BLIND SEPARATION OF LINEAR MIXTURES OF DIGITAL SIGNALS USING SUCCESSIVE INTERFERENCE CANCELLATION ITERATIVE LEAST SQUARES

Tao Li and Nicholas D. Sidiropoulos

Department of Electrical Engineering
University of Virginia
Charlottesville, VA 22903, USA
email : (tl7d,nikos)@virginia.edu

ABSTRACT

We consider blind separation of linear mixtures of digital communication signals in noise. When little or nothing can be assumed about the mixing matrix, signal separation may be achieved by exploiting structural properties of the transmitted signals. ILSP and ILSE are two iterative least squares (ILS) separation algorithms that exploit the finite-alphabet property. ILSE is monotonically convergent and performs very well, but its complexity is exponential in the number of signals; ILSP is computationally cheaper, but is not guaranteed to converge monotonically, and leaves much to be desired in terms of BER-SNR performance relative to ILSE. We propose two computationally efficient and provably monotonically convergent ILS blind separation algorithms based on an optimal scaling Lemma. The signal estimation step of both algorithms is reminiscent of Successive Interference Cancellation (SIC) ideas. For well-conditioned data and moderate SNR, the proposed algorithms attain the performance of ILSE at the complexity cost of ILSP.

1. INTRODUCTION AND DATA MODELING

Consider the instantaneous multiple-input multiple-output model:

$$\mathbf{X} = \mathbf{AS} + \mathbf{V} \tag{1}$$

where $\mathbf{X}$ is the observable $m \times N$ data matrix, $\mathbf{A}$ is an $m \times d$ mixing matrix, $\mathbf{S}$ is a $d \times N$ signal matrix, $\mathbf{V}$ is an $m \times N$ matrix of i.i.d. Gaussian random variables, and it is assumed that $m \geq d$, $N \geq d$, and $\mathbf{A}$ and $\mathbf{S}$ are full rank (d). This model arises, e.g., in the context of antenna array reception of d narrowband sources impinging on an array of m antennas, whereby Equation (1) describes the discrete-time baseband equivalent model after down-conversion, matched filtering, and sampling at the symbol rate, assuming small delay spread, Nyquist pulse shaping, and fixed propagation environment over N symbols. In this scenario, the i^{th} row of $\mathbf{S}$ contains the symbol sequence corresponding to the i^{th} source.

In the absence of noise, the objective of blind source separation is to factor $\mathbf{X}$ into $\mathbf{A}$ and $\mathbf{S}$ by exploiting known properties of either (or both) of $\mathbf{A}$, $\mathbf{S}$. One approach is to constrain $\mathbf{S}$ to satisfy known structural properties, e.g., finite alphabet or constant modulus [1, 3, 4, 5, 6, 7, 8]; let us denote this by $\mathbf{S} \in \mathbf{\Phi}$. Given $\mathbf{X}$, d, and $\mathbf{\Phi}$ a key issue is whether or not the factors are unique (modulo the inherent permutation and scale ambiguity); this is addressed in [1] for the finite-alphabet (FA) property. In the presence of noise, an *optimal* factorization is sought, e.g., in the least squares (LS) sense:

$$\min_{\mathbf{A},\mathbf{S} \in \Phi} \|\mathbf{X} - \mathbf{AS}\|_F^2 \tag{2}$$

which coincides with conditional maximum likelihood (treating $\mathbf{A}$ and $\mathbf{S}$ as deterministic unknowns).

Even though the model in Equation (1) appears to be quite restrictive due to its memoryless nature, more complicated convolutive models can be reduced to Equation (1) by means of blind equalization techniques [3].

2. PRIOR ART

The optimization problem in (2) is a bilinear nonparametric regression subject to structural constraints on one of the factors. Iterative Least Squares (ILS - also known as Alternating Least Squares, ALS) is one means of solving (2). The basic idea behind ILS is simple: each time compute a LS update for *one* of the unknown factors, conditioned on a previously obtained estimate for the other factor; proceed to update the other factor in a similar fashion; repeat until convergence. Global (from an arbitrary starting point) convergence to a feasible solution that achieves (at least) a local minimum of the cost function is guaranteed, by virtue of the fact that each (conditional LS) update may either improve or maintain, but cannot worsen the fit. The quality (fit) of the final output is generally dependent on the initialization.

The identifiability of the model in Equation (1), and associated ILS algorithms for the important special case that $\mathbf{\Phi}$ is a finite-alphabet constraint (usually denoted by Ω) have been considered in [1]. Two ILS algorithms have been proposed: Iterative Least Squares with Enumeration (ILSE), which is a true ILS algorithm, and Iterative Least Squares with Projection (ILSP), which is a much simpler pseudo-ILS algorithm. These are reviewed next.

Iterative Least Squares with Enumeration(ILSE):

1. Given $\mathbf{A}_0, k = 0$
2. $k = k + 1$
   - Let $\mathbf{A} = \mathbf{A}_{k-1}$ in Equation (3) below, and minimize for $\mathbf{S}_k$ (enumeration).
   - $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^H(\mathbf{S}_k\mathbf{S}_k^H)^{-1}$
3. Repeat 2. until $(\mathbf{A}_k, \mathbf{S}_k) \approx (\mathbf{A}_{k-1}, \mathbf{S}_{k-1})$.

$$\min_{\mathbf{S}\in\mathbf{\Omega}} \|\mathbf{X} - \mathbf{AS}\|_\mathbf{F}^\mathbf{2} = \min_{\mathbf{s}(1)\in\Omega} \|\mathbf{x}(1) - \mathbf{As}(1)\|_F^2 + \cdots + \min_{\mathbf{s}(N)\in\Omega} \|\mathbf{x}(N) - \mathbf{As}(N)\|_F^2 \tag{3}$$

In the above, $(\cdot)^H$ stands for Hermitian transpose, and each of the N minimizations in (3) is carried out by enumeration over all possible finite-alphabet d-tuples[1]. ILSE is a true ILS algorithm (guaranteed to converge, actually in a finite number of steps in this case [1]) but prohibitively complex even for moderate d. ILSP is a computationally cheaper alternative.

Supported by NSF/CAREER CCR-9733540

[1] We use Ω to denote the FA restriction on vectors and matrices alike.

Iterative Least-Squares with Projection (ILSP):

1. Given $\mathbf{A}_0, k=0$
2. $k=k+1$
   - $\widehat{\mathbf{S}}_k = (\mathbf{A}_{k-1}^H \mathbf{A}_{k-1})^{-1}\mathbf{A}_{k-1}^H \mathbf{X}$
   - $\mathbf{S}_k = proj_\Omega[\widehat{\mathbf{S}}_k]$
   - $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^H(\mathbf{S}_k\mathbf{S}_k^H)^{-1}$
3. Repeat 2. until $(\mathbf{A}_k, \mathbf{S}_k) \approx (\mathbf{A}_{k-1}, \mathbf{S}_{k-1})$.

Here $proj_\Omega[\cdot]$ projects its matrix argument onto the finite alphabet element-wise. In ILSP, the computationally demanding enumeration step is replaced by a finite-alphabet projection of the unconstrained LS update. Unfortunately, the resulting iteration is not a true ILS algorithm: the two-step update is not necessarily LS-optimal, and it may actually worsen the fit. This means that ILSP is not guaranteed to converge in general (it usually does in practice). ILSP is more prone to spurious minima than ILSE, and it tends to provide measurably worse results. On the other hand, the complexity of ILSP is $\mathcal{O}(Nmd)$ per iteration, while ILSE requires $\mathcal{O}(NmdL^d)$ per iteration (recall that both m, N are $\geq d$), where L is the size of the FA.

3. PROPOSED ALGORITHMS

ILSE computes a true conditional LS update for $\mathbf{S}$ by enumeration over the finite alphabet for each individual column of $\mathbf{S}$ (taking advantage of the decomposability of the Frobenious norm) - thus computing a simultaneous LS projection for all source symbols corresponding to a given time index. On the other hand, ILSP projects the unconstrained LS update of $\mathbf{S}$ onto the finite alphabet. Note that ILSP updates all columns of $\mathbf{S}$ simultaneously, but in a suboptimal fashion. The core idea behind both algorithms proposed herein can be summarized as follows. Instead of updating $\mathbf{S}$ suboptimally as a whole, or optimally one column at a time (the latter being very complex), update one row of $\mathbf{S}$ at a time, conditioned on $\mathbf{A}$ and the remaining rows of $\mathbf{S}$. This is reminiscent of Successive Interference Cancellation (SIC) ideas [11], since it uses previously obtained estimates of other users to "cancel" the multiuser interference and obtain an improved estimate for a user of "current interest". The difference with SIC is that our problem is blind (the mixing is unknown), and the process shifts back and forth between estimating the mixing matrix and updating the estimated user symbol streams. Interestingly, it turns out that the optimal update of one row of $\mathbf{S}$ conditioned on all other rows is easy to compute - in fact it is equivalent to projecting the unconstrained LS row update to the finite alphabet. Contrast this with the ILSE update of one column of $\mathbf{S}$ at a time - which is optimal, but requires enumeration over all possible finite-alphabet d-tuples. The optimality of projecting unconstrained LS row updates is a ramification of the following *optimal scaling Lemma* (a simple proof appears in [2]):

Lemma 1 *Let $\mathbf{X}$ be a given $m \times N$ matrix, and $\mathbf{a} \neq \mathbf{0}$ be a given $m \times 1$ vector. The problem $\min_{\mathbf{s}\in\Phi} \|\mathbf{X} - \mathbf{a}\mathbf{s}^H\|_2^2$ is equivalent to $\min_{\mathbf{s}\in\Phi} \|\mathbf{b} - \mathbf{s}\|_2^2$, where $\mathbf{b}$ stands for the* unconstrained *minimizer of $\|\mathbf{X} - \mathbf{a}\mathbf{s}^H\|_2^2$ with respect to $\mathbf{s}$, i.e., $\mathbf{b} \triangleq \frac{1}{\|\mathbf{a}\|_2^2}\mathbf{X}^H\mathbf{a}$. The above holds for general Φ (not necessarily finite-alphabet constraints). Note that, depending on Φ, the constrained solution may or may not be unique; we denote $proj_\Phi(\mathbf{b}) \triangleq arg\min_{\mathbf{s}\in\Phi} \|\mathbf{b} - \mathbf{s}\|_2^2$, with the understanding that it stands for "*an *argument that minimizes ...".*

To see how the above Lemma applies to the problem at hand, isolate one row of $\mathbf{S}$, say row r, and denote it by $\mathbf{s}_r^H$. Let $\mathbf{a}_r$ be the corresponding column of $\mathbf{A}$, and consider the LS update for $\mathbf{s}_r^H$ conditioned on everything else:

$$\min_{\mathbf{s}_r\in\Omega} \|\mathbf{X} - \mathbf{A}\mathbf{S}\|_F^2 = \min_{\mathbf{s}_r\in\Omega} \|\mathbf{X} - \mathbf{A}^{(r)}\mathbf{S}^{(r)} - \mathbf{a}_r\mathbf{s}_r^H\|_F^2$$
$$= \min_{\mathbf{s}_r\in\Omega} \|\widetilde{\mathbf{X}}^{(r)} - \mathbf{a}_r\mathbf{s}_r^H\|_F^2$$

where $\mathbf{A}^{(r)}$: $m \times (d-1)$ consisting of all but the r^{th} column of $\mathbf{A}$, $\mathbf{S}^{(r)}$: $(d-1) \times N$ consisting of all but the r^{th} row of $\mathbf{S}$, and $\widetilde{\mathbf{X}}^{(r)}$ is the equivalent data matrix. It follows that the LS update for $\mathbf{s}_r$ is given by $proj_\Omega(\frac{1}{\|\mathbf{a}_r\|_2^2}(\widetilde{\mathbf{X}}^{(r)})^H\mathbf{a}_r)$. This leads to the following algorithm.

Successive Interference Cancellation Iterative Least-Squares (SIC-ILS):

1. $k=0$; $\mathbf{S}_0$ = random FA; $\mathbf{A}_0 = \mathbf{X}\mathbf{S}_0^H(\mathbf{S}_0\mathbf{S}_0^H)^{-1}$.
2. $k=k+1$
   - $\mathbf{S}$-Update:
     - $\mathbf{S}_k = \mathbf{S}_{k-1}$;
     - for $r=1$ to d,
     - $\mathbf{a}_r = r^{th}$ column of $\mathbf{A}_{k-1}$;
     - $\mathbf{s}_r^H = r^{th}$ row of $\mathbf{S}_k$;
     - $\widetilde{\mathbf{X}}^{(r)} = \mathbf{X} - (\mathbf{A}_{k-1}\mathbf{S}_k - \mathbf{a}_r\mathbf{s}_r^H)$;
     - r^{th} row of $\mathbf{S}_k = proj_\Omega(\frac{1}{\|\mathbf{a}_r\|_2^2}\mathbf{a}_r^H\widetilde{\mathbf{X}}^{(r)})$;
     - end
   - $\mathbf{A}$-Update: $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^H(\mathbf{S}_k\mathbf{S}_k^H)^{-1}$
3. Repeat 2 until $(\mathbf{A}_k, \mathbf{S}_k) \approx (\mathbf{A}_{k-1}, \mathbf{S}_{k-1})$.

Notice that the update of a given row depends on all previously obtained updates of all other rows. An interesting twist is that different row update orders may give rise to different trajectories in the search space, exhibiting different convergence rates and BER-SNR performance characteristics. One way of resolving this issue is as follows. At any given point in time, one may *rank* possible row updates according to the resulting improvement in fit; pick the one that provides the best improvement; repeat. This is well-motivated from an optimization viewpoint (it results in a step-wise *steepest* descent), and it also makes sense in a near-far situation, since row updates corresponding to more powerful users are likely to lead to more significant improvements in fit. This idea gives rise to the following algorithm.

Ranked Successive Interference Cancellation Iterative Least-Squares (RSIC-ILS):

1. $k=0$; $\mathbf{S}_0$ = random FA; $\mathbf{A}_0 = \mathbf{X}\mathbf{S}_0^H(\mathbf{S}_0\mathbf{S}_0^H)^{-1}$.
2. $k=k+1$
   - $\mathbf{S}$-Update:
     - $\mathbf{S}_k = \mathbf{S}_{k-1}$;
     - Designate all rows as being "active";
     - while active rows exist,
     - for each active row (say, row-r):
     - $\mathbf{T}_r = \mathbf{S}_k$;
     - $\mathbf{a}_r = r^{th}$ column of $\mathbf{A}_{k-1}$;
     - $\mathbf{s}_r^H = r^{th}$ row of $\mathbf{S}_k$;
     - $\widetilde{\mathbf{X}}^{(r)} = \mathbf{X} - (\mathbf{A}_{k-1}\mathbf{S}_k - \mathbf{a}_r\mathbf{s}_r^H)$;
     - r^{th} row of $\mathbf{T}_r = proj_\Omega(\frac{1}{\|\mathbf{a}_r\|_2^2}\mathbf{a}_r^H\widetilde{\mathbf{X}}^{(r)})$;

– cost(r) = $\|\mathbf{X} - \mathbf{A}_{k-1}\mathbf{T}_r\|_F^2$;
– end for;
– Pick the $\widehat{r}$ with minimum cost(r); update $\mathbf{S}_k = \mathbf{T}_{\widehat{r}}$; de-activate the $\widehat{r}^{th}$ row;
– end while

- $\mathbf{A}$-Update: $\mathbf{A}_k = \mathbf{X}\mathbf{S}_k^H(\mathbf{S}_k\mathbf{S}_k^H)^{-1}$

3. Repeat 2 until $(\mathbf{A}_k, \mathbf{S}_k) \approx (\mathbf{A}_{k-1}, \mathbf{S}_{k-1})$.

Complexity: The per-iteration complexity of ILSP is $\mathcal{O}\ (Nmd)$, while that of ILSE is $\mathcal{O}\ (NmdL^d)$ [1]. The corresponding figures for SIC-ILS and RSIC-ILS are $\mathcal{O}(Nmd)$, and $\mathcal{O}(Nmd^2)$ respectively. One should keep in mind the assumption that m, N are $\geq d$ to properly interpret the order notation. The complexity claim may not be obvious from the pseudo-code listings: it requires updating the $\mathbf{A}_{k-1}\mathbf{S}_k$ matrix by subtracting the rank-1 contribution of row r before its update, and adding the rank-1 contribution of row r after its update (instead of actually computing the product, as listed in the pseudo-code for clarity of exposition).

3.1. Convergence

Theorem 1 *Both SIC-ILS and RSIC-ILS are globally monotonically convergent in a finite number of steps.*

The proof is a consequence of Lemma 1: each row update may either improve or maintain, but cannot worsen the fit. Thus both SIC-ILS and RSIC-ILS are true ILS algorithms, guaranteed to converge to (at least) a local minimum. Convergence in a finite number of steps follows from the fact that the cost function is decreasing, and there is only a finite number of distinct possibilities for $\mathbf{S}$ (due to the finite-alphabet constraint), each one of which is paired with one LS update for $\mathbf{A}$. In the worst case, the iteration will cycle over all the distinct possibilities once. In practice, the number of iterations before convergence is usually quite small (under 10 iterations). Notice that the same result applies (as shown in [1]) to ILSE, but not to ILSP.

3.2. Incorporating FEC Constraints

Notice that Lemma 1 may be used to incorporate forward error correction (FEC) constraints as well. To this end, one has to have efficient means of computing $\min_{\mathbf{s}\in\mathcal{C}} \|\mathbf{b} - \mathbf{s}\|_2^2$, where $\mathcal{C}$ is the FEC *codebook*. In other words, one has to have an efficient algorithm for computing the projection of a hypothetical "received data" sequence onto the codebook. If the signals are convolutionally encoded, the sought algorithm is the well-known "soft" Viterbi decoder for a AWGN channel. Similar algorithms are available for many block codes as well. Note that (LS) optimality of the decoding algorithm is crucial for maintaining monotone convergence of the overall blind source separation iteration - suboptimal pseudo-projections onto the codebook will not do. The main point here is that Lemma 1 allows us to take advantage of FEC constraints (meant to guard against noise) to improve blind source separation.

4. MONTE-CARLO RESULTS

We conducted a series of Monte-Carlo experiments to assess the relative performance of ILSP, ILSE [1], SIC-ILS, and RSIC-ILS. In our simulations, $\mathbf{A}$ corresponds to a ULA of $m = 4$ sensors ($\lambda/2$ sensor spacing) receiving $d = 3$ BPSK signals arriving from $[10^o, 30^o, 50^o]$ relative to the array broadside. $N = 100$, the elements of $\mathbf{S}$ are $+1$ or -1 ($E_b = 1$), and $\mathbf{S}$ is held fixed during the simulation. 10,000 Monte Carlo trials were conducted for each $\frac{E_b}{N_o}$-BER datum reported. ILSP and ILSE are initialized with $\mathbf{A}_0 = \mathbf{I}_{m\times d}$, whereas SIC-ILS and RSIC-ILS are initialized with random $\mathbf{S}_0$. A maximum of two re-initializations (random $\mathbf{A}_0$ for ILSP/ILSE, random $\mathbf{S}_0$ for SIC/RSIC-ILS) per trial were allowed for each algorithm.

Figures (1), (2), and (3) present BER versus $\frac{E_b}{N_o}$ results for ILSP, SIC-ILS, and ILSE, respectively. Each Figure depicts three separate curves, one for each source ("user"). Notice that, for low to moderate SNR, SIC-ILS attains the performance of ILSE at the complexity cost of ILSP. Also notice that, in contrast to ILSP, SIC-ILS demodulates all three users at the same BER. We remark that ILS algorithms (including ILSP and ILSE) exhibit a BER bounce-back effect at higher SNR due to local minima [3].

Figures (4), (5) present the results of SIC-ILS, RSIC-ILS, respectively, in a near-far scenario. Error rates are plotted versus the $\frac{E_b}{N_o}$ of the weakest user. The other two users are 6 and 12 dB, respectively, above the weakest user. The error rates for the strongest user in Figure (5) drop below the statistical significance threshold.

Figure (6) presents coded SIC-ILS results for rate $\frac{1}{2}$ repetition-coded signals. Error rates are plotted versus $\frac{E_b}{N_o}$ per *coded* bit. As expected, the incorporation of FEC constraints helps at low SNR. This is particularly useful when the users are power-controlled, but require a higher quality of service than what is available without coding.

MATLAB code for all algorithms is available on-line at: `http://www.people.virginia.edu/~tl7d,~nds5j`.

5. DISCUSSION AND CONCLUSIONS

We have proposed two algorithms for blind separation of linear mixtures of digital communication signals, SIC-ILS, and RSIC-ILS. Both exploit the FA property, and an optimal scaling Lemma. SIC-ILS and RSIC-ILS feature moderate complexity, global monotonic convergence, and cover the intermediate ground in performance in-between ILSP and ILSE. A bonus feature is that they allow easy incorporation of FEC constraints into the basic iteration. Future work includes the investigation of these ideas within the framework of trilinear regression [9].

6. REFERENCES

[1] S. Talwar, M. Viberg, and A. Paulraj, "Blind Separation of Synchronous Co-Channel Digital Signals Using an Antenna Array - Part I: Algorithms", *IEEE Trans. Signal Processing*, 44(5):1184–1197, May 1996.

[2] R. Bro and N. D. Sidropoulos, "Least Squares Algorithms Under Unimodality and Non-negativity Constraints", *Journal of Chemometrics* 1998, to appear.

[3] A-J van der Veen , "Analytical Method for Blind Binary Signal Separation", *IEEE Trans. Signal Processing*, vol. 45, pp.1078-1082, April 1997.

[4] A-J van der Veen and A. Paulraj, "An Analytical Constant-Modulus Algorithm", *IEEE Trans. Signal Processing*, vol. 44, pp.1136–1155, May 1996.

[5] A-J van der Veen, S. Talwar and A. Paulraj, "Blind estimation of multiple digital signals transmitted over FIR channel", *IEEE Signal Processing Let.*, vol.2, pp. 99-102, May 1995.

[6] D. Yellin and B. Porat, "Blind identification of FIR systems excited by discrete-alphabet input", *IEEE Trans. Signal Processing*,vol.41,no.3,pp.1331-1339,1993.

[7] A. Swindlehurst, S. Daas and J. Yang, "Analysis of a decision directed beamformer", *IEEE Trans. Signal Processing* , vol.43, no.12, pp.2920-2927, 1995.

[8] K. Anand, G. Mathew, and V. U. Reddy, "Blind separation of multiple co-channel BPSK signals arriving at an antenna array", *IEEE Signal Proc. Letters*, vol.2, no.9, pp. 176-178, Sept. 1995.

[9] N. D. Sidiropoulos, G. B. Giannakis, and R. Bro, "Deterministic Waveform-Preserving Blind Separation of DS-CDMA Signals Using an Antenna Array", *Proc. 1998 IEEE Workshop on Statistical Signal and Array Processing (SSAP98)*, Sept. 14-16, Portland, Oregon.

[10] G. H. Golub, and C. F. Van Loan, *Matrix Computations* , *3rd edition* , The John Hopkins Univ. Press ,1996.

[11] S. Verdu, *Multiuser Detection*, Cambridge University Press, Cambridge, 1998.

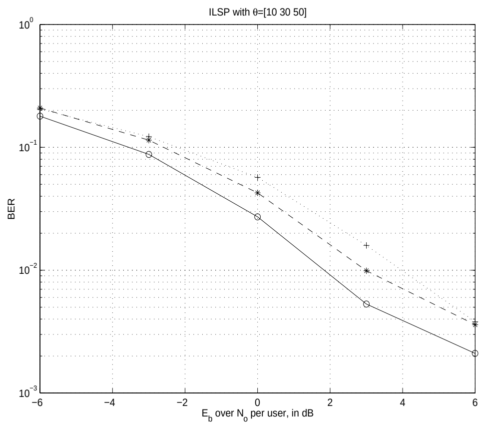

Figure 1: BER vs. $\frac{E_b}{N_o}$ curves for ILSP.

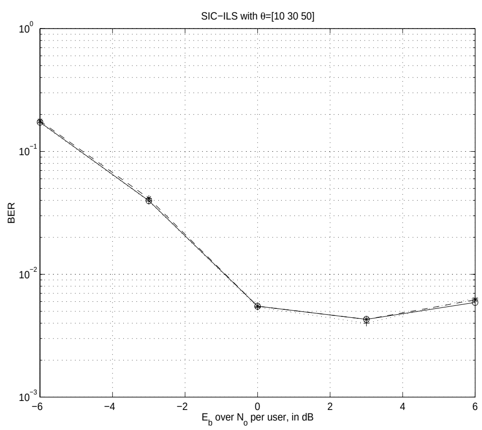

Figure 2: BER vs. $\frac{E_b}{N_o}$ curves for SIC-ILS.

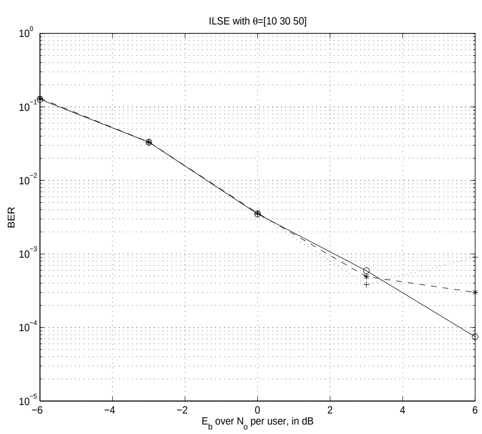

Figure 3: BER vs. $\frac{E_b}{N_o}$ curves for ILSE.

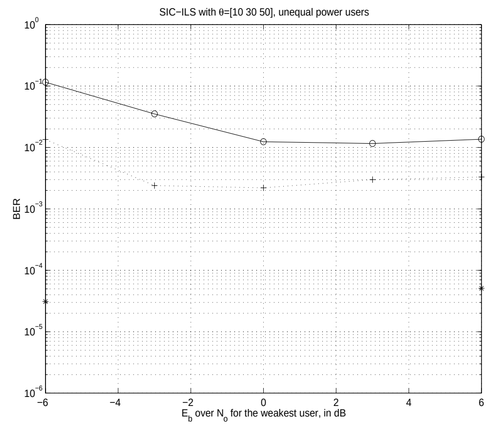

Figure 4: BER vs. $\frac{E_b}{N_o}$ curves for SIC-ILS (near-far scenario).

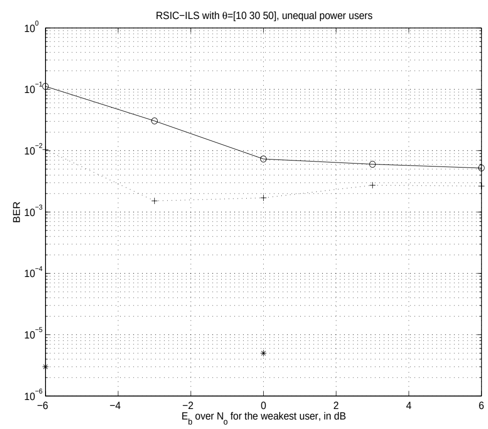

Figure 5: BER vs. $\frac{E_b}{N_o}$ curves for RSIC-ILS (near-far scenario).

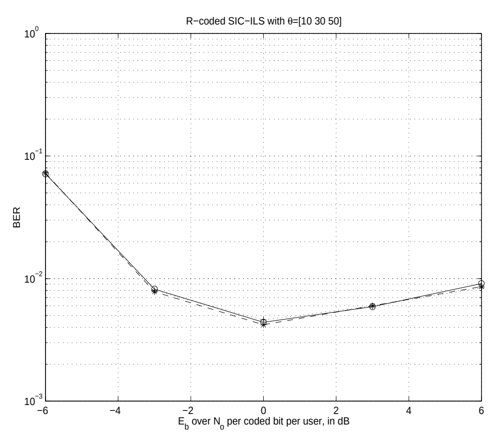

Figure 6: BER vs. coded $\frac{E_b}{N_o}$ curves for rate-$\frac{1}{2}$ repetition coded SIC-ILS.