THEOREM 12.5 *Let n_c be the number of occurrences of a symbol c in the input string s, whose total length is $n = |s|$. We denote by $\rho_{MTF}(s)$ the average number of bits per symbol used by the compressor that squeezes the string s^{MTF} using the γ-code over its integers. It is $\rho_{MTF}(s) \leq 2H + 1$, namely that compressor can be no more than twice worse than the entropy of the source, and thus it cannot be more than twice worse than the Huffman compressor.*

Proof Let $p_1, \ldots, p_{n_c}$ be the positions in s where symbol c occurs. Clearly, between any two consecutive occurrences of c in s, say p_i and p_{i-1}, there may exist no more than $p_i - p_{i-1}$ distinct symbols (including c itself). So the index encoded by the `MTF`-compressor for the occurrence of c at position p_i is at most $p_i - p_{i-1}$. In fact, when processing position p_{i-1} the symbol c is moved to the front of the list, then it can move (at most) one position back per symbol processed subsequently, until we reach the occurrence of c at position p_i. This means that the integer emitted for the occurrence of c at position p_i is $\leq p_i - p_{i-1}$ (number of symbols processed). This integer is then encoded via γ-code, thus using no more than $\gamma(p_i - p_{i-1}) \leq 2(\log_2(p_i - p_{i-1})) + 1$ bits. As far as the first occurrence of c is concerned, we can assume that $p_0 = 0$, and thus encode it with at most $\gamma(p_1) \leq 2(\log_2 p_1) + 1$ bits. Overall the cost in bits for storing the occurrences of c in string s is

$$
\begin{aligned}
&\leq \gamma(p_1) + \sum_{i=2}^{n_c} \gamma(p_i - p_{i-1}) \\
&\leq 2\log_2(p_1) + 1 + \sum_{i=2}^{n_c} (2\log_2(p_i - p_{i-1}) + 1) \\
&\leq \sum_{i=1}^{n_c} (2\log_2(p_i - p_{i-1}) + 1).
\end{aligned}
\tag{12.1}
$$

By applying Jensen's inequality we can move the logarithm function outside the summation, so that a telescopic sum comes out:

$$
\begin{aligned}
&\leq n_c \left(2\ \log_2\left(\frac{1}{n_c}\left(\sum_{i=1}^{n_c}(p_i - p_{i-1})\right)\right) + 1\right) \\
&= n_c \left(2\ \log_2\left(\frac{p_{n_c}}{n_c}\right) + 1\right) \\
&\leq n_c \left(2\ \log_2\left(\frac{n}{n_c}\right) + 1\right)
\end{aligned}
\tag{12.2}
$$

where the last inequality comes from the simple observation that $p_{n_c} \leq n$. If now we sum for every symbol $c \in \Sigma$ and divide for the string length n, because the Theorem is stated as number of bits per symbol in s, we get:

$$
\rho_{MTF}(s) \leq 2\left(\sum_{c\in\Sigma} \frac{n_c}{n} \log_2\left(\frac{n}{n_c}\right)\right) + 1 \ \leq\ 2H + 1 \tag{12.3}
$$

The thesis follows because H lower bounds the average codeword length of Huffman's code. ■

There do exist cases for which the `MTF`-based compressor performs much better than Huffman's compressor.