Computationally Efficient MIMO Demodulation

William A. Gardner, David A. Burgess, Chris W. Reed
Statistical Signal Processing, Inc
1909 Jefferson Street
Napa, CA 94559 wag@statsig.com

Abstract

It is shown that multiple complementary techniques for trading BER performance for cost reduction in the Viterbi algorithm for multiple-input/multiple-output joint demodulation can be merged to obtain substantial and efficient cost reduction. Quantitative results for application to the separation of GSM cochannel signals are presented. It is shown that up to six signals can be separated at a GSM basestation with BER of 10% before decoding gain with only a single dual-polarized antenna and E_b/N_o = 10dB in real time on a Compaq 1GHz ES-40 server, with cost savings factors up to 8700. It is also shown that for a specified BER and number of signals, the cost of MIMO demodulation decreases as the number of sensors increases.

1. Introduction

It is a well-known fact that a joint demodulator for a single sensor can separate two cochannel interfering (CCI) signals and sometimes more depending on SNR, SIR, and other parameters, such as fraction of symbols that are known at the receiver. It follows that, more generally, a joint demodulator for multiple sensors can separate a number of signals that exceeds the number of sensors, and offers a distinct advantage over linear combining sensor outputs followed by demodulation. This advantage accrues primarily from more effective utilization of the increased channel diversity resulting from multiple sensors, but little is known about how many signals can be separated with a specified BER and number of sensors. The primary reason for the lack of progress in this area of multiple-input/multiple-output (MIMO) demodulation is computational cost. Even if the optimally efficient Viterbi algorithm is used to implement the MIMO demodulator, computational cost still grows exponentially in the number of signals. The computational cost can be so high that even simulations present a substantial challenge. Consequently, efficient means for trading off BER performance for computational cost reduction are required in order to move forward with MIMO demodulation R&D.

The purpose of this paper is to report on progress made on developing efficient techniques for trading BER performance for cost reduction.

2. MIMO Demodulation

In the conventional application of antenna arrays to the extraction of digital messages from cochannel interfering signals, the received multichannel data is linearly combined to effectively steer beams and nulls in the synthesized antenna pattern. Then the output of each particular linear combiner is fed into a conventional (non-joint) demodulator for the particular signal targeted by the particular linear combiner. The maximum-likelihood demodulator (for the simplified AWGN channel model and IID equally-likely-symbol sequence) implements a least-squares fit of the modulated digital-symbol sequence to the linear combiner output.

In contrast to this, the MIMO demodulator implements a joint least-squares fit of all the interfering signals to all the individual sensor outputs. The sensor outputs are not linearly combined; they are not really nonlinearly combined either. They are simply used to form the MIMO model-fitting cost function that is minimized by a trellis search over the multiple symbol sequences. This trellis search is optimally implemented by the Viterbi algorithm. Nevertheless, we shall, to contrast this with linear combining, sometimes refer to it loosely as "nonlinear combining".

The joint demodulators described in this paper employ three distinct and complementary techniques for decreasing BER and/or trading BER for reduced computational cost, relative to the baseline Viterbi algorithm. All three techniques can be used in conjunction, simultaneously, and together they provide a powerful means for computation reduction and/or BER reduction.

The new MIMO demodulator is a modification of the conventional Viterbi algorithm for multi-sensor JMLSD (joint maximum-likelihood sequence detection). The normal Viterbi algorithm [1] has two phases: spawning and pruning. In the spawning phase, existing joint symbol sequences are extended with permutations of possible input symbols to create a new set of candidate joint sequences. In the pruning phase, this set of candidate sequences is pruned down to a survivor set according to dynamic programming principles and an analysis of their sum-of-squared errors relative to the input. The pruning process is optimum in the sense that it is guaranteed not to remove any candidate sequence that lies along the maximally likely symbol sequence path through the trellis. The survivors of the pruning are then spawned again and pruned again with the next symbol's worth of samples of the input signal, and the algorithm repeats through its phases.

The Viterbi algorithm offers optimal BER performance for IID equally-likely-symbol sequences, but is prohibitively costly for most joint demodulation applications. The number of candidate sequences reaches a size of B, raised to the power L(K+1), where B is the alphabet size, L is the number of signals, and K is the number of symbols in the state associated with each signal (the channel memory). Example values for some existing systems are as follows: numbers of candidates range from 4 to 262,144 for BPSK with K going from 1 to 2 and L from 1 to 6; from 8 to 262,144 for QPSK with K going from 1 to 2 and L from 1 to 3; and from 64 to 134,217,728 for 8PSK with K going from 1 to 2 and L from 1 to 3. Because of its optimal BER performance, we use the Viterbi algorithm to provide a baseline for BER measurements in those cases where it can be computed at reasonable cost ($L \leq 5$).

The first modification used in our MIMO demodulator applies constraints in the spawning phase. Known symbol values in the signals (due to header sequences, midambles, etc.) are taken into account to prevent the spawning of candidate sequences that will violate the signal format. In selecting survivor sequences for joint demodulation, a bit-error in one sequence tends to induce bit-errors in the other sequences. This happens because the change in the sum-of-squared-model-fitting-errors (SSE) cost due to a bit-error in one sequence can (depending on the channel parameters) be partially compensated for by changing a bit value in another sequence. The result is that the SSE cost of a joint sequence with multiple errors can be lower than that of a joint sequence with a single error. Constrained spawning can improve the BER performance of the Viterbi algorithm because it exploits some of the non-random nature of communications signals. It yields the true MLSE solution for the more accurate model that recognizes the existence of known symbols. This non-statistical constrained spawning produces both a decrease in computational cost and an improvement in BER performance by eliminating candidate sequences that are known to be invalid. The degree of improvement depends on the amount of known structure in the signals being demodulated. As an example, approximately 20% of the symbols are known in GSM signals, and somewhat fewer (closer to 10%) are known in IS-136 signals.

The second and more important modification used in our MIMO demodulator adds a third phase to the conventional two-phase (spawning/pruning) Viterbi algorithm. In this third phase, many sequences that survived the pruning phase are removed with selection based on an analysis of the distribution of changes in sum-of-squared-error values. Unlike the pruning phase, this phase of sequence removal is non-optimum, in the sense that sequences along the maximally likely path might be deleted accidentally. However, the probability of such an accidental deletion is controlled and can be adjusted to trade BER performance for computational cost. Furthermore, this statistical thinning can be generalized to accomplish even more aggressive survivor reduction in applications where it is desired to extract only a subset of the interfering signals. In such a case, higher BER in the joint demodulator can be tolerated for signals not to be extracted, thereby enabling further survivor reduction. For the objective of trading off BER for computational cost in the most efficient manner possible, this statistical thinning (ST) algorithm introduced here has several advantages over all prior approaches that we have been able to find in the literature, including well-known reduced-state sequence estimation (RSSE) methods [7] - [11], the M-algorithm [2] - [5], and the T-algorithm [6].

Typically, RSSE techniques reduce computation by grouping the trellis states into a smaller

number of state classes, and allowing only one survivor sequence in each class. The partitioning algorithm used to construct these state classes is, in effect, changing the structure of the trellis. This change can greatly reduce computation, but must be planned carefully to maintain BER performance. There does not seem to be any work on class definition that is directly relevant to joint demodulation with general channels, although the approach described in [11] might be applicable here.

In contrast to RSSE, the M, T, and ST algorithms do not alter the trellis -- they remove individual survivor sequences that are unlikely to form viable paths. This reduces the number of states that are visited by the algorithm at any given time. The M and T algorithms do this in a deterministic manner, whereas the ST algorithm does it non-deterministically. The M-algorithm is probably the best-known survivor reduction algorithm. The survivor selection function for the M-algorithm keeps the M lowest-cost candidate sequences. The T-algorithm is a lesser-known survivor reduction algorithm. It allows the survival of candidate sequences whose costs fall below a certain threshold, relative to the current minimum cost sequence. The ST algorithm is essentially a variation on the T-algorithm in which the threshold changes with the distribution of candidate costs according to a specific adaptation procedure.

Simply replacing the survivor selection function in the conventional Viterbi algorithm with the new survivor reduction function for either the M-algorithm or T-algorithm did not yield an efficient tradeoff of BER for cost. We found that both cost and BER performance could be improved considerably with hybrid algorithms, in which the optimal pruning of the Viterbi algorithm is followed by a survivor reduction phase. At any given BER, the added cost of the Viterbi pruning function is more than compensated for by its reduction of the candidate set.

However, it is reported in [6] that, for trellis decoding applications (which are distinct from the CCI separation applications addressed herein), a more efficient BER/cost tradeoff is obtained by using the T-algorithm without Viterbi pruning but with aggressive path truncation (forced decisions). A comparison of this approach to the Viterbi-augmented T-algorithm reported on herein has not been made for the applications of interest here.

The M, T, and ST algorithms are more effective for the applications of interest here than are the RSSE techniques. Our investigation has revealed that the Viterbi-augmented T algorithm and the ST algorithm typically provide more efficient tradeoffs than does the Viterbi-augmented M algorithm; and the ST algorithm has the advantage of using an adaptive threshold within the thinning criterion that has the potential to better accommodate severely nonstationary channels, including fading and fluctuating numbers of interfering signals.

The third modification used in our MIMO demodulator introduces per-survivor decision feedback [12]. In the conventional Viterbi algorithm, the length of the pulse model P, is $K + 1$, where K is the number of symbols of channel memory. (The additional symbol is the current one.) By making the model parameter P independent of the algorithm parameter K, we can reduce K while still accurately modeling the channel pulse. This is equivalent to feeding tentatively decided symbol values (those decisions within survivors) back into the pulse model. For any given values of K and P, we have $P - K$ - 1 feedback symbols. Because the size of the candidate set is an exponential function of $K + 1$, a reduction in K produces a great reduction in computational cost; moreover, the BER performance of the demodulator can often be preserved by keeping P large enough. Likewise, increasing P can sometimes improve the BER performance of a joint demodulator without significantly increasing the algorithm's computational cost. In addition to the per-survivor decision feedback, the MIMO demodulators evaluated in this paper also use path truncation at 5P symbols.

The advantage of linear combining (beamforming) over non-linear combining (MIMO demodulation) is that it offers a lower per-candidate processing cost, but this computational savings brings a BER penalty because so much information about the joint channel is discarded. This loss results from the fact that linear combining does not use the increased channel capacity resulting from the channel diversity due to multiple sensors in an effective manner. This loss is especially

significant if the sensors provide independent views of the environment, as is the case with orthogonally polarized sensors in an environment with a wide range of signal polarizations. By utilizing all available channel information, dual-pol joint demodulation can provide a substantial BER performance improvement. When combined with survivor reduction, the total cost of dual-pol joint demodulation can be lower than that of linear combining for a given BER. As the number of sensors increases, the channel capacity increases due to increased spatial diversity. This increased capacity enables more aggressive survivor reduction for a specified BER. The enhanced survivor reduction can result in a decrease in computational cost that more than offsets the increase in cost due to the increase in the number of sensors.

3. Simulations

Experiment 1: One and Two Sensors.

The GSM signal environments used in this study were constructed using cochannel combinations of six signals, all of which were beacon channels, fully loaded (no dummy bursts), with pseudorandom bits filling the data fields of each normal burst, carrier frequency offsets ranging from 5Hz to 20Hz, and timing offsets ranging from 1 symbol to 738.46 symbols. These signals were combined with powers in 0 dB and 3dB steps to form co-channel interference test environments. The GSM signal generator uses the known approximation of GMSK by complex PAM. The error of this model is about –22dB. Test cases where this modeling error may be significant are noted. The middle symbol of the GSM pulse contains only about 50% of the pulse energy. The middle three symbols contain over 95% of the energy. Most test cases were run with P = 4 to ensure a full pulse support and to allow for timing offsets among CCI signals of up to ±0.5 symbol. All of the GSM test environments were generated with a sample rate of four times the GSM symbol rate, and with additive white Gaussian noise extending over the full sampled bandwidth.

For all tests, the noise power was set at 10 dB below the strongest signal in the environment. TSNR figures indicate the ratio of total signal power to in-band noise power, where the GSM bandwidth is assumed to be the same as the symbol rate. The relationship between TSNR and primary-signal E_b/N_0 depends on the number of CCI and their relative powers. For 0dB SIR among all CCI, individual SNRs are 4.8dB, 6dB, 7dB, and 7.8dB below TSNR for 3,4,5, and 6 CCI, respectively.

All GSM tests were run for 54,166 symbol periods (0.2 sec). This test length was chosen because it allowed an integer number of cycles for all of the frequency offsets chosen for these test.

The T-algorithm performance curves are not included in this paper because of their similarity to those for the ST algorithm in the environments studied here where the only nonstationarity is due to differential Doppler shifts.

Bit-errors were detected by comparing the joint demodulator output against the modulator input bits. This comparison was performed for the data fields only. Fixed bit sequences defined by the GSM standard (and exploited by the constrained Viterbi algorithm) are not included in BER calculations.

Because of the similarity of the algorithms under consideration, we can justify expressing and comparing their computational costs in terms of the size of each candidate set scaled by the number of sensors. This policy is based on three assumptions: (1) Running time is proportional to the number of candidates, N, in software implementations. (In practice, running time is dominated by a term linear in N.) (2) For a given number of sensors, the per-candidate processing cost is about the same for all of these algorithms. (3) Computational cost rises linearly with the number of sensors, M. (In practice, running time rises sub-linearly in M.)

The results are presented in Figures 1 to 20 as graphs that plot average BER against average number of candidates. The “sweet spot” of such a plot is at the origin – 0 BER for 0 cost. The effectiveness of a survivor-reduced joint demodulator can be judged by how close its performance curve comes to this point. Where applicable, each graph is also marked with a horizontal dashed line that shows the approximate BER limit for good voice copy; a horizontal solid line that shows the baseline BER for the Viterbi algorithm with K = 2 (when available); and a vertical dashed line that shows the computational limit for real-time joint demodulation on a Compaq 1GHz ES-40 server.

# Of CCI	Signal Spacing	Primary E_b/N_0	TSNR	MIN BER	AV EBER	MAX BER
3	0 dB	4.2 dB	9 dB	15.1	15.7	16.2
3	0 dB	7.2 dB	12 dB	7.71	8.68	9.32
3	0 dB	10.2 dB	15 dB	3.28	4.51	5.21
3	0 dB	13.2 dB	18 dB*	1.24	2.47	3.12
3	3 dB	6.6 dB	9 dB	7.56	14.5	21.0
3	3 dB	9.6 dB	12 dB	3.39	7.0	10.1
3	3 dB	12.6 dB	15 dB	0.96	2.33	3.37
3	3 dB	15.6 dB	18 dB	0.23	.73	1.04
4	0 dB	6.0 dB	12 dB	17.2	18.8	19.5
4	0 dB	9.0 dB	15 dB	9.04	11.7	12.7
4	0 dB	12.0 dB	18 dB*	4.31	7.20	8.32
4	0 dB	15.0 dB	21 dB*	1.87	4.83	6.06
4	3 dB	9.3 dB	12 dB	7.11	17.2	27.6
4	3 dB	12.3 dB	15 dB	3.90	10.0	16.4
4	3 dB	15.3 dB	18 dB	1.57	4.52	7.49
4	3 dB	18.3 dB	21 dB*	0.46	1.76	3.06
5	0 dB	8.0 dB	15 dB*	18.3	20.8	21.9
5	0 dB	11.0 dB	18 dB*	10.6	14.1	15.4
5	0 dB	14.0 dB	21 dB*	5.68	9.74	11.2
5	0 dB	17.0 dB	24 dB*	3.13	7.15	8.59
5	3 dB	12.3 dB	15 dB	6.58	20.0	32.9
5	3 dB	15.3 dB	18 dB	4.18	13.9	23.5
5	3 dB	18.3 dB	21 dB*	2.26	8.75	15.0
5	3 dB	21.3 dB	24 dB*	1.14	5.45	9.48

Table 1: BER performance baselines for 3-CCI, 4-CCI, and 5-CCI GSM environments. * indicates those test cases where PAM approximation error for GSM may be comparable to the noise level, thus causing an inflation of BER. BER expressed in percentages.

The following abbreviations are used to label curves on the graphs: **VA** is the Viterbi algorithm, **C-VA** is the constrained Viterbi algorithm, **M** is the M-algorithm, and **ST** is the statistical thinning algorithm. Although not explicitly denoted in these labels, per-survivor feedback is used whenever $K < P - 1$.

For 3-CCI, 4-CCI, and 5-CCI GSM with K =2, the cost of the Viterbi algorithm is 512, 4096, and 32,768 candidates per symbol period, respectively. Running the Viterbi algorithm for a single sensor, without constraints, decision feedback, or survivor reduction, we get the BER performance figures shown in Table 1.

From this table, we can see that 4-CCI GSM environments are only marginally copyable (BER ≤ 10%) with a single sensor at 10dB E_b/N_0, and 5-CCI GSM environments are not copyable at all with a single sensor at 10dB E_b/N_0. For the 6-CCI cases, the cumulative error in the GSM PAM approximation can be as large as –14dB relative to the total signal power. At the same time, 6-CCI joint demodulation requires a TSNR on the order of 18dB. It is difficult to find 6-CCI GSM environments that can produce less than 10% BER from a single sensor, regardless of E_b/N_0 and TSNR. For this reason, single sensor 6-CCI test cases and baselines are omitted from this paper. On the other hand, dual-pol processing makes 6-CCI environments copyable, although the computational requirements are high enough to make this copy only marginally feasible with present-day hardware.

Dual-Polarization: The most striking effect of dual-pol joint demodulation is the improvement in BER. In the 4-CCI cases, environments that were marginal at best with single sensors yielded BERs < 2% when dual-pol processing was applied. In the 5-CCI and 6-CCI cases, dual polarization processing turned impossible environments into copyable ones with reasonably low BER.

For single and dual-pol processing of environments with 3 to 5 equal-power CCI GSM signals, we find that the BER improvements from dual-pol joint demodulation are equivalent to a 6 dB to 12 dB improvement in E_b/N_0. This is considerably better than the 3 dB improvement one might expect from linear combining of two sensors by simple addition.

Another effect of dual-pol processing is that it allows survivor reduction to operate more

efficiently, yielding smaller candidate sets. This happens because the additional channel capacity makes minimum-cost sequences more distinct, reducing the difficulty of the joint demodulation problem. Because the candidate sets can shrink by more than half, the total cost of dual-pol demodulation can be lower than that of single-sensor demodulation, even though the per-candidate cost may be twice as high.

Per-Survivor Decision Feedback: Changing K from 2 to 1 degraded single-sensor BER by roughly 1% in copyable environments. For dual-pol environments, BER losses were less than 1%. For the full VA, going from $K = 2$ to $K = 1$ speeds the algorithm by a factor of 2^L. When used in conjunction with survivor reduction, we observed speedups, due to reduction of K by one, by roughly a factor of 2.

Survivor Reduction: In all cases, statistical thinning matched or exceeded the performance of the M-algorithm in terms of computational efficiency. The difference was especially pronounced when survivor reduction was used in conjunction with significant decision feedback (K=1) or dual-pol processing.

Experiment 2: More Than Two Sensors

As illustrated here, somewhat surprisingly, the computational cost increments associated with adding more sensors are often more than offset by the resultant more-effective survivor reduction. The use of additional sensors can often decrease the total cost of demodulating a particular environment at a specified target BER. This result is demonstrated in the following simulations with 2 to 5 sensors and 2 to 6 CCI.

The simulated sensor arrays had the following characteristics: Sensors were arranged in symmetric circular arrays of 2, 3, 4, and 5 elements, each with a diameter of 2 wavelengths. Both isotropic and cartioid sensor patterns were simulated. Cartioid sensors were arranged with lobes facing away from the center of the array. Each cartioid sensor was modeled as having a gain of +3dB in the direction its lobe.

Each array configuration was tested in environments with the following properties: GSM beacon signals with nominal 960 MHz carriers were used for all environments. Environments contained 2 to 6 cochannel emitters, each having random carrier phase and carrier offset in the range of -40 Hz to +40 Hz (normal distribution). Emitters were placed in a plane, distributed uniformly, at distances of 30.5 meters (100 wavelengths) to 6.9 kilometers (6.25 symbol periods) from the center of the sensor array. Power control was simulated so that all signals arrived at the center of the array have received powers within 3dB of a reference level. In each environment, multipath interference was simulated by placing one image of each emitter in the above specified planar region. Each emitter's multipath image had an effective radiated power of 0 dB to -10 dB relative to its source and a distance from the array of 1 to 4 times that of its source. Image distance and power were selected randomly from uniform distributions. In each environment, noise was added at a test-specified level relative to the power control reference level.

The geometry of the environment used for this test is shown in Figure 21. (For environments of $n < 6$ CCI, the first n signals are used.) In Figure 21, filled circles represent transmitters and open circles represent multipath images. The receiving array is at the origin.

For these joint demodulation simulations, we used the constrained Viterbi-augmented ST algorithm with per-survivor decision feedback, with the degree of thinning adapted to provide a constant average BER of 10%, and with a pulse model length of $P = 4$ symbol periods and a channel memory of $K = 1$. Both the TSNR and the number of sensors were varied, and the computational cost of achieving the target 10% BER was observed. For brevity, only the 6-CCI test cases are presented here. There are two such test cases: one for isotropic sensors and one for cartioid sensors.

The results of the 6-CCI tests are shown in Figures 22 and 23. For each curve the candidate counts are scaled up by the number of sensors to obtain the computational cost in equivalent signle-sensor candidate counts. Each curve is defined over only a limited range of TSNR. Below this range, a 10% BER is not possible at any cost. Above this range, the demodulator produces a BER of less than 10%, even when operating at the minimum cost of one survivor per symbol period. Many of these curves have overlapping ranges, and in these overlapping ranges it is plain to see that in each case the receiver with a larger number of sensors has a lower cost. As examples, for omnidirectional antennas, going from 2 sensors to 3 sensors enables SNR to drop by 3dB (from

12dB to 9dB) with no change in cost; going from 3 sensors to 4 sensors enables cost to be cut in half while reducing SNR required by about 1dB. (Recall that the average SNR for each signal in these six-signal environments is about 7.8dB below the TSNRs shown in Figures 22 and 23.)

We also found that the use of cartioid field patterns provided a computational cost reduction roughly equivalent to 3dB TSNR improvement, which can be seen when comparing the two 6-CCI curve sets in Figures 22 and 23. This performance improvement is presumably due to the increased channel diversity afforded by the non-isotropic sensors.

4. Conclusions

The results of the simulations performed support the following conclusions:

- Statistical thinning and the T-algorithm offer a better BER-cost tradeoff than the M-algorithm under the following conditions:
  - Smaller alphabet – Performance differences were greater for GSM than for IS-136. (IS-136 results not shown)
  - Smaller signal spacing – Performance differences were usually greatest for equal-power environments. Both equal-power and 1dB-spaced (not shown) environments generally produced greater performance differences than 3dB or 5dB-spaced (not shown) environments.
  - Decision feedback – Performance differences were greater for the K = 1 cases than for the K = 2 cases.
  - Dual-pol – Performance differences were greater for dual-pol reception than for single-sensor reception.
- Dual-pol joint demodulation produces BER performance, for the GSM environments in this study, that is equivalent to an E_b/N_0 improvement of 8 to 12dB relative to single-sensor.
- With survivor reduction, the total cost of dual-pol joint demodulation can actually be lower than that of single-sensor processing (at the same BER).
- With survivor reduction, the use of per-survivor decision feedback roughly halves the computational cost of joint demodulation for GSM, regardless of the number of CCI. The BER loss is typically about 1%
- GSM should be copyable (BER ≤ 10%)for up to 6-CCI with dual-pol reception (at 10dB E_bN_0) and current-generation computers.
- Non-isotropic antennas enable more efficient BER/cost tradeoff than do isotropic antennas, presumably due to increased channel diversity.
- Required SNR as well as computational cost to meet a specified BER target can both be decreased by increasing the number of antennas
- Cost reduction factors achieved range from 2 to 4 orders of magnitude. The cost reduction factors realized by the ST C-VA algorithm with per-survivor decision feedback (P=4, K=1), relative to the baseline VA (P=3, K=2) are 6557 for 6 CCI and 1725 for 5 CCI, both at 5% BER, 407 for 4 CCI at BER = 2%, and 85 for 3 CCI at BER = 1%, all for dual-pol reception and 0 dB signal spacing.
- Although exact channel estimates were used in all simulations reported here, other simulations performed show that accurate estimates of all channel parameters needed by the MIMO demodulator can be obtained in the presence of ten or more CCI by properly exploiting all the known bit sequences in GSM.

5. Acknowledgment

The research reported herein was supported in part by the National Reconnaissance office Director's Innovation Initiative, contract NRO-000-99-R-0176, and the National Science Foundation, Small Grants for Exploratory Research, NSF 0003284.

6. References

[1] G. D. Forney, "The Viterbi Algorithm," *Proc. IEEE*, Vol. 61 pp. 268-278, Mar. 1973.

[2] J. B. Anderson and S. Mohan, "Sequential coding algorithms: A survey and cost analysis," *IEEE Trans. Comms*., vol. COM-32, pp. 169-176, Feb. 1984.

[3] T. M. Aulin, "Breadth-first maximum-likelihood sequence detection: Basics," *IEEE Trans. Comms*., vol. 47, pp. 208-216, Feb. 1999.
[4] T. M. Aulin, "Breadth-first maximum-likelihood sequence detection," Chalmers Univ. of Tech. Report, Oct. 1992.
[5] N. Seshadri and J. B. Anderson, "Decoding of severely filtered modulation codes using the (M, L) algorithm," *IEEE J. Sel. Areas. Comms*., vol. 7, no.6, Aug. 1989.
[6] S. J. Simmons, "Breadth-first trellis decoding with adaptive effort," *IEEE Trans Comms*., vol. 38, no. 1, Jan. 1990.
[7] M. V. Eyuboglu and S. U. H. Qureshi, "Reduced-state Sequence Estimation with Set Partitioning and Decision Feedback", *Proceedings of Global Telecommunications Conference*, *GLOBCOM '86*, Houston Texas, USA, pp.1023-1028, Dec. 1986; also in *IEEE Trans. Comm*., vol. COM-36, pp. 13-20, Jan. 1988.
[8] F.L. Vermuelen and M.E. Hellman, "Reduced-State Viterbi Decoding for Channels with Intersymbol Interference," *Proceedings of Intern. Conf. On Comm*., ICC '74, Minneapolis, MN, pp. 37B.1-37B.9, June 1974.
[9] R. E. Kamel and Y. Bar-Ness, "Reduced-Complexity Sequence Estimation Using State Partitioning," *IEEE Trans. Comms.*, Vol. 44, no 9, Sept. 1996, pp. 1057-1063.
[10] B.E. Spinnler and J.B. Huber, "Design of Hyper States For Reduced-State Sequence Estimation," *Intl. Jour. Electron. Commun*., Vol 50, No 1, pp. 17-26, 1996.
[11] Torbjorn Larsson. *A State-Space Partitioning Approach to Trellis Decoding*, PhD dissertation, Chalmers University, Goteborg, Sweden, 1991.
[12] R. Raheli, et. al. "Per-survivor processing: A general approach to MLSE in uncertain environments," *IEEE Trans. Comms.*, Vol. 43, April 1995, pp 354-364.

7. Figure Captions

Figure 1. Results for 3-CCI @ 0dB Spacing, K = 1, Single Sensor
Figure 2. Results for 3-CCI @ 0dB Spacing, K = 2, Single Sensor.
Figure 3. Results for 3-CCI @ 3dB Spacing, K = 1, Single Sensor.
Figure 4. Results for 3-CCI @ 3dB Spacing, K = 2, Single Sensor.
Figure 5. Results for 3-CCI @ 0dB Spacing, K = 1, Dual-Pol
Figure 6. Results for 3-CCI @ 0dB Spacing, K = 2, Dual-Pol
Figure 7. Results for 3-CCI @ 3dB Spacing, K = 1, Dual-Pol
Figure 8. Results for 3-CCI @ 3dB Spacing, K = 2, Dual-Pol
Figure 9. Results for 4-CCI @ 0dB Spacing, K = 1, Dual-Pol
Figure 10. Results for 4-CCI @ 0dB Spacing, K = 2, Dual-Pol
Figure 11. Results for 4-CCI @ 3dB Spacing, K = 1, Dual-Pol
Figure 12. Results for 4-CCI @ 3dB Spacing, K = 2, Dual-Pol
Figure 13. Results for 5-CCI @ 0dB Spacing, K = 1, Dual-Pol
Figure 14. Results for 5-CCI @ 0dB Spacing, K = 2, Dual-Pol
Figure 15. Results for 5-CCI @ 3dB Spacing, K = 1, Dual-Pol
Figure 16. Results for 5-CCI @ 3dB Spacing, K = 2, Dual-Pol
Figure 17. Results for 6-CCI @ 0dB Spacing, K = 1, Dual-Pol
Figure 18. Results for 6-CCI @ 0dB Spacing, K = 2, Dual-Pol
Figure 19. Results for 6-CCI @ 3dB Spacing, K = 1, Dual-Pol
Figure 20. Results for 6-CCI @ 3dB Spacing, K = 2, Dual-Pol
Figure 21. Emitter and reflector placement
Figure 22. Computational cost of joint demodulation for 10% BER vs total SNR for various numbers of isotropic antennas in an environment with 6 GSM CCI.
Figure 23. Computational cost of joint demodulation for 10% BER vs total SNR for various numbers of cartioid antennas in an environment with 6 GSM CCI.

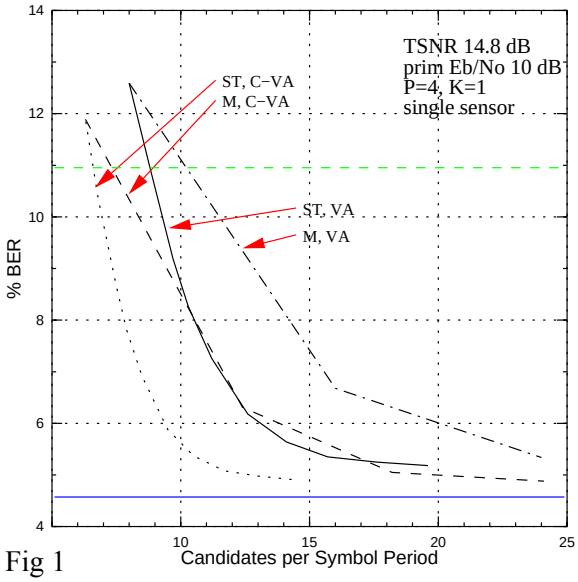

Fig 1

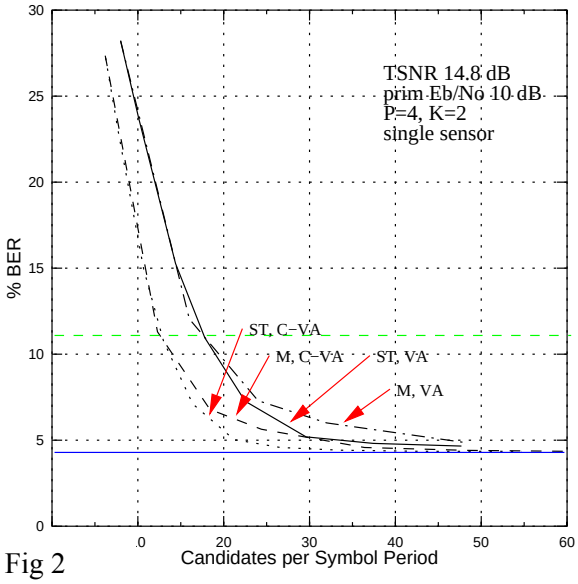

Fig 2

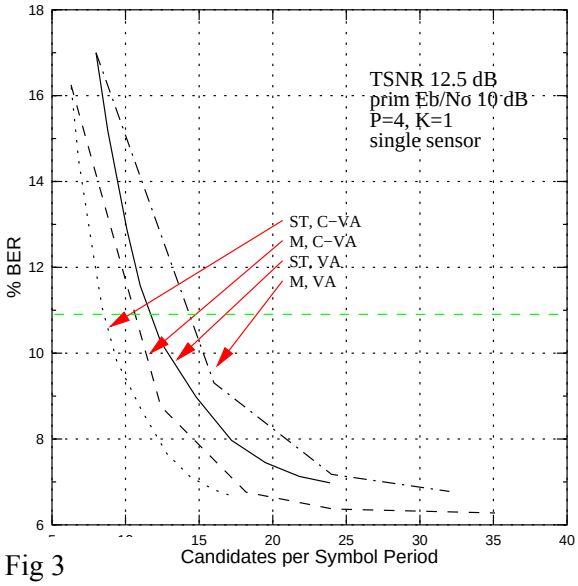

Fig 3

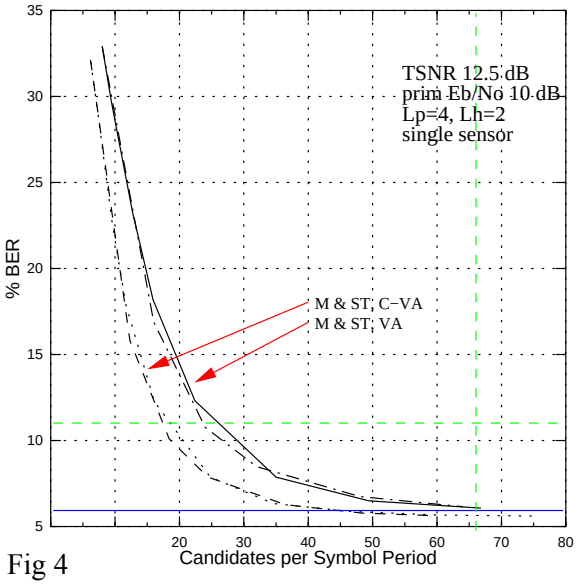

Fig 4

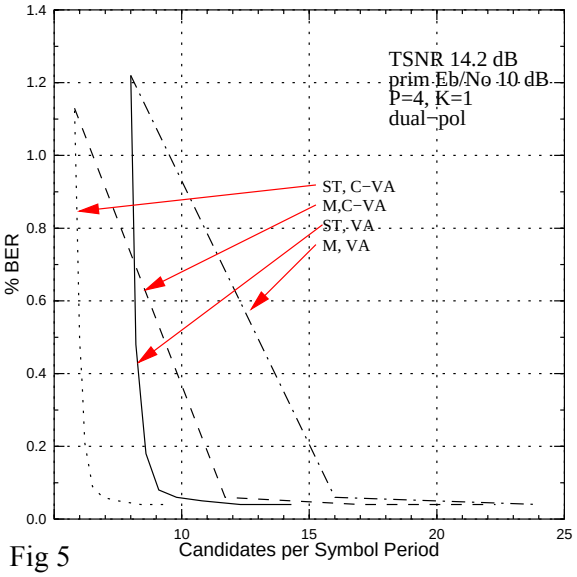

Fig 5

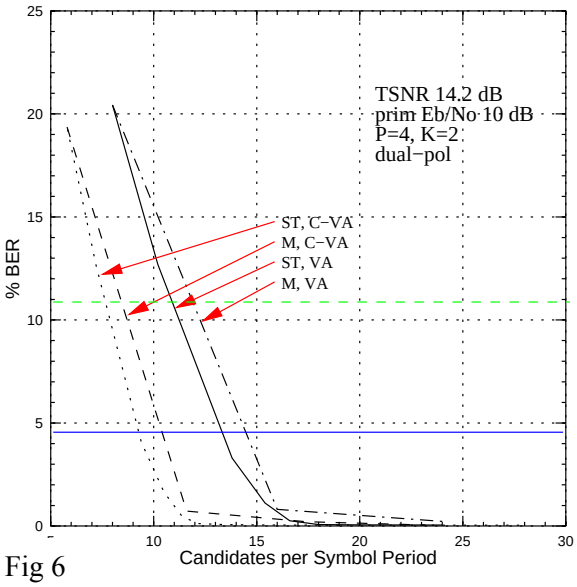

Fig 6

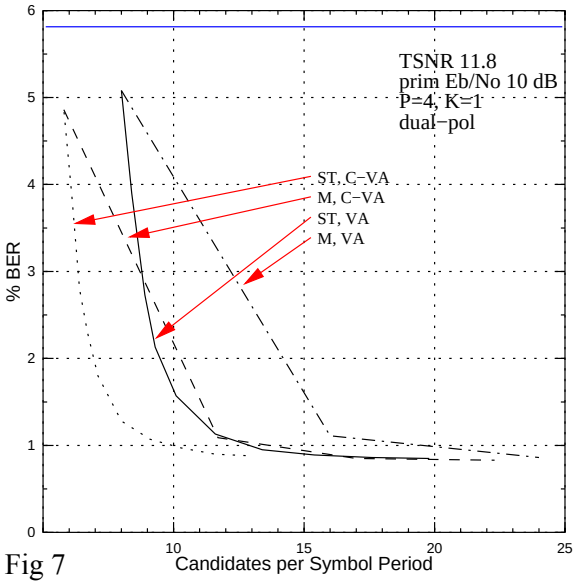

Fig 7

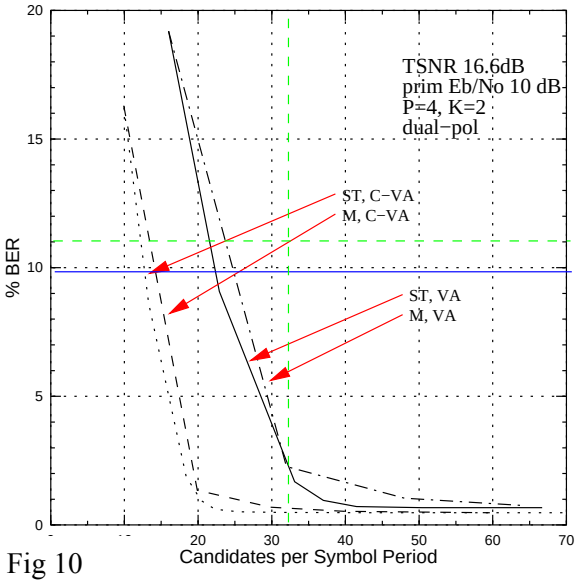

Fig 10

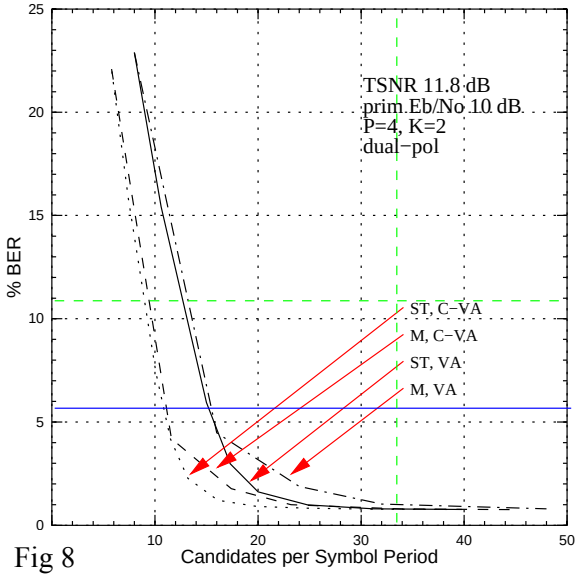

Fig 8

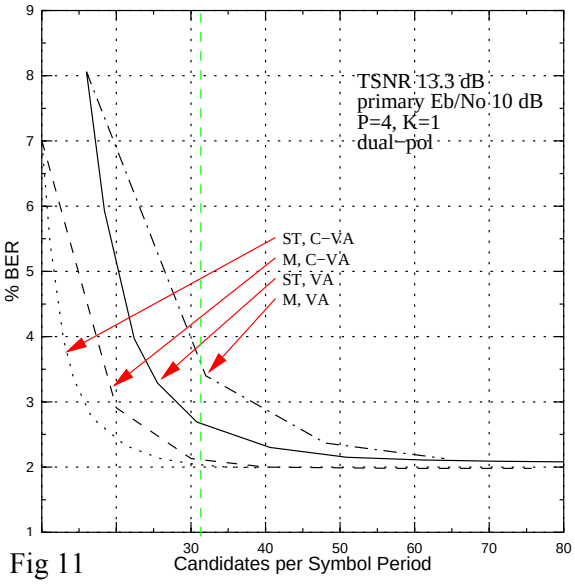

Fig 11

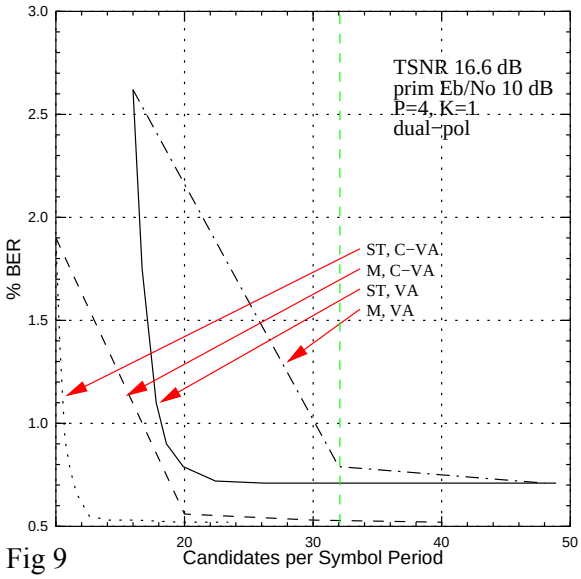

Fig 9

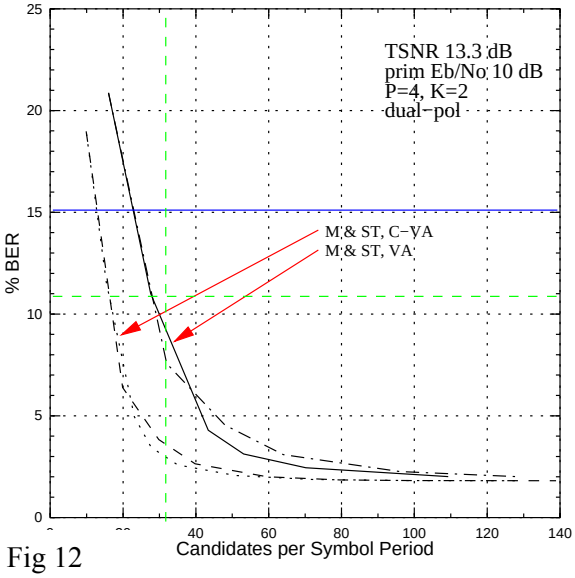

Fig 12

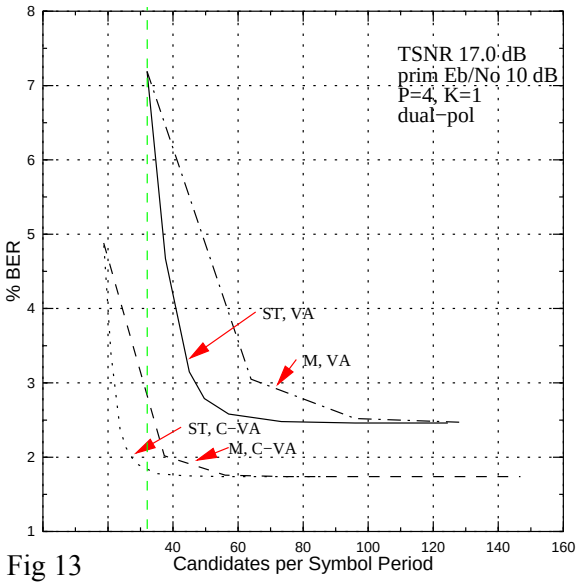

Fig 13

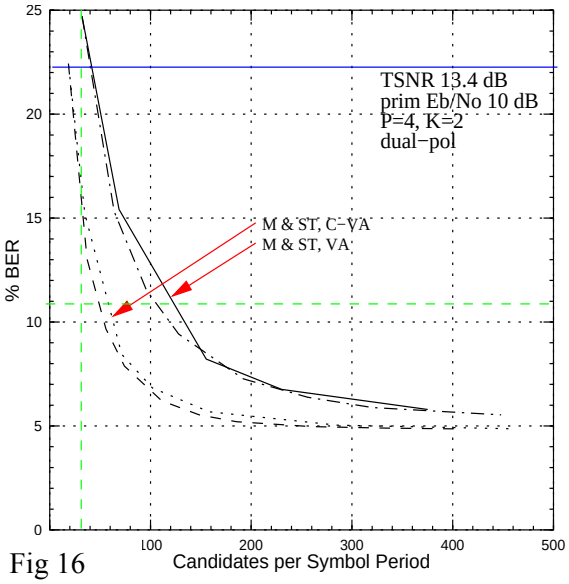

Fig 16

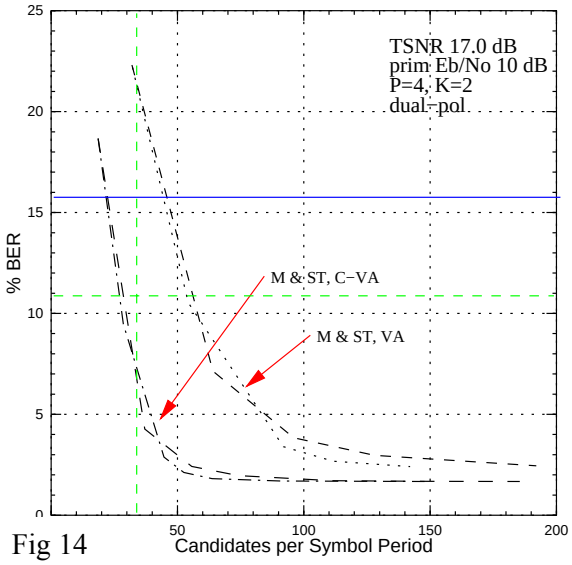

Fig 14

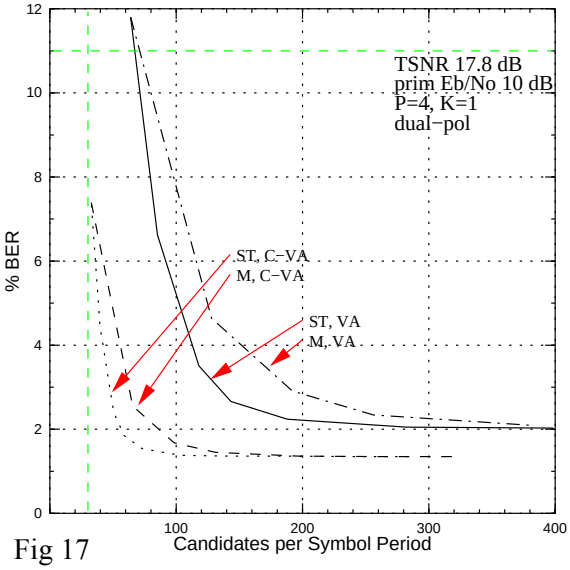

Fig 17

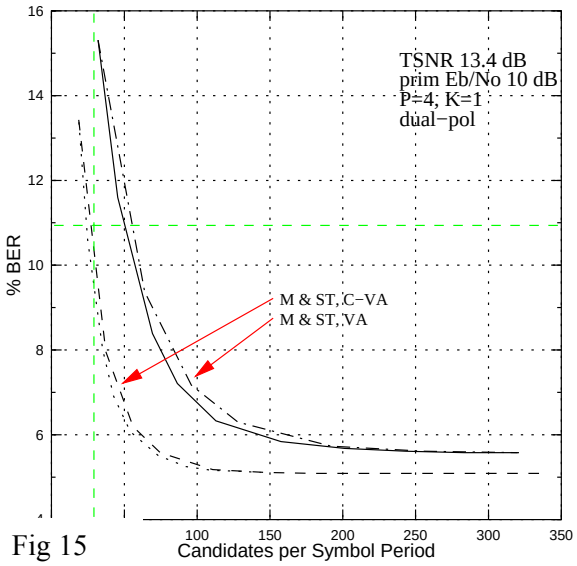

Fig 15

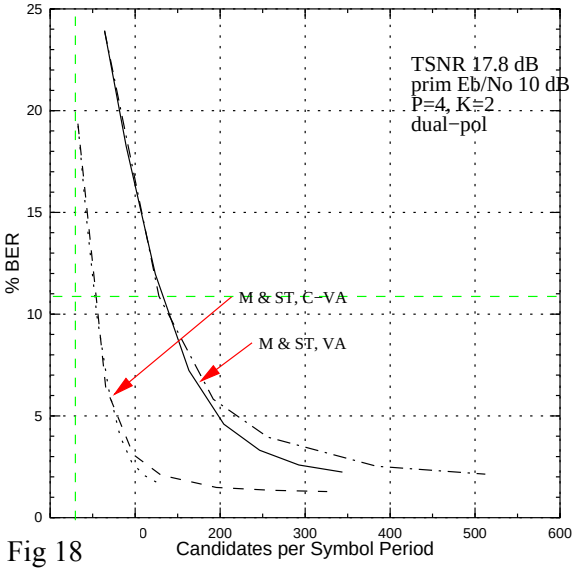

Fig 18

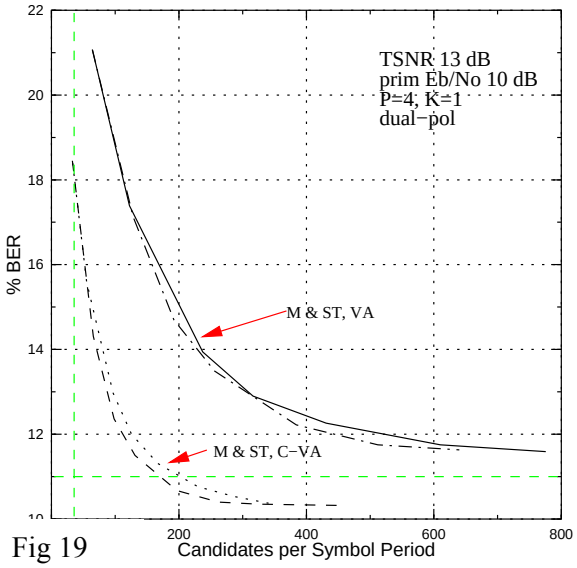

Fig 19

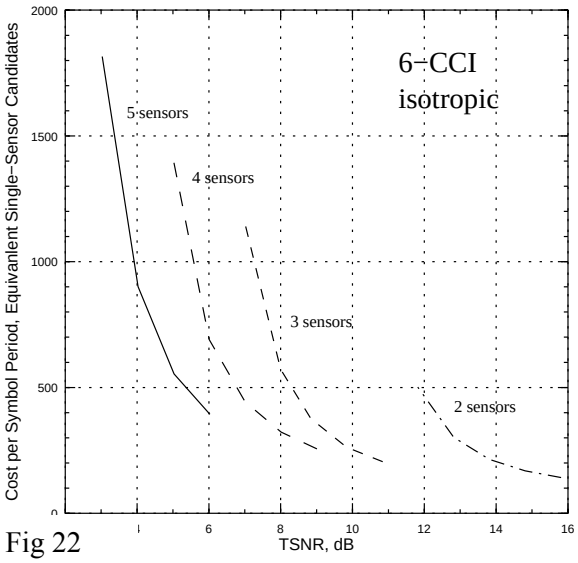

Fig 22

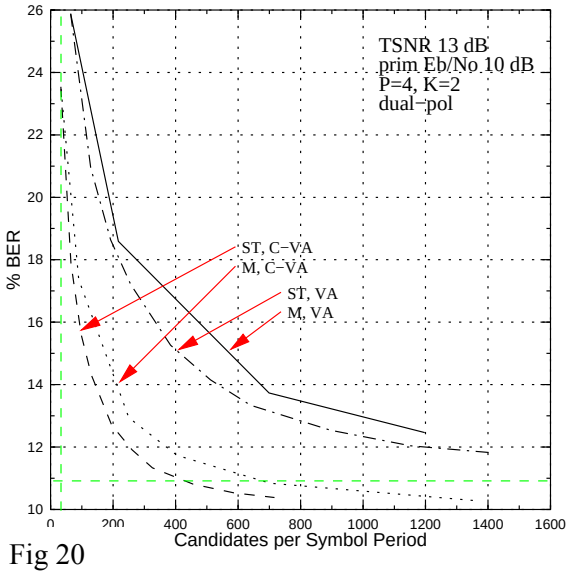

Fig 20

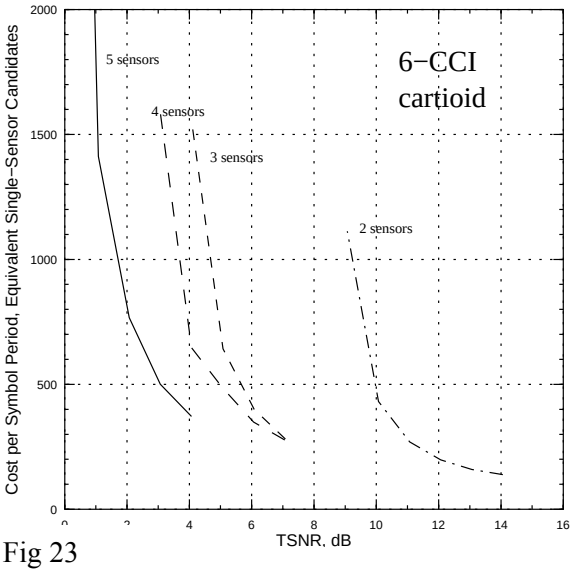

Fig 23

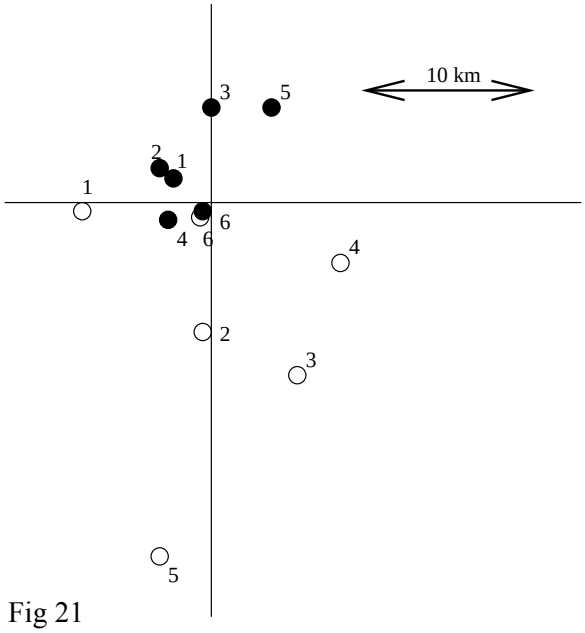

Fig 21