and random space-time interleaving; sub-optimal turbo-like receiver that performs iterative decoding of the RLST codes and estimation of the channel matrix in an iterative and, most important, simple fashion. The net result is a new transceiver that is not only computationally efficient compared with the optimal maximum likelihood decoder, but it also yields a probability of error performance that is orders of magnitude smaller than traditional BLAST schemes for the same operating conditions. This paper also presents experimental results using real-life indoor channel measurements demonstrating the high-spectral efficiency of TURBO-BLAST.**

***Index Terms*—High-rate layered space-time methods, multitransmit multireceive antennas, turbo principle.**

I. Introduction

In 1993, Berrou *et al.* developed the revolutionary iterative "turbo" receiver for decoding two-dimensional (2-D) product-like codes [1]. Two properties constitute the hallmark of turbo codes.

- The error performance of the turbo decoder improves with the number of iterations of the decoding algorithm.
- The turbo decoder is capable of approaching the Shannon limit of channel capacity in a computationally feasible manner.

What is even more profound is the fact that the "turbo learning principle" has been successfully applied not only to channel decoding but also to channel equalization, coded modulation, multiuser detection, and joint source and channel decoding [2]. In this paper, we describe another novel application of the turbo learning principle to wireless communications using multiple antennas at both the transmitting and receiving ends of the wireless channel. This antenna structure is called Bell Labs layered space-time (BLAST) architectures [3].

The multitransmit multireceive (MTMR) communications structures, popularized as *BLAST architectures*, have received considerable attention in the literature as they could provide the basis for very high data-rate communication over wireless channels for a fixed total transmit power. For a single user, the architecture provides tremendous spectral efficiency compared to other multiplexing (wideband) schemes such as code-division multiple access (CDMA) and orthogonal frequency-division multiplexing (OFDM) by having each transmitting antenna operate in a co-channel manner and use the entire channel bandwidth.

The spatial diversity provided in MTMR structures establishes a strong link between the transmitter and receiver. However, MTMR schemes rely on a rich scattering matrix channel. In a highly correlated channel environment, the major source of channel impairment in these schemes is co-antenna interference (CAI). To mitigate the degrading effects of CAI and block fading, we require the use of a *robust* MTMR antenna system. In particular, the architecture must be designed with an appropriate coder and decoder to make the probability of code-word error arbitrarily small with practical coding and decoding complexity. Space-time coding [4] and layered space-time coding are the most popular coding techniques for MTMR schemes. However, space-time coding proposed in [4] and [5] is typically designed in two-dimensional (2-D) space, and it is not well suited for high-information rate transmission due to its decoding complexity. In contrast, the layered space-time coding uses an elegant layered space-time concept, in which the 2-D space-time coding design is achieved by layering one–dimensional (1-D) channel codes. Most importantly, the layered space-time concept introduced in BLAST architectures allows the multidimensional decoding problem to be optimally solved by using 1-D receiver processing.

The first BLAST proposed in the literature is the Diagonal-BLAST (D-BLAST) architecture [3], which has a diagonal layering space-time coding with sequential nulling and interference cancellation decoding. D-BLAST suffers from boundary wastage at the start and end of each packet, which become significant for a small packet size. Designing elegant diagonal layered space-time coding techniques that eliminate the boundary wastage present an open research problem; indeed, they have become a popular research topic. Vertical-Blast (V-BLAST) overcomes the limitation of D-BLAST by using independent horizontal layered space-time coding scheme; unfortunately, it does not utilize the transmit diversity and, therefore, suffers from the problem of reduced information capacity [6]. The developments of D-BLAST and V-BLAST motivated the authors of this paper to focus on another layered space-time architecture, hereafter called Turbo-BLAST. This new system is based on a random

Manuscript received May 7, 2001; revised April 30, 2002. This work was supported by Natural Science and Engineering Research Council (NSERC) of Canada. The associate editor coordinating the review of this paper and approving it for publication was Dr. Bertrand M. Hochwald.

M. Sellathurai was with McMaster University, Hamilton, ON L8S 4K1, Canada. She is now with the Satelite Communications Research Branch, Communications Research Centre, Ottawa, ON K2H 8S2, Canada.

S. Haykin is with McMaster University, Hamilton, ON L8S 4K1, Canada (e-mail: haykin@mcmaster.ca).

Publisher Item Identifier 10.1109/TSP.2002.803327.

layered space-time (RLST) code and an iterative detection and decoding (IDD) receiver [7]–[9].

Several versions of IDD receivers for space-time codes, BLAST, hybrid BLAST, and space-time codes have been proposed in the literature [7]–[16]. The first such architectures addressed in the literature are Turbo-BLAST [7] and threaded space-time (TST) architecture [10]. In the Turbo-BLAST system, the transmit diversity is introduced through a random space-time permuter following independent encoding of each substream, using either block or convolutional forward error-correction (FEC) codes. The combination of two things [1) independent encoding of the substreams and 2) random space-time interleaving] can be viewed as a RLST code. Most importantly, the RLST concept allows for an IDD receiver, in which the multidimensional decoding problem is solved by using successive 1-D decoding stages. It is interesting to note that the threaded space-time (TST) architecture proposed in [10] and vertical layered space-time (VLST) architecture proposed in [11]–[13] have similar coding and decoding structures as Turbo-BLAST. However, the design of TST code is more general than Turbo-BLAST in that it is based on 2-D space-time coding principles that maximize both the spatial and temporal diversity. The RLST and VLST codes can be viewed as special classes of TST code designs. A detailed exposition of the TST code design for any transmit diversity is presented in [14]. Moreover, in [15] and [16], a combination of horizontally coded BLAST and space-time block code [5] with IDD is studied. In this scheme, transmit diversity is achieved within each group by the use of space-time block codes but at the expense of a reduced information rate.

Our paper on Turbo-BLAST builds on previous work [7]–[16] in two major ways.

1) The IDD receivers need channel estimates *a priori*. With short training sequences, it is difficult to achieve good channel estimates in MTMR systems with a large number of transmit and receive antennas. We show that by jointly estimating the channel matrix and the interference at each iteration of the IDD using the minimum mean-squared error (MMSE) principle, Turbo-BLAST can achieve a performance close to the corresponding system with channel knowledge *a priori*. At the first iteration, we use a short training sequence to produce a preliminary estimate of the channel matrix. Subsequently, we re-estimate the channel matrix using more reliably estimated symbols of each packet at each subsequent iteration. Experimentally, we show that this iterative channel estimation procedure using successively decoded symbols of the entire block at each iteration does significantly improve the performance of the receiver.
2) We show that Turbo-BLAST architectures can handle any configuration of transmit and receive antennas, including the case of fewer receive antennas than transmit antennas. From a practical perspective, the ability to work with fewer receive antennas than transmit antennas is necessary in most cellular systems because the base station can usually accommodate more transmit antennas than mobile transceivers. We demonstrate this ability for Turbo-BLAST wireless systems by using real-life channel measurements for an indoor environment.

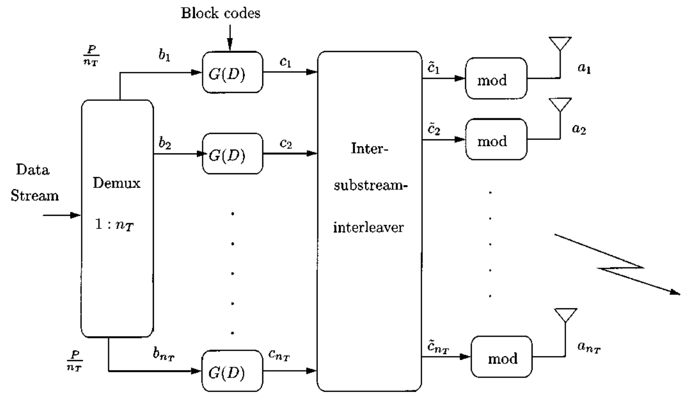

Fig. 1. Turbo-BLAST transmitter.

II. TURBO-BLAST: BASIC TRANSMITTER CONSIDERATIONS

We consider a MTMR system that has n_T transmitting and n_R receiving antennas. Throughout this paper, we assume that the n_T transmitters operate with synchronized symbol timing at a rate of $1/T$ symbols per second and that the sampling times of n_R receivers are symbol synchronous. The channel variation is assumed to be negligible over M symbol periods, comprising a packet of symbols. Moreover, we only consider a narrowband frequency-flat communication environment, i.e, no delay spread. The extension of this scheme to a frequency-selective environment is straightforward.

Fig. 1 shows a high-level description of the Turbo-BLAST architecture, having n_T transmitting and n_R receiving elements. The encoding process involves the following.

- We demultiplex the user information bits into n_T substreams $\{\bar{\mathbf{b}}_k\}_{k=1}^{n_T}$ of equal data rate.
- We independently block-encode each data substream, which uses the same predetermined linear block FEC code with a weighted distribution of minimum weight equal to $d_{\min}$

$$\mathbf{C} = \left[\bar{\mathbf{b}}_1\mathbf{G}, \bar{\mathbf{b}}_2\mathbf{G}, \ldots, \bar{\mathbf{b}}_{n_T}\mathbf{G}\right] = \left[\bar{\mathbf{c}}_1, \bar{\mathbf{c}}_2, \ldots, \bar{\mathbf{c}}_{n_T}\right]^T \quad (1)$$

  where $\mathbf{G}$ is $K \times L$ binary code generator, the $\{\bar{\mathbf{b}}_k\}_{k=1}^{n_T}$ are K-dimensional information sequences, and the $\{\bar{\mathbf{c}}_k\}_{k=1}^{n_T}$ are L-dimensional code sequences.
- The encoded substreams are bit-interleaved using a random space-time permuter Π. We use $\tilde{\mathbf{C}} = \{\tilde{\mathbf{c}}_k\}_{k=1}^{n_T}$ to denote the permuted substreams, where

$$\tilde{\mathbf{C}} = \Pi(\mathbf{C}). \quad (2)$$

  The random space-time interleaver is independent of the incoming data streams, and its design must guarantee the use of an entire subchannel by each independently coded substream in an equal manner, thereby permitting the use of an off-line design procedure. In the rest of the paper, we consider a space interleaver based on diagonal layering of each independently coded substream, as shown in Fig. 2,

Fig. 2. Diagonal space interleaver.

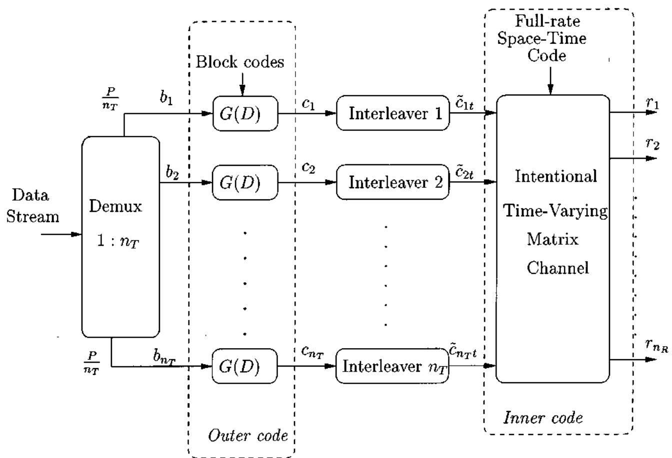

Fig. 3. RLST codes as serially concatenated codes.

which is then followed by random time interleavers to generate the RLST codes. The space interleaving procedure is simply a permutation operation over the L columns, according to the interleaver. Note that unlike D-BLAST, we do not experience any boundary wastage in this diagonal layering structure due to the cyclic nature of the encoding process.

- The space-time interleaved substreams are independently mapped into symbols $\mathbf{A} = \{\bar{\mathbf{a}}_k\}_{k=1}^{n_T}$, where

$$\mathbf{A} = f(\tilde{\mathbf{C}}). \quad (3)$$

Each interleaved substream is transmitted using a separate antenna. The transmitted signals are received on n_R receiving antennas, whose output signals $\mathbf{R} = \{\bar{\mathbf{r}}_k\}_{k=1}^{n_R}$ are fed to an iterative receiver.

A. Intentional Time-Varying Channel

The combined use of block codes and interleaving provides the basis for the *random block codes*, namely, parallel and serially concatenated turbo codes. Using this principle for MTMR systems, we produce RLST block codes by concatenating block codes and space-time interleavers.

Fig. 3 illustrates another view of the proposed turbo space-time block codes under a quasistatic Rayleigh fading environment. In this representation, we include the effect of diagonal space-time interleaving with the quasistatic Rayleigh matrix channel. The combination of diagonal-interleaving and the quasistatic Rayleigh matrix channel introduces an "intentional" time-varying channel.

The channel shown in Fig. 4(a) is generated for a (16,16)-BLAST system; note that each subchannel is static within the packet of interest. In Fig. 4(b), we show the time-varying subchannels generated by the intersubstream interleaving process, that is, by combining the space-time interleavers and the channel shown in Fig. 4(a). Only three subchannels (out of 16) are shown here for simplicity. For a sufficiently large number of transmitters, a highly time-varying channel can be achieved even in delay-limited and nonergodic systems. Moreover, the time averages of each independent channel in Fig. 4(b) will approach their corresponding ensemble averages in the limit as the observation interval T and the number of transmit antennas n_T approach infinity; thus, the space-time interleaver generates an artificial ergodic process from the nonergodic quasistatic Rayleigh fading MTMR channels. Note that in Fig. 4(a), each subchannel is nonergodic since it does not change with time.

III. TURBO-BLAST DECODER: BASIC CONSIDERATIONS

It is well known that the optimal signal-decoding problem in intersubstream-encoded MTMR schemes has a computational complexity that is exponential in the number of substreams, the constellation size, and the block size. Even though it is possible to model the proposed RLST block code as a single Markov process and a trellis can be formed to include the effect of space-time interleaving, optimal decoding of such a trellis representation is extremely complex and does not lend itself to feasible decoding algorithms [17].

This section proposes a practical sub-optimum detection scheme based on iterative "turbo" detection principles. The intersubstream coding proposed as independent encoding and space-time interleaving can be viewed as a serially concatenated code as illustrated in Fig. 3: Outer code—n_T parallel channel codes; inner code—time-varying channel matrix. The inner and outer codes are separated by n_T parallel interleavers. The concatenated code can be decoded using a lower complexity iterative receiver similar to the iterative schemes proposed for serially concatenated turbo codes. In the iterative decoding scheme, we separate the optimal decoding problem into two stages (two low complex sub-trellises) and exchange the information learned from one stage to another iteratively until the receiver converges. The two decoding stages are 1) *Inner decoder:* soft-input/soft-output (SISO) channel detector; 2) *Outer decoders:* A set of n_T parallel SISO channel decoders. The detector and decoder stages are separated by space-time interleavers and deinterleavers. The interleavers and deinterleavers are used to compensate for the interleaving operation used in the transmitter as well as to decorrelate the correlated outputs before feeding them to the next decoding stage. The iterative receiver produces new and better estimates at each iteration of the receiver and repeats the information-exchange process a number of times to improve the decisions and channel estimates. Note that the design of our intersubstream coding uses independent coding of each substream; hence, the receiver

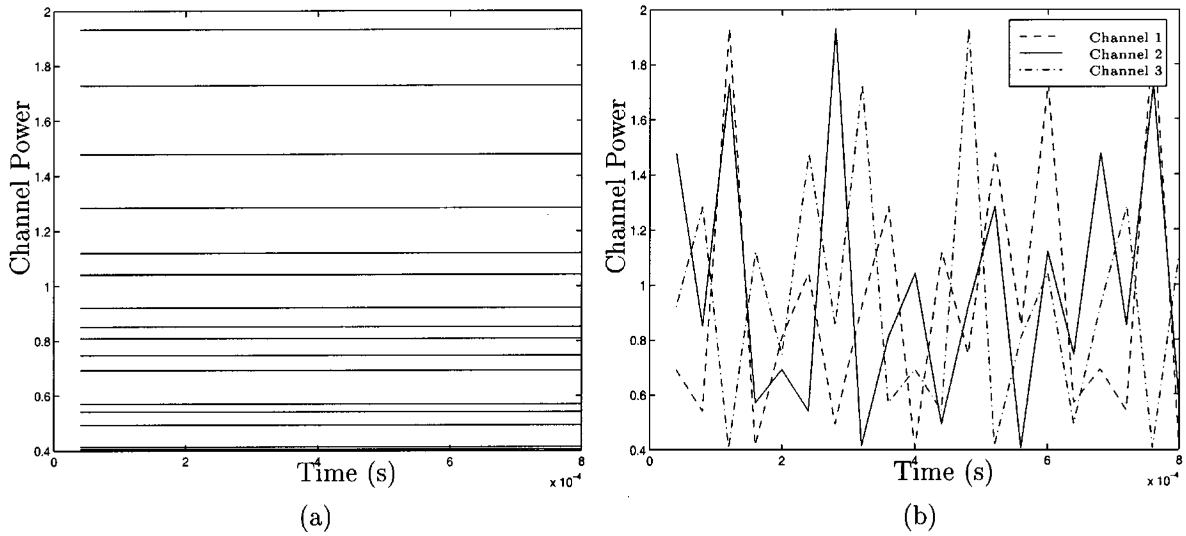

Fig. 4. Intentional time-varying channel. (a) Channel response before interleaving. (b) Channel response after interleaving.

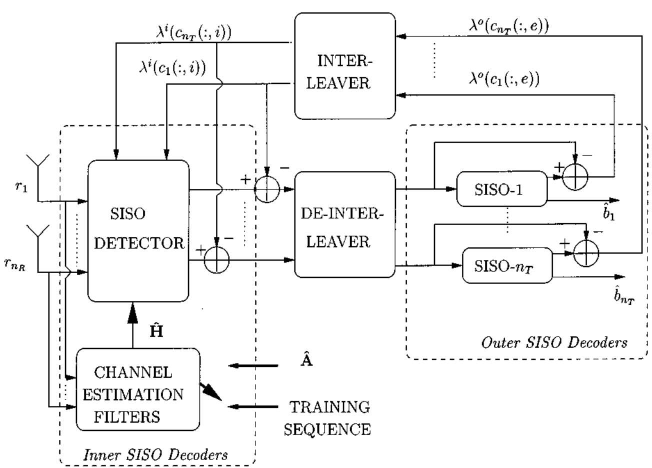

Fig. 5. Iterative decoder.

needs to select only one of 2^L sequences for each n_T sequence separately without increasing the probability of symbol error significantly. The iterative decoder is shown in Fig. 5.

A. Iterative Decoding Algorithm

The iterative decoding structure of serially concatenated turbo codes provides the principal model for the iterative decoding algorithm. The following notations are used to explain the algorithm: the log-likelihood ratios (LLR) λ^i and λ^o, with superscripts i and o, denote the LLR associated with the inner decoder and the outer decoder of the decoding process, respectively. The symbols $\lambda(:,i)$, $\lambda(:,e)$, and $\lambda(:,p)$ at the output and input of the SISO modules refer to *intrinsic*, *extrinsic*, and *a posteriori* information formulated as log-likelihood ratios.

First, we define the *a posteriori* log-likelihood ratio (LLR) of a transmitted bit symbol $c_j(l)$, $j = 1, 2, \ldots n_T$, and $l = 1, 2, \ldots, L$:

$$\lambda(c_j(l);p) = \log \frac{P\{c_j(l) = +1|\mathbf{r}\}}{P\{c_j(l) = -1|\mathbf{r}\}}. \tag{4}$$

Using Bayes' rule, (4) can be rewritten as

$$\begin{aligned}\lambda(c_j(l);p) =& \log \frac{P\{\mathbf{R}|c_j(l) = +1\}}{P\{\mathbf{R}|c_j(l) = -1\}} \\ &+ \log \frac{P\{c_j(l) = +1\}}{P\{c_j(l) = -1\}} \\ =& \lambda(c_j(l);e) + \lambda(c_j(l);i).\end{aligned} \tag{5}$$

The first term $\lambda(c_j(l);e)$ in (5) constitutes *extrinsic* information, and the second term $\lambda(c_j(l);i)$ constitutes *intrinsic* information of the code bit $c_j(l)$.

The iterative decoder, which is illustrated in Fig. 5, depicts message passing between the inner/detector and outer/decoder SISO modules:

1) The SISO detector (inner SISO module) generates soft estimates of the code bits $c_j(l)$ conditional on the received signal $\mathbf{r}(t)$, as well as the *intrinsic* information about all the code bits $c_k(l)$, $\forall\, k$, $k \neq j$, and $c_j(t)$, $\forall\, t$, $t \neq l$. Note that the soft information on $c_j(l)$, as computed by the SISO detector, is influenced by the *intrinsic* information of $\lambda(c_j(l);i)$ from the previous stage.

- Estimate the *a posteriori information*

$$\lambda^i(c_j(l);p) = \log \frac{P\{c_j(l) = +1|\mathbf{R}, \lambda^i(\mathbf{C};i)\}}{P\{c_j(l) = -1|\mathbf{R}, \lambda^i(\mathbf{C};i)\}}, \quad \forall\, j,l. \tag{6}$$

  During the first iteration, the initial *intrinsic* probabilities of all symbol bits are assumed to be 1/2 (i.e., equally likely). Thus, $\lambda(\mathbf{c}_j(l);i) = 0, \forall\, j,l$.

- Compute the extrinsic information

$$\lambda^i(\mathbf{C};e) = \lambda^i(\mathbf{C};p) - \lambda^i(\mathbf{C};i) \tag{7}$$

  where $\lambda^i(\mathbf{C};e)$ is the extrinsic information about the set of code bits $\mathbf{C}$ of the SISO detector, which is fed back to the outer decoder as the intrinsic information of its coded bits. Before application to the outer decoder, the extrinsic information is reordered to compensate for the pseudo-random interleaving introduced in the turbo encoder, yielding

$$\lambda^o(\mathbf{C};i) = \Pi^{-1}\{\lambda^i(\mathbf{C};e)\}. \tag{8}$$

2) The n_T outer SISO modules, in turn, process the soft information $\lambda^o(\mathbf{c}_j(l);i)$ and compute refined estimates of soft information on both code $\mathbf{c}_j(l)$ and information bits $\mathbf{b}_j(l)$, based on the trellis structure of the channel codes, which is the channel code constraint.

- The *a posteriori information* for information and code bits is, respectively, shown in (9) and (10) at the bottom of the page. The input $\lambda^o(\mathbf{B},i)$ is always initialized to zero, assuming equally likely source information bits.
- The extrinsic information of information and code bits are, respectively

$$\lambda^o(\mathbf{B};e) = \lambda^o(\mathbf{B};p) - \lambda^o(\mathbf{B};i) \tag{11}$$
$$\lambda^o(\mathbf{C};e) = \lambda^o(\mathbf{C};p) - \lambda^o(\mathbf{C};i). \tag{12}$$

The output, that is, the *extrinsic* information of the n_T outer decoders provides *intrinsic* information to the inner/detector SISO module after reordering to compensate for the random interleaving; thus

$$\lambda^i(\mathbf{C};i) = \Pi\{\lambda^o(\mathbf{C};e)\}. \tag{13}$$

Steps 1) and 2) are repeated until the decoding algorithm converges.

3) An estimate of the message bits $\mathbf{B}$ is obtained by hard limiting the LLR $\lambda^o(\mathbf{B};p)$ at the output of the outer decoder

$$\hat{\mathbf{B}} = \mathrm{sgn}\{\lambda^o(\mathbf{B};p)\}. \tag{14}$$

Note that the outer decoder of the iterative decoding algorithm is made up of n_T parallel SISO channel decoders, implemented by using the generalized BCJR algorithm. A detailed explanation of the generalized BCJR algorithm is presented in [1]. An issue of interest is the criterion used to optimize the inner SISO module in the iterative decoders. We design the inner SISO module using the mean-square error minimization (MMSE), as described next.

B. Minimum Mean-Square Error Receiver

The received signal $\mathbf{r}(i) \in \mathcal{C}^{n_R \times 1}$ at the receive array at time i is

$$\mathbf{r}(i) = \mathbf{H}(i)\mathbf{a}(i) + \mathbf{v}(i) \tag{15}$$

where $\mathbf{H}(i) \in \mathcal{C}^{n_R \times n_T}$, $\mathbf{a}(i) \in \mathcal{C}^{n_T \times 1}$, and $\mathbf{v}(i) \in \mathcal{C}^{n_R \times 1}$. Let $a_k(i)$ be the desired signal

$$\mathbf{r}(i) = \mathbf{h}_k(i)a_k(i) + \mathbf{H}_k(i)\mathbf{a}_k(i) + \mathbf{v}(i) \tag{16}$$

where $\mathbf{H}_k(i) = [\mathbf{h}_1(i), \mathbf{h}_2(i), \ldots, \mathbf{h}_{k-1}(i), \mathbf{h}_{k+1}(i), \ldots, \mathbf{h}_{n_T}(i)] \in \mathcal{C}^{n_R \times n_T - 1}$, and $\mathbf{a}_k = [\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_{k-1}, \mathbf{a}_{k+1}, \ldots, \mathbf{a}_{n_T}]$. The decision statistic of the kth substream using a linear filter $\mathbf{w}_k$ is

$$y_k(i) = \underbrace{\mathbf{w}_k^H \mathbf{h}_k a_k(i)}_{d_k} + \underbrace{\mathbf{w}_k^H \mathbf{H}_k \mathbf{a}_k(i)}_{u_k} + \underbrace{\mathbf{w}_k^H \mathbf{v}(i)}_{\bar{v}_k} \tag{17}$$

where d_k, u_k and $\bar{v}_k$ are the desired response obtained by the linear beamformer, the CAI, and phase-rotated noise, respectively.

We propose a multisubstream detector based on the MMSE principle and soft interference cancellation, which optimizes the interference estimate and the weights of the linear detector jointly in a manner similar to the multiuser receivers proposed in [18] and [19].

We remove CAI from the linear beamformer output y_k and write

$$x_k = \mathbf{w}_k^H \mathbf{r} - u_k \tag{18}$$

where u_k is the linear combination of interfering substreams, and x_k is the improved estimate of transmitted symbol a_k. For brevity, we omit the sampling index (i). The performance of the estimator is measured by the error $e_k = a_k - x_k$. The weights $\mathbf{w}_k \in \mathcal{C}^{n_T \times 1}$ and the interference estimate u_k are optimized by minimizing the mean-square value of the error between each substream and its estimate.

Problem 1: Given (15) and (18), find the weight vectors $\mathbf{w}_k =$ and u_k by minimizing the cost function

$$(\hat{\mathbf{w}}_k, \hat{u}_k) = \arg \min_{(\mathbf{w}_k, u_k)} \mathcal{E}\left[\|a_k - x_k\|^2\right] \tag{19}$$

where the expectation $\mathcal{E}$ is taken over the noise and the statistics of the data sequence.

Solution 1: The solution to Problem 1 is given by

$$\hat{\mathbf{w}}_k = (\mathbf{P} + \mathbf{Q} + \Sigma_{n_R})^{-1}\mathbf{h}_k \tag{20}$$
$$\hat{u}_k = \mathbf{w}_k^H \mathbf{z} \tag{21}$$

where

$$\begin{aligned}
\mathbf{P} &= \mathbf{h}_k \mathbf{h}_k^H \in \mathcal{C}^{n_R} \\
\mathbf{Q} &= \mathbf{H}_k \left[\mathbf{I}_{(n_T-1)} - \mathrm{Diag}\left(\mathcal{E}[\mathbf{a}_k]\mathcal{E}[\mathbf{a}_k]^H\right)\right] \mathbf{H}_k^H \in \mathcal{C}^{n_R} \\
\Sigma_{n_R} &= \sigma^2 \mathbf{I}_{n_R}, \in \mathcal{C}^{n_R}, \sigma^2 > 0 \\
\mathbf{z} &= \mathbf{H}_k \mathcal{E}[\mathbf{a}_k] \in \mathcal{C}^{n_R \times 1}.
\end{aligned}$$

We used standard minimization techniques to solve the optimization problem formulated in (19) (see the Appendix). In arriving at this solution, we used

$$\begin{aligned}
\mathcal{E}[\mathbf{v}\mathbf{v}^T] &= \sigma^2 \mathbf{I}_{n_R}; \quad \mathcal{E}[\mathbf{a}\mathbf{v}] = \mathbf{0}; \\
\mathcal{E}[a_i a_j] &= \mathcal{E}[a_i]\mathcal{E}[a_j] \quad \forall\, i \neq j.
\end{aligned} \tag{22}$$

$$\lambda^o(b_j(l);p) = \log \frac{P\{b_j(l) = +1 | \lambda^o(\mathbf{C};i), \lambda^o(\mathbf{B};i), \text{decoding}\}}{P\{b_j(l) = -1 | \lambda^o(\mathbf{C};i), \lambda^o(\mathbf{B};i), \text{decoding}\}}, \quad \forall\, j,l \tag{9}$$

$$\lambda^o(c_j(l);p) = \log \frac{P\{c_j(l) = +1 | \lambda^o(\mathbf{C};i), \lambda^o(\mathbf{B};i), \text{decoding}\}}{P\{c_j(l) = -1 | \lambda^o(\mathbf{C};i), \lambda^o(\mathbf{B};i), \text{decoding}\}}, \quad \forall\, j,l. \tag{10}$$

These conditions are achieved by independent and different space-interleaving and time-interleaving applied at the transmitter.

- For the first iteration, we assume $\mathcal{E}[\mathbf{a}_k] = \mathbf{0}$, in which case, (18) reduces to the linear MMSE receiver for substream k

$$x_k = \mathbf{h}_k^H (\mathbf{H}\mathbf{H}^H + \sigma^2 \mathbf{I})^{-1} \mathbf{r}. \tag{23}$$

- With increasing number of iterations, we assume that in the limit, $\mathcal{E}[\mathbf{a}_k] \to \mathbf{a}_k$, in which case, (18) simplifies to a perfect interference canceler

$$x_k = \left(\mathbf{h}_k^H \mathbf{h}_k + \sigma^2\right)^{-1} \mathbf{h}_k^H (\mathbf{r} - \mathbf{H}_k \mathbf{a}_k). \tag{24}$$

Solution 2: The MMSE solution to the weight vector $\mathbf{w}_k$ requires inversion of $n_R \times n_R$ matrices. A sub-optimum solution to Problem 1 is obtained by ignoring the matrix $\mathbf{Q}$ in $\mathbf{w}_k$ as

$$\begin{aligned} x_k =& \mathbf{h}_k^H \left(\left(\mathbf{h}_k \mathbf{h}_k^H + \sigma^2 \mathbf{I} \right)^{-1} \right)^H (\mathbf{r} - \mathbf{H}_k \mathcal{E}[\mathbf{a}_k]) \\ =& \left(\left(\mathbf{h}_k^H \mathbf{h}_k + \sigma^2 \right)^{-1} \right) \mathbf{h}_k^H (\mathbf{r} - \mathbf{H}_k \mathcal{E}[\mathbf{a}_k]). \end{aligned} \tag{25}$$

This solution requires a scalar inversion only. Note that the matrix $\mathbf{Q}$ represents the variance-covariance of the residual interferences.

In practice, we need the channel matrix. Finding a good estimate of the channel matrix is a critical issue in MTMR scheme with large transmit and receive antennas. The use of soft-interference cancellation suffers from large error floors when there channel estimation errors are present. We propose a bootstrapping channel estimation procedure to avoid the error floor. During the first iteration of the receiver, we use a short training sequence to estimate an initial channel matrix. We re-estimate the channel matrix using reliably estimated symbols of each packet at each subsequent iteration and used by the detector to estimate spatial matched filter weights and interferences. The reliably estimated symbols are found by setting a threshold on the output LLRs. If the LLRs of the symbols exceed the threshold, then we use the hard decision of those symbols to update the channel values.

To acquire the expectations of interfering substreams, we use n_T-parallel SISO decoders to provide the *intrinsic* probabilities of the transmitted bit streams. The *intrinsic* probabilities are obtained from the decoder soft outputs of the previous iterations using the relationship

$$\begin{aligned} P(c_j = +1) =& \frac{\exp(\lambda(c_j))}{1 + \exp(\lambda(c_j))} \\ P(c_j = -1) =& \frac{1}{1 + \exp(\lambda(\{c_j\}))} \end{aligned} \tag{26}$$

where $\lambda(c_j)$ is the soft output (formalized as log-likelihood ratio) of symbol c_j provided by the SISO decoder. The expectations of the transmitted bits are

$$\begin{aligned} \mathcal{E}[c_j] =& \frac{(+1)\exp(\lambda(c_j))}{1 + \exp(\lambda(c_j))} + \frac{(-1)}{1 + \exp(\lambda(c_j))} \\ =& \tanh\left(\frac{\lambda(c_j)}{2}\right), \quad j = 1, 2, \ldots, n_T \end{aligned} \tag{27}$$

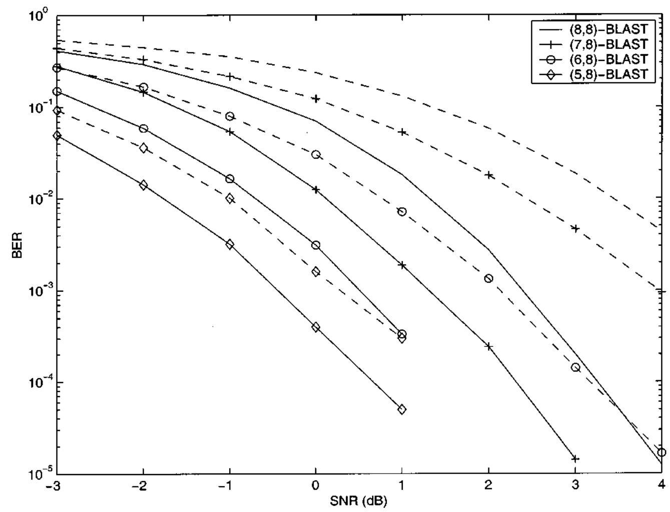

Fig. 6. Bit-error performance for $n_T = 5$, 6, 7, and 8 and $n_R = 8$, using convolutional code with rate $R = 1/2$ and constraint length 3 and QPSK modulation.

IV. Experimental Results

This section compares the performance of QPSK-modulated Turbo-BLAST with that of a correspondingly horizontal coded V-BLAST using real-life indoor channel measurements on various MTMR configurations. The channel measurements were acquired using the narrowband test-bed at Bell Labs of Lucent Technologies, Crawford Hill, NJ, in an indoor environment. At the transmit end, each substream of 100 information bits is independently encoded using a rate-1/2 convolutional code generator (7,5) and then interleaved using space-time interleavers. The space interleavers are designed using diagonal layering interleavers (Fig. 2). The time interleavers are chosen randomly, and no attempt is made to optimize their design. We refer to horizontal coded V-BLAST when each of the substreams is independently coded using rate 1/2 convolutional code with generator (7,5) and QPSK modulated. We synthesize the received signal using the measured channel characteristics and evaluate the performance of Turbo-BLAST over a wide range of SNRs using various BLAST combinations. For the first two experiments, we evaluate the Turbo-BLAST system with the exact channel matrix. In the third experiment, we show the performance with channel estimation using a short training sequence and iterative channel estimates.

Experiment 1—Turbo-BLAST versus V-BLAST, $n_T = 5$, 6, 7, and $n_R = 8$: We consider BLAST configurations with fewer transmit antennas than receive antennas. Fig. 6 compares the bit-error rate performance of Turbo-BLAST (solid trace) and coded V-BLAST (broken trace) for antenna configurations of eight receive and five to eight transmit antennas. Note that the Turbo-BLAST gives the best performance obtained within the first ten iterations. The bit-error performance of both V-BLAST and Turbo-BLAST improves with decreasing number of transmitters with Turbo-BLAST outperforming V-BLAST in all four cases. In terms of V-BLAST performance, a substantial gain in BER performance is realized with fewer transmit antennas. In particular, V-BLAST falls short of Turbo-BLAST performance by a wide margin for more transmit antennas. For ex-

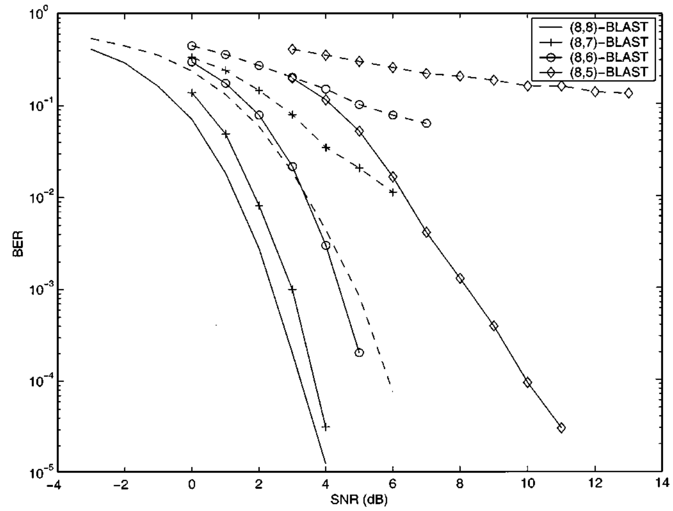

Fig. 7. Bit-error performance for $n_T = 8$ and $n_R = 5$, 6, 7, and 8, using convolutional code with rate $R = 1/2$ and constraint length 3 and QPSK modulation.

ample, Turbo-BLAST achieves 2–3 dB gain over V-BLAST for $n_T = 7$ and $n_R = 8$, whereas only 0.5 dB gain is attained when $n_T = 5$ and $n_R = 8$.

Experiment 2—Turbo-BLAST versus V-BLAST, $n_T = 8$, and $n_R = 5, 6, 7$: We next consider BLAST configurations with fewer receive antennas than transmit antennas. Fig. 7 compares the BER performance of Turbo-BLAST (solid trace) with that of horizontal coded V-BLAST (broken trace). With antenna configurations of eight transmit and five to eight receive antennas, Turbo-BLAST gives the best performance within the first ten iterations. The figure reveals a major limitation of V-BLAST system: the inability to work efficiently with fewer receive antennas than transmit antennas. In the context of Turbo-BLAST, the following observations can be made from Fig. 7: First, the bit-error performance of Turbo-BLAST improves with increasing number of receivers, with Turbo-BLAST outperforming V-BLAST in all four cases. Increasing the number of receivers from seven to eight offers little benefit.

Experiment 3—Turbo-BLAST versus V-BLAST, $n_T = n_R = 8$ and Iterative Channel Estimates: In Figs. 8 and 9, we compare the performances of the following decoders: 1) an iterative decoder with initial channel estimation using 16 training symbols only and 2) an iterative decoder with initial channel estimation and iterative refined channel estimation. The BER performance results are compared for Turbo-BLAST architectures with perfect channel knowledge and with perfect channel and interference knowledge. Fig. 8 shows the BER performance versus SNR of IDD receivers under various conditions considered at iteration 1 and iteration 9. Fig. 9 shows the convergence of the IDD receivers at $(\mathrm{SNR}) = 3$ dB. Although, the BER performance of the decoder with iterative channel estimation is initially (at first iteration) worse than the decoder with channel knowledge, in the fifth iteration of the decoder, it reaches very close to the performance of the decoder with channel knowledge. Moreover, both decoders converge close to the decoder, which has knowledge of both the channel and the interference. The BER performance of the decoder with initial channel estimates only is about 2–4 dB worse than the other schemes because of channel estimation errors.

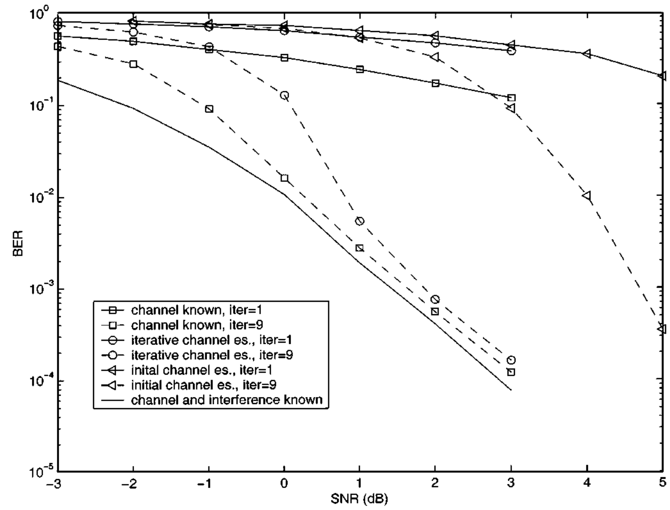

Fig. 8. Bit-error performance with iterative channel estimation for $n_T = 8$ and $n_R = 8$, using convolutional code with rate $R = 1/2$ and constraint length 3 and QPSK modulation.

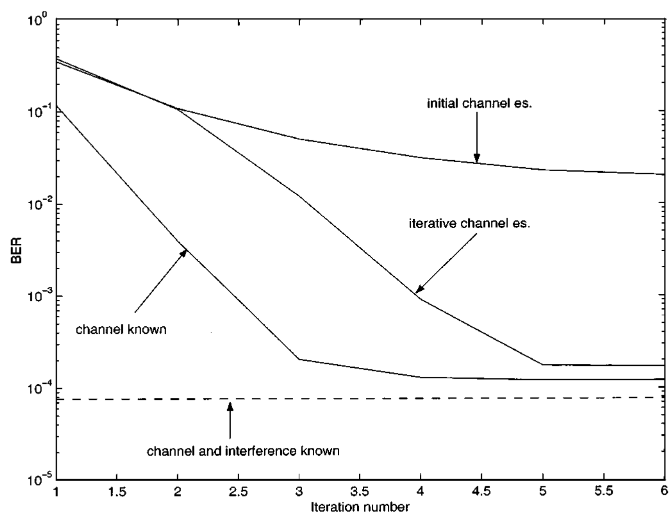

Fig. 9. Convergence behaviors of IDD receivers under various conditions. Bit-error performance with number of iterations for $n_T = 8$ and $n_R = 8$, using convolutional code with rate $R = 1/2$ and constraint length 3 and QPSK modulation.

V. Conclusions

In this paper, we studied Turbo-BLAST and showed that the combination of BLAST and turbo principles provides a reliable and practical solution to high data-rate transmission for wireless communication. Turbo-BLAST is built with 1) simple layered space-time code by using 1-D channel codes and space-time interleavers and 2) iterative detection and decoding receivers. In particular, the decoding algorithm considered in this paper updates the channel estimate at each iteration based on reliable interim hard-decisions of the iterative decoder.

We demonstrated the performance of Turbo-BLAST using real-life wireless channel data with various antenna configurations, including the case of fewer receive antennas than transmit antennas in an indoor environment. The iterative detection decoding receiver with iterative channel estimation improves the BER performance at each iteration rapidly and converges close to a decoder with knowledge of the channel and interference within four to five iterations. Moreover, we have shown that by using real-life data, at a target BER of 10^{-3}, a power gain of 2 to 4 dBs is achieved over the correspondingly coded V-BLAST system.

APPENDIX

Given (15) and (18), find the weight vectors $\mathbf{w}_k$ and u_k by minimizing the cost (convex) function

$$(\hat{\mathbf{w}}_k, \hat{u}_k) = \arg \min_{(\mathbf{w}_k, u_k)} \mathcal{E}\left[\|a_k - x_k\|^2\right] \tag{28}$$

where the expectation is taken over noise and the statistics of the data sequence.

A. Proof

The cost function is written as

$$\begin{aligned} C =& \mathcal{E}\left[\|x_k - a_k\|^2\right] = \mathcal{E}\left[\left\|\left(\mathbf{w}_k^H \mathbf{r} - u_k - a_k\right)\right\|^2\right] \\ =& \mathbf{w}_k^H \mathcal{E}[\mathbf{r}\mathbf{r}^H]\mathbf{w}_k - \mathbf{w}_k^H \mathcal{E}[\mathbf{r}(u+a)^*] \\ &- \mathcal{E}\left[\mathbf{r}(u_k + a_k)^*\right]^H \mathbf{w}_k + \mathcal{E}\left[(u_k + a_k)^2\right] \end{aligned} \tag{29}$$

where

$$\begin{aligned} \mathcal{E}\left[\mathbf{r}\mathbf{r}^H\right] =& \mathcal{E}\Big[\left[\mathbf{h}_k a_k + \bar{\mathbf{H}}_k \mathbf{a}_k + \mathbf{v}\right] \\ & \times \left[\mathbf{h}_k a_k + \bar{\mathbf{H}}_k \mathbf{a}_k + \mathbf{v}\right]^H\Big] \\ =& \mathbf{h}_k \mathbf{h}_k^H + \bar{\mathbf{H}}_k \mathcal{E}\left[\mathbf{a}_k \mathbf{a}_k^H\right] \bar{\mathbf{H}}_k^H + \mathcal{E}[\mathbf{v}\mathbf{v}^H] \end{aligned} \tag{30}$$

and

$$\begin{aligned} \mathcal{E}\left[\mathbf{r}(u_k + a_k)^*\right] =& \mathcal{E}\left[[\mathbf{h}_k a_k + \bar{\mathbf{H}}_k \mathbf{a}_k + \mathbf{v}](u_k + a_k)^*\right] \\ =& \mathbf{h}_k + \bar{\mathbf{H}}_k \mathcal{E}[\mathbf{a}_k] u_k^*. \end{aligned} \tag{31}$$

By assuming that the soft outputs of different substreams are independent, we obtain

$$\mathcal{E}[\mathbf{a}_k \mathbf{a}_k^H] = \mathbf{I}_{n_T - 1} - \mathrm{Diag}\left[\mathcal{E}[\mathbf{a}_k][\mathbf{a}_k]^H\right] + \mathcal{E}[\mathbf{a}_k]\mathcal{E}[\mathbf{a}_k]^H. \tag{32}$$

We use standard minimization techniques to solve the optimization problem formulated in (28). By setting $\partial C/\partial u_k = 0$ and $\partial C/\partial \mathbf{w}_k = 0$ and using (22), we get

$$\begin{aligned} u_k - \mathbf{w}_k^H \bar{\mathbf{H}}_k \mathcal{E}[\mathbf{a}_k] =& 0 \\ u_k =& \mathbf{w}_k^H \mathbf{z} \end{aligned} \tag{33}$$

and

$$\begin{aligned} &\left[\mathbf{h}_k \mathbf{h}_k^H + \bar{\mathbf{H}}_k \left(\mathcal{E}\left[\mathbf{a}_k \mathbf{a}_k^H\right]\right) \bar{\mathbf{H}}_k^H + \mathcal{E}[\mathbf{v}\mathbf{v}^H]\right] \\ &\quad \times \mathbf{w}_k - \bar{\mathbf{H}}_k \mathcal{E}[\mathbf{a}_k] u_k^* = \mathbf{h}_k \\ &(\mathbf{P} + \mathbf{S} + \Sigma_{n_R})\mathbf{w}_k - \mathbf{z} u_k^* = \mathbf{h}_k. \end{aligned} \tag{34}$$

Solving (33) and (34), we get

$$\mathbf{w}_k = (\mathbf{P} + \mathbf{Q} + \Sigma_{n_R})^{-1} \mathbf{h}_k. \tag{35}$$

This completes the proof of (20).

ACKNOWLEDGMENT

The authors wish to express their gratitude to Lucent Technologies, Crawford Hill, NJ, for providing real-life BLAST data, without which the results presented herein would not have been possible.

REFERENCES

[1] C. Berrou and A. Glavieux, “Near optimum error- correcting coding and decoding: Turbo codes,” *IEEE Trans. Commun.*, vol. 44, pp. 1261–1271, Oct. 1996.

[2] J. Hagenauer, “The turbo principle: Tutorial introduction and state of the art,” in *Proc. Int. Symp. Turbo Codes*, Best, France, Sept. 1997.

[3] J. G. Foschini, “Layered space-time architecture for wireless communication in a fading environment when using multi element antennas,” *Bell Labs Tech. J.*, vol. 2, pp. 41–59, Autumn 1996.

[4] V. Tarokh, N. Seshadri, and A. R. Calderbank, “Space-time codes for high data rate wireless communications: Performance criterion and code construction,” *IEEE Trans. Inform. Theory*, vol. 44, pp. 744–765, Mar. 1998.

[5] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, “Space-time block codes from orthogonal design,” *IEEE Trans. Inform. Theory*, vol. 45, pp. 1456–1567, July 1999.

[6] G. D. Golden, J. G. Foschini, R. A. Valenzuela, and P. W. Wolniansky, “Detection algorithm and initial laboratory results using V-BLAST space-time communication architecture,” *Electron. Lett.*, vol. 35, pp. 14–15, Jan. 1999.

[7] M. Sellathurai and S. Haykin, “A nonlinear iterative beamforming technique for wireless communications,” in *33rd Asilomar Conf. Signals, Syst., Comput.*, vol. 2, Pacific Grove, CA, Oct. 1999, pp. 957–961.

[8] ——, “A simplified diagonal BLAST architecture with iterative parallel-interference cancellation,” in *Proc. IEEE Int. Conf. Commun.*, vol. 10, June 2001, pp. 3067–3071.

[9] ——, “Turbo-BLAST: A novel technique for multi-transmit multi-receive wireless communications,” in *Multiaccess, Mobility, and Teletraffic for Wireless Communications*. Boston, MA: Kluwer, 2000, vol. 5, pp. 13–24.

[10] H. El Gamal and A. R. Hammons Jr., “New approach for space-time transmitter receiver design,” in *Proc. 37th Allerton Conf. Commun., Contr., Comput.*, Oct. 1999, pp. 186–194.

[11] S. L. Ariyavisitakul, “Turbo space-time processing to improve wireless channel capacity,” *IEEE Trans. Commun.*, vol. 48, pp. 1347–1359, Aug. 2000.

[12] X. Li, H. Huang, G. J. Foschini, and R. A. Valenzuela, “Effects of iterative detection and decoding on the performance of BLAST,” *Proc. IEEE Globecom*, vol. 1, pp. 1062–1066, Nov. 2000.

[13] W. Choi, K. Cheong, and J. M. Cioffi, “Iterative soft interference cancellation for multiple antenna system,” in *Proc. Wireless Commun. Network Conf.*, vol. 1, 2000, pp. 304–309.

[14] H. El Gamal and A. R. Hammons Jr., “New approach to layered space-time coding and signal processing,” *IEEE Trans. Inform. Theory*, vol. 47, pp. 2321–2334, Sept. 2001.

[15] T. Mexia and S. K. Cheng, “Low complexity post-ordered iterative decoding for generalized layered space-time coding systems,” in *Proc. IEEE Int. Conf. Commun.*, vol. 4, June 2001, pp. 1137–1141.

[16] S. Baro, G. Bauch, A. Pavlic, and A. Semmler, “Improving BLAST performance using space-time block codes and turbo decoding,” *Proc. IEEE Globecom*, vol. 2, pp. 1061–1066, Nov. 2000.

[17] M. Moher, “An iterative multiuser decoder for near capacity communications,” *IEEE Trans. Commun.*, vol. 46, pp. 870–880, July 1998.

[18] X. Wang and V. Poor, “Iterative (TURBO) soft interference cancellation and decoding for coded CDMA,” *IEEE Trans. Commun.*, vol. 47, pp. 1046–1061, July 1999.

[19] H. El Gamal and E. Geraniotis, “Iterative multiuser detection for coded CDMA signals in AWGN and fading channel,” *IEEE J. Select. Areas Commun.*, vol. 18, pp. 30–41, Jan. 2000.

Mathini Sellathurai (S'95–M'02) received the Tech. Lic. degree in electrical engineering from the Royal Institute of Technology, Stockholm, Sweden, in 1997 and the Ph.D. degree in electrical engineering from McMaster University, Hamilton, ON, Canada, in 2001.

She is currently with Communications Research Centre Canada, Ottawa, as a Senior Research Scientist. She was with Lucent Technologies, Bell Labs, as a Visitong Researcher, working in wireless communications. Her research interests include adaptive signal processing, space-time and multimedia wireless communications, information theory, and channel coding.

Dr. Sellathurai was awarded the doctoral price in engineering and computer sciences from the Natural Sciences and Engineering Research Council of Canada for her Ph.D. dissertation.

Simon Haykin (F'86) received the B.Sc. degree with First Class Honors in 1953, the Ph.D. degree in 1956, and the D.Sc. degree in 1967, all in electrical engineering, from the University of Birmingham, Birmingham, U.K.

He is a University Professor with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada, which is the highest academic rank at the University. His current research interests include the applications of learning and adaptive signal processing to space-time wireless communications, radar, and hearing systems for the hearing impaired.

Dr. Haykin is a Fellow of the Royal Society of Canada and the recipient of the Honorary degree of Doctor of Technical Science from ETH, Zurich, Switzerland.