A Note on Polynomial-Complexity Optimal Multiuser Detection for Non-Orthogonal CDMA

Tan Bien Aik
Electrical & Computer Engineering
National University of Singapore
tbienaik@dso.org.sg
20 Science Park Drive Singapore 118230
HP 96651762 Fax 68720977

Mehul Motani
Electrical & Computer Engineering
National University of Singapore
motani@nus.edu.sg
10 Kent Ridge Cresent Singapore 119260

Lim Teng Joon
Electrical & Computer Engineering
University of Toronto
limtj@comm.utoronto.ca

Technical Areas (In order of Preference): Spread Spectrum, Signal Processing for Communications.

A Note on Polynomial Complexity Optimal Multiuser Detection for Non-Orthogonal CDMA Signals

Tan Bien Aik
Electrical & Computer Engineering
National University of Singapore

Mehul Motani
Electrical & Computer Engineering
National University of Singapore

Lim Teng Joon
Electrical & Computer Engineering
University of Toronto

***Abstract*--** **Optimum multiuser detection (MUD) in code division multiple access (CDMA) systems have a complexity that increases exponentially with the number of users in order to achieve maximum likelihood performance. In [1] and [2], it was realized that if the signal cross-correlations between users are constant, then maximum likelihood performance can be obtained with polynomial-complexity. Another polynomial-complexity algorithm is proposed in [3] and [4] using graph theory. It transforms the MUD problem into one that solves for a minimum cut in a graph or network. Also, a minimum cut problem is equivalent to a maximum flow problem [5] and there are many polynomial-complexity max flow or min cut algorithms available. Some of their complexities are dependent on the number of edges. An alternative method to transform the MUD problem into a network, that sees a reduction in the number of edges by 50 percent, is proposed. This results in minimizing the already polynomial-complexity of some max flow algorithms.**

Index Terms—CDMA, Multiuser Detection, Polynomial Complexity.

I. INTRODUCTION

The basic synchronous CDMA signal of K users consists of the sum of antipodally modulated spreading sequence waveforms and additive white Gaussian noise. It is expressed mathematically in Equation (1). It should be noted that this signal is continuous-time.

$$y(t)=\sum_{k=1}^{K}A_k b_k s_k(t)+n(t) \qquad t\in[0,T] \qquad (1)$$

where

- T is the bit interval
- $s_k(t)$ the deterministic spreading sequence of length N chips assigned to the k^{th} user
$$\| s_k \|^2=\int_0^T s_k^2(t)dt=1 \qquad (2)$$
- A_k is the received amplitude of the k^{th} user signal. A_k^2 is the energy of the k^{th} user.
- $b_k\in$ {-1,+1} is the antipodal bit signal transmitted by the k^{th} user.
- $n(t)$ is the white Gaussian noise with zero mean and uniform power spectral density of $N_o/2$.

Equation (2) simply states that the spreading sequence waveforms have unit energy over $[0,T]$. Another parameter that is commonly used here is the cross-correlation of the spreading sequence waveforms (see Equation (3)). It quantifies the similarity between two spreading sequence waveforms.

$$\rho_{ij}=<s_i,s_j>=\int_0^T s_i(t)s_j(t)dt \qquad (3)$$

The continuous-time signal $y(t)$ can be converted into the discrete form by correlating it with deterministic signals of spreading sequence waveforms and conventional sampling. This is done by passing $y(t)$ through a bank of matched filters each matched to the spreading sequence waveform of different users. The output of the matched filter is

$$\mathbf{y}=\mathbf{RAb}+\mathbf{n} \qquad (4)$$

where

$\mathbf{y}=[y_1,............,y_K]^{\mathrm{T}}$

$\mathbf{R}=\{\rho_{\mathrm{ij}}\}$ is a K by K cross correlation matrix

$\mathbf{A}=diag[A_1,............,A_K]$

$\mathbf{b}=[b_1,............,b_K]^{\mathrm{T}}$

and $\mathbf{n}=[n_1,............,n_K]^{\mathrm{T}}$ is a zero mean Gaussian random vector with covariance matrix of

$$\mathrm{E}[\mathbf{nn}^{\mathrm{T}}]=\frac{N_0}{2}\mathbf{R} \qquad (5)$$

Maximum likelihood (ML) multiuser detection will result in the lowest error probability achievable for equally likely priors. It computes the conditional probability $\boldsymbol{P}_{\mathbf{y}|\mathbf{b}}(\mathbf{y}|\mathbf{b})$ for all possible combinations of $\mathbf{b}$ and chooses the one that gives the maximum $\boldsymbol{P}_{\mathbf{y}|\mathbf{b}}(\mathbf{y}|\mathbf{b})$. This particular $\mathbf{b}$ that maximizes $\boldsymbol{P}_{\mathbf{y}|\mathbf{b}}(\mathbf{y}|\mathbf{b})$ is known as maximum likelihood estimate. It is the most likely $\mathbf{b}$ that was being transmitted based on the observed vector $\mathbf{y}$. The maximum likelihood estimate can be expressed as [6, pg162].

$$\hat{\mathbf{b}}_{ML}=\underset{\mathbf{b}\in\{-1,+1\}^K}{\arg\max}\; p_{\mathbf{y}|\mathbf{b}}(\mathbf{y}\,|\,\mathbf{b})=\underset{\mathbf{b}\in\{-1,+1\}^K}{\arg\max}\left[2\mathbf{y}^{\mathrm{T}}\mathbf{Ab}-\mathbf{b}^{\mathrm{T}}\mathbf{ARAb}\right]$$
$$=\underset{\mathbf{b}\in\{-1,+1\}^K}{\arg\max}\quad \Omega(\mathbf{b}) \qquad (6)$$

Looking at Equation (6), there are 2^K possible values of $\mathbf{b}$ and hence, there are 2^K possible outcomes or computations. In other words, a detector with a computational complexity, $O(2^K)$, that increases exponentially with the number of users is needed to get maximum likelihood performance. The exponential complexity may put a cap on the capacity and data rate of the CDMA system, as increasing the number of users can increase computational time for the maximum likelihood estimate.

II. POLYNOMIAL-COMPLEXITY MULTIUSER DETECTION USING GRAPH THEORY

The usual {-1,+1} multiuser detection (MUD) problem had been converted into a {0,1} quadratic programming problem so that it can be eventually transformed into a maximum flow and minimum cut network problem [3, 4]. The reason is simply that there is a bag of polynomial-complexity max flow and min cut algorithms available [7, 8]. However, one condition needed for these to work is that cross correlations between users must be non-positive.

From [3] and [4], the problem of maximizing the $\Omega(\mathbf{b})$ function for $\mathbf{b}\in$ $\{-1,+1\}^K$ has been converted to a problem of minimizing the $\Omega(\mathbf{x})$ quadratic function for $\mathbf{x}\in$ $\{0,1\}^K$ in order to calculate the maximum likelihood estimate (See Equation (7)).

$$
\begin{aligned}
\hat{\mathbf{x}}_{ML} &= \arg\min_{\mathbf{x}\in\{0,1\}^K}[\Omega(\mathbf{x})] \\
&= \arg\min_{\mathbf{x}\in\{0,1\}^K}[\mathbf{x}^{\mathrm{T}}\mathbf{Hx}+(-\mathbf{u}^{\mathrm{T}}\mathbf{H}-\mathbf{y}^{\mathrm{T}}\mathbf{A})\mathbf{x}] \\
&= \arg\min_{\mathbf{x}\in\{0,1\}^K}[\mathbf{x}^{\mathrm{T}}\mathbf{Hx}+\mathbf{px}] \qquad (7)
\end{aligned}
$$

$$
\Omega(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{Hx}+\mathbf{px} = \sum_{i=1}^{K}\sum_{j=1}^{K} h_{ij}x_i x_j + \sum_{i=1}^{K} p_i x_i \qquad (8)
$$

where $\mathbf{x}$ is a K elements column vector with $x_i \in \{0,1\}$, $\mathbf{u}$ is a column vector with all K elements equal to one and $\mathbf{H}=\mathbf{ARA}=\{h_{ij}\}$ is the unnormalized cross correlation matrix. Note that h_{ii} can be assumed to be zero as both h_{ii} in $\mathbf{H}$ and $\mathbf{p}$ cancels each other out.

Given a directed graph, G=[V,E] where

$$
\mathrm{V} = \{v_0, v_1, v_2, \ldots, v_K, v_{K+1}\}
$$

is the set of vertices and $v_o=a$ is the source and $v_{K+1}=z$ is the sink. E is the set of edges connecting the vertices.

The capacity of a cut was transformed into a function of vector $\mathbf{x}$ and manipulated in [3] and [4] to be:

$$
\begin{aligned}
C(\mathbf{x}) &= \sum_{i=1}^{K}\left[C_{i\,K+1} - C_{0i} + \sum_{j=1}^{K} C_{ij}\right]x_i - \sum_{i=1}^{K}\sum_{j=1}^{K} C_{ij}x_i x_j \\
&\quad + \sum_{j=1}^{K} C_{0j} + C_{0\,K+1} \qquad (9)
\end{aligned}
$$

where C_{ij} is the capacity of the directed edge from i to j.

By comparing the Equations (8) and (9), the conditions for defining the capacities can be determined. And because the number of edges is more than the conditions given, the assignment of capacities can sometimes be arbitrary. The conditions set by [3] and [4] were:

- $C_{ij} = -h_{ij} \quad \forall i, j \in \{1,2,\ldots,K\}$ (10)
- $\left[C_{i\,K+1} - C_{0i} + \sum_{j=1}^{K} C_{ij}\right] = p_i$ (11)
- $\sum_{j=1}^{K} C_{0j} + C_{0\,K+1} = 0$ (12)

Note that both [3] and [4] have negative (source incident) capacities in their resultant network conversion due to Condition (12). In this paper, it is desired to have only non-negative capacities so that not only min cut algorithms but polynomial max flow algorithms can also be implemented. Max flow algorithms work within the boundary of $0 \le F_{ij} \le C_{ij}$ for i not equal j. On top of that, it is possible to reduce the number of edges by half in exploiting the symmetry of $\mathbf{H}$ and Condition (11). The implication will be a reduction in any graph algorithm's complexity that is edge dependent.

An Alternative Approach

Since $\mathbf{H}$ is symmetric, $h_{ij}=h_{ji}$ *and* $h_{ii}=0$. Equation (8) can be rewritten as

$$
\Omega^*(\mathbf{x}) = 2\sum_{i=1}\sum_{j>i} h_{ij}x_i x_j + \sum_{i=1}^{K} p_i x_i + \xi \qquad (13)
$$

The number of terms in the first summation has been reduced from K^2 to $(K^2\text{-}K)/2$. It will be clear in the later part of this section that this will significantly reduce the number of edges not incident to the source and the sink by half. ξ is an arbitrary constant and is added so that Condition (12) is not equal to zero when Equation (13) and (9) are compared, this will result non-negative capacities unlike in [3] and [4]. Note that the addition of ξ to the minimizing equation will not affect the result of Equation (10), the maximum likelihood estimate.

Comparing Equation (9) and Equation (13), it is now clear that if we choose the edge capacities according to

- $C_{ij} = -2h_{ij} \quad \forall i, j \in \{1,2,\ldots,K\}$ and $i < \mathrm{j}$ (14)
- $\left[C_{i\,K+1} - C_{0i} + \sum_{j=1}^{K} C_{ij}\right] = p_i$ (15)
- $\sum_{j=1}^{K} C_{0j} + C_{0\,K+1} = \xi$ (16)

then

$$
C(\mathbf{x}) = \Omega^*(\mathbf{x}) \;\; \forall\, \mathbf{x} \in \{0,1\}^K \qquad (17)
$$

Thus, finding $\mathbf{x}$ that minimizes $\Omega^*(\mathbf{x})$ or $\Omega(\mathbf{x})$ is equivalent to finding a minimum cut in a network G. The minimum cut problem is also a maximum flow problem as both maximum flow condition and minimum cut condition must co-exist together which is proven by the max flow min cut theorem [5].

Given the parameters of $\mathbf{p}$ and $\mathbf{H}$ of the {0,1} quadratic programming problem, the corresponding network can be constructed by satisfying the three conditions set by Conditions (14) to (16).

Condition (14) assigns the capacities, C_{ij} that are not incident to the source and the sink and are properly oriented, to $-2h_{ij}$ for $i \neq j$. The improperly oriented ones that are not incident to the source and the sink are assigned to zero, that is, the edge does not exist. Because $h_{ij} = A_i A_j \rho_{ij}$ where A_i A_j are positive and $C_{ij} \ge 0$ then h_{ij} need to be non-positive. This means ρ_{ij} need to be non-positive too. In other words, all cross correlation between different users must be non-positive for Condition (14) to be satisfied. Note that in [3] and [4], the number of capacities not incident to the sink and source is $(K^2\text{-}K)$. Here, the number of capacities have been reduced to $(K^2\text{-}K)/2$.

As for Condition (15), the equation can be further simplified into:

$$
\begin{aligned}
&\left[C_{i\,K+1} - C_{0i} + \sum_{j=1}^{K} C_{ij}\right] = p_i \\
&\left[C_{i\,K+1} - C_{0i} - 2\sum_{j>i}^{K} h_{ij}\right] = -2\sum_{j>i}^{K} h_{ji} - y_i A_i \\
&[C_{i\,K+1} - C_{0i}] = 2\sum_{j>i}^{K} h_{ij} - 2\sum_{j>i}^{K} h_{ji} - y_i A_i \qquad (18)
\end{aligned}
$$

Note C_{0i} represents all the capacities from the source to other vertices except the sink and there are K of them. $C_{i\,K+1}$ represents all the capacities from all the vertices except the source to the sink and there are K of them too. It is desired for $C_{ij} \ge 0$. So if the right hand side (RHS) of Equation (18) is positive, then let $C_{i\,K+1} > C_{0i}$. To minimize the number of edges, let $C_{i\,K+1}$ = RHS of Equation (18) and $C_{0i} = 0$. However if the RHS of Equation (18) is negative, then let $C_{i\,K+1} < C_{0i}$. Again, to minimize the number of edges, let $C_{i\,K+1} = 0$ and C_{0i} = RHS of Equation (18). This also results in the reduction in the number of edges incident to the source or the sink from $2K$ to K.

Finally, $C_{0\,K+1}$ is assigned zero in Condition (16). In summary, a MUD problem that has been converted into a graph with K^2+K+1 edges, has been reduced to a graph with $(K^2+K)/2$ edges. All the edges will only have non-negative capacities. In Figure 3, there are two curves representing the resultant number of edges when a MUD problem is converted into a network problem. The number of edges depends on the number of users. In the legend, Network-C means the network derived from the MUD problem using the conventional method in [3] and [4]. And Network-A means the network derived from the MUD problem using the alternative approach presented here in this paper.

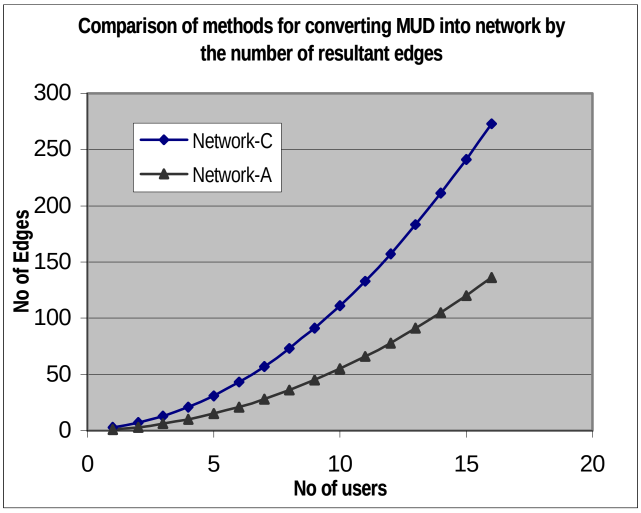

Figure 3 Comparing Methods for Converting MUD into Network Problem

It clearly shows that the reduction in the number of edges for the alternative method as compared to the conventional method is slightly more than 50 percent. If any polynomial-complexity algorithm is directly proportional to the number of edges, then the complexity is reduced by 50 percent when Network-A is used as compared to Network-C. Examples of edge dependent algorithms are successive shortest path and the highest-label preflow-push algorithms with complexity of $O(n^2m)$ (50 percent reduction in the order of complexity) and $O(n^2\sqrt{m})$ (30 percent reduction in the order of complexity) respectively where n is the number of vertices and m is the number of edges [8].

III. A 3-USER MUD EXAMPLE

The objective of this section is to show, by example, that the min cut or max flow of Network-A and Network-C, derived from a MUD problem, leads to the same minimized function $\Omega(\hat{\mathbf{x}}_{ML})$ and maximum likelihood estimate $\hat{\mathbf{x}}_{ML}$. For Network-C, it is already shown in [4], using an example, that solving the min cut problem is equivalent to solving $\mathbf{x}$ that minimizes function $\Omega(\mathbf{x})$. In this section, it will be shown by example that solving a min cut problem in Network-A is also equivalent to that of minimizing $\Omega(\mathbf{x})$. A numerical example, each for Network-A and Network-C, is also included to illustrate that computing the maximum flow is also equivalent to solving the MUD problem, that is, $\Omega(\hat{\mathbf{x}}_{ML})$ and $\hat{\mathbf{x}}_{ML}$.

Consider a 3-user (K=3) MUD example with the unnormalized cross correlation matrix,

$$\mathbf{H} = \begin{bmatrix} 0 & -A_1A_2\rho_{12} & -A_1A_3\rho_{13} \\ -A_2A_1\rho_{21} & 0 & -A_2A_3\rho_{23} \\ -A_3A_1\rho_{31} & -A_3A_2\rho_{32} & 0 \end{bmatrix} \quad (19)$$

where $\rho_{21} = \rho_{12}$, $\rho_{13} = \rho_{31}$ and $\rho_{32} = \rho_{23}$.

Note that the diagonal elements have been equated to zero as it is insensitive to the multiuser detection problem shown in Section III. Based on Equation (13), the minimizing function is:

$$\Omega^*(\mathbf{x}) = 2\sum_{i=1}\sum_{j>i} h_{ij}x_ix_j + \sum_{i=1}^{K} p_ix_i + \xi$$

Computing the minimizing function for all possible manifestations of $\mathbf{x}$, we have:

$$\Omega^*(0,0,0) = \xi$$

$$\Omega^*(0,0,1) = \xi + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3$$

$$\Omega^*(0,1,0) = \xi + A_1A_2\rho_{12} + A_2A_3\rho_{23} - y_2A_2$$

$$\Omega^*(0,1,1) = \xi + A_1A_2\rho_{12} + A_1A_3\rho_{13} - y_2A_2 - y_3A_3$$

$$\Omega^*(1,0,0) = \xi + A_1A_2\rho_{12} + A_1A_3\rho_{13} - y_1A_1$$

$$\Omega^*(1,0,1) = \xi + A_1A_2\rho_{12} + A_2A_3\rho_{23} - y_1A_1 - y_3A_3$$

$$\Omega^*(1,1,0) = \xi + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_1A_1 - y_2A_2$$

$$\Omega^*(1,1,1) = \xi - y_1A_1 - y_2A_2 - y_3A_3$$

Using the details furnished in the sub-section "*The Alternative Approach*", the 3-user MUD problem can be converted into a network. Condition (14) will result in $C_{12} = 2A_1A_2\rho_{12}$, $C_{13} = 2A_1A_3\rho_{13}$ and $C_{23} = 2A_2A_3\rho_{23}$. While Condition (15) will have the following equations:

$$[C_{14} - C_{01}] = -A_1A_2\rho_{12} - A_1A_3\rho_{13} - y_1A_1$$

Assuming that the RHS of the equation is negative, let $C_{14} = 0$ and $C_{01} = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1$.

$$[C_{24} - C_{02}] = A_1A_2\rho_{12} - A_2A_3\rho_{23} - y_2A_2$$

Assuming that the RHS of the equation is positive, let $C_{02} = 0$ and $C_{24} = A_1A_2\rho_{12} - A_2A_3\rho_{23} - y_2A_2$.

$$[C_{34} - C_{03}] = A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3$$

Assuming that the RHS of the equation is positive, let $C_{03} = 0$ and $C_{34} = A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3$.

From Condition (16), let $C_{04} = 0$ and $\xi = C_{01} = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1$.

Hence, the resultant network can be shown in the following:

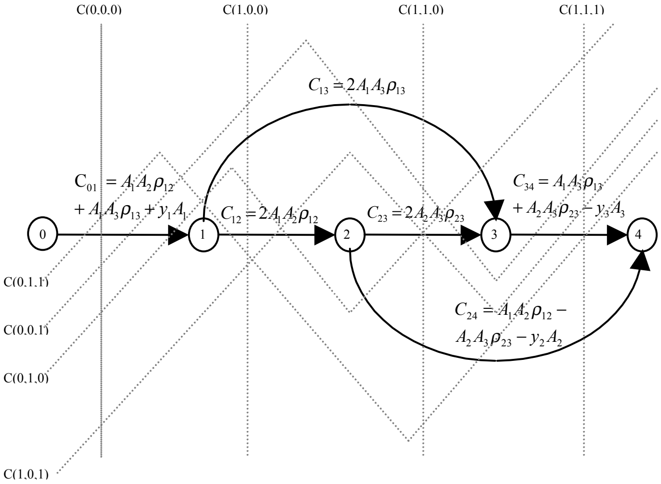

Figure 4 Network Derived from MUD Problem of *K*=3

Computing the capacities of cut, *C*(x), for all possible manifestations of **x**, we have:

$$C(0,0,0) = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1 \\ = \xi$$

$$C(0,0,1) = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1 + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3 \\ = \xi + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3$$

$$C(0,1,0) = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1 + 2A_2A_3\rho_{23} + A_1A_2\rho_{12} \\ - A_2A_3\rho_{23} - y_2A_2 \\ = \xi + A_1A_2\rho_{12} + A_2A_3\rho_{23} - y_2A_2$$

$$C(0,1,1) = A_1A_2\rho_{12} + A_1A_3\rho_{13} + y_1A_1 + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3 \\ + A_1A_2\rho_{12} - A_2A_3\rho_{23} - y_2A_2 \\ = \xi + A_1A_2\rho_{12} + A_1A_3\rho_{13} - y_2A_2 - y_3A_3$$

$$C(1,0,0) = 2A_1A_2\rho_{12} + 2A_1A_3\rho_{13} \\ = \xi + A_1A_2\rho_{12} + A_1A_3\rho_{13} - y_1A_1$$

$$C(1,0,1) = 2A_1A_2\rho_{12} + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_3A_3 \\ = \xi + A_1A_2\rho_{12} + A_2A_3\rho_{23} - y_1A_1 - y_3A_3$$

$$C(1,1,0) = 2A_1A_3\rho_{13} + A_2A_3\rho_{23} + A_1A_2\rho_{12} - y_2A_2 \\ = \xi + A_1A_3\rho_{13} + A_2A_3\rho_{23} - y_1A_1 - y_2A_2$$

$$C(1,1,1) = A_1A_3\rho_{13} - y_3A_3 + A_1A_2\rho_{12} - y_2A_2 \\ = \xi - y_1A_1 - y_2A_2 - y_3A_3$$

Notice the equivalence of the minimizing function $\Omega^*(\mathbf{x})$ and the capacity cut function *C*(**x**). Hence, $C(\mathbf{x}) = \Omega^*(\mathbf{x})\ \forall\, \mathbf{x} \in \{0,1\}^K$ is illustrated here and solving the min cut problem is equivalent to solving **x** that minimizes $\Omega(\mathbf{x})$. Note that $\Omega^*(\mathbf{x}) = \Omega(\mathbf{x}) + \xi$.

IV. Conclusion

The optimum MUD is known to be NP-hard, that is, its computational complexity increases exponentially, ($O(2^K)$), with the number of users, *K*. One perspective of solving the multiuser detection problem with polynomial-complexity is found in [3] and [4]. It involves converting the MUD problem into a network problem that can be solve with polynomial-complexity max flow or min cut algorithms provided that the cross-correlations between users are non-positive. Some of these polynomial algorithms are edge dependent forming the motivation to find an alternative method to convert the MUD into a network with minimal number of edges. The result is a 50 percent reduction in number of edges for the network using this new method as compared to the one suggested in [3] and [4]. Hence, edge dependent polynomial algorithms such as highest label preflow push max flow algorithm, which is known to be the most effective algorithm [8], can benefit from such improvements (30 percent reduction in the order of complexity).

References

[1] Christian Schlegel, Alex Grant, “Polynomial complexity optimal detection of certain multiple-access systems”, IEEE Transactions on Information Theory, Vol.: 46, Issue: 6, Page: 2246 -2248, Sept. 2000.

[2] Mehul Motani, “On Lower Complexity Optimum Multiuser Detection for CDMA”, Unpublished Paper, 1999.

[3] Chandrasekar Sankaran, Anthony Ephremides, “Solving a Class of Optimum Multiuser Detection Problems with Polynomial Complexity”, IEEE Transactions on Information Theory, Vol.: 44, Issue: 5, Page:1958-1961, Sept 1998.

[4] Sennur Ulukus, Roy D. Yates, “Optimum Multiuser Detection is Tractable for Synchronous CDMA Systems Using M-Sequences”, IEEE Communications Letters, Vol.: 2, Issue: 4, Page: 89-91, April 1998.

[5] Richard Johnsonbaugh, “Discrete Mathematics”, Macmillan, USA, 1993.

[6] S Verdu, “Multiuser Detection”, Cambridge University Press, New York, USA, 1998.

[7] R Motwani and P Raghavan, “Randomized Algorithms”, Cambridge University Press, New York, USA, 1995.

[8] Ravindra K Ahuja, Thomas L Magnanti, James B Orlin, “Network Flows Theory, Algorithms, and Applications”, Prentice Hall, New Jersey, USA, 1993.