Markov Chain Monte Carlo: Applications to MIMO detection and channel equalization

Rong-Rong Chen, Ronghui Peng, and Behrouz Farhang-Boroujeny
Department of Electrical and Computer Eng.
University of Utah
Salt Lake City, UT 84093, USA
Email: {rchen, peng, farhang}@ece.utah.edu

***Abstract*— In this paper, we present an overview of recent work on the applications of Markov Chain Monte Carlo (MCMC) techniques to both multiple-input and multiple-output (MIMO) detection and channel equalization. In the setting of MIMO detection, we have shown that, even for very large antenna systems with high spectral efficiencies of 24 bits/channel use (8 transmit and 8 receive antennas with 64 QAM modulation), the MCMC MIMO detector can bring us within 2 dB of the channel capacity with a greatly reduced complexity compared to several versions of sphere decoding based detectors. For frequency selective channels, we demonstrate that MCMC-based equalizers yield excellent performance even for severe inter-symbol-interference (ISI) channels. The MCMC equalizer achieves significant performance gain over minimum mean square error (MMSE) linear equalizer and performs closely to the optimal maximum *a posteriori* probability (MAP) equalizer. We will also discuss new approaches that effectively alleviate the well-known high SNR problems in existing MCMC detectors.**

I. Introduction

In this paper, we review recent results on the application of Markov Chain Monte Carlo (MCMC) techniques to multiple-input multiple-output (MIMO) detection and channel equalization. The MCMC simulation is a mathematical tool that can be used to draw samples from an arbitrary and possibly unknown distribution. The MCMC detectors follow a statistical search procedure called the Gibbs sampler (GS) to randomly generate a small sample set containing the most likely transmitted signal vectors. A distinguished feature of the MCMC detectors considered here is that they can achieve excellent performance with a very small sample set. This makes the low-complexity MCMC detectors highly attractive when the complexity of the optimal maximum *a posteriori* probability (MAP) detector grows exponentially with the number of antennas, constellation size, and channel memory.

We first discuss the MCMC MIMO detectors studied in [1]–[7]. The MCMC detectors demonstrate superior performance than linear detectors such as minimum mean square error (MMSE) and zero-forcing (ZF) [8] detectors, and the more sophisticated list sphere decoding (LSD) based detectors and its variants [9]–[11]. The linear detectors achieve low computational complexity, however, at the expense of significant performance loss. The widely used LSD detectors, while perform well, have a complexity that grows exponentially as the number of transmit antenna increases [12]. The MCMC detectors are shown to achieve near capacity performance at high spectral efficiencies with much lower complexity than the LSD based detectors. As opposed to LSD detectors that perform well at high signal-to-noise ratio (SNR), the MCMC detector suffers from the high SNR problem. Possible remedies for addressing the high SNR problem of MCMC detectors can be found in recent work [5], [7].

We will also review recent work on channel equalization using MCMC techniques [13], [14]. For frequency selective channels with inter-symbol-interference (ISI), [14] shows that, as opposed to the bit-wise GS or symbol-wise GS that generates one bit, or one symbol at a time, a more complex version of the GS that examines a group of symbols at a time is necessary to achieve good performance for channels with significant ISI. Such MCMC detectors demonstrate superior performance than the MMSE-based equalizer of [15].

II. MIMO Detection

We consider a MIMO channel with t transmit and r receive antennas. The channel model is given by :

$$\mathbf{y} = \sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d} + \mathbf{n}, \tag{1}$$

where $\mathbf{d} = (d_1, \cdots, d_t)^T$ and d_j is the complex symbol transmitted from the j-th antenna. The notation $(\cdot)^T$ denotes the transpose operator. $\mathbf{y} = (y_1, \cdots, y_r)^T$ is the received signal vector; $\mathbf{H}$ is the r by t channel gain matrix with i.i.d. complex Gaussian $\mathcal{CN}(0,1)$ distributed entries; $\mathbf{H}$ is also assumed to be independent over time; the channel noise vector $\mathbf{n}$ has i.i.d. $\mathcal{CN}(0,1)$ distributed entries; ρ denotes the SNR per receive antenna. Throughout this paper, we use lowercase bold letters to represent vectors and uppercase bold letters to represent matrices.

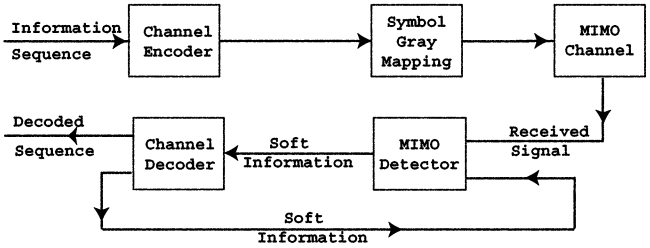

Fig. 1. A block diagram of the MIMO system.

Fig. 1 shows a block diagram of the MIMO system. At the transmitter side, the channel encoder encodes a sequence of information bits into a sequence of coded bits, which is then mapped to a sequence of complex symbols through Gray mapping. Here, we assume that the size of the complex constellation is $M = 2^{M_c}$ so that every M_c bits is mapped to a complex symbol. The sequence of complex symbols is then divided into blocks of t symbols and sent through t transmit antennas over the MIMO fading channel. At the receiver end, for each set of received signal samples from the r receive antennas, the MIMO detector computes the log-likelihood-ratios (LLRs) of the coded bits. The "extrinsic" part of these LLRs is passed to the channel decoder for decoding. At the next iteration, the LLRs produced by the channel decoder are fed back to the MIMO detector to produce updated LLRs. In this manner, joint MIMO detection and channel decoding proceeds iteratively through soft information exchange. After a pre-determined number of iterations, decisions are made at the output of the channel decoder to generate the decoded bit sequence.

A. Capacity-approaching MCMC MIMO detector

Assume that the bit sequence corresponding to $\mathbf{d}$ is denoted by $\mathbf{b} = (b_0, b_1, \cdots, b_{K-1})^T$, where $K = tM_c$. Let $\boldsymbol{\lambda} = (\lambda_0, \cdots, \lambda_{K-1})^T$, where λ_i is the LLR of the i-th bit provided by the channel decoder. In a MAP detector, given $\mathbf{y}$ and $\boldsymbol{\lambda}$, the LLR value of a particular bit b_k is given by

$$\begin{aligned} \gamma_k &= \ln \frac{P(b_k = 0|\mathbf{y}, \boldsymbol{\lambda})}{P(b_k = 1|\mathbf{y}, \boldsymbol{\lambda})} \\ &= \ln \frac{\sum\limits_{\bar{\mathbf{b}}_k} P(b_k = 0, \bar{\mathbf{b}}_k|\mathbf{y}, \boldsymbol{\lambda})}{\sum\limits_{\bar{\mathbf{b}}_k} P(b_k = 1, \bar{\mathbf{b}}_\mathbf{k}|\mathbf{y}, \boldsymbol{\lambda})} \end{aligned} \tag{2}$$

where $\bar{\mathbf{b}}_k = (b_0, \cdots b_{k-1}, b_{k+1}, \cdots, b_{K-1})$; $b_j \in \{0, 1\}$.

Computation of each summation in (2) is over all combinations of $\bar{\mathbf{b}}_k$ which is equal to 2^{K-1}. Hence, the complexity of the optimal MAP detector grows exponentially in K. The MCMC detector reduces the detection complexity by utilizing Gibbs samplers to generate a small sample set $\mathcal{B}$ containing most likely bit sequences. The output LLRs are computed from samples in $\mathcal{B}$. Hence, the computational complexity is reduced greatly compared to the MAP detector.

The main steps of the bit-wise MCMC detector (b-MCMC) [6] are summarized in Algorithm 1. We run D Gibbs sampler in parallel. Each Gibbs sampler performs I iterations to generate I bit vectors $\{\mathbf{b}^{(n)}, n = 1, \cdots I\}$. Hence, a total of $D \cdot I$ bit vectors will be generated and they constitute (excluding repetitious samples) the sample set $\mathcal{B}$.

To compute the output LLR for bit k, we define an expanded set $\mathcal{B}^k$ which includes not only all bit vectors in $\mathcal{B}$, but also new bit vectors that are obtained by flipping the k-th bit of the vectors in $\mathcal{B}$. We then let $\mathcal{B}^k_{+1}$ and $\mathcal{B}^k_{-1}$ denote the set of bit vectors in $\mathcal{B}^k$ whose k-th bit is $+1$ and -1, respectively. Considering the larger set $\mathcal{B}^k$ instead of $\mathcal{B}$ assures that the number of elements in the subset $\mathcal{B}^k_{+1}$ and $\mathcal{B}^k_{-1}$ are the same.

Algorithm 1: b-MCMC MIMO detector

Input : Prior LLR for transmitted symbols (from the decoder)
Output: Extrinsic LLR for channel decoder

Use bit-wise Gibbs sampler to generate samples
  `// D parallel Gibbs sampler`
  repeat
    Generate initial sequence $\mathbf{b}^{(0)}$
    `// I iterations`
    for $n \leftarrow 1$ **to** I **do**
      Generate $b_0^{(n)}$ from the distribution $P(b_0^{(n)} = b|b_1^{(n-1)}, b_2^{(n-1)}, \cdots, b_{K-1}^{(n-1)}, \mathbf{y}, \lambda_0)$
      Generate $b_1^{(n)}$ from the distribution $P(b_1^{(n)} = b|b_0^{(n)}, b_2^{(n-1)}, \cdots, b_{K-1}^{(n-1)}, \mathbf{y}, \lambda_1)$
      $\vdots$
      Generate $b_{K-1}^{(n)}$ from the distribution $P(b_{K-1}^{(n)} = b|b_0^{(n)}, b_1^{(n)}, \cdots, b_{K-2}^{(n)}, \mathbf{y}, \lambda_{K-1})$
      Save $\mathbf{b}^{(n)}$ into sample set $\mathcal{B}$
  until *D times* ;

Compute the LLR
  for $k \leftarrow 0$ **to** $K - 1$ **do**
    Compute extrinsic LLR for b_k using (3) or (4).

This is necessary for successful operation of the various Max-Log and Log-MAP detectors studied in this paper.

For Max-Log b-MCMC, the output LLR for bit k is computed as

$$\begin{aligned} \mathrm{LLR}_k \approx & \max_{\{\mathbf{b}:\, \mathbf{b} \in \mathcal{B}^k_{+1}\}} \left\{ -\left\|\mathbf{y} - \sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\right\|^2 + \frac{1}{2}\boldsymbol{\lambda}^T\mathbf{b} \right\} \\ & - \max_{\{\mathbf{b}:\, \mathbf{b} \in \mathcal{B}^k_{-1}\}} \left\{ -\left\|\mathbf{y} - \sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\right\|^2 + \frac{1}{2}\boldsymbol{\lambda}^T\mathbf{b} \right\}. \end{aligned} \tag{3}$$

where $\mathbf{d}(\mathbf{b})$ denotes the symbol vector corresponding to the bit vector $\mathbf{b}$.

Alternatively, [6] shows that replacing the Max-Log algorithm by the more accurate Log-MAP algorithm with table look-up (Log-MAP-tb) yields superior performance while requiring less number of samples. For the Log-MAP-tb b-

MCMC, the output LLR is computed using

$$\mathrm{LLR}_k \approx \ln \sum_{\{\mathbf{b}:\, \mathbf{b}\in \mathcal{B}_{+1}^k\}} \exp\Big\{ -\Big\|\mathbf{y}-\sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\Big\|^2 + \frac{1}{2}\boldsymbol{\lambda}^T\mathbf{b}\Big\} - \ln \sum_{\{\mathbf{b}:\, \mathbf{b}\in \mathcal{B}_{-1}^k\}} \exp\Big\{ -\Big\|\mathbf{y}-\sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\Big\|^2 + \frac{1}{2}\boldsymbol{\lambda}^T\mathbf{b}\Big\}. \tag{4}$$

To further reduce the complexity, we approximate (4) using the Jacobian logarithm [16]

$$\begin{aligned}\ln(e^{\delta_1}+e^{\delta_2}) &= \max(\delta_1,\delta_2) + \ln(1+e^{-|\delta_2-\delta_1|}) \\ &= \max(\delta_1,\delta_2) + f_c(|\delta_1-\delta_2|),\end{aligned} \tag{5}$$

where the correction function $f_c(|\delta_1-\delta_2|)$ is approximated using a one-dimensional pre-computed table. Our simulation results show that it suffices to use a very small table which stores only ten values with $|\delta_1-\delta_2|$ equally spaced between 0 and 5. We refer to the b-MCMC detector that utilizes (4) based on table look-up (5) as the Log-MAP-tb b-MCMC.

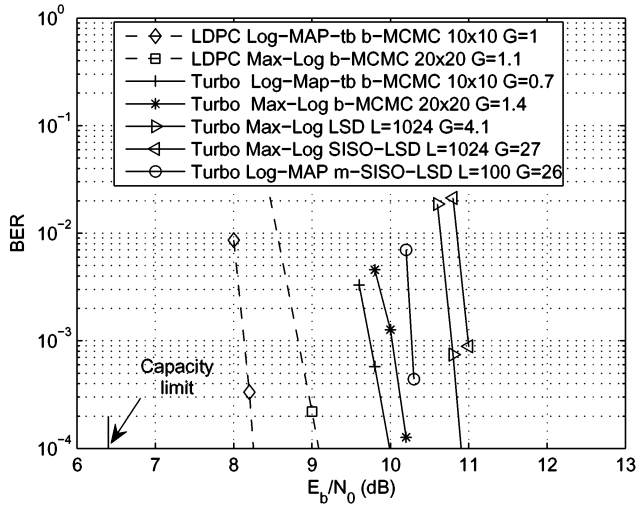

Fig. 2. Performance of turbo and LDPC coded TX8 64QAM systems.

Fig. 2 shows performance curves of b-MCMC detectors for an eight transmit and eight receive antenna system with 64QAM modulation. We compare the MCMC detectors with several versions of the LSD based detectors including the Max-Log LSD [9] ($L = 1024$), the Max-Log soft-in soft-out LSD (SISO-LSD) [10] ($L = 1024$), and the modified versions of these two detectors, denoted by Log-MAP-tb LSD ($L = 100$) and Log-MAP-tb SISO-LSD ($L = 100$), obtained by replacing the Max-Log by Log-MAP-tb. Here L denotes the list size. We use G to denote the normalized simulation time of each system against the LDPC coded system that employs the 10x10 Log-MAP-tb b-MCMC detector with $D = I = 10$. The LDPC code used here is optimized for the 10x10 Log-MAP-tb b-MCMC detector using the EXIT chart method [17]. Please refer to [6] for the details of the code parameters and discussions on code design. Fig. 2 shows that for both turbo coded and LDPC coded systems, the same 10x10 Log-MAP-tb b-MCMC detector gives the best performance with the lowest complexity. The LDPC coded system using 10x10 Log-MAP-tb b-MCMC achieves within 1.8 dB of the channel capacity (6.4 dB), which reduces the previously known capacity gap (4 dB [1]) by more than 50%. Its simulation time is $G = 0.7 \sim 1$ for the turbo coded system and the LDPC coded system, respectively. In comparison, simulation time of LSD based detectors are $G = 4.1 \sim 27$ for the turbo coded system. Since the simulation time of LSD based detectors increases even further at lower SNR, it becomes prohibitive to find good LDPC codes and evaluate LDPC coded performance for these detectors. This result clearly demonstrate the MCMC detector as the detector of choice for approaching channel capacity in both performance and complexity.

B. MCMC MIMO detectors for high SNR

It is well-known [1], [5], [18] that MCMC methods become less effective as the operating SNR increases. This is because at higher SNRs, the Markov chain associated with the GS is likely to become reducible so that it tends to get stuck in certain states and fail to reach the states corresponding to small distances $\left\|\mathbf{y}-\sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\right\|^2$. One approach to alleviate this problem is to assume a higher noise variance than the actual noise variance when running the GS [1]. This increases the chances that the GS converge to the optimal solution. In [5], it is shown that detection performance can be further improved by initializing one of the GSs using the MMSE or ZF solution. Recently, [7] proposes a constrained MCMC detector for high SNR scenarios. The main idea is as follows: First, the GS is initialized from either a ZF solution $b_{\mathrm{ZF}}^{(0)}$, or a ML solution $b_{\mathrm{ML}}^{(0)}$. The ML solution can be found by running a hard sphere decoder [9]. Assume that the GS is initialized using $b_{\mathrm{ML}}^{(0)}$. Now consider an arbitrary bit k. When there is no prior from the channel decoder, the exact Max-Log algorithm for computing the LLR of bit k is given by

$$\begin{aligned}\mathrm{LLR}_k = &\max_{\{\mathbf{b}:\, b_k=1\}} \Big\{ -\Big\|\mathbf{y}-\sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\Big\|^2\Big\} \\ &- \max_{\{\mathbf{b}:\, b_k=-1\}} \Big\{ -\Big\|\mathbf{y}-\sqrt{\frac{\rho}{t}}\mathbf{H}\mathbf{d}(\mathbf{b})\Big\|^2\Big\},\end{aligned} \tag{6}$$

If $b_{\mathrm{ML},k}^{(0)} = 1$, then the first maximum in (6) is achieved by $b_{\mathrm{ML}}^{(0)}$. We refer to the vector that attains the second maximum in (6) as the N-ML solution. The N-ML solution needs to be found in order to compute the exact LLR_k. In [7], it is shown that we can get good approximations of the N-ML solution by running constrained MCMC. As shown in Algorithm 2, for each bit k, we can run a GS starting from an initial vector that is modified from $b_{\mathrm{ML}}^{(0)}$. Then we apply the GS to update one bit at a time while keeping the value of bit k constrained (let $b_k = -1$ if $b_{\mathrm{ML},k}^{(0)} = 1$). At high SNR, it is shown in [7] that a few iterations are sufficient for the GS to get close to the N-ML solution.

Algorithm 2: ML-C-MCMC detector

Input : Prior LLR for transmitted symbols (from the decoder)

Output: Bit sequences used for computing extrinsic LLR

Use bit-wise Gibbs sampler to generate sample sets

  Generate initial ML vector $\mathbf{b}_{\text{ML}}^{(0)}$ using hard output sphere decoding, save into important set $\mathcal{B}$

  for $k \leftarrow 0$ **to** $K-1$ **do**

    $d = \lfloor k/M_b \rfloor$

    Flip the k-th bit and change the values of remaining bits belonging to the d-th symbol so that the resulting symbol is the closest to the d-th symbol in $\mathbf{b}_{\text{ML}}^{(0)}$. Denote the new vector by $\mathbf{b}^{(0)}$

    `// I iterations`

    for $n \leftarrow 1$ **to** I **do**

      Generate $b_0^{(n)}$ from the distribution $P(b_0 = b|b_1^{(n-1)}, b_2^{(n-1)}, \cdots, b_{K-1}^{(n-1)}, \mathbf{y})$

      Generate $b_1^{(n)}$ from the distribution $P(b_1 = b|b_0^{(n)}, b_1^{(n-1)}, \cdots, b_{K-1}^{(n-1)}, \mathbf{y})$

      $\vdots$

      Fix the value of the k-th bit: $b_k^{(n)} = b_k^{(n-1)}$

      Generate $b_{k+1}^{(n)}$ from the distribution $P(b_{k+1} = b|b_0^{(n)}, \cdots, b_k^{(n)}, b_{k+2}^{(n-1)}, \cdots, b_{K-1}^{(n-1)}, \mathbf{y})$

      $\vdots$

      Generate $b_{K-1}^{(n)}$ from the distribution $P(b_{K-1} = b|b_0^{(n)}, b_1^{(n)}, \cdots, b_{K-2}^{(n)}, \mathbf{y})$

      Save $\mathbf{b}^{(n)}$ into the important set $\mathcal{B}$

Compute the LLR

  for $k \leftarrow 0$ **to** $K-1$ **do**

    Compute extrinsic LLR for b_k using (3).

In Fig. 3 we compare performance of ZF initialized MCMC (ZF-MCMC) [5], the ZF initialized constrained MCMC (ZF-C-MCMC), the ML initialized constrained MCMC (ML-C-MCMC) [7], LSD, and the exact Max-Log detector. It is assumed that MIMO detection is performed only once, followed by channel decoding. The ZF-MCMC runs 11 GSs with 3 iterations each. For both ZF-C-MCMC and ML-C-MCMC, 16 constrained GSs are ran (each corresponding to a fixed bit k) with 2 iterations each. It is clear that the ZF-C-MCMC is at least one order of magnitude better than the non-constrained ZF-MCMC [5]. The ML-C-MCMC improves performance of ZF-C-MCMC at the cost of additional complexity needed to find the ML solution. We also compare the performance of

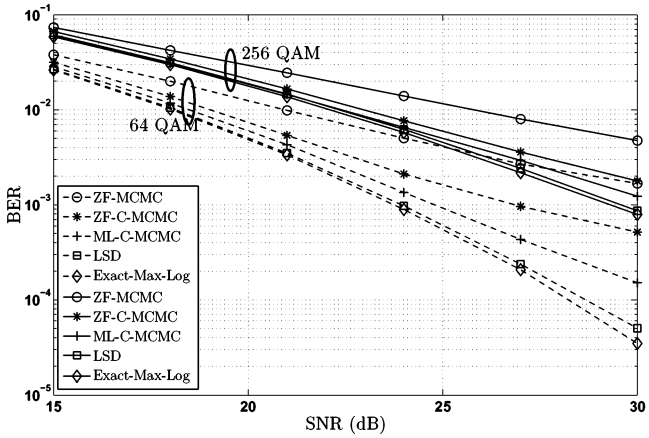

Fig. 3. Performance comparisons of ZF-MCMC, ZF-C-MCMC ML-C-MCMC, LSD, and exact Max-Log detector. Assume a two transmit, two receive MIMO channel with 64 QAM or 256 QAM constellations.

the MCMC detectors with the LSD detector of [9], using the same list size as in [9]. The LSD detector performs comparable to the ML-C-MCMC when the QAM alphabet is large. For lower order QAM modulations, the LSD still outperforms the ML-C-MCMC detector at the expense of much higher implementation complexity. Finally, the performance curve of the exact Max-log detector is presented as a performance benchmark. The ML and N-ML solutions for each bit b_k can be found by running a sphere decoder presented in [9] or [10].

III. Channel Equalization

We consider a single antenna ISI channel given by

$$y_n = \sum_{l=0}^{L} h_l x_{n-l} + z_n, \quad n = 0, 1, \cdots, N+L-1, \tag{7}$$

where L is the channel memory, h_l denotes the channel gain of the l-th tap, $x_0, \cdots, x_{N-1}$ are the transmitted symbols, y_n is the received signal at time n. The channel noise $\{z_n\}$ is i.i.d, complex Gaussian with zero mean and variance of $\sigma^2 = N_0/2$ per dimension. For notational convenience, in (7), we let $x_n = 0$ for $-L \le n \le -1$ and $N \le n \le N+L-1$.

Assume that the channel gain $\{h_l\}$ is known perfectly to the receiver. Let $\mathbf{y} = \{y_0, y_1, \cdots, y_{N+L-1}\}, \mathbf{x} = \{x_0, x_1, \cdots, x_{N+L-1}\}$ and $\mathbf{x}_{j-L:j} = \{x_{j-L}, x_{j-L+1}, \cdots, x_j\}$. Since the channel has a finite memory of L, the conditional pdf of the $\mathbf{y}$ given $\mathbf{x}$ can be written as

$$\begin{aligned} p(\mathbf{y}|\mathbf{x}) &= \prod_{j=0}^{N+L-1} p(y_j|\mathbf{x}_{j-L:j}) \\ &= \prod_{j=0}^{N+L-1} \exp\Big(-\frac{1}{2\sigma^2}|y_j - \sum_{l=0}^{L} h_l x_{j-l}|^2\Big) \end{aligned} \tag{8}$$

In [14], various versions of MCMC based channel equalizers are proposed. In particular, a group-based MCMC equalizer, denoted by g-MCMC, is shown to greatly improve the

convergence rate of the GS. It outperforms bit-wise MCMC equalizer under channels with significant ISI.

The main steps of the g-MCMC equalizer is summarized in Algorithm 3. The key idea is to group every G symbols $\mathbf{x}_{i:i+G-1} := \{x_i, \cdots, x_{i+G-1}\}$ together and update these symbols simultaneously inside the Gibbs sampler. We first identify a sample set $\mathcal{S} = \{\mathbf{u}^1, \cdots, \mathbf{u}^r\}$, where each $\mathbf{u}^l$ is a symbol vector of length G. For small values of $G \leq 2$, we let $\mathcal{S}$ contain all possible choices of $\mathbf{x}_{i:i+G-1}$ and the size of $\mathcal{S}$ is $r = 2^{M_b G}$. For larger values of $G \geq 3$, $\mathcal{S}$ is found by applying the QRD-M algorithm [19] with the M parameter equals r. Each time a random sample vector $\mathbf{x}_{i:i+G-1}$ is generated from the sample set $\mathcal{S}$ following a probability mass function (PMF) $\{\gamma_l, l = 1, \cdots, r\}$ over $\mathcal{S}$. We compute $\{\gamma_l\}$ according to

$$\gamma_l = C \prod_{j=i}^{i+L+G-1} p(y_j | \mathbf{x}_{j-L:j}^l) P(\mathbf{x}_{i:i+G-1} = \mathbf{u}^l), \quad (9)$$

where C is a scaling constant such that $\sum_{l=1}^{r} \gamma_l = 1$, and $\mathbf{x}^l = (x_0^{(n)}, \cdots x_{i-1}^{(n)}, \mathbf{u}^l, x_{i+\mathrm{G}}^{(n-1)}, \cdots x_{N-1}^{(n-1)})$.

In Algorithm 3, $G_{\max}$ is the maximal number of symbols allowed for group updating, $a\%b$ denotes the remainder of a divided by b, and $\lfloor a \rfloor$ denotes the maximal integer less than a. Note that line 3 of Algorithm 3 allows us to group different adjacent symbols over iterations. As shown in simulation results, this is necessary to speed up the mixing rate of GS.

Assume that the GS produces a sample set $\mathcal{B}$. Each element in $\mathcal{B}$ is a bit sequence of length $M_b N$. Assume that bit k is mapped to symbol x_i, then the received signals that are affected by b_k are $\mathbf{y}_{i:i+L}$. Since $\mathbf{y}_{i:i+L}$ depends only on bits $\{b_l,\ i_1 = M_b(i-L) \leq l \leq M_b(i+L) = i_2\}$, we find that when computing the output LLR for b_k, it is sufficient to truncate each sequence in $\mathcal{B}$ to take into account only bits $\{b_l,\ i_1 \leq l \leq i_2\}$. We denote the set that contains the truncated sequences by $\mathcal{B}_{i_1:i_2}$. For each $0 \leq k \leq M_b N - 1$, we construct a larger set $\mathcal{B}_{i_1:i_2}^k$ which includes all sequences in $\mathcal{B}_{i_1:i_2}$, together with new sequences that are obtained by flipping the k-th bit of each sequence in $\mathcal{B}_{i_1:i_2}$. Repetitious sequences are removed from $\mathcal{B}_{i_1:i_2}^k$. Furthermore, we let $\mathcal{B}_{i_1:i_2}^{k,+1}$ and $\mathcal{B}_{i_1:i_2}^{k,-1}$ denotes sequences in $\mathcal{B}_{i_1:i_2}^k$ whose k-th bit equals 1 and -1, respectively. The LLR for bit b_k is then computed as

$$\gamma_k = \ln \frac{\sum_{\mathbf{b}_{i_1:i_2} \in \mathcal{B}_{i_1:i_2}^{k,+1}} p(\mathbf{y}_{i:i+L} | \mathbf{b}_{i_1:i_2}) \prod_{l=i_1}^{i_2} P(b_l)}{\sum_{\mathbf{b}_{i_1:i_2} \in \mathcal{B}_{i_1:i_2}^{k,-1}} p(\mathbf{y}_{i:i+L} | \mathbf{b}_{i_1:i_2}) \prod_{l=i_1}^{i_2} P(b_l)} \quad (10)$$

In Fig. 4 we compare performance of the proposed MCMC equalizers with the optimal MAP equalizer and the MMSE equalizer of [15]. The system diagram is similar to the one shown in Fig. 1 with the MIMO channel replaced by an ISI channel and the MIMO detector replaced by an equalizer. We consider a channel with significant ISI (taken from [20]) whose

Algorithm 3: Group-wise Gibbs sampler

Input : Prior LLR for transmitted symbols (from decoder)

Output: Bit sequences used for computing extrinsic LLR

1 Generate initial sequence $\mathbf{x}^{(0)}$;
2 **for** $n \leftarrow 1$ **to** I **do**
3 $G_0 \leftarrow n\%G_{\max}$, $J \leftarrow \lfloor (N - G_0)/G_{\max} \rfloor$;
4 Generate $\mathbf{x}_{0:G_0-1}^{(n)}$ from the distribution
5 $P(\mathbf{x}_{0:G_0-1} | x_{G_0}^{(n-1)}, \cdots, x_{N-1}^{(n-1)})$
6 Generate $\mathbf{x}_{G_0:G_0+G_{\max}-1}^{(n)}$ from the distribution
7 $P(\mathbf{x}_{G_0:G_0+G_{\max}-1} | x_0^{(n)}, \cdots, x_{G_0-1}^{(n)}, x_{G_0+G_{\max}}^{(n-1)}, \cdots, x_{N-1}^{(n-1)})$
8 $\vdots$
9 Generate $\mathbf{x}_{G_0+JG_{\max}:N-1}^{(n)}$ from the distribution
10 $P(\mathbf{x}_{G_0+JG_{\max}:N-1} | x_0^{(n)}, \cdots, x_{G_0+J\cdot G_{\max}-1}^{(n)})$
11 **SubFunction: Generate $\mathbf{x}_{i:i+G-1}$ from the distribution P**
12 **if** $G < 3$ **then**
13 **foreach** *combination of* $\boldsymbol{x}_{i:i+G-1}$ **do**
14 Compute the PMF using (9)
15 Select one combination randomly according to PMF $\{\gamma_1, \cdots, \gamma_r\}$.
16 **else**
17 Perform QRD-M over to select r combinations with maximal probabilities
18 Generate one sample from the selected r combinations randomly according to PMF (9).
19
20 **Compute the LLR**
21 **for** $k \leftarrow 0$ **to** $M_b N - 1$ **do**
22 Construct $\mathcal{B}_{i_1:i_2}^{k,+1}$ and $\mathcal{B}_{i_1:i_2}^{k,-1}$.
23 Compute extrinsic LLR for b_k using (10)
24

channel impulse response is given by

$$h_1[n] = 0.227\delta[n] + 0.46\delta[n-1] + 0.688\delta[n-2] + 0.46\delta[n-3] + 0.227\delta[n-4]$$

We assume that the receiver knows the channel response perfectly. The average energy per bit to noise ratio is defined as:

$$\frac{E_b}{N_0} \triangleq \frac{E_s}{N_0 R M_c} = \frac{E(|y_n|^2)}{N_0 R M_c} = \frac{\sum_{l=0}^{L-1} |h_l|^2}{2\sigma^2 R M_c}$$

We use a rate $1/2$ convolutional code with generator polynomial $(1 + D^2, 1 + D + D^2)$. The code length of 4098. The bit sequence is mapped to a sequence of 8-PSK symbols using gray mapping. The channel interleaver is obtained by an S-random interleaver [21] with $S = 0.5\sqrt{0.5K_c}$, where K_c is the number of coded bits. In Fig. 4, we plot the performance curves for the case of separate equalization and

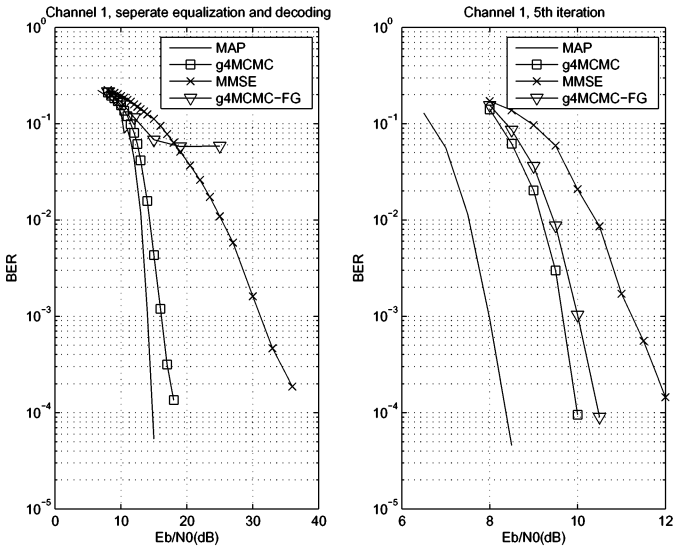

Fig. 4. Performance comparisons of MAP, MMSE, and MCMC equalizers over a channel with significant ISI.

decoding, where equalization is performed only once, followed by channel decoding, and the case of five iterations of joint equalization and decoding. The g-MCMC equalizer uses 10 parallel Markov chain and each chain runs 20 iterations. The maximal group updating symbols of the g-MCMC is 4, QRD-M is used to reduce the complexity and $r = 8$. Fig. 4 shows that the proposed g-MCMC equalizer significantly outperforms the MMSE equalizer. For separate equalization and decoding, the MMSE equalizer has a huge gap (more than 20 dB) from the MAP detector, while the g-MCMC is less than 4 dB worse. After 5 iterations, the g-MCMC is only 1.7 dB away from the MAP equalizer and still performs 2 dB better than the MMSE equalizer at BER 10^{-4}. The g4MCMC-FG detector shown in Fig. 4 assumes fixed grouping, i.e., (G_0 is always set to be zero in Algorithm 3). It is clear that g4MCMC-FG performs much worse than the g4MCMC that groups different symbols over iterations. The performance gap between g4MCMC-FG and g4MCMC is significant for the case of separate equalization and decoding.

IV. Conclusion

In this paper, we summarize our recent results on soft-in soft-out MCMC detectors for MIMO channels and frequency selective channels. It is shown that in both cases, MCMC techniques are highly effective in achieving good performance with an acceptable (relatively low) receiver complexity.

It will be interesting to further explore MCMC techniques for continuously time-varying channels with imperfect channel state information. Theoretical analysis of the convergence rate of MCMC detectors remains to be a challenging research direction.

References

[1] B. Farhang-Boroujeny, H. Zhu, and Z. Shi, "Markov chain Monte Carlo algorithms for CDMA and MIMO communication systems," *IEEE Trans. Signal. Process.*, vol. 54, no. 5, pp. 1896–1909, May 2006.

[2] H. Zhu, Z. Shi, and B. Farhang-Boroujeny, "MIMO detection using Markov Chain Monte Carlo techniques for near capacity performance," in *Proc. IEEE Int. Conf. on Acoustic, Speech, and Signal Processing (ICASSP'05)*, Philadelphia, PA, Mar. 2005, pp. 1017–1020.

[3] Z. Shi, H. Zhu, and B. Farhang-Boroujeny, "Markov chain Monte Carlo techniques in iterative detectors," in *Proc. IEEE Globecom'04*, Dallas, TX, Nov./Dec. 2004, pp. 325–329.

[4] H. Zhu, B. Farhang-Beroujeny, and R. R. Chen, "On performance of sphere decoding and Markov Chain Monte Carlo methods," *IEEE Signal Processing Letters*, vol. 12, no. 10, pp. 669–672, Oct. 2005.

[5] X. Mao, P. Amini, and B. Farhang-Boroujeny, "Markov Chain Monte Carlo MIMO Detection Methods for High Signal-to-Noise Ratio Regimes," in *Proc. IEEE Globe Conf. on Comm. (Globecom'07)*, Nov. 2007, pp. 3979–3983.

[6] R.-R. Chen, R. Peng, A. Ashikhmin, and B. Farhang-Beroujeny, "Approaching MIMO capacity using bitwise Markov Chain Monte Carlo detecion," *Submitted to IEEE Trans. Commun.*, Otc. 2008.

[7] S. Akoum, R. Peng, R.-R. Chen, and B. Farhang-Boroujeny, "Soft detection using constrained markov chain monte carlo simulations," *To appear Proc. IEEE International Conf. on Comm. (ICC'09)*.

[8] G. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas," *Bell Labs Technical Journal*, vol. 1, no. 2, pp. 41–59, Aug. 1996.

[9] B. M. Hochwald and S. ten Brink, "Achieving near-capacity on a multiple antenna channel," *IEEE Trans. Commun.*, vol. 51, no. 3, pp. 389–399, Mar. 2003.

[10] H. Vikalo, B. Hassibi, and T. Kailath, "Iterative decoding for MIMO channels via modified sphere decoding," *IEEE Trans. on Wireless Commun.*, vol. 3, no. 6, pp. 2299–2311, Nov. 2004.

[11] R. Wang and G. B. Giannakis, "Approaching MIMO channel capacity with soft detection based on hard sphere decoding," *IEEE Trans. on Communications*, vol. 54, no. 4, Apr. 2006.

[12] J. Jalden and B. Ottersten, "On the complexity of sphere decoding in digital communications," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1474–1484, 2005.

[13] R. Peng, R.-R. Chen, and B. Farhang-Boroujeny, "Low complexity markov chain monte carlo detector for channels with intersymbol interference," *To appear Proc. IEEE International Conf. on Comm. (ICC'09)*.

[14] R. Peng, R.-R. Chen, and B. Farhang-Beroujeny, "Low Complexity Markov Chain Monte Carlo Detector for Channels with Intersymbol Interference," *To be submitted to IEEE Trans. Commun.*, Feb. 2009.

[15] M. Tuchler, A. C. Singer, and R. Koetter, "Minimum mean squared error equalization using a priori information," *IEEE Trans. Signal Processing*, vol. 50, no. 3, pp. 673–683, March 2002.

[16] P. Robertson, E. Villebrun, and P. Hoeher, "A comparison of optimal and sub-optimal MAP decoding algorithms operating in the Log domain," in *Proc. IEEE Int. Conf. on Comm.*, vol. 2, Seattle, Jun. 1995, pp. 1009–1013.

[17] S. ten Brink, G. Kramer, and A. Ashikhmin, "Design of low-density parity-check codes for modulation and detection," *IEEE Trans. on Commun.*, vol. 52, no. 4, pp. 670–678, Apr. 2004.

[18] S. A. Laraway and B. Farhang-Boroujeny, "Implementation of a Markov Chain Monte Carlo Based Multiuser/MIMO Detector," *To appear IEEE Trans. on Circuits and Systems*, 2009.

[19] K. J. Kim and J. Yue, "Joint channel estimation and data detection algorithms for MIMO-OFDM systems," in *Thirty-Sixth Asilomar Conference on Signals, Systems and Computers*, 2002, pp. 1857–1861.

[20] J. Proakis, *Digital Communications, 3rd Ed.* McGraw Hill, 1995.

[21] C. Heegard and S. Wicker, *Turbo coding*. Boston, MA: Kluwer, 1999.