THE UTILIZATION OF THE HUFFMAN CODE TO IMPROVE MANAGERIAL PERFORMANCE FOR ECONOMIC ORGANIZATIONS USING TOP-LEVEL TECHNOLOGY

Marian Sorin IONESCU
Politehnica University of Bucharest
313 Splaiul Independenţei, Bucharest, Romania
marian.ionescu@man.ase.ro

Abstract. *The new challenges arising from the confrontation of the economic organizations with the broadest and complex process generated by modern human civilization, the globalization, the meeting of the needs and desires of the more and more educated end customers, with the skills of use and profound understanding of the new technologies, the responses for vectors of competing economic organizations, require innovative approaches at the level of top organizational decisional-management. In their complexity, the present economic phenomena bring to the human decident an increasing amount of data with particular volumes and structures characteristic of the types of processes represented, so the process of decision making becomes a particularly difficult, interdisciplinary task, the human responsible is required a high standard of professional training. Any economic organization that proposes in its developing strategies to maximize generated added value and profitability, should develop at the level of* $I.T.$ *compartment policies and methods of prevailing, managing and storing information as effectively and optimally as possible. Compressing this data without losses and preserving its quality is one of the arguments that make the difference in an economy of the successful decisional-management competition. We present in this paper, possibilities of saving the memory spaces allocated to the information used in the operational processes, possibly later extrapolated strategically, with a percentage rate of* 20%, *up to* 90%, *depending on the dependence on the type and characteristics of the compressed data in the studied processes. For operational, concrete cases, solutions such as variable length code tables are proposed, thus source code symbols are coded, whose interpretation is essential in developing optimized managerial decision generating organizational performance. In this paradigm of approach, the data used is perceived as character sequences, with the "$HUFFMAN\ Code$" algorithm developed, the approach selected in the study is a succession of tabloid structures, is developed step by step, the optimal way of representation to each character as a binary string. For a top organizational decisional-manager this is a competitive alternative in developing an improved, efficient and successful decision-making process. The selected example involves the design of a binary code, in which each character is represented by a string, also binary, unique. The selected operation, a business model used in an economy of performance, substantiates the structures within the "$HUFFMAN\ Code$".*

Keywords: Memory Spaces; Compressing Data; Business Model; Binary Code; $HUFFMAN\ Code$.

Introduction

Developed on theoretically well-founded principles, they analyze the advantages but also the vulnerabilities induced for this type of alternative approach, of coding, and finally data compression, using fewer bits than symbols which have a much lower frequency. The efficiency level of this mathematical method, the "*HUFFMAN Code*" method, offers alternatives for efficient implementations, when developing and presenting a linear code, identifiable in a well-defined temporary process, implicitly a sorting process being imperatively developed at input of data. Among the methods that separately encode the symbols used in data processing, the algorithm has an optimality character, but this type of method is not always the optimal choice in the highly diversified portfolio of current data compression methods that are easy to subsequent transpose operationally, is not always the optimal choice.

The present scientific paper, we consider it to be a concrete, pragmatic point of view, with the subsequent possibility of extrapolation, in the case of the practice of the top deciding managers for understanding, modeling, processing, implementing the informational processes, involving a significant number of data, with connotations and valences operational-strategic specific to them. Efficient management, accessibility to the data sequences used, the speed at which such a phenomenon is approached become decisive factors in the hierarchy of business models in a modern business macro-environment, with the totality of its characteristics.

In an economy of competition, the promotion of advanced technologies and avant-garde innovative managerial concepts are vectors for the extensive and intensive consolidation and development of the economic organization in the economic macro - environment. The selection, analysis and understanding of data are fundamental approaches in increasing the performance of the business model selected to be transposed into operationality by top decisional management. The value of working with extrinsical informational flows but also intrinsical organizational flows is the factor that delimitates the managerial performance of failure, the key element that provides the competitive advantage, fundamentally the most solid of future success. The lack of development of the informational component within the business model deprives the economic entity of a powerful and aggressive vector of strategic development in well-defined time horizons.

The phenomenology of sampling, optimal analysis and storage of data provides organizational decision makers with the ability to understand the internal operationalization of the business in a multitude of well-defined states over time. Each informational data sequence must be stored, the totality of transactions recorded and ordered according to predefined criteria, thus all the aspects of the data complex are presented in a complete and clear picture to the decision makers and responsible organizational factors, allowing the improving to optimization of the underlying strategies and operationalities. The information must be sorted according to criteria clearly defined by the top decisional-management and registered, the memory space at the economic organization level has a finite capacity, given by the allocated hardware infrastructure, compaction-compression of the data thus becoming an obvious option for the elaboration of a process of generating superior added value, performance of the selected business model, implemented and subsequently operationalized.

The systemic data analysis process is the strategic vector that offers the possibility to identify the most favorable decision for the positive evolution of the economic organization. Data analysis from a conceptual point of view is generally an area of informational science that has become extremely useful in the quasi-totality of the market segments that make up the modern, globalized and hyper-dynamic business macro-environment. The way of sampling, analysis, sorting, understanding and storage are the factors that condition the present managerial performance but also strategic, on well-defined time horizons, this systemic decision-maker effort provides an exhaustive vision of the entire business. At the same time, it is possible to identify the strengths of the business models, innovative trends and model alternatives, accurately identify the deficiencies and possible directions of model development, changes in strategies and operationality to generate an improving trend towards optimization. In addition to improving and quantifying economic performance by mathematical indicators, data analysis, their compaction and compression, it provides software and hardware facilitation to anticipate and develop predictions regarding the performance of the economic organization in terms of performance, with constraints and rigor of the business macro-environment, as well as internal structural limitations.

The clear and profound understanding of the business model, which elaborates the organizational strategy and the operationalization of the economic entity's activities, is the premise of developing the most correct managerial decisions. The organization's resources at the conceptual software and hardware logistics level require systemic compression-compaction of data processes as viable alternatives to performance identification.

HUFFMAN code

The concept and paradigms that are subsequently positioned make the "$HUFFMAN\ Code$" an algorithm, a way of compressing data extremely efficient. Approaching the data compression issue for business models is equivalent to implementing a special type of optimized prefix codes, developing the compression process in its entirety does not induce any kind of informational losses. The $HUFFMAN\ coding$ is not conceptually encrypted, this feature is engineered with structure customized and dedicated to operationalized business models, by modifying the $Mainly\ Mutated\ HUFFMAN\ TABLE, MHT$, and $Chaotic\ HUFFMAN\ TABLE,\ CHT$ chaotic mass (Atallah, 1999; Cormen, Leiserson & Rivest, 2009).

In systemic data compaction processes used by the top organizational decision management for the elaboration of managerial decision, these are operable with or without losses, including subsequent storage processes admitting successive improvements, at this level some distortions are identifiable. Only certain information has value and influence on the implemented business models, therefore a clear and transparent selection process is absolutely necessary, a relevant amount of data encoded under a particular bits structure is declared to be useful to the intended purpose. In the case of data-loss compaction phenomena, these must be positioned within a tolerated error range such that the re-built replica for the original data sequence does not become a corrupted or false information provider. To select certain types of data compression algorithms, adaptable and useful in understanding, adapting, implementing organizational business models, these are quantified as an initial approach. Concepts such as the operating speed on a certain type of computer, the mathematical complexity of the functional entities that make up the algorithm, the calculation procedures and structures, the amount of data compression and the time allocated for such processes. The similitude degree after the reconstruction of the information from the original, are criteria for choosing data compression algorithms for economic business organizations. Subsequent distortion of inward data is evaluated using loyalty and informational quality coefficients, the degree of similarity between the reconstructed version and the original sequence of their succession. Systemic data compression and storage phenomena are directly reflected in the analysis and control of organizational strategic operational costs. Thus they identify their value and usefulness of effective elements at the service of the top decisional-management.

The $HUFFMAN\ code$ has a specific "greedy algorithms" structure, the process itself takes place by recording the emergence frequencies of each symbol in a file, and then an optimization process is developed to represent the symbols as a binary string. The fundamental idea of managing information for an economic organization is that of using shorter binary codes. The problem of binary characters code representation ("code", for ease) induces, presents each character as a unique binary string, called "codeword". In the case of a fixed length code, the number of bits which need to be assigned to represent a number of characters is of a very large order. Thus the necessity to enter the variable length code occurs, from our point of view, the most adapted requirements coming from the business environment. The typicality of this new approach is the frequency of emergence of short codeword characters and the rather rare emergence of long codeword characters. Concrete calculations show a 25% saving of memory space allocated within the file used. We only analyze the codes in which code words and a prefix of a certain word code are not identifiable. This concept is referred to in the specialty literature as "prefix-free codes" or in a standard way "prefix-codes".

For economic information used in managerial decision processes concatenation is a process with an improved degree of simplification for any binary code, concatenating codewords defining each character of the storage file. The usage of code prefixes simplifies the reverse process of decoding the information. We find that no wordcode, in operational business practice, is a prefix of another wordcode with which a file begins, the decoding process is unambiguous, not having more interpretations. We identify the initial codeword translated into the original characters, the process is repetitive for the rest of the decoded file. The decoding process requires a conventional representation for a prefix code, the initial codeword can be used easily at any time, the selected binary tree has the characteristics required for such representation. The binary representation for a character codeword is the way, the pattern which needs to be traced from the root of the tree to the character, the truth value "0" involves moving to the lower left branch, the truth value "1" involves moving to the lower right branch. An optimal code for an informational file used by the economic organization admits representation as a complete binary tree, in which each non-final node has two options ("two child branches").

We introduce the formalism in which we consider C as the alphabet for which we use the characters used within the binary tree. All the characters with a certain frequency are positive. It results that the tree for an optimal prefix code has exactly $|C|$ leaf-nodes, one for each letter of the alphabet, exactly $|C| - 1$, internal nodes, after which the search process goes up to the end nodes.

In the practice of the $HUFFMAN\ code$, management decision at the level of the $I.T.$, department, in order to solve the occurred challenge, the developed model starts with the identification of a T tree corresponding to a prefix code, the development of algorithm and calculation procedures it follows, to quantify the number of bits required for coding the file in which the information underlying the process is stored and structured.

For each character c in the C alphabet, the $c.freq$ attribute shows the c frequency within the file, let $d_T(c)$ the depth of the "leaf" c within the tree, also to be mentioned that $d_T(c)$ represents the length of code-word for the character selected from the alphabet c. (Cormen, Leiserson, Rivest & Stein, 2009, p.431)

We identify the number of bits needed to encode the file,

$$B(T) = \sum_{c \in C} c.freq \cdot d_T(c), \quad (1)$$

the formula is, from an economic point of view, the value of the binary T tree cost browsing.

Operational implementation, approach through HUFFMAN Code

The algorithm of the $HUFFMAN\ Code$ is a particular case, but from our point of view, flexible and adaptable to business issues, the classic "greedy" type algorithms. Within this algorithm we identify the construction of an optimal code-prefix, which provides exactly the solution and the finality for the complex problem facing the top organizational decisional-management. Considering C selected set with n characters, each character $c \in C$ is a target with an attribute $c.freq$ attached to it, given its frequency of occurrence, the algorithm develops a T - graph tree which corresponds to the optimal coding for a scroll-up process. The process is initiated with the help of a set having the cardinal $|C|$ by the nodes ("leaves") of the graph, going through a sequence $|C| - 1$ of a merging operation leading to the creation of the final tree.
We identify in the implementation conditions of the algorithm the existence of a minimum priority queue,Q ordered by the "$freq$" attribute, which identifies the objects with the lowest frequency in the merging process.

The result of the development of a merging operational process for business models implemented for economic organizations is the appearance of a new systemic object with an individualized frequency equal to the sum of the frequencies of each of the two objects that participated in the merger phenomenon. An executable pseudocode structure for processes and their transposition into operationality previously presented has the structure below.

HUFFMAN calculation algorithm (C)
$1 \rightarrow \mathrm{n} = |\mathrm{C}|$
$2 \rightarrow \mathrm{Q} = \mathrm{C}$
$3 \rightarrow \mathrm{for\ i} = 1 \mathrm{\ to\ } (\mathrm{n} - 1)$
$4 \rightarrow$ Allocation of a new nodes
$5 \rightarrow \mathrm{z.left} = \mathrm{x} = \mathrm{EXTRACT} - \mathrm{MIN(Q)}$
$6 \rightarrow \mathrm{z.right} = \mathrm{y} = \mathrm{EXTRACT} - \mathrm{MIN(Q)}$
$7 \rightarrow \mathrm{z.freq} = \mathrm{x.freq} + \mathrm{y.freq}$
$8 \rightarrow \mathrm{INSERT\ (Q, z)}$
$9 \rightarrow \mathrm{return\ EXTRACT} - \mathrm{MINQ}$
Return the root of the tree
(Cormen, Leiserson, Rivest & Stein, 2009, p.431)

Beneficial for the implementation and operational transposition of compression (compaction) of the data used by organizational business models is the development of possible changes to the $HUFFMAN\ coding\ tree$, depending on the main target task. Modeling the Huffman coding tree-specific processes demonstrates that encryption and subsequent data compaction operations are performed only together. The data encoding approach used by the business models specific to economic organizations employs moving processes and algorithms adapted to $HUFFMAN\ coding$, these are transfers between labeled branches of the tree. Changing the $HUFFMAN\ encoding\ tree$ structure provides a high degree of compression accuracy of the data used in the decision making process, this has a dual structure, is used in the encryption process (coding), but also in the key switching mechanisms. The process induces a series of multiple mutations, the data streams are shared in segments considered to be representative, "key", depending on the information encoding sequences in the form of representative bit structures.

We note with $T(k)$ the mutations performed in the tree graph level. Each data segment (information) has a number of bits dependent on the number of nodes in the Huffman tree, the segment-level moving process (transfer) is applied to the tree in the case of logical value $1 - "truth"$, $0 - "false"$ occurance, the node is not modified.

The analysis of the proposed Huffman algorithm

The degree of accuracy of the Huffman algorithm, its implementation and operational transposition mode are essential for continuous improvement, up to optimizing business models for economic organizations.
The problem of determining the optimal prefix code has a greedy-type exposure for the data which need to be compacted in the optimal structure, in subsequent mathematical formalisms approached, this approach is detailed.

Observation 1: We consider C a selected alphabet, each $c \in C$ character has an expressible frequency by $c.freq$, considering $x, y \in C$ two characters with the lowest frequency, then there is an optimal prefix code in which the previously defined words, "Codewords" for x, y have the same length, the last bit is the only different element.
Comment: We present a using frequency table of the elements (letters) of Latin alphabet, for data sequences transmitted in English to the studied economic entity (organization), functioning on a particular business model that undergoes a compaction process.

$\boldsymbol{Character}$	$\boldsymbol{Frequency}$ (%)	$\boldsymbol{Character}$	$\boldsymbol{Frequency}$ (%)
$\boldsymbol{A}$	$\mathbf{8.167\%}$	$\boldsymbol{N}$	$\mathbf{6.75\%}$
$\boldsymbol{B}$	$\mathbf{1.492\%}$	$\boldsymbol{O}$	$\mathbf{7.51\%}$
$\boldsymbol{C}$	$\mathbf{2.782\%}$	$\boldsymbol{P}$	$\mathbf{1.93\%}$
$\boldsymbol{D}$	$\mathbf{4.253\%}$	$\boldsymbol{Q}$	$\mathbf{0.10\%}$
$\boldsymbol{E}$	$\mathbf{12.70\%}$	$\boldsymbol{R}$	$\mathbf{5.99\%}$
$\boldsymbol{F}$	$\mathbf{2.23\%}$	$\boldsymbol{S}$	$\mathbf{6.33\%}$
$\boldsymbol{G}$	$\mathbf{2.02\%}$	$\boldsymbol{T}$	$\mathbf{9.06\%}$
$\boldsymbol{H}$	$\mathbf{6.09\%}$	$\boldsymbol{U}$	$\mathbf{2.76\%}$
$\boldsymbol{I}$	$\mathbf{6.97\%}$	$\boldsymbol{V}$	$\mathbf{0.98\%}$
$\boldsymbol{J}$	$\mathbf{0.15\%}$	$\boldsymbol{W}$	$\mathbf{2.36\%}$
$\boldsymbol{K}$	$\mathbf{0.77\%}$	$\boldsymbol{X}$	$\mathbf{0.15\%}$
$\boldsymbol{L}$	$\mathbf{4.03\%}$	$\boldsymbol{Y}$	$\mathbf{1.97\%}$
$\boldsymbol{M}$	$\mathbf{2.41\%}$	$\boldsymbol{Z}$	$\mathbf{0.07\%}$

Consider the T tree that represents an optimal code prefix and we modify it to become another optimal code prefix such that the x, y characters appear as adjacent Leaves positioned at maximum depth in the new tree graph, we study whether it is possible to develop a tree for which "codewords" of x and y have the same length, the difference between the two data informational structures is made only by the last bit.
We consider a, b two characters which are Leaves positioned on the same level, "sibling leaves" as depth in the T tree. Without loss of generals, suppose a. $a.freq \leq b.freq$ and $x.freq \leq y.freq$, until the two values $x.freq$ and $y.freq$ become the leaves positioned with the lowest frequency level, in order, $a.freq$ and $b.freq$ having two random frequencies (arbitrary), in order, result $x.freq \leq a.freq, y.freq \leq b.freq$. There is the possibility of occurrence of the point case, $x.freq = a.freq$ or $y.freq = b.freq$, if the equality $x.freq = b.freq \rightarrow a.freq = b.freq = x.freq = y.freq$, the observation previously presented becomes obviously true.

Scientific analysis continues for $x.freq <> b.freq \rightarrow x <> b$.
A process of changing the location of a and x in the T tree leads to the occurrence of the T' tree, when the position of b and y in the T'tree is changed, the T'' tree is generated in which x, y are "sibling leaves" (the same level of importance) at maximum depth, if $x = b$ but $y <> a$, then the T'' tree x, y does not have "sibling" leaves at the maximum depth in the graph tree, the assumption that $x <> b$, the situation is not likely to occur in the operationalities of the selected business model.

Starting from mathematical relational (1), the cost difference between T and T' trees is:
$B(T) - B(T') = \sum_{c \in C} c.freq \cdot d_T(c) - \sum_{c \in C} c.freq \cdot d_{T'}(c) = x.freq \cdot d_T(x) + a.freq \cdot d_T(a) - x.freq \cdot d_{T'}(x) - a.freq \cdot d_{T'}(a) = x.freq \cdot d_T(x) + a.freq \cdot d_T(a) - a.freq \cdot d_T(x) = (a.freq - x.freq)(d_T(a) - d_T(x) \geq 0$, because both relationships, $a.freq - x.freq$ si $d_T(a) - d_T(x)$ are non-negative, $a.freq - x.freq$ is non negative because x is the "leaf" of the minimal frequency tree, and $d_T(a) - d_T(x)$ is non negative because a is the maximum depth "leaf" within the T tree.
Analogously it is possible to change $y \rightarrow b$, which does not induce any change in cost, because $B(T') - B(T'')$ is non negative, thus $B(T'') \leq B(T)$.
As long as T is optimal, results $B(T) \leq B(T'') \rightarrow B(T'') = B(T)$, so$T''$ is an optimal tree in which x, y, appear as "sibling leaves" at maximum depth in the studied tree, which confirms the correctness of the mathematical observation made and the correctness of the developed logical reasoning.
We notice that in compaction of data-information accessed for business models, in order to save and optimally use the used memory space.

The problem of optimal code prefixes is a mathematical property of the optimal graph arborescent substructures. (Cormen, Leiserson, Rivest & Stein, 2009, p.434)

Observation 2: Considering C an alphabet with the $c.freq$ frequency defined for each character $c \in C$, we choose x, y two characters of C with the minimum frequency.
We consider C' to alter the alphabet C in which the characters x, y are displaced and a new character z is added such that the equality $C' = C - \{x, y\} \cup \{z\}$ tooks place, we define $freq$ for C' as for C we identify the mathematical relation $z.freq = x.freq + y.freq$, we consider T' any tree representing an optimal prefix code for the C' alphabet, then the T tree is obtained from T' replacing the "Leaf" node for z with an internal node

having the x, y "children" within the tree-graph structure, this is the representation of an optimal prefix code for the C alphabet.

Comment. It is to be discussed how the cost $B(T)$of the T tree in terms of the cost $B(T')$ of the T' tree, considering the component costs of equation (1), for each character $c \in C - \{x, y\}$, , result the relationality $d_T(c) = d_{T'}(c)$, therefore $c.freq \cdot d_T(c) = c.freq \cdot d_{T'}(c)$ until $d_T(x) = d_T(y) = d_{T'(z)} + 1$, results $x.freq \cdot d_T(x) + y.freq \cdot d_T(y) = (x.freq + y.freq)(d_{T'}(z) + 1) = z.freq \cdot d_{T'}(z) + (x.freq + y.freq)$, from where results:

$$B(T) = B(T') + x.freq + y.freq < -> B(T') = B(T) - x.freq - y.freq$$

Suppose that T is not an optimal prefix code for the alphabet C, then there exists an optimal T'' tree such that $B(T'') < B(T)$, with no generality restriction of the analysis and reasoning developed, T'' have x, y as "sibling leaves". We consider T''' the T'' tree with common parts x, y replaced by the z graph leaf, with $z.freq = x.freq + y.freq$ frequency, results that:

$$B(T''') = B(T'') - x.freq - y.freq < B(T) - x.freq - y.freq = B(T')$$

What reveals a contradiction by subtracting the hypothesis that T' represents a code with the prefix optimal for C'. We conclude that T represents an optimal code prefix for any type of C initially selected alphabet.

Observation 3: The operational callout to Huffman's procedural structure type, generates as an immediate consequence an optimal prefix code (Cormen, Leiserson, Rivest & Stein, 2009).

The operational computing structure for the implementation of the mutational process

The business models for which data is compacted (compressing) using the Huffman code method, are characterized by a variable operating speed of work with the accessed informational structures. The mutation, modification, of the arborescent graph technology using the S.H.T. ("Swapped Huffman Tree Coding"), therefore of transformation-modification of the Huffman codes, for operationalized business models with relative low speeds, is implementable at the level of different algorithms and procedure software, subsequent digitally transposed. The computer structure below is an operational-functional presentation of the mutational process of compaction-compression of data used in the service of business models - the development of top organizational management in the elaboration of the decision

.

```
OPERATIONAL − FUNCTIONAL STRUCTURE
INPUT: Original − Code, Encryption − Key
char*function mutation
{
check = maxlenght(OCs);
while(maxlenght(maked) == check)
{
making
(" ", "0", "1", "00", "01", "11", "000", "001", … )
```

$$
\begin{gathered}
\}\\
\text{for}(i = 0; i < \text{count}(\text{maked});\ i^{+++})\\
\{\\
\text{if}\big((\text{Oc}! = \text{maked})\text{or}(\text{OC\_and maked\_is\_completely\_same})\big)\\
\{\\
\text{that\_maked\_delete;}\\
\}\\
\}\\
\text{divided Encryption} - \text{Key according to count (maked);}\\
\text{for}(j = 0; j < \text{count}(\text{maked}); j^{++})\\
\{\\
\text{if}(\text{OC\_left} == \text{maked})\\
\{\\
\text{OC\_left}'^{s-}\text{next} = \text{toggle;}\\
\}\\
\}\\
\text{return OC;}\\
\text{goto NextCode;}\\
\}
\end{gathered}
$$

For a correct understanding of the functioning of the pseudo-code entity, we proceed to a brief description in the structural steps of the previous computer entity. The pursued characteristics are the structure that has been developed, but also its transposition into operationality with the totality of the consequences implicitly derived from it.

$$
\begin{gathered}
\text{Stage 1 } \rightarrow \text{ The coding key is defined in M segments}\\
\text{Stage 2 } \rightarrow \text{ We are in the process of developing a confirmation} - \text{verification list}\\
\text{"S" of the } T^{(n)} \text{ tree}, n = \{0, \dots, M-1\}\\
\text{The maximum length of each symbol in the checklist is defined}\\
l = k - 1 \text{ bits, k the maximal number of bits from } T^{(n)}\\
\text{Confirmation} - \text{verification list is defined starting from:}\\
\{\text{" "}, 0, 1, 00, 01, 10, 11, 000, 001, \dots\}\\
\text{Confirmation} - \text{verification list created, improving it}\\
\text{Comparison process with the corresponding tree symbol list}\\
\text{Replace Huffman codes with identical ones}\\
\text{Or that does not exist in the list of symbols}\\
\text{Stage 3 } \rightarrow \text{The confirmation} - \text{verification list from "Stage 2" is accessed}\\
\text{Corresponding to the key encryption segment of "Stage 1"}\\
\text{Value of Os in the cipher key segment ignores the mutation, for 1s mutated effected}\\
\text{Moving verified lists } \rightarrow \text{ every symbol belonging to the verified lists}\\
\text{M.S.B.} = \text{ list of the most significant bits, Most Significant Bit}\\
\text{Comparison process with the most significant bit } \in \text{M.S.B.}\\
\text{If the symbol is the same, we switch to the next bit in M.S.B.}, 0 \rightarrow 1, 1 \rightarrow 0\\
\text{Stage 4 } \rightarrow \text{ Return to Stage 2 and repeat the process}\\
\text{Until the last key segment is reached}\\
\text{After generating } T^{(1)} \text{we go from step c to stage 2}\\
\text{We use } T^{(1)} \text{ instead of } T^{(0)} \text{ as a base tree to generate } T^{(2)}\\
\text{Repeat process until the end segment of the cipher key is reached}
\end{gathered}
$$

Redundancy of Huffman codes, main results

The enormous amount of information with which the selected business model is confronted within the macroeconomic environment, makes the sorting processes and its subsequent compression to store two fundamental choices for achieving the highest added value and generating organizational performance. The Huffman codes are subjected to a redundancy of a certain intensity; their subsequent operationalization is conditioned by solving this type of challenge. Scientific studies show that a characteristic of Huffman codes is the existence of a relatively "tight" upper limit, the lower limit of this process of data compression resulting from derivation.

We consider a non-memory source S to belong to a finite N dimensional alphabet, of the structure $S = \{s_1, \dots, s_N\}$ and a multiple set of probabilities, which may contain repeating members due to different source symbols but with equal probabilities, $P = \{u_1, \dots, u_N\}$, $u_i \Pr(S = s_i)$.
A source code with the variable length D for the S source is defined by the one-to-one coding function, $f_C: S \rightarrow \{0, 1, \dots, (D-1)\}$, thus we can calculate the average length of a code, C specific to the compression Huffman processes with

$$L(C) := \sum_{i=1}^{N} u_i l_C(u_i) \qquad (2)$$

One of the fundamental goals pursued in data compression processes for business models is that of developing minimal length codes, the average minimum length of a code without prefix cannot be less than the entropy source, regardless of the degree of code efficiency, the entropy in the base D is represented by the symbolism

$$H(s) \coloneqq -\sum_{i=1}^{N} u_i log_D(u_i) \qquad (3)$$

an important indicator of quantification of source code performance is redundant, redundant is the difference between the mean length and the entropy of the informational source (of data), expressible by

$$R(C) \coloneqq L(C) - H(S) \qquad (4)$$

The Huffman coding algorithm and data compression algorithm provides an optimal prefix code for a memory-free source of information, no code for the P distribution can not have a shorter length than the Huffman code. (Cormen, Leiserson, Rivest & Stein, 2009; Dasgupta, Papadimitriou & Vazirani, 2008; Lewis & Papadimitriou, 1998)

A D −Huffman code is represented by a $D - T$ tree whose leaves are the correspondence of the source symbols, the sides of D that go from each intermediate node of T are labeled with one or more D -sides of the previous presented alphabet, the word code is the "codeword" corresponding to the symbol is the (indefinite) label vector positioned along the path from root to leaf in the corresponding tree graph. Thus, we identify the situation where, for selected organizational business models, Huffman's compaction-compression data algorithm is a functional recursive bottom-up construction of T, where at each step of D there are nodes with the least probability combined in a a new node, thus adding a number of symbols from an inactive source having zero probability. The number of N source symbols is expressed in mathematical form $k(D-1)$for k integer, from now on they represent intermediate nodes in the tree.

In the majority of applications of the economic models, Huff man's operationalized codes are binary type, $D = 2$, Huffman's redundancy within the used business model analysis is always non-negative, but neither exceeds numerical value 1. The upper limit of $R(T)$ redundancy by addressing binary Huffman codes when $p_1 := max_i u_i$ and u_i correlates with the probability that the most commonly used symbol of the source is known. The problem of the superior margin of redundancy in probability terminations in mathematical formalism $p_N := min_i u_i$. The probability that the smallest operationalized symbol of the source is known, the upper margin, $R(T)$ in the case of the Huffman type codecs, is well defined when both probabilities for extremes p_1 and p_N are known. The upper limit of the redundancy margin is derived from a function having the probability conditioned by the two source symbols p_{N-1} and p_N.

In the following is presented a well-controlled upper limit of the redundancy addressed with the help of Huffman codes, for a source that contains a symbol with a certain given probability p.

Observation 4: We consider a Huffman code for a finite alphabet source, that includes a symbol with the probability p, otherwise being arbitrary.
Redundancy of this code is superior to

$$R_{max}(p) := \begin{cases} 2 - p - H(p), daca\ 0.5 \leq 1 < 1 \\ 1 + p - H(p), daca\ 0 \leq p < 0.5 \end{cases}$$

where $H(p) := -plogp - (1-p)\log(1-p)$ is a binary entropy function, the relation shows a strong interconnection, so are identifiable distributive sequences whose redundant Huffman converge to the value $R_{max}(p)$.
This observation shows the superior margin in the case of Huffman redundancy.

Observation 5: We consider the Huffman code for a finite alphabet source, which includes a symbol with the probability p, otherwise being arbitrary.
Redundancy of this code is inferior limited by

$$R_{min}(p) := mp - H(p) - (1-p)\log(1-2^{-m})$$

where $m > 0$, let the value $[-logp]$ or $[-logp]$ which minimizes expression, we identify the interconnection between the distribution source of the symbol having the probability p with the Huffman code redundancy equal to $R_{min}(p)$.

Observation 6: The redundancy of a Huffman code $D - set$ contains a letter with probability p has a "tightly" limitation, computable by

$$R_{min,D(p)} = mp - H_D(p) - (1-p)log_D(1-D^{-m})$$

Observation 7: We consider p the characteristic probability of any source symbol, then the corresponding redundant for the upper margin of the Huffman code is

$$R_{ub} := \begin{cases} 2 - p - H(p), daca\ 0.5 \leq p < 1 \\ 0.5, daca\ \pi_0 < p < 0.5 \\ 1 + p - H(p), daca\ p \leq \pi_0 \end{cases}$$

we note that $\pi_0 \cong 0.18$ is the smallest root of the equation $1 + p - H(p) = 0.5$

The calculation of the superior redundancy for a known probability source is one of the concerns that influence the phenomenology approach of compressing-compaction data with the help of Huffman code methodology.

Observation 8: Considering p_1 the symbol probability of the most known source, the Huffman redundant of this source is superior margined by the function

$$f(p_1) = \begin{cases} 2 - p_1 - H(p_1), daca\ 0.5 \leq p_1 < 1 \\ 3 - 5p_1 - H(2p_1), daca\ \pi_1 \leq p_1 < 0.5 \\ \gamma, daca\ p_1 \leq \pi_1 \end{cases}$$

where $\gamma = R_{max}\left(\frac{1}{3}\right) = 1 + \frac{1}{3} - H\left(\frac{1}{3}\right) \cong 0.415, \pi_1 \cong 0.491$ is the root of $3 - 5x - H(2x) = \gamma$

Observation 9: If p_1 is the probability of a letter in a source, if its value $p_1 \geq 0.4$, then the length corresponding to the codeword in the Huffman coding approach is one that satisfies the condition $l(p_1) = 1$. (Dasgupta, Papadimitriou & Vazirani, 2006)

Conclusions

Modern economic organizations, actors in the modern economic macro-environment, regardless of the strategic operational business model selected, are confronted with an enormous amount of data, these coming from the intrinsic but especially the extrinsic economic entities. Relevant conducted business activity information needs to be sorted, analyzed, interpreted and subsequent transposed into arguments for the taken decisions. The programs for encryption, compression and storage of data useful to top organizational management are conditions, facilities that make the most of generating added value, sustainable growth vectors in modern, globalized, and particularly dynamic economic competition. The Huffman coding and compression algorithms do not induce data loss at file level, thus the organizational economic history is not affected, the determining role in their operational transposition is the frequency of the occurrence of a symbol in the compression file.

Statistical coding is fundamental to the usage of the Huffman algorithm in the complex economic issues, therefore the probability of symbols has a direct influence on the length of its representation, the more probable the occurrence of a symbol, the shorter the representation of the bits sequence. Comparing with other types of approaches, A.S.C.I.I. coding, for example, fixed length code, the Huffman encoding is a variable length system, assigns smaller codes for characters with a high degree of usage, smaller codes for less frequently used characters, thus the files are reduced and the data transfer process is much simplified.
Several studies (Lewis & Papadimitriou, 1998). Economic competition, strengthening and developing the position of the economic organization in the specialized market, where it is an actor, becomes in the digital age a competition for the understanding, analysis and operational transposition of the information. Organizational software and hardware structures are active parts of the systemic process to achieve economic performance through correct, optimized managerial decisions.

Huffman's data compression methods are a fundamental argument in achieving these central targets, as well as fundamental research trends, subsequent applicative transposion and quantification in increasing profit returns demonstrating their importance.

References

Aki, S.G. (1989). The design and analysis of parallel algorithms. Prentice Hall.
Atallah, M. J. (1998). Algorithms and theory of computation handbook. CRC press.

Avidan, S., & Shamir, A. (2007, August). Seam carving for content-aware image resizing. In *ACM Transactions on graphics (TOG)* (Vol. 26, No. 3, p. 10). ACM.

Baase, S. (2009). Computer algorithms: introduction to design and analysis. Pearson Education India.

Bellman, R. (1957). *Dynamic programming*. Princeton University Press.

Bentley, J.L. (1982). *Writting Efficient Progmams*. Prentice Hall.

Chvatal, V. (1983). *Linear Programming*. W.H. Freeman and Company.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms*. MIT press.

Dasgupta, S., Papadimitriou, C., & Vazirani,U. (2008). *Algorithms.* McGraw-Hill.

Drake, A.W. (1967). Fundamentals of Applied Probability Theory. McGraw-Hill.

Fredman, M., & Saks, M. (1989, February). The cell probe complexity of dynamic data structures. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing* (pp. 345-354). ACM.

Gabow, H. N., & Tarjan, R. E. (1985). A linear-time algorithm for a special case of disjoint set union. *Journal of computer and system sciences*, 30(2), 209-221.

Gass,S. (1975). *Linear Programming: Methods and Applications*. International Thomson Publishing.

Goldberg, A.V. (1995). Scaling algorithms for the shortest paths problem. *SIAM Journal on Computing*, 24(3), 494-504.

Golub, G.H., & Van Loan, C.F. (2012). *Matrix computations* (Vol. 3). JHU Press.

Gonzalez, R.C., & Woods, R.E. (1992). *Digital image processing.* Addison-Wesley.

Goodrich, M.T., & Tamassia, R. (1998). *Data structures and algorithms in Java.* John Wiley & Sons.

Graham, R.L., Knuth, D.E., & Patashnik, O. (1994). *Concrete Mathematics, A Foundation for Computer Science*, Reading, MA: Adison-Wesley.

Karloff, H. (1991). *Linear programming*. Birkhauser.

Lewis, H.R., & Papadimitriou, C.H. (1998). *Elements of the Theory of Computation*. Prentice Hall.

Manber, U. (1989). *Introduction to algorithms: a creative approach* (Vol. 4). Reading, MA: Addison-Wesley.

Pratt, W. K. (2007). *Digital image processing: PIKS Scientific inside* (Vol. 4). Hoboken, NJ: Wiley.

Sipser, M. (2006). Introduction to the Theory of Computation, Thomson Course Technology.

Skiena, S.S. (1998). *The algorithm design manual: Text* (Vol. 1). Springer.

Weiss, M. A. (2007). *Data Structure and Problem Solving Using C++*, third edition, Addison-Wesley.