The length of the byte-oriented STM-M frame is $\tau_f = 125 \mu s = M \times 270 \times 9$ bytes. The first $M \times 9$ bytes are not scrambled and include the frame alignment signal, a sequence of $6 \times M$ bytes. This sequence consists of $3 \times M$ consecutively repeated $A_7$=11110110 bytes, followed by $3 \times M$ consecutive $A_7$=00101000 bytes. Other deterministic frame overhead contents can also be used for synchronisation, but this possibility is out of the scope of this paper. Synchronisation monitoring circuit generates a local framing signal and compares it to the received signal at the presumed correct position. The maximum OOF time in case of a synchronization slip is $T_{\text{maxITU-T}}=625 \mu s = 5$ frames, if tested within the random data region [10]. This region is supposed to be the worst case for accidental FAW simulation at incorrect position. However, this assumption is confirmed just for a carefully chosen bifix-free framing signal, checked as a single FAW [16, 17, 18]. The comparison can be made all at once, as in classical framing systems, or FAW by FAW within the same frame, if the framing signal is split into $N_s$ consecutive FAWs of length $L$ bytes (or $L_e=L\cdot8$ bits).

A detailed analysis dealing with this problem if $M=1$ (STM-1 frame) is performed in [12, 13], where both the expected value and Probability Density Function (p.d.f.) of the mean monitoring time were derived. Probability Density Function (p.d.f.) of the number of frames needed to achieve OOF event can be expressed in terms of a complex recursive relation, as shown in [15].

Figure 1 shows the mean holding time for STM-M frame ($M=4$) when no error per FAW allowed ($e=0$), when a single error per FAW allowed ($e=1$) and when two errors per FAW allowed ($e=2$). The effects of frame-alignment signal splitting are easily seen – the shorter the FAW, the longer the holding time, the less false alarms and therefore less subsequent searches during which data would be lost. However, long framing signal has more possibilities for multiple checks within a single frame, so its lengths should, obviously, chosen “per bona cause”.

Within the overlap region a local framing signal (locally generated framing signal) partially overlaps its received counterpart (slip, offset $O$ of the framing signal is within the region where local framing signal partially overlaps its received counterpart), as shown in Figure 2. If framing signal is split into $N_s$ consecutive FAWs of length $L$ bytes (or $L_e=L\cdot8$ bits), different events can occur. Some of the local FAWs can be compared to the received random data bits. The comparison would succeed with the well-known simulation probability (index $i$ denotes FAW’s order within the synchronization signal, i.e. order within a frame) [19].
allowed (consequence of the repetitive structure of the synchronization and when two errors per FAW allowed (1)).

Where $e$ denotes the allowed number of errors per byte. If a local FAW is compared to error-free received but slipped synchronization signal (the second and the third test in Figure 2), Hamming distance $D_j$ between them is:

$$D_j = \sum_{i=0}^{L} a_{L-i} \oplus a_{L-i+j}$$

(2)

Where $a_i \in \{0, 1\}$, $s = 1, ..., S$, $S_e$ denotes the particular $s^{th}$ bit of the synchronization signal and $\oplus$ is ex-or function. This distance might change due to channel bit-errors (with probability $P_e$). So, if $k$ errors are assumed to affect the $L_e-D_j$ received synchronization bits that would otherwise have matched, and $D_j+k-j$ errors has occurred at the $D_j$ synchronization bits already in disagreement, new distance would be exactly $j$. If $j$ is less or equal to the allowed value $L_e$, FAW is simulated with probability:

$$q(O) = \sum_{j=0}^{L_e} \sum_{k=0}^{m} \binom{L_e-D_j}{D_j+k-j} (1-P_e)^k P_e^{L_e-k}$$

(3)

At last, if local FAW overlaps $R_i$ received data bits and $L_e-R_i$ received framing bits (the first test in Figure 4), the simulation probability is [16,17]:

$$q(O) = \sum_{j=0}^{L_e} \left\{ \frac{1}{2^k} \sum_{l=m(L_e-j)}^{l=0} \binom{L_e-R_i-D_j}{D_j+k-l} Q_e^{D_j+k-l} \right\}$$

(4)

Small mean OOF time, $T_{OOF}=E\{O_{OOF}\}$, guarantees faster alarm announcement and reduced amount of data lost. As the simulation probabilities $q(O)$ change from test to test, classical tools for $T_{OOF}$ evaluation [2,4,12,13] cannot be used within the overlap region and an alternative method is derived [19] (matrix method [9] can be used instead). The derived method applied to STM-M frames yielded amazing, but not unexpected $T_{OOF}$ values. An example is shown in Figure 3 - $L=1$ for STM-1 frame when no error per FAW allowed ($e=0$), when a single error per FAW allowed ($e=1$) and when two errors per FAW allowed ($e=2$).

Large peaks, not infinite due to channel errors only, are a consequence of the repetitive structure of the synchronization signals – e.g. slip of one byte ($O=\pm 8$ bits) causes that, 4 local FAWs out of 6 ones perfectly match the received signal disabling the announcement of synchronization loss.

### III ASYMMETRICAL RAKE-LIKE FAWS

Yet, the idea of framing signals split into short FAWs (multiple FAWs) should not be abandoned, as it prevents frequent false alarms that cause unnecessary data loss during the following resynchronization. The right way to find a better frame alignment signal is shown in [19]. It would be ideal if this optimising method for sequences intended for splitting into shorter FAWs could be implemented. Unfortunately, the SDH recommendations [10] cannot be changed retroactively. As SDH synchronization signals cannot be optimised retroactively, a possible solution must cope the fact that at the most critical offset position, $O=\pm 8$ bits, local signal differs from the received one at 6 bit positions only; 8 bits might be different with probability 0.5; and as much as $M \times 48-14$ (!) bits are exactly the same!}

![Figure 2 STM-1 Synchronization signals within the overlap region (L=2, N_S = 3 consecutive FAWs)](image)

![Figure 3 OOF time within overlap region for STM-1](image)
the proposal of asymmetric rake-like frame-alignment: to each FAW, a portion of a byte (or of consecutive bytes, if \( N_S > 8 \) for \( M=1 \)) is assigned as evenly as possible; the bits that are, or might be, different at the critical \( O=\pm 8 \) position are also assigned uniformly to each FAW. This process should be done in such a way to minimize the total sum of simulation probabilities within the overlap region [16, 17]:

\[
P_{TS} = \sum_{O=\pm 8} \sum_{i=1}^{N_S} q_i(O)
\]  

(5)

Unfortunately, there is no other way to perform it but the computer search.

![Figure 4](image)

**Figure 4** Three rake-like FAWs of STM-1 frame

As an example, rake-like structure of \( L=2 \) FAWs with two errors per byte allowed (\( e=2 \)) is shown in Figure 4 (this values of \( L \) and \( e \) are proposed for STM-1, [12, 13]). Each of \( N_S=3 \) FAWs consists of 2 or 3 bits from each of the bytes. To each one 2 of 6 certainly different bits, and 2 or 3 ones out of 8 randomly different bits at troublesome \( O=\pm 8 \) positions are assigned. For the remaining 34 bits (identical at \( O=\pm 8! \)), a search is performed to ensure that they minimize the total sum of simulation probabilities within the overlap region [16, 17]. Finally, each FAW gets 2 bits out of 6 certainly different ones at troublesome \( O=\pm 8 \) positions, and 2 or 3 bits from other framing bytes.

The outcome is shown in Figure 5. This figure shows \( T_{OOF} \) peaks at \( O=\pm 8 \), still existing due to numerous identical bits, are reduced below the ITU-T maximum (it is impossible to lower them below this value if \( L=1 \)). But more significant \( T_{OOF} \) improvements are achieved for higher-order frames [15]. An improvement over the classical check of the optimised sequence is shown in Figure 6, for STM-M frame (\( M=1 \)) and \( L=2 \) FAWs when a single error per FAW is allowed (\( e=1 \)).

The rake-like solution improves holding time allowing shorter FAW lengths. Out of overlap region, rake-like FAWs have the same properties as consecutive-bytes FAWs of same length. In [20] is demonstrated that using distributed sequences (synchronization sequence is not continuous, but is interleaved with the data symbols) reduces the probability of false synchronization over traditional methods (frame synchronization is traditionally achieved by inserting a given sequence periodically into the data stream) in the presence of additive white Gaussian noise. Thus, frame synchronization performance using asymmetrical rake-like framing structure (a kind of distributed framing sequence) is significantly better.

**IV AMOUNT OF LOST FRAMES**

It is well known that two type errors events might occur while monitoring the frame alignment: random errors might cause the false loss of alignment alarm, requiring unnecessary re-alignment (resynchronization); and various disturbances might cause the real synchronization loss. The portion of frames that are lost, in respect to the total number of frames, can obtain using the following approximate equation [21]:

\[
\text{LOSS} \% = \frac{\pi_{OOF} + \pi_{RES}}{\pi_{OOF} + \pi_{RES} + \pi_{SYNC}} \cdot 100 = \frac{p_2 \cdot \tau_{OOF} + (p_1 + p_2) \cdot \tau_{RES} + \tau_{F}}{p_2 \cdot \tau_{OOF} + (p_1 + p_2) \cdot \tau_{RES} + \tau_F} \cdot 100
\]  

, where \( \tau_{OOF} \) is the value of Out-of-Frame confirmation time; \( \tau_{RES} \) is the value of maximum resynchronisation time (it is already known [21] that, for long bifix-free sequences, the time \( \tau_{RES} \) equals to the number of verifications); false loss of alignment, requiring unnecessary re-alignment, occurs with probability \( p_1 = 1/E[HOLD] \) where \( E[HOLD] \) is the synchronization holding time; \( p_2 \) is the probability of the disturbance that may cause the real synchronization loss; \( \tau_F \) is the length of the byte-oriented STM-M frame. Exact equation (shown in [22]) is very complicate and differs slightly from the approximate ones.

![Figure 5](image)

**Figure 5** Mean OOF time at \( O=\pm 8 \) for STM-1 frame

![Figure 6](image)

**Figure 6** Mean OOF time for optimal type of FAW (STM-1)
Frame Alignment Word (FAW) allowed. Number of verifications equals three ($t_R = 3-t_p$) and the probabilities that a disturbance would cause the real synchronization loss $p_2$ are 0.0000001 and 0.1. The channel bit error ratio (Poisson type) $P_e$ is 0.001.

> Figure 7 Approximate percentages of lost frames for STM-1

It is obvious that slip of one byte ($O = \pm 8$) causes the greatest amount of lost data for ITU-T sequence, not 100% due to channel errors only.

V CONCLUSION

Framing signal split into independent subsets (shorter frame-alignment words) considerably improves synchronization holding time (prolongs the holding time between false alarms), but might cause infinite out-of-frame detection time if non-optimal sequence is applied. The signal splitting cannot be applied straightforwardly to STM-M synchronization signal, as its repetitive structure prevents the detection of synchronization loss at certain slip positions. A method for avoiding this problem is proposed and applied to the existing Synchronous Digital Hierarchy (SDH) framing sequences. A rake-like method of synchronization signal splitting enables the usage of short FAWs that guarantees longer holding time. The analysis of rake-like checked and proposed framing sequence structure is based both upon the statistical parameters (normalized mean OOF time, $T_{OOF}/T_{maxITU}$) and upon the average amount of lost frames (data).

REFERENCES

[22] D. Bajić, V. Šenk, I. Stanojević: “Synchronization procedures for cyclic code”.