(Section from the book "Computer networks", authors: S.I.Samoilenko, A.A. Davidov, V.V.Zolotarev, E.I.Tretjakova, "Science", Moscow, 1981, p.278)

Fundamental principles of multithreshold decoding

The comparison of different decoding methods demonstrates, that the most effective algorithms, for example, realizing Viterbi decoders for lengthy codes are extremely complex, and the characteristics of simple algorithms in channels with a large noise are rather unsatisfactory.

Fundamentals of a multithreshold decoding method (MTD) are described below for linear codes permitting to increase efficiency of threshold decoders essentially. Let's describe at first qualitatively idea of suggested algorithm.

Let's take as an example a convolution orthogonal code (SOC) with **R=1/2.** Customary threshold decoder (TD) with a feedback makes a decision about decoding symbols on the basis of **J=d-1** check bits keeping error in **(d-1) (d-2)/2** information symbols, which ones were not decoded yet, the same number of the decoder errors and **(d-1)** errors in control characters. In this case low veracity of checks at large **d** is explained to that almost always checks are distorted owing to errors in information symbols, and if $\mathbf{p_0}$ is small enough, i. e. $\mathbf{P_1 (e) << p_0}$, then in the main they are errors in symbols received from a channel and yet not corrected in TD.

Let's admit now, that the decoding circuit is made, in which convolutional code information symbols, corrected by first TD_1, and also the check symbols from communication channel become again input for other TD_2 of the same type. The second TD_2 behavior will be determined by two factors. On the one hand, the veracity of checks in TD_2 is in the average higher, as the error probabilities in information symbols most hardly distorted checks, now became much lower. It can provide more high quality of decoding with usage of two TD chain as contrasted to the first one alone. It is clear, that number of TD can be more large, than 2. The lower limit of such scheme efficiency would be determined by veracity of a reception for those control symbols, which one enter in checks going at a threshold switch of TD. Completely similar speculations result in the same conclusions and for block codes.

On the other hand, all errors and checks of the second decoder appear now dependent, and even the probability of the first error $P_1(e)$ for second TD_2 can not be counted so simply any more, as for first TD_1. In the worst case it is possible, that the errors on the output of TD_1 are packed so, that while TD_1 does not make errors, TD_2, apparently, is not required. But if after each error in TD_1 due to a strong packaging (because of error propagation effect, EP) arises such an error burst, that second

TD_2, which due to code selection is oriented a correction of independent accidental errors, can not correct it at all. Therefore and in this case TD_2 can appear useless.

Below it will be shown, that, nevertheless, efficiency of considered further multithreshold decoding (MTD) can be very high, if simple conditions are taken into account. Most of further conclusions are fair both for convolution, and for block codes.

Algorithm of multithreshold decoding

Let's consider binary linear systematic block or convolutional code with code rate **R=k/n**, where **k** - number of information symbols, **n** - code word length.

After transmission through binary symmetric channel (BSC) without memory the optimum decoder (OD), minimizing mean probability of an error, among set of $\mathbf{2^k}$ the equiprobable code words **{Ā}** selects such a code vector**,** for which Hamming distance **r =** $|\overline{Q} \oplus \widetilde{A}|$**,** where $\overline{Q}$ **-** , $\oplus$- **mod2** addition, would be minimum on all set **{Ā}.**

Let any binary vector of length **n** we shall represent as a of vectors with length **k** and **(n-k)**, relating to information and control parts of vector accordingly: $X = (\overline{X}_I, \overline{X}_V)$.

Then in the supposition, that the check matrix of a code is submitted in a systematic view $H = (C : I)$, they may prove

Lemma.

For any code vector $\overline{A}$ and received word $\overline{Q}$ it is fair

$$\overline{A} \oplus \overline{Q} = (\overline{D}, H(\overline{Q}_I \oplus \overline{D}, \overline{Q}_V)), \tag{1}$$

where the vector $\overline{D}$ of length **k** is determined by a relation

$$\overline{A}_I = \overline{Q}_I \oplus \overline{D}. \tag{2}$$

The proof. By virtue of code linearity

$$\overline{S} = H(\overline{Q}_I \oplus \overline{D}, \overline{Q}_V) = H(\overline{A}_I, \overline{A}_V \oplus \overline{A}_V \oplus \overline{Q}_V) = H \cdot \overline{A} \oplus H(\overline{0}_I, \overline{A}_V \oplus \overline{Q}_V),$$

where $\overline{0}_I$ - zero informational word.

Since $H\overline{A} = 0$ for $\overline{A}$ - code word, and $H(\overline{0}_I, \overline{A}_V \oplus \overline{Q}_V) = \overline{A}_V \oplus \overline{Q}_V$, because $(\overline{A}_V \oplus \overline{Q}_V)$ is multiplied only on a submatrix **I** (it consists of diagonal "ones") of a check matrix **H**, one get, that the vector $\overline{S}$ is

$$\overline{S} = \overline{A}_V \oplus \overline{Q}_V. \tag{3}$$

Carrying out in right part of (1) changes with allowance of (2), we discover, that

$$(\overline{D}, \overline{S}) = (\overline{D}, \overline{A}_V \oplus \overline{Q}_V) = (\overline{D} \oplus \overline{Q}_I \oplus \overline{Q}_I, \overline{A}_V \oplus \overline{Q}_V) = \overline{A} \oplus \overline{Q}.$$

The lemma is proved.

Its contents is encompass in that the difference $\overline{B} = \overline{Q} \oplus \overline{A}$ for any received vector $\overline{Q}$ and any code word $\overline{A}$ is determined by a couple of vectors $(\overline{D}, \overline{S})$. By exhaustive search of all vectors $\overline{A}$ it is possible to find vector $\hat{A}$, minimizing $|\overline{B}|$ and being the solution of the optimum decoder (OD). By virtue of definition at

$\overline{D}=0$ vector $\overline{S}$ - is a customary syndrome of the received vector $\overline{Q}$: $\overline{S}$ =**H** $\cdot \overline{Q}$. For a simplicity of presentation we shall name $\overline{S}$ hereinafter and for $\overline{D} \neq 0$ as a syndrome, as this generalization is natural and does not result hereinafter in any contradictions. Let's mark also, that at each $\overline{A}$ change there is no necessity to calculate anew all components of a syndrome. It is enough at each change step to invert only those components of $\overline{S}$, which one contain odd number of errors in changed informational symbols.

Let's consider now new decoding algorithm, which one is very close to the threshold one. Let at the first preliminary stage the decoder executes for $\overline{D}=0$ calculus and storage of vector $\overline{S}$. Then the fulfillment of main decoding procedure itself starts. At each step the decoder calculates the customary sum a component of a syndrome **s_{jk}** keeping as components an error in a decoding symbols (i. e. the sum of checks $s_{jk} \in \{S_j\}$, where $\{S_j\}$ - set of checks components for **e_j**, conforming to a decoding symbol **i_j**, and symbol **d_j**, component of vector $\overline{D}$, also relating to a symbol **i_j**:

$$L_j = \sum_{s_{jk} \in \{S_j\}} s_{jk} + d_j. \tag{4}$$

Thus we shall suppose, that originally, that $\overline{D}=0$ because before the beginning of decoding operations in memory of the decoder there is an only received vector $\overline{Q}$ and the decoder has no any other more preferential hypotheses about the received vector.

Let's select a threshold **T** to be equal to half of all addends in (4). For SOC this number is equal **T=d/2 = (J+1) /2**. Let, at last, all **J=d-1** checks, i_j and d_j invert at $L_j > T$ and remain invariable at $L_j \leq T$.

The offered procedure at the start attempt of decoding, while every **d_j**=0 , coincides with customary algorithm for TD. Let's call hereinafter decoder realizing suggested algorithm, multithreshold decoder (MTD). The selection of such name will be clear from further consideration.

Thus the theorem 1 is fair.

Theorem 1. <u>The main theorem of multithreshold decoding</u>.

If at arbitrary **j**-th a step MTD changes a decoding informational symbol **i_j** , then:

a) thus MTD finds the new code word $\overline{A}_2$, more close to the received vector $\overline{Q}$ than that code word $\overline{A}_1$, corresponding to **i_j** with its value before **j**-th step of decoding

$$|\overline{B}_1| \overset{\Delta}{=} |\overline{A}_1 \oplus \overline{Q}| > |\overline{A}_2 \oplus \overline{Q}| \overset{\Delta}{=} |\overline{B}_2|;$$

б) after the **j**-th step decoding termination the processing of any next symbol $i_k, k \neq j$, is possible, so that at its change the further approaching to the received vector $\overline{Q}$ will be carried out.

The proof. Before the beginning of symbol $\mathbf{i_j}$ decoding in accordance with lemma it is fair

$$\left(\overline{D}_1,\overline{S}_1\right)=\left(\overline{A}_{1I}\oplus\overline{Q}_1,H\left(\overline{Q}_I\oplus\overline{D}_1,\overline{Q}_V\right)\right)=\overline{A}_1\oplus\overline{Q},$$

where

$$\overline{A}_1=\left(\overline{A}_{1I},\overline{A}_{1V}\right),\overline{A}_{1I}=\overline{Q}_I\oplus\overline{D}_1.$$

Weight of vector $\overline{B}_1$ before this step, equal $|\overline{B}_1|=|\overline{D}_1|+|\overline{S}_1|$, is possible to be presented by a customary sum of weights $W_1=L_{1j}+X$, where L_{1j} is determined by expression (4) and is equal to sum of all checks and symbol d_j, at a threshold switch; X - weight a remaining component $\overline{S}_1$ and $\overline{D}_1$, not included in L_{1j}.

Let's consider vector $\overline{A}_2$, distinguished from $\overline{A}_1$ only in one character $\mathbf{i_j}$, and difference $\overline{B}_2=\overline{A}_2\oplus\overline{Q}$, conforming to it. As $\overline{B}_1$ and $\overline{B}_2$ differ among themselves only in those components, which are inputs for a threshold switch, $|\overline{B}_2|=L_{2j}+X$ where $L_{1j}+L_{2j}=J+1$, because by virtue of a code linearity each check and the symbol $\mathbf{d_j}$, are exactly in one of two vectors, are equal 1.

As MTD changes i_j, if $L_{1j}>T$, for this purpose it is necessary, that it was $L_2<L_1$ and, therefore, $|\overline{B}_1|>|\overline{B}_2|$. So the point a) of the theorem is proved.

Further, apparently, if the symbol $\mathbf{i_j}$ did not change, it is possible to decode any other symbol $i_k,k\neq j$, as thus the conditions of a lemma are saved. In a case of $\mathbf{i_j}$ change pursuant to the rules of MTD working, after decoding $\mathbf{i_j}$ the equations $\overline{A}_{2I}=\overline{Q}_I\oplus\overline{D}_2$ and $\overline{S}_2=H\left(\overline{Q}_I\oplus\overline{D}_2,\overline{Q}_V\right)$ take place, where $\overline{D}_2$ differs from D_1 in a symbol $\mathbf{d_j}$, as at change (according to original TD algorithm - through a feedback from a threshold switch) checks attributing to $\mathbf{i_j}$, those components $\overline{S}_1$ are inverted, in which ones $\overline{S}_2$ differs from $\overline{S}_1$. From here we receive, that after change for defined above vectors $\overline{D}_2$, $\overline{A}_2$ и $\overline{S}_2$ appears true next relation:

$$\left(\overline{D}_2,\overline{S}_2\right)=\left(\overline{A}_2\oplus\overline{Q}\right),$$

similar to that one, which took place before change $\mathbf{i_j}$. Thereby at the subsequent steps of decoding and further changes of symbols $i_k,k\neq j$, the approaching to the message $\overline{Q}$, received from a channel will implement also.

The main MTD theorem is proved.

We have shown, that MTD at each change of decoding symbols comes closer to vector $\overline{Q}$, i.e. at each change of symbols it finds the new code word, more close to the OD solution. MTD all the time compares these more and more verisimilar vectors $\overline{A}_i$ only with that the new possible solutions, which one differ from the next current solution only in one information symbol. In any time moment the informational part of one of compared code words is in the decoder. In a case if the new possible code word will appear closer to the received message, than that, which one is in MTD, the decoder chooses it and the further comparisons are made already with this new vector $\overline{A}_i$. It is clear, that basically it is possible to realize large enough number of decoding attempts and approaching to the OD solution - vector $\widetilde{A}$. It is in essence important, that at final number of reviews (decoding iterations I) the decoder of total received vector with **k** information symbols, that, , is certainly, always true, MTD complexity is the same, as for customary TD - linear. It is apparent, because at number of iterations **I** the decoder executes **N=I·k** of attempts of symbol decoding in a received message.

Let's allow further, that MTD has reached the OD solution, i.e. in the information register MTD there are symbols of vector $\tilde{A}$. Then it is fair

Consequence from the main MTD theorem.

MTD will not change the OD solution.

The proof. If MTD has changed at any step though one symbol in vector $\tilde{A}$, it would mean, that other code vector was found, which one is closer to $\overline{Q}$, than $\tilde{A}$, that is impossible, because, by definition the nearest word to $\overline{Q}$ is the vector $\tilde{A}$.

The consequence is proved.

The principled moment is that the consequent demonstrates the MTD solution stability relatively the optimum solution: having reached it, MTD will remain in it. It is very important when the algorithm has a capability of repeated symbol changing.

It is possible also to note, that in the proof of the main MTD theorem the uniqueness of a decoding symbol $\mathbf{i_j}$ was not used essentially. So that given procedure may be used at once and to decode a group of informational symbols.

The description of MTD operation principles allows to understand the causes of name selection for a new algorithm. At implementation of a convolutional code it is necessary simply to lengthen all decoder shift registers, on which one was plotted customary TD, and on them to add still some of threshold switches working the same manner as and TD of a first stage. So, all code symbols pass through certain set of threshold switches, being step-by-step cleaned from errors.

For block codes the received symbols move through the cyclically convolute registers and too multiply miss one or several threshold switches. This version of MTD application is shown in computer movie about MTD, suggested for you on our site. This common property of MTD, fair also for non-binary and many other codes also determine name selection of this error correction method at large noise of a channel.

MTD efficiency

To get estimations of MTD efficiency only by computational methods is extremely difficult. Therefore for large noise levels it is more expedient a part of researches to conduct with the help of simulation, because thus simply enough to receive indispensable statistics at simultaneously more precise direct estimation of a main decoder specification, which one is of interest - mean probability of an error per bit $\mathbf{P_b(e)}$ at the MTD output.

For an estimation of MTD working quality and the interpretations of received simulation outcomes following items of information are useful that are introduced below.

Let as a result of a gradual approaching to the OD solution MTD has find code vector $\widetilde{A}^*$, distinguished from $\widetilde{A}$ only in one information symbol $\mathbf{i_j}$. Conditions below are formulated, at which one MTD will reach the solution of OD.

The theorem 2.

Let MTD executes decoding timelagged, т. е. it will change $\mathbf{i_j}$, only if the check sum, conforming to it, is more, than any sum of checks relating other symbols. Then if the solution $\widetilde{A}$ for the given received vector $\overline{Q}$ is unique, after completion of MTD decoding procedure will change $\mathbf{i_j}$, and its solution will coincide with optimum one.

The proof. Let vectors at the MTD input correspond to the solution distinguished from the optimum solution in a symbol $\mathbf{i_j}$, and the sum, relating it, on a threshold switch is $a_j, a_j > T$. Let's show, that it is strictly more then sum $\mathbf{a_k}$ relating any other symbol $i_k, k \neq j$.

Let's demonstrate it by contradiction.

Let there will be $i_k, k \neq j$, such, that $a_k \geq a_j$. Then, as in a symbol $\mathbf{i_j}$ $\widetilde{A}^*$ is distinct of $\widetilde{A}$, $a_j = T + b_j > T, b = 0{,}5; 1{,}5\ldots$ and $a_k = T + b_k > T, b_k \geq b_j$.

Let's designate $W_j = a_j + x = |\overline{B}_j| = |\overline{Q} \oplus \widetilde{A}^*|$, where $\mathbf{x}$ - weight of all others components $\overline{S}$ and $\overline{D}$, that did not come at a threshold switch input.

If $b_k \geq b_j$, after $\mathbf{i_k}$ inverse and conforming to it the components of vectors $\overline{S}$ and $\overline{D}$ the sum on a threshold decreases at value $\Delta = a_k - (J + 1 - a_k) = 2a_k - J - 1 = 2T + 2b_k - J - 1 = 2b_k > 0$. After inverse $\mathbf{i_j}$ the sum at a threshold would decrease at $\mathbf{2b_j}$. As with input vector, which one would change in both these cases, was $\widetilde{A}^*$, it corresponds to a case, when the code vector $\widetilde{A}_k^*$ distinguished from $\widetilde{A}^*$ in a symbol $\mathbf{i_k}$, is closer to $\overline{Q}$, than $\widetilde{A}$, that is impossible. By virtue of $\widetilde{A}$ uniqueness a case $\mathbf{a_k{=}a_j}$, is impossible as well. But it means, that the sum a_j is maximum among all a_i and MTD really will change $\mathbf{i_j}$.

The theorem is proved.

Let's mark, that most of self-orthogonal convolutional codes correspond to introduced limitations in this theorem. The given outcome is fair irrespective of valid or erroneous decision about $\mathbf{i_j}$ was made by OD. So the considered multithreshold decoder does not degrade the optimum decision and can correct single deviations from the optimum decision. This property MTD is naturally to call as a stability relatively the OD decision.

For systematic convolutional codes the theorem 2 can be reformulated so, that the sum $\mathbf{a_j}$ should be maximum among all $\mathbf{a_j}$ only within the limits of constraint code length $\mathbf{n_A}$ to $\mathbf{i_j}$. It is connected with the fact that the sums relating information symbols, which one are at farther distances, appear necessarily consisting of a different component of vectors $\overline{S}$ and $\overline{D}$ and, therefore, not dependent from each other. Thus MTD of a convolutional code will correct all single deviations from the optimum solution, disjointed by interval more then $\mathbf{2n_A}$.

At last, we shall formulate yardsticks of quality of decoding MTD, which one can be utilized at the analysis of outcomes of simulation of activity of decoders.

The theorem 3.

Let MTD repeatedly decodes in a certain order each information symbol of the message. Then, if it has made an alone error in a symbol $\mathbf{i_j}$ and can not correct it at secondary attempt of decoding, OD also will make error in this message.

The proof. Let error is accomplished in a symbol $\mathbf{i_j}$. As any symbols were not changed at following attempt of decoding, the sums at a threshold switch in all cases were less T. But it means, that the exact code word with inverted $\mathbf{i_j}$ can not be the solution, because the sum will be in this case more then T and, means, it will be farther from $\overline{Q}$, than the decision corresponding to informational register of MTD. They can find other code words distinguished from true code vector in some number of informational symbols and located closer to $\overline{Q}$ than decision of MTD. But it means, that OD will made an error too.

The theorem is proved.

Last theorem allows easily to identify those errors arising at usage MTD, which one would make and OD: all single errors MTD, not corrected at repeated decoding, result in errors and at usage OD.

Let's remark, that the greatest necessity of coding appears in channels with a moderate and considerable noise. Behavior TD at a large noise level is much more difficult to investigate and the advantages of application sequentially connected decoders in a convolutional code or repeated symbol review for a block code is not so obvious. Therefore most fast way to get the probabilistic characteristics MTD at large noise levels as a result of computer simulation. Certainly, and for all other powerful algorithms of decoding the computer simulation is an single method of analysis of their efficiency at a large noise level.

=========