3

Advances in Intelligent Systems

Sergey Y. Yurish
Editor

Advances in Intelligent Systems

Book Series, Volume 3

S.Yurish
Editor

Advances in Intelligent Systems

Book Series, Volume 3

IFSA International Frequency Sensor Association Publishing

S. Yurish, *Editor*
Advances in Intelligent Systems, Book Series, Vol. 3

Published by IFSA Publishing, S. L., 2025
E-mail (for print book orders and customer service enquires):
ifsa.books@sensorsportal.com

Visit our Home Page on http://www.sensorsportal.com

Neither the authors nor International Frequency Sensor Association Publishing accept any responsibility or liability for loss or damage occasioned to any person or property through using the material, instructions, methods or ideas contained herein, or acting or refraining from acting as a result of such use.

ISBN: 978-84-09-72041-5
BN-20250505-XX
BIC: UYD

Contents

Preface

I am pleased to present the 3rd volume of the '*Advances in Intelligent Systems*' Book Series, a continuation of our effort to disseminate high-quality and forward-looking research in the rapidly evolving domains of artificial intelligence, control theory, cybersecurity, and intelligent infrastructure. This volume showcases six contributions from international researchers who explore both theoretical frameworks and practical implementations of intelligent systems across diverse application areas.

Chapter 1 introduces an original method of synthesizing neural networks using algebraic and spectral approaches. Rather than minimizing implementation errors via traditional weight tuning, this approach constructs neural structures with threshold activation functions and binomial hyperfilters. The result is a cascade architecture that efficiently implements high-dimensional logical functions. The work advances neural synthesis theory by combining elements of logical function decomposition, spectral analysis, and modular hardware-friendly designs.

In Chapter 2 the focus shifts to chaos stabilization in nonlinear dynamical systems. The author outlines classical and modern methods for suppressing or inducing chaotic behavior using Lyapunov exponents and feedback control. A unique feature of this contribution is its emphasis on both centralized and decentralized control strategies, supported by a case study involving a three-machine power system model.

Chapter 3 presents a novel semantic approach to modeling the capabilities of industrial field devices. By leveraging ontologies and SWRL inference rules, the authors design a reasoning engine capable of automatically generating device capability descriptions in an Industry 4.0 context. The work is significant for its contribution to the standardization and interoperability of smart devices in automated systems.

Cybersecurity is addressed in Chapter 4. This extensive chapter provides both foundational knowledge and practical frameworks for enhancing infrastructure resilience. The authors propose methods for integrating OSINT (Open Source Intelligence) tools into cybersecurity monitoring platforms, offering detailed case studies and implementation guidelines for utility sectors such as energy and water.

Chapter 5 investigates the challenges of data communication reliability in railway systems. It combines signal modeling, simulation of impulse interference, and experimental data to assess the quality of service in train-to-train communications. The findings offer insights into the robustness of M2M communication protocols in high-noise environments.

Finally, Chapter 6, explores statistical forecasting in railway operations. The authors develop and validate a Markov-based predictive model to estimate train delays during peak hours. The approach is data-driven and is grounded in real-time analytics, offering actionable outputs for transport scheduling and operational efficiency.

Altogether, this volume underscores the interdisciplinary character of intelligent systems research. From neural architectures and control algorithms to semantic modeling and cyber resilience, the chapters collectively demonstrate how intelligent methods are transforming engineering and technology domains. We hope that the content will be of interest to academics, practitioners, and students working in these fields, and we look forward to continuing this Series with future volumes that reflect the dynamic landscape of intelligent system design.

Sergey Y. Yurish

Editor
IFSA Publishing   Barcelona, Spain

Contributors

Victor Chavez
Aachen University of Applied Sciences, Institute for Applied Automation and Mechatronics, Aachen, Germany

Vladimir Chebotarev
Computing Center of the Far Eastern Branch of the Russian Academy of Sciences, Khabarovsk, Russia

Furkan Çolhak
CCIP, Center for Cyber Security and Critical Infrastructure Protection, Kadir Has University Istanbul, Turkey

Reiner Creutzburg
CCIP, Center for Cyber Security and Critical Infrastructure Protection, Kadir Has University Istanbul, Turkey

Boris Davydov
Far Eastern State Transport University, Khabarovsk, Russia E-mail: msidorovich60@gmail.com

Hasan Dağ
CCIP, Center for Cyber Security and Critical Infrastructure Protection, Kadir Has University Istanbul, Turkey

Mert İlhan Ecevit
CCIP, Center for Cyber Security and Critical Infrastructure Protection, Kadir Has University Istanbul, Turkey. E-mail: mertilhan.ecevit@khas.edu.tr

Yulia Ponomarchuk
Far Eastern State Transport University, Khabarovsk, Russia, E-mail: msidorovich60@gmail.com

Vladimir N. Shashikhin
Peter the Great St. Petersburg Polytechnic University, Russia

Maksim Sidorovich
Far Eastern State Transport University, Khabarovsk, Russia, |E-mail: msidorovich60@gmail.com

Alexander N. Sychev
Candidate of Technical Sciences, Associate Professor, Riga, Latvia

Jörg Wollert
Aachen University of Applied Sciences, Institute for Applied Automation and Mechatronics, Aachen, Germany

1.

Synthesis of Cascade Neural Structure with Binomial Hyperfilters

Alexander Nikolaevich Sychev

1.1. Introduction

A lot of tasks, which are solved with the help of artificial neural networks (NN), could be adequately represented by logical functions (LF). For example, when we have different methods of processing the information for pattern recognition, in soft computing algorithm, etc. That is why, it is quite appropriate to explore the questions of neural networks' synthesis implementing the corresponding functions.

The accepted theory of an artificial neuron is that there is a nonlinear response of the device to the sum of the weights of the input signals. It turns out that separate neurons realize effectively only those linear functions which according to the Post's classification possess complete monotony. If we consider power, then the class of such linear functions is rather small. Therefore, neurons are united into networks, the functional capabilities of which are slightly higher.

But the stumbling block for synthesis of NN is LF $f(x)$:

- more or less chaotic distribution of unit and zero constituents in space X of definition of LF, $X = \{x\}$;
- LF with characteristic signs of partial anti-self-duality, for example, autocorrelation characteristics (ACC) B_f and $B_{\overline{f}}$, in accordance with function $f(x)$ and its inversion $\overline{f(x)} = 1 - f(x)$ of the definite rank that does not contain coefficients with zero values: $\forall\omega\left\{\ B_f(\omega) \neq 0\ \&\ B_{\overline{f}}(\omega) \neq 0\right\}$;

Alexander Nikolaevich Sychev
Candidate of Technical Sciences, Associate Professor, Riga, Latvia

- LF with expressed characteristics of linearity in the deductions over field GF(2).

In the synthesis of neural networks, in accordance with the established techniques, the number of neurons is determined in advance, also this or that structure of connections between neurons is determined, some sigmoid neuron activation function is selected, and by adjusting the weights of their inputs, the amount of inevitable implementation error is minimized. As a result, the expected aim of the synthesis – the exact implementation by the network specified LF is replaced by a search for a hypothetical global minimum of the implementation error.

As an alternative, attention should be paid to the construction of neural networks with a threshold activation function. Algebraic methods are effective in this direction, also, spectral representations of linear functions are applicable, and so the achievement of accurate implementation results for problems of arbitrary complexity is ensured. In the works of [1, 2], on the basis of the mathematical concept of a tangent bundle of space X, the concept of hyper neuron and binomial hyperfilter is constructed. There examples are given that allow to evaluate the efficiency of their application as part of neurostructures.

The aim of the work is to describe a method for constructing a cascade neurostructure on the basis of bipolar neurons with binomial hyperlinear transformations of input signals involving the operation of self-dual transformations of the fragments realized by LF.

1.2. Tangent Bundle of Space Definition of the Logical Function

Let X, the definition space of LF $f(x)$, $x=(x_1,x_2,\ldots,x_n)$, be divided into non-overlapping fragments: $X=\bigcup\limits_{i=1\div m} X_i, X_i \cap X_j = \varnothing, \quad i \neq j.$ In accordance with the properties of Walsh spectral representations [3] the spectrum S of the sum, for example, of two functions $f_a(x)+f_b(x)$ is equal to the sum of spectra S_a, S_b of each of these functions: $S=S_a+S_b$. Consequently, spectrum f (if its value $\{0,1\}$ is interpreted by the values $\{-1,1\}$) can be represented by the sum $S=\sum\limits_{i=1\div m} S_i$, where

S_i is the spectrum of fragmentary partial function f_i, that is defined by the expression

$$f_i(x)=\begin{cases} f(x), x\in X_i \\ "-", \ x\notin X_i \end{cases} \tag{1.1}$$

Here the symbol "–" means an undefined value not taken into account that in case of spectrum calculation is represented as"0". Thus, the coefficients $s(\omega)$, $s_i(\omega)$ of the corresponding spectra are connected by the relation

$$s(\omega)=\sum_{i=1\div m} s_i(\omega), \tag{1.2}$$

where $s(\omega)=\sum_x f(x)\cdot wal(\omega,x)$, $s_i(\omega)=\sum_x f_i(x)\cdot wal(\omega,x)$, $s(\omega)\in S$, $s_i(\omega)\in S_i$, $i=1\div m$, $wal(\omega,x)$ – of Walsh function with the number ω, that serves as an argument of the spectral representation. At that it may turn out that for a certain value of argument $\omega^*\neq 0$ the signs of the corresponding coefficients within the spectrum of one or several fragments differ from the sign of the sum. For example, $s(\omega^*)\leq 0$, while $s_i(\omega^*)>0$, $s_j(\omega^*)>0$, for some X_i, X_j. This fact could be used for the increase in the efficiency of linearization of neural structures.

Linear filters $\lambda(x)$ are defined by argument values $\omega\neq 0$, $\omega=(\omega_1,\omega_2,\dots,\omega_n)$ of the coefficients $s(\omega)$ according to the rule of scalar product of vectors: $\lambda(x)=\langle\omega, x\rangle=\omega_1x_1\oplus\omega_2x_2\oplus\dots\oplus\omega_nx_n$, where $\oplus$ is the addition sign according to $\mathrm{mod}\,2$. While choosing filter λ effective for the neuron structure, the corresponding to this filter coefficient $s(\omega)$ is selected among spectral coefficients maximal in absolute value, $s(\omega)\in\max\{|s(\omega)|\}$. Thus, in order to increase the efficiency of the filter, it is necessary to manipulate the output signal $\lambda(x)$, taking into consideration coefficient signs $s_i(\omega)$, $s_j(\omega)$, of the corresponding fragments X_i, X_j. For example, if $x\in X_i$, and the sign $s_i(\omega)$ is opposite to sign $s(\omega)$, then, while defining,

$$\lambda^{\delta}(x)=\lambda(x)\oplus\delta_i(x),\ \delta_i(x)=\begin{cases}1, \ x\in X_i \\ 0, \ x\notin X_i\end{cases}, \tag{1.3}$$

the increase in the absolute value of the modified coefficient $s_i^{\delta}(\omega)$, corresponding to λ^{δ} : $\left|s_i^{\delta}(\omega)\right| = \left|s(\omega)\right| + 2$ can be achieved.

Space partitioning X is modeled in a simple way with the help of a regular decoder. Decoder (*DC*) with t inputs provide splitting X into $K = 2^t$ fragments. If linear filters $\lambda_1(x), \lambda_2(x), \dots, \lambda_t(x)$, defined by a linearly independent set of arguments $\{\omega_1, \omega_2, \dots, \omega_t\}$ are connected to DC inputs, then the output signals of the decoder will correspond to the tangent bundle $X = X_0 \cup X_1 \cup \dots \cup X_{K-1}$ of the space X based on *T*, which in this case is the subspace in *X*, $T \lhd X$, with the basis $\{\omega_1, \omega_2, \dots, \omega_t\}$. Herewith, linear independence of the arguments which compose the basis provide the orthogonality of the partition for each argument, and therefore, the equal thickness of the tangent layers $|X_0| = |X_1| = \dots = |X_{K-1}|$. The correcting signal $\delta_i(x) = 1$, $i = 0 \div (K-1)$, if $x \in X_i$ is formed on *i* is an input of DC. Now, if for some filter $\lambda^{\nu} = \langle \omega, x \rangle$, selected during the synthesis process of the neuron structure by value $\left|s^{\nu}(\omega)\right|$, the signs of the sum $s^{\nu}(\omega)$ and its member $s_i^{\nu}(\omega)$ are different, so i is an input of DC it is necessary to connect via diode and adder $m2$ with the input of the linear filter λ^{ν}, Fig. 1.1.

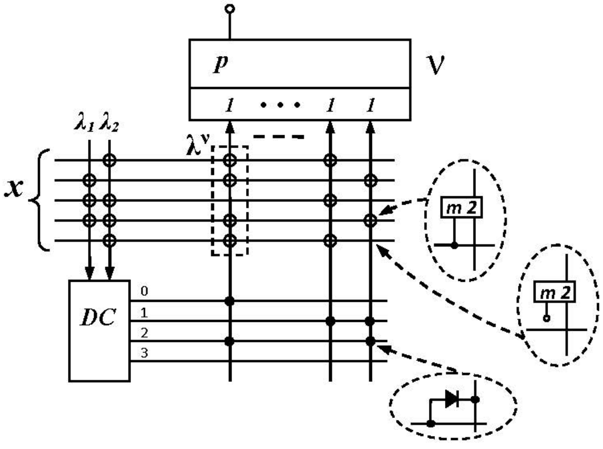


Fig. 1.1. The Block Diagram of Hyperlinear Transformation of Neuron Input Signals: **v** is the neuron with threshold activation and structure vector, $A = (p;1,1,\dots,1)$; ***DC*** is the decoder with linear filters $\lambda_1(x)$, $\lambda_2(x)$ at the inputs; ***m2*** is the two-way modulo 2 adder; $\lambda^{\mathbf{v}}$ is the linear filter at the neuron input.

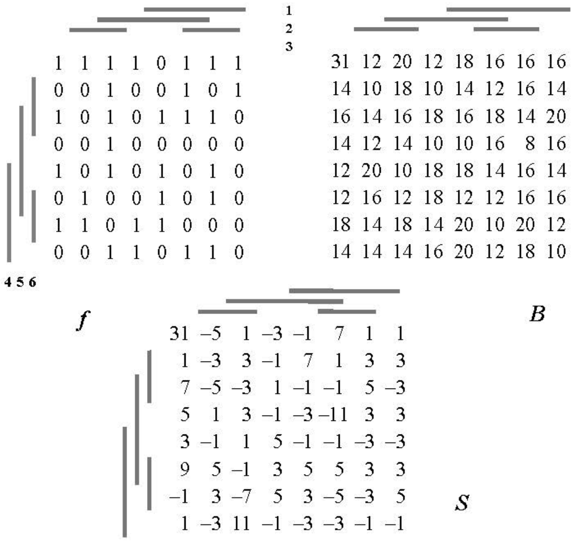
The selection of a linear independent set of arguments $\{\omega_1, \omega_2, \dots, \omega_t\}$ for the base T should be made taking into consideration two factors. First of all, taking into account the values of spectral coefficients $s(\omega_j) \in S,\ j = 1 \div t,$ and, secondly, taking into account the values of autocorrelation coefficients $b(\omega_j) \in B,$ which in the work of [3] are determined by the ratio

$$b(\omega) = \sum_x f(x) \cdot f(x \oplus \omega) \tag{1.4}$$

Absolute values $|s(\omega_j)|,\ \omega_j \in \{\omega_1, \omega_2, \dots, \omega_t\},$ should be minimal in order to provide a proportional distribution of one and zero constituents of function f by fragments $X_i,\ i = 0 \div (K-1).$ This is necessary for equalization of fragmentary coefficients $s_i(\omega) \in S_i,\ i = 0 \div (K-1),$ for the choice of filter $\lambda^{\nu}(x),$ common for all fragments. But, contrary, the values $b(\omega_j)\ \omega_j \in \{\omega_1, \omega_2, \dots, \omega_t\}$ should be maximal in order to reduce the non-monotony of partial fragmentary functions $f_i(x).$ As it is clear that for every $\omega_j \in \{\omega_1, \omega_2, \dots, \omega_t\}$ arguments x and $x' = x \oplus \omega_j$ belong to different fragments X_i. Therefore, for the fragment X, the number of anti-self-duality (non-monotonicity), characterized by conditions $f(x) = f(x \oplus \omega) = 1 \text{ и } (x \in X_i) \& (x \oplus \omega \in X_i)$ decreases. For example, Fig. 1.2 shows LF $f(x_1, x_2, \dots, x_6)$ with $n = 6$ variables, its spectrum is $S(\omega_1, \omega_2, \dots, \omega_6)$ and ACC $B(\omega_1, \omega_2, \dots, \omega_6).$ Here, binary codes of arguments x, ω with variables $x_i, \omega_i,\ i = 1 \div 6,$ in Karnaugh tables are modeled by the combination of lines with numbers which correspond to indices i.

The selection of basis $\{\omega_1, \omega_2, \dots, \omega_t\},\ t \le n,$ of base $T,$ according to the above mentioned rule, should be determined from a set $n = 6$ of linear independent arguments $\omega,$ relative to elements in Karnaugh table, for example, with the numbers {13, 17, 7, 33, 37, 56}.

The numbers are matched with the elements in a table by columns, from top down, beginning from the top left element. The values {-1, 1, -1, -1, -1, -1} correspond to the selected elements in the table ***S***, and in table ***B*** they correspond to the values {20, 20, 18, 18, 18, 18}.



1	1	1	1	0	1	1	1
0	0	1	0	0	1	0	1
1	0	1	0	1	1	1	0
0	0	1	0	0	0	0	0
1	0	1	0	1	0	1	0
0	1	0	0	1	0	1	0
1	1	0	1	1	0	0	0
0	0	1	1	0	1	1	0

31	12	20	12	18	16	16	16
14	10	18	10	14	12	16	14
16	14	16	18	16	18	14	20
14	12	14	10	10	16	8	16
12	20	10	18	18	14	16	14
12	16	12	18	12	12	16	16
18	14	18	14	20	10	20	12
14	14	14	16	20	12	18	10

31	–5	1	–3	–1	7	1	1
1	–3	3	–1	7	1	3	3
7	–5	–3	1	–1	–1	5	–3
5	1	3	–1	–3	–11	3	3
3	–1	1	5	–1	–1	–3	–3
9	5	–1	3	5	5	3	3
–1	3	–7	5	3	–5	–3	5
1	–3	11	–1	–3	–3	–1	–1

Fig. 1.2. Representation of LF $f(x)$ and Spectrum $S(\omega)$ and AKX $B(\omega)$ by Karnaugh Tables.

In general, besides linear method, other techniques for forming a layering base T could be used. For example, between the outputs of filters $\lambda_1(x), \lambda_2(x), \dots, \lambda_n(x)$ and decoder DC with t inputs, it is possible to place additionally t neurons $v_1, v_2, \dots, v_t$, that have $m_1, m_2, \dots, m_t$ inputs, $m_1 + m_2 + \dots + m_t \le n.$ At that, it is necessary that the rank of each neural function satisfied the condition $|v_i(x)| = 2^{n-1}$, naturally, in order to provide the conditions for equal power of fragments $|X_i|$, $i = 1 \div (K-1)$.

In order to illustrate this, let us consider two variants of forming the base of the tangent bundle, Fig. 1.3.

Let, in the first case, Fig. 1.3a, the arguments of the basis $\{\omega_1, \omega_2\}$ of the linear base *T*, elements {13, 33} be chosen. Correspondingly, $\omega_1 = (001110)$, $\omega_2 = (010000)$ and 4 tangent layers or subspace X_0, X_1, X_2, X_3 are formed. In Karnaugh table, as we can see in Fig. 1.4 ***X0-3,*** these subspaces are indicated by numbers of the corresponding indices.

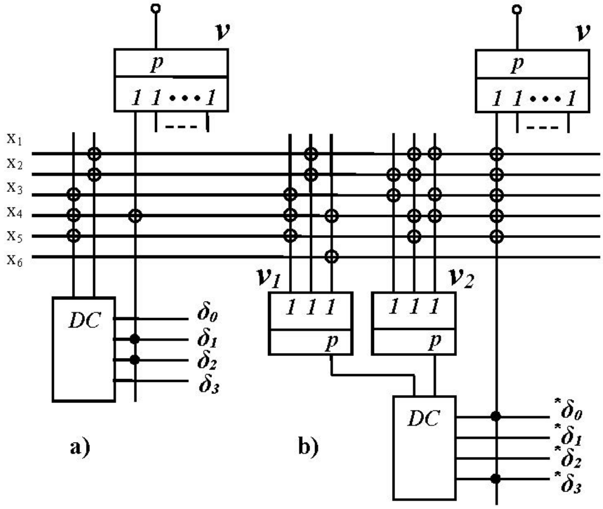
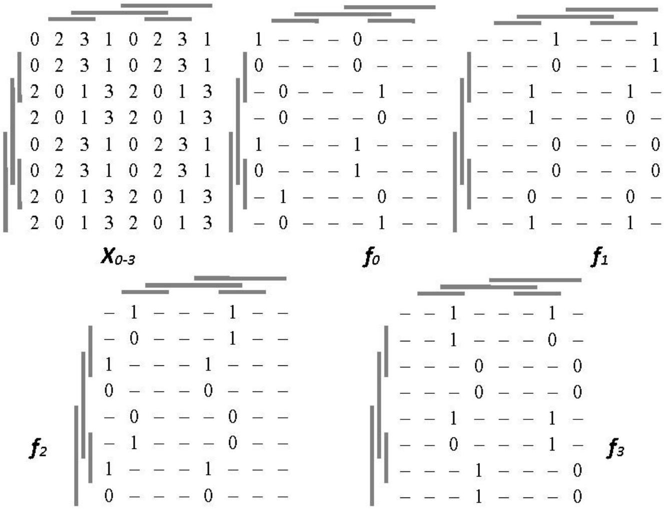


Fig. 1.3. Examples of Technical Formation of the Base of the Tangent Bundle of the Space of Definition of a Function.



$X_{0\text{-}3}$

0	2	3	1	0	2	3	1
0	2	3	1	0	2	3	1
2	0	1	3	2	0	1	3
2	0	1	3	2	0	1	3
0	2	3	1	0	2	3	1
0	2	3	1	0	2	3	1
2	0	1	3	2	0	1	3
2	0	1	3	2	0	1	3

f_0

1	–	–	–	0	–	–	–
0	–	–	–	0	–	–	–
–	0	–	–	–	1	–	–
–	0	–	–	–	0	–	–
1	–	–	–	1	–	–	–
0	–	–	–	1	–	–	–
–	1	–	–	–	0	–	–
–	0	–	–	–	1	–	–

f_1

–	–	–	1	–	–	–	1
–	–	–	0	–	–	–	1
–	–	1	–	–	–	1	–
–	–	1	–	–	–	0	–
–	–	–	0	–	–	–	0
–	–	–	0	–	–	–	0
–	–	0	–	–	–	0	–
–	–	1	–	–	–	1	–

f_2

–	1	–	–	–	1	–	–
–	0	–	–	–	1	–	–
1	–	–	–	1	–	–	–
0	–	–	–	0	–	–	–
–	0	–	–	–	0	–	–
–	1	–	–	–	0	–	–
1	–	–	–	1	–	–	–
0	–	–	–	0	–	–	–

f_3

–	–	1	–	–	–	1	–
–	–	1	–	–	–	0	–
–	–	–	0	–	–	–	0
–	–	–	0	–	–	–	0
–	–	1	–	–	–	1	–
–	–	0	–	–	–	1	–
–	–	–	1	–	–	–	0
–	–	–	1	–	–	–	0

Fig. 1.4. Representation of Subspaces $X_{0\text{-}3}$ and the Corresponding Fragmentary Functions f_i by Karnaugh Tables.

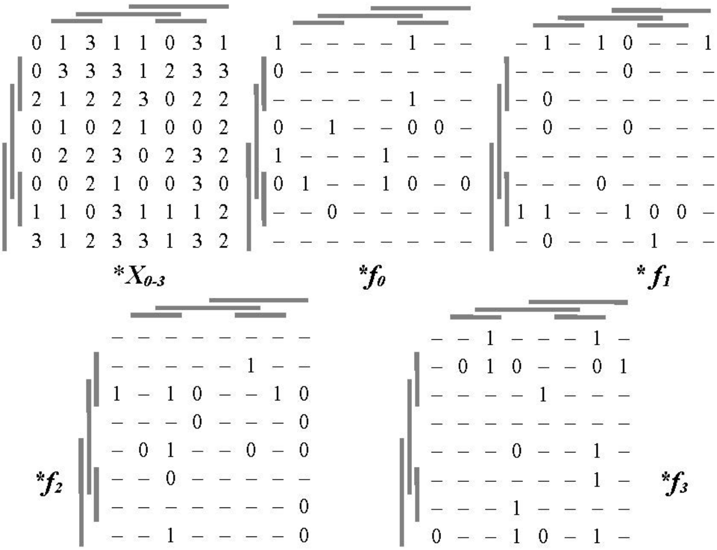
Corresponding to them fragmentary or partial functions f_l are given in tables f_0, f_1, f_2, f_3. In the second case, Fig. 1.3b, we shall place linearized neurons $v_1(\lambda_1,\lambda_2,\lambda_3)$ and $v_2(\lambda_4,\lambda_5,\lambda_6)$ at the inputs of the decoder, these linearized neurons have structural vectors equal to: $A_1=\{2;1,1,1\}$, $A_2=\{2;1,1,1\}$. The input filters of neurons $\{\lambda_1,\lambda_2,\lambda_3\}$, $\{\lambda_4,\lambda_5,\lambda_6\}$ are determined by arguments ω with numbers {13, 33, 7} and {17, 37, 56}. Here

$$\lambda_1 \Leftrightarrow \omega_1 = (001110), \lambda_2 \Leftrightarrow \omega_2 = (010000),$$

$$\lambda_3 \Leftrightarrow \omega_3 = (000101), \lambda_4 \Leftrightarrow \omega_4 = (011000),$$

$$\lambda_5 \Leftrightarrow \omega_5 = (110110), \lambda_6 \Leftrightarrow \omega_6 = (101100)$$

The second bundle base $^*T$, formed in this way, forms subspaces $^*X_0$, $^*X_1$, $^*X_2$, $^*X_3$, which are represented in Fig. 1.5 in the table $^*X_{0-3}$. Partial functions $^*f_i$, corresponding to them are given in tables $^*f_0$, $^*f_1$, $^*f_2$, $^*f_3$.



$^*X_{0-3}$

0	1	3	1	1	0	3	1
0	3	3	3	1	2	3	3
2	1	2	2	3	0	2	2
0	1	0	2	1	0	0	2
0	2	2	3	0	2	3	2
0	0	2	1	0	0	3	0
1	1	0	3	1	1	1	2
3	1	2	3	3	1	3	2

$^*f_0$

1	–	–	–	–	1	–	–
0	–	–	–	–	–	–	–
–	–	–	–	–	1	–	–
0	–	1	–	–	0	0	–
1	–	–	–	1	–	–	–
0	1	–	–	1	0	–	0
–	–	0	–	–	–	–	–
–	–	–	–	–	–	–	–

$^*f_1$

–	1	–	1	0	–	–	1
–	–	–	–	0	–	–	–
–	0	–	–	–	–	–	–
–	0	–	–	0	–	–	–
–	–	–	–	–	–	–	–
–	–	–	0	–	–	–	–
1	1	–	–	1	0	0	–
–	0	–	–	–	1	–	–

$^*f_2$

–	–	–	–	–	–	–	–
–	–	–	–	–	1	–	–
1	–	1	0	–	–	1	0
–	–	–	0	–	–	–	0
–	0	1	–	–	0	–	0
–	–	0	–	–	–	–	–
–	–	–	–	–	–	–	0
–	–	1	–	–	–	–	0

$^*f_3$

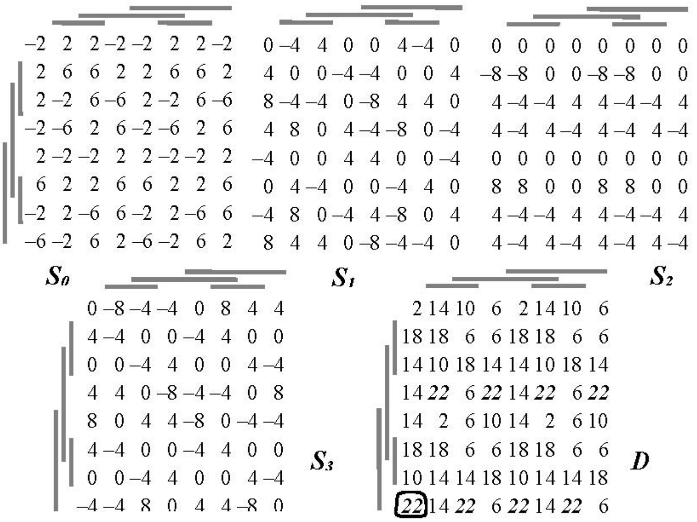
–	–	1	–	–	–	1	–
–	0	1	0	–	–	0	1
–	–	–	–	1	–	–	–
–	–	–	–	–	–	–	–
–	–	–	0	–	–	1	–
–	–	–	–	–	–	1	–
–	–	–	1	–	–	–	–
0	–	–	1	0	–	1	–

Fig. 1.5. Representation of Subspaces ***X*₀₋₃** and the Corresponding Fragmentary Functions ***fᵢ*** by Karnaugh Tables.

It is necessary to use the spectra of partial LF for further calculations. In the work [4] the rule for their calculations is given, this rule is based on the preliminary presentation of the partial LF $f(x)$ corresponding to the ternary function of the binary argument $u(x)$. In this example, in order to make it easier to understand, and not to lose the efficiency of the calculations, every partial LF $f_i(x)$ is represented by the ternary function according to the rule

$$u_i(x) = \begin{cases} r/k_1^i, \ if \ f_i(x) = 1; \\ -r/k_0^i, \ if \ f_i(x) = 0\ , \\ 0, \ if \ \ f_i(x) = "-" \end{cases} \qquad (1.5)$$

where r is a certain arbitrary normalizing multiplier, the same for all the fragments $f_i(x)$, k_1^i is the number of single constituents $k_1^i = \left|\{x : f_i(x) = 1\}\right|$, and k_0^i is the number of zero constituents $k_0^i = \left|\{x : f_i(x) = 0\}\right|$ of the function $f_i(x)$, $i = 0 \div 3$. In the first variant with base T, in view of almost equal ratios of constituents, $\{k_1^0, k_1^1, k_1^2, k_1^3\} = \{7, 8, 8, 8\}$, $\{k_0^0, k_0^1, k_0^2, k_0^3\} = \{9, 8, 8, 8\}$, we shall further simplify and accept that $u_i(x) = 1$, if $f_i(x) = 1$ and $u_i(x) = -1$ if $f_i(x) = 0$. The spectra of fragmentary representing functions u_0, u_1, u_2, u_3 in Fig. 1.6 are presented in tables ***S₀S₁S₂S₃***.



–2	2	2	–2	–2	2	2	–2
2	6	6	2	2	6	6	2
2	–2	6	–6	2	–2	6	–6
–2	–6	2	6	–2	–6	2	6
2	–2	–2	2	2	–2	–2	2
6	2	2	6	6	2	2	6
–2	2	–6	6	–2	2	–6	6
–6	–2	6	2	–6	–2	6	2

S₀

0	–4	4	0	0	4	–4	0
4	0	0	–4	–4	0	0	4
8	–4	–4	0	–8	4	4	0
4	8	0	4	–4	–8	0	–4
–4	0	0	4	4	0	0	–4
0	4	–4	0	0	–4	4	0
–4	8	0	–4	4	–8	0	4
8	4	4	0	–8	–4	–4	0

S₁

0	0	0	0	0	0	0	0
–8	–8	0	0	–8	–8	0	0
4	–4	–4	4	4	–4	–4	4
4	–4	4	–4	4	–4	4	–4
0	0	0	0	0	0	0	0
8	8	0	0	8	8	0	0
4	–4	–4	4	4	–4	–4	4
4	–4	4	–4	4	–4	4	–4

S₂

0	–8	–4	–4	0	8	4	4
4	–4	0	0	–4	4	0	0
0	0	–4	4	0	0	4	–4
4	4	0	–8	–4	–4	0	8
8	0	4	4	–8	0	–4	–4
4	–4	0	0	–4	4	0	0
0	0	–4	4	0	0	4	–4
–4	–4	8	0	4	4	–8	0

S₃

2	14	10	6	2	14	10	6
18	18	6	6	18	18	6	6
14	10	18	14	14	10	18	14
14	***22***	6	***22***	14	***22***	6	***22***
14	2	6	10	14	2	6	10
18	18	6	6	18	18	6	6
10	14	14	18	10	14	14	18
$\boxed{22}$	14	***22***	6	***22***	14	***22***	6

D

Fig. 1.6. Representation of Spectra ***S₀ – S₃*** and Spectral Density ***D***.

In the event that for one or more functions it turns out that $k_1^i = 0$ or $k_0^i = 0,$ then it is necessary to calculate $k_1 = \left|\{x : f(x) = 1\}\right|,$ $k_0 = \left|\{x : f(x) = 0\}\right|$ for the generating function $f(x),$ and determine the ternary functions according to the rule

$$u_i(x) = \begin{cases} r/k_1, \ if\ f(x) = 1 \\ -r/k_0, \ if\ f(x) = 0 \\ 0, \ if\ \ f(x) = "-" \end{cases} \qquad (1.6)$$

In the second variant with the base bundle ${}^*T$ the ratio of constituents are different $\{{}^*k_1^0, {}^*k_1^1, {}^*k_1^2, {}^*k_1^3\} = \{8, 7, 6, 10\},$ $\left\{{}^*k_0^0, {}^*k_0^1, {}^*k_0^2, {}^*k_0^3\right\}$ = $\{8, 9, 10, 6\}$, we shall form ternary functions, defining the value $r = 5.$ The spectra of fragmentary representing functions ${}^*h_0, {}^*h_1, {}^*h_2, {}^*h_3$ in Fig. 1.7 are presented in tables ****S₀***, ****S₁***, ****S₂***, ****S₃***.

In order to determine the corrective action of $\delta_0 \div \delta_3$ and ${}^*\delta_0 \div {}^*\delta_3,$ it is necessary to memorize the signs of the spectral coefficients for each variant of base T and ${}^*T.$ It could be done following the rules

$$\Delta(\omega) = \sum_{i=0\div 3} 2^i \cdot sign^0(s_i(\omega)), \qquad (1.7)$$

$${}^*\Delta(\omega) = \sum_{i=0\div 3} 2^i \cdot sign^0({}^*s_i(\omega)), \qquad (1.8)$$

where

$$sign^0(z) \ = \begin{cases} 1, if\ z > 0 \\ 0, if\ z \le 0 \end{cases} \qquad (1.9)$$

As a result, in Fig. 1.8 in tables Δ and ${}^*\Delta,$ the numbers corresponding to four-bit binary codes $\delta = (\delta_3\,\delta_2\,\delta_1\,\delta_0)$ and ${}^*\delta = ({}^*\delta_3\,{}^*\delta_2\,{}^*\delta_1\,{}^*\delta_0),$ determining the corrective connections of decoder outputs with linear filters on the inputs of neuro structures are presented.

In each variant, in order to select an argument ω^v, defining the first filter λ^v of the neurostructure, it is necessary to calculate the total spectral densities

0	0	–2.50	–2.50	2.50	0	–2.50	0	
2.50	2.50	5.00	0	0	2.50	0	2.50	
2.50	–2.50	0	0	2.50	0	2.50	0	
0	0	2.50	2.50	0	–2.50	5.00	–2.50	
2.50	0	–2.50	0	2.50	2.50	0	0	
5.00	2.50	5.00	2.50	5.00	0	7.50	–2.50	$^*S_0$
–2.50	0	–2.50	0	0	0	–2.50	2.50	
0	–2.50	5.00	–2.50	–2.50	–2.50	0	0	

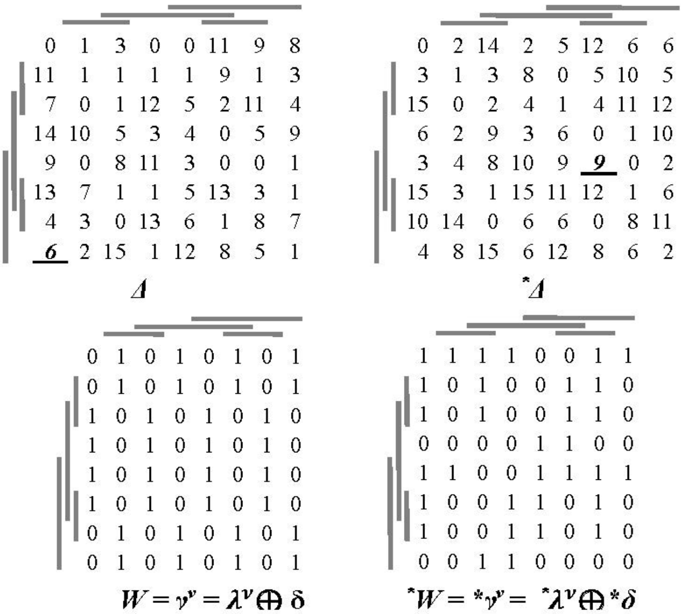
0	1.27	3.17	1.27	–0.63	–1.59	0.95	1.27	
1.27	–1.27	1.27	–1.27	–3.81	–4.13	3.49	–1.27	
1.27	–1.27	1.27	–1.27	–1.59	–1.90	1.27	–3.49	
4.44	1.27	–1.27	1.27	1.59	–3.81	–1.27	3.49	
0.95	–3.17	–1.27	3.49	–1.90	–3.81	–1.27	1.27	
4.13	3.81	–3.81	1.59	1.27	–1.27	–3.81	3.81	$^*S_1$
1.90	1.59	–1.59	3.81	1.27	–1.27	–3.81	3.81	
–1.27	–0.95	0.95	1.27	–6.35	–3.81	3.17	1.27	

0	–5.33	1.33	–2.67	1.33	7.33	2.00	2.67	
–2.67	0	0	–2.67	–2.67	2.00	–3.33	2.67	
2.67	0	0	2.67	–0.67	1.33	0	0.67	
1.33	0	0	–2.67	0.67	–5.33	0	–0.67	
–0.67	4.00	0	–0.67	–1.33	–5.33	0	–2.67	
0.67	0	0	0.67	–2.67	5.33	0	2.67	
–4.00	3.33	–3.33	2.67	2.67	–5.33	0	–2.67	$^*S_2$
2.67	–2.00	2.00	2.67	2.67	0	1.33	–2.67	

0	–2.67	1.33	0	–4.00	2.00	–0.67	–2.67	
0	–5.33	–2.67	2.67	–2.67	–0.67	2.00	0	
2.67	–2.67	–5.33	0	–3.33	–1.33	2.67	0.67	
–1.33	0	2.67	–2.67	–4.67	–2.67	0	2.00	
–0.67	–1.33	5.33	2.00	1.33	5.33	–2.67	–2.67	
3.33	0	–2.67	0.67	2.67	2.67	0	0	
4.00	0.67	–3.33	0	0	0	2.67	2.67	$^*S_3$
0	3.33	4.67	–2.67	2.6	2.67	–4.00	0	

0	9.27	8.34	6.44	8.47	10.92	6.12	6.60	
6.44	9.10	8.94	6.60	9.14	9.29	8.83	6.44	
9.10	6.44	6.60	3.94	8.09	4.57	6.44	4.83	
7.11	1.27	6.44	9.10	6.92	14.31	6.27	8.66	
4.79	8.51	9.10	6.16	7.07	$\boxed{\mathbf{16.98}}$	3.94	6.60	
13.13	6.31	11.48	5.42	11.60	9.27	11.31	8.98	$^*D$
12.40	5.59	10.75	6.48	3.94	6.60	8.98	11.64	
3.94	8.79	12.62	9.10	14.18	8.98	8.51	3.94	

Fig. 1.7. Representation of Spectra ***S_0* – *S_3*** and Spectral Density ****D***.



0	1	3	0	0	11	9	8
11	1	1	1	1	9	1	3
7	0	1	12	5	2	11	4
14	10	5	3	4	0	5	9
9	0	8	11	3	0	0	1
13	7	1	1	5	13	3	1
4	3	0	13	6	1	8	7
$\underline{\mathbf{6}}$	2	15	1	12	8	5	1

Δ

0	2	14	2	5	12	6	6
3	1	3	8	0	5	10	5
15	0	2	4	1	4	11	12
6	2	9	3	6	0	1	10
3	4	8	10	9	$\underline{\mathbf{9}}$	0	2
15	3	1	15	11	12	1	6
10	14	0	6	6	0	8	11
4	8	15	6	12	8	6	2

$^*\Delta$

0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1

$W = \gamma^{\nu} = \lambda^{\nu} \oplus \delta$

1	1	1	1	0	0	1	1
1	0	1	0	0	1	1	0
1	0	1	0	0	1	1	0
0	0	0	0	1	1	0	0
1	1	0	0	1	1	1	1
1	0	0	1	1	0	1	0
1	0	0	1	1	0	1	0
0	0	1	1	0	0	0	0

$^*W = {}^*\gamma^{\nu} = {}^*\lambda^{\nu} \oplus {}^*\delta$

Fig. 1.8. Representation of Correction Codes Δ, *Δ and Hyperfilter Functions γ^{ν}, $*\gamma^{\nu}$ Corresponding to the Bases of Bundles T, *T.

$$D = D_0 + D_1 + D_2 + D_3 = |S_0| + |S_1| + |S_2| + |S_3|, \qquad (1.10)$$

$$^*D = {}^*D_0 + {}^*D_1 + {}^*D_2 + {}^*D_3 = |{}^*S_0| + |{}^*S_1| + |{}^*S_2| + |{}^*S_3| \qquad (1.11)$$

It is necessary to find one of the maximum coefficients $d(\omega^{\nu}) \in \max\{d(\omega) \in D\}$ and $^*d(^*\omega^{\nu}) \in \max\{{}^*d(^*\omega) \in {}^*D\}$ in the corresponding tables. Arguments ω^{ν}, $^*\omega^{\nu}$ at that will define filters $\lambda^{\nu} = \langle \omega^{\nu}, x \rangle$, $^*\lambda^{\nu} = \langle {}^*\omega^{\nu}, x \rangle$. The selected coefficients are highlighted in tables D и $^*D$. According to them $\lambda^{\nu} = x_4$, $^*\lambda^{\nu} = x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5$. The corresponding numbers 6 and 9 in tables Δ and $^*\Delta$ determine the correction codes $(\delta_3 \delta_2 \delta_1 \delta_0) = (0110)$ and $(^*\delta_3 {}^*\delta_2 {}^*\delta_1 {}^*\delta_0) = (1001)$. The block diagrams, according to the calculations, are presented in Fig. 1.3 by the variants a) and b).

Filters λ^{ν} which generate a signal at the input i of the neuron increase the excitation level of w neuron ν by a_i of the input weight. This happens or does not happen depending on the values of the argument x function f being implemented and filter function $\lambda^{\nu}(x)$, that is if $\lambda^{\nu}(x)=1$. It is clear that for a set of neuron inputs and a corresponding set of filters the distribution of levels w is made according to the values of argument x, and it can be characterized by the matrix $W=\{w(x)\}$. Implementation of a given LF $f(x)$ synthesized by the neurostructure is achieved if $w(x^1)\geq p$ and $w(x^0)<p$, where x^1, x^0 are single and zero constituents $x^1\in\{x:f(x)=1\}$, $x^0\in\{x:f(x)=0\}$. If this condition is not implemented, then it is necessary to continue synthesis adding neuron inputs and the corresponding filters. Therefore, in the process of building a neurostructure, it is necessary to analyze the distribution of constituents in matrix W.

For the variants of neurostructures under consideration selected by the first filters $\lambda^{\nu}, {}^{*}\lambda^{\nu}$ and weight values $a_1=1$, ${}^{*}a_1=1$, the distribution matrices have the form W, ${}^{*}W$ in Fig. 1.8. The corresponding to them constituent distributions of the original function $f(x)$ are displayed in Table 1.1, where k_1, k_0 represent the numbers of the single and zero constituents, $k_1=\left|\{x^1\}\right|$, $k_0=\left|\{x^0\}\right|$ as a variant for base T, and ${}^{*}k_1=\left|\{x^1\}\right|$, ${}^{*}k_0=\left|\{x^0\}\right|$ as a variant for base ${}^{*}T$, distributed according to the excitation levels w, ${}^{*}w\in\{0,1\}$.

Table 1.1. Distribution of Constituents by Levels of ***w*** and ****w*** Excitation of Neurons.

w	*0*	*1*		${}^{*}w$	*0*	*1*
k_1	*10*	*21*		${}^{*}k_1$	*9*	*22*
k_0	*22*	*11*		${}^{*}k_0$	*23*	*10*

The variant of the tangent bundle of space X, that uses additional nonlinear transformations decoder inputs provided a little bit better result of the constituent distribution. However, the efficiency of such transformations will increase along with the increase in the number of K layers of space X stratification and the increase in the dimension n of the implemented function f.

1.3. Building a Binomial Hyperfilter

A binary k-bit, $k = 2^t$, correction code $\delta = (\delta_{k-1}, \delta_{k-2}, \dots, \delta_0)$, attributed to the linear filter λ, by the single digit values, for example, $\delta_i = 1, \delta_j = 1$, $i, j = 0 \div (K-1)$, determines the action of inverse filter value $\overline{\lambda} = 1 - \lambda$ in the corresponding tangent layers X_i, X_j. Hence, code δ can be interpreted as a binary function $\delta(x)$, determined in space X according to the rule

$$\delta(x) = \begin{cases} 1, if\ (\delta_i = 1) \,\&\, (x \in X_i) \\ 0, if\ (\delta_i = 0) \,\&\, (x \in X_i) \end{cases}, \tag{1.12}$$

and the conjugate action of the filter λ and code δ shall be considered a hyperfilter $\gamma(x) = \lambda(x) \oplus \delta(x)$.

It is noted in the work [2] that the level of LF approximation $f(x)$ via neuron function $v(x)$ can be increased by input linearization of σ for m inputs of neuron v. Linearization of σ generates function $v^{\sigma}(x) = v(\sigma \otimes x)$, where σ is a binary $m \times n$ matrix, $\otimes$ is a multiplication sign according to mod2. The lines σ_i, $i = 1 \div m$ of matrix σ are defined sequentially according to arguments $\omega^i \neq 0$ of the dominant coefficients $s(\omega^i)$ of the spectrum S of function $|s(\omega^i)| = \max|S|$. The same arguments define linear filters $\lambda_1(x), \lambda_2(x), \cdots, \lambda_m(x)$ at the inputs of neuron v.

With the increase in the dimension and complexity of $f(x)$ the efficiency of application of only filters $\lambda_i(x)$ decreases and this makes it difficult to solve the approximation task. At the same time, relying on the neuron linearization, additional methods using spectral representations of LF can be applied.

Firstly, as described above, it is necessary to carry out hype linearization using the tangent bundle of space X. In case of its software implementation, all decoder output connections DC with inputs of linear filters $\lambda_1(x), \lambda_2(x), \cdots, \lambda_m(x)$ are represented by a binary matrix η of dimension $K \times m$, where K is the number of decoder outputs. The calculation of the value of the set of hyperfilters is determined by the expression $\gamma = \sigma \otimes x \oplus \eta(\sigma_{DC} \otimes x)$, where $\sigma_{DC} \otimes x$ represents the number of the line in matrix η.

Secondly, you should use local, linear hyperfilters $\mu(x)$ of multiplicity k of the type $\mu(x) = \prod_{j=1}^{k} \gamma^{j}(x)$, moreover, the multiplicity is not necessarily be the same. As a result, we shall get a hyper linearized neural function $v^{\mu}(x) = v(\mu^{1}(x), \mu^{2}(x), \cdots, \mu^{m}(x))$, that will approximate a given function $f(x)$ more efficiently. Here appears an opportunity to accurately implement $f(x)$ using the binary sum of a simple series of similar hyper linearized neurons $f(x) = v_{1}^{\mu\,1}(x) \oplus v_{2}^{\mu\,2}(x) \oplus \cdots \oplus v_{d}^{\mu\,d}(x)$. But, in this case, with a further increase in the dimension of complex functions $f(x)$, the corresponding rows are significantly lengthened.

The matter is that every subsequent series $(i+1)$ neurofunction $v_{i+1}^{\mu\,i+1}(x)$ must approximate the residual function $f_{i+1}(x) = f(x) \oplus v_{1}^{\mu 1}(x) \oplus v_{2}^{\mu 2}(x) \oplus \cdots \oplus v_{i}^{\mu\,i}(x)$ with an ever decreasing rank as the serial number $(i+1)$ increases. As a result, it turns out that it is necessary to use an increasing number of the local filters μ^{i+1} with increasing multiplicity and that results in a significant increase in the total number of hyperfilters $\gamma(x)$ that are used.

Therefore, thirdly, localizing binomial combinations of hyperfilters of the kind $\phi(x) = \prod_{j=1}^{k} \beta^{j}(x) = \prod_{j=1}^{k} (\gamma^{j1}(x) \vee \gamma^{j2}(x))$ should be used. Such a binomial hyperfilter β of multiplicity k contains $2k$ of simple hyperfilters γ, being equivalent at that to the sum 2^{k} of the local hyperfilters μ of the multiplicity k, which contain $k \cdot 2^{k}$ simple hyperfilters γ of the multiplicity 1.

In order to implement this method it is necessary to carry out the calculation of elementary binomials $\beta^{j}(x) = \gamma^{j1}(x) \vee \gamma^{j2}(x)$ for a number of functions $f'(x)$, that are partially defined fragments of the original function f in the areas $X' \subset X$. These actions for each elementary binomial are performed in the following order:

- calculation of the spectral density D' of function f' according to base T;
- choice $d'(\omega) = \max\{D'\}$ and a definition of hyperfilter $\gamma^{j1}(x),$ according to the value of argument ω of the coefficient $d'(\omega);$
- definition of the region $X'' = \{x : (x \in X') \,\&\, (\gamma^1(x) = 0)\}$ and the corresponding to it fragment $f''(x),\ x \in X''$ to the function $f(x),$ in this case $f'' \subset f';$
- calculation of the spectral density D'' of the function f'' according to base T;
- choice $d''(\omega) = \max\{D''\}$ and the determination of hyperfilter $\gamma^{j2}(x),$ in accordance with the value of the argument ω of the coefficient $d''(\omega).$

If it turns out that $f'' = 0,$ then area $X'' = \{x : (x \in X') \,\&\, (\gamma^1(x) = 1)\}$ and the corresponding fragment $f'' \subset f'$ are redefined. In this case, $\beta^j(x) = \gamma^{j1}(x)$ and further on the calculation of binomial $\beta^{j+1}(x)$ for the fragment $f''(x)$ of the function f' is carried out on $X^{(j)} = \{x : (x \in X') \,\&\, (\gamma^1(x) \vee \gamma^2(x) = 1)\},$ reduced to the region approximated by a binomial $\beta^j(x).$

Thus, the calculation of each of the subsequent elementary binomials $\beta^{j+1}(x)$ in the hyperfilter β is carried out in successively contracting areas $X^{(j)} = \bigcap_{r=1}^{j} X^{(r)}$ and the corresponding to them fragments of the function $f^{(j)} \subset f'.$

Let us consider an example of constructing a binomial hyperfilter ϕ of multiplicity 2, $\phi = \beta_1 \cdot \beta_2,$ according to base ${}^*T$ for the above mentioned LF $f(x_1, x_2, \ldots, x_6).$ In the first binomial $\beta_1 = \gamma^{11} \vee \gamma^{12}$ the first term of binomial, hyperfilter $\gamma^{11}(x) = \lambda^{11}(x) \oplus \delta^{11}(x),$ has already been calculated. Its components $\langle \lambda^{11}, \delta^{11} \rangle$ are determined by a pair of numbers $\langle 45, 9 \rangle.$ The first number indicates the number of the spectral argument ω value in Karnaugh ${}^*D$ table in Fig. 1.7, and the second, in the corresponding position 45 in the table ${}^*\Delta,$ in Fig. 1.8, represents a binary code, the single digits of which indicate the connection of the linear filter λ^{11} with the corresponding output of the decoder DC. In order to define hyperfilter γ^{12} a fragmentary function is highlighted

$$g(x) = \begin{cases} f(x), \ if \ \ \gamma^{11}(x) = 0 \\ "-", \ \ if \ \ \gamma^{11}(x) = 1 \end{cases}, \qquad (1.13)$$

and is presented in Fig. 1.9 in table g. The corresponding tables of correction connections δ_g and the spectral density D_g are calculated according to rules (1.7) and (1.10). Wherein spectra of ternary functions $h_0 \div h_3$ representing in correlation with g, its tangent fragmentary functions $g_0 \div g_3$ are calculated according to rule (1.5), where the value of $r = 5$. The choice of one of the maximum values in table D_g and the corresponding to it value in table δ_g is determined by hyperfilter γ^{12} by a couple $\langle 18, 7 \rangle$.

–	–	–	–	0	1	–	–
–	0	–	0	0	–	–	1
–	0	–	0	1	–	–	0
0	0	1	0	–	–	0	0
–	–	1	0	–	–	–	–
–	1	0	–	–	0	–	0
–	1	0	–	–	0	–	0
0	0	–	–	0	1	1	0

g

0	17.24	8.86	11.05	5.33	17.24	11.52	16.38
11.05	16.38	***23.05***	***23.05***	11.05	8.38	17.71	17.71
11.05	11.05	***23.05***	17.71	11.05	5.71	17.71	15.05
8.86	17.71	11.05	11.52	16.86	17.71	16.38	16.86
17.24	5.33	16.38	11.52	17.24	0	11.05	8.86
8.38	11.05	17.71	17.71	16.38	11.05	***23.05***	***23.05***
5.71	11.05	15.05	17.71	11.05	11.05	17.71	***23.05***
17.71	16.86	16.86	16.38	17.71	8.86	11.52	11.05

Dg

Δg

6	0	14	8	3	12	2	5
5	0	*7*	***8***	0	4	10	12
9	4	***2***	5	5	1	15	9
3	5	9	15	7	0	13	2
5	10	12	11	9	15	1	7
13	9	5	3	9	12	***1***	***14***
8	12	0	6	13	0	12	***11***
9	14	11	4	12	10	6	0

2	1	1	1	0	1	2	2
1	0	2	0	1	1	1	1
1	1	1	1	0	1	2	0
1	0	1	0	1	2	0	1
2	1	1	1	1	2	2	2
1	1	0	2	2	0	1	0
1	1	0	1	2	0	2	0
0	0	2	2	1	1	1	1

Wβ1

Fig. 1.9. Representation by Karnaugh Tables of the Function g, its Spectral Density D_g, Correction Δ_g and the Arithmetic Sum of the Binomial Filter $W_{\beta 1}$.

Consequently, $\lambda^{12}(x) = x_2 \oplus x_3 \oplus x_6$, $\delta^{12} = (0111)$, and according to the rule for constructing a decoder

$\delta^{12}(x)=\bar{\nu}_1(x)\cdot\bar{\nu}_2(x)\vee\bar{\nu}_1(x)\cdot\nu_2(x)\vee\nu_1(x)\cdot\bar{\nu}_2(x)$. Then $\gamma^{12}(x)=\lambda^{12}(x)\oplus\delta^{12}(x)$. In table $W_{\beta 1}$ the arithmetic sum of $\gamma^{11}(x)+\gamma^{12}(x)$ is presented, and in table β_1, in Fig. 1.12, is represented by a binomial $\beta_1(x)=\gamma^{11}(x)\vee\gamma^{12}(x)$. The distribution of constituents x^1 and x^0 of the function $f(x)$ by weight values $W_{\beta 1}$ is shown in Table 1.2a, where $k_1=\left|\{x^1\}\right|, k_0=\left|\{x^0\}\right|$ is the corresponding number of constituents.

Table 1.2. Distribution of Constituents According to Hyperfilter Signal Values.

***W**β1*	***0***	***1***	***2***
k1	*1*	*19*	*11*
k0	*15*	*13*	*5*

a)

β1	***1***
k1	*30*
k0	*18*

b)

γ21	***0***	***1***
k1	*10*	*20*
k0	*14*	*4*

c)

***W**β2*	***0***	***1***	***2***
k1	*1*	*21*	*8*
k0	*10*	*5*	*3*

d)

φ1	***0***	***1***
k1	*2*	*29*
k0	*25*	*8*

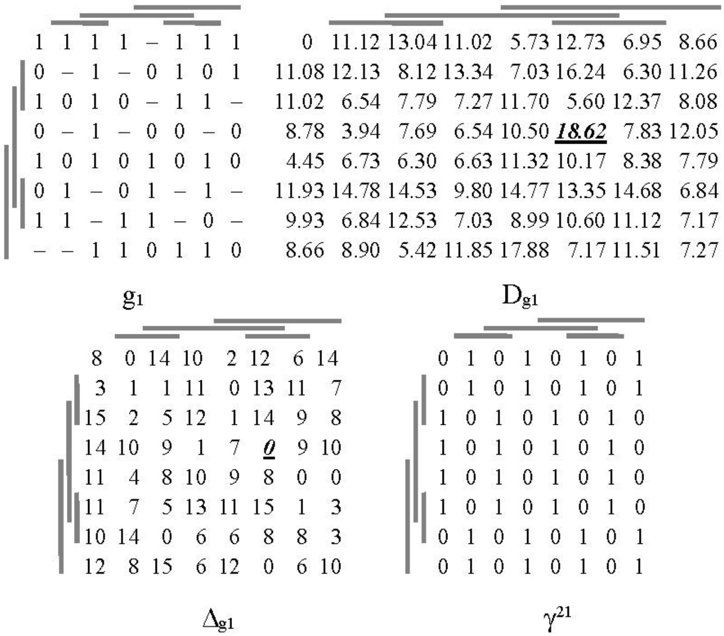
e)

The calculation of the second binomial $\beta_2=\gamma^{21}\vee\gamma^{22}$ begins by isolating a fragmentary function

$$g_1(x)=\begin{cases} f(x),\ if\ \ \beta_1(x)=1 \\ "-",\ \ if\ \ \beta_1(x)=0 \end{cases} \qquad (1.14)$$

The ratio of this function constituents is shown in the Table 1.2b. In general, the same computational operations as in the previous case are repeated. According to function g_1 and base ${}^{*}T$ tangent fragment functions $g_{10}\div g_{13}$ are defined and also the corresponding ternary representatives $h_{10}\div h_{13}$, their spectra $S_{10}\div S_{13}$, spectral densities $D_{10}\div D_{13}$, total spectral density D_{g1} and corrective action matrix Δ_{g1}.

The results of the operations are given in tables g_1, D_{g1}, Δ_{g1} in Fig. 1.10.



1	1	1	1	–	1	1	1
0	–	1	–	0	1	0	1
1	0	1	0	–	1	1	–
0	–	1	–	0	0	–	0
1	0	1	0	1	0	1	0
0	1	–	0	1	–	1	–
1	1	–	1	1	–	0	–
–	–	1	1	0	1	1	0

g_1

0	11.12	13.04	11.02	5.73	12.73	6.95	8.66
11.08	12.13	8.12	13.34	7.03	16.24	6.30	11.26
11.02	6.54	7.79	7.27	11.70	5.60	12.37	8.08
8.78	3.94	7.69	6.54	10.50	***18.62***	7.83	12.05
4.45	6.73	6.30	6.63	11.32	10.17	8.38	7.79
11.93	14.78	14.53	9.80	14.77	13.35	14.68	6.84
9.93	6.84	12.53	7.03	8.99	10.60	11.12	7.17
8.66	8.90	5.42	11.85	17.88	7.17	11.51	7.27

D_{g1}

8	0	14	10	2	12	6	14
3	1	1	11	0	13	11	7
15	2	5	12	1	14	9	8
14	10	9	1	7	***0***	9	10
11	4	8	10	9	8	0	0
11	7	5	13	11	15	1	3
10	14	0	6	6	8	8	3
12	8	15	6	12	0	6	10

Δ_{g1}

0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0
0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1

γ^{21}

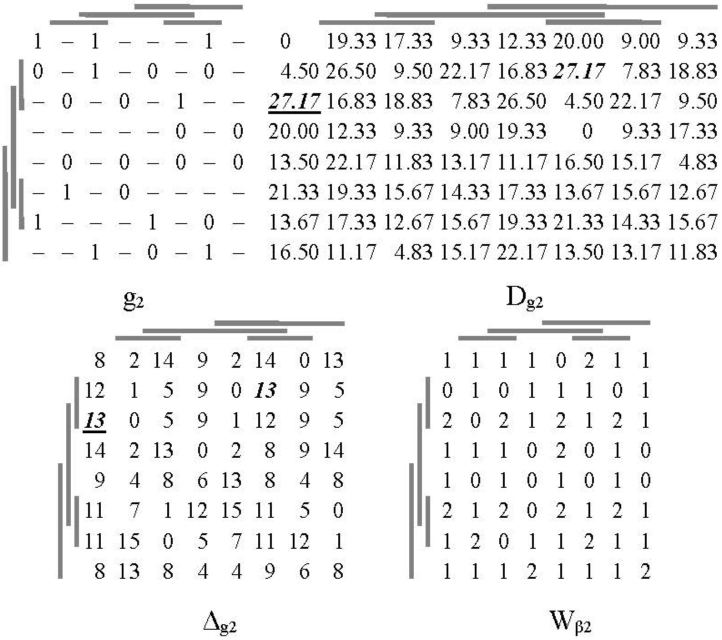
Fig. 1.10. Representation by Karnaugh Tables of the Function g_1, its Spectral Density D_{g1}, Correction Δ_{g1} and the Filter γ^{21}.

The function of binomial hyperfilter ϕ_1, defined by conjunction $\phi_1(x) = \beta_1(x) \& \beta_2(x)$, is shown in table $\phi_1 = \beta_1 \circ \beta_2$. The distribution of constituents x^1 and x^0 of the original function $f(x)$ in areas $X^0 = \{x : \phi_1(x) = 0\}$ and $X^1 = \{x : \phi_1(x) = 1\}$ is indicated in Table 1.2e. On the basis of them the defining pair $\langle 44, 0 \rangle$ for the hyperfilter γ^{21} is found in a similar way. The function of filter $\gamma^{21}(x)$ is presented in the table $W\gamma^{21}$. It corresponds to the linear function $\lambda^{21}(x) = x_1 \oplus x_2 \oplus x_3 \oplus x_5$ and the corrective action $\delta^{21}(x) = 0$.

The calculation of the second hyperfilter γ^{22} in the binomial β_2 is carried out using a fragment function

$$g_2(x) = \begin{cases} g_1(x), \ if \ \gamma^{21}(x) = 0 \\ "-", \quad if \ \gamma^{21}(x) = 1 \end{cases} \tag{1.15}$$

The ratio of constituents of the function $g_2(x)$ is defined, Table 1.2c, in column "0". Having carried out the previous procedure of calculations, based on the results in tables D_{g2}, δ_{g2}, Fig. 1.11, we find the defining pair $\langle 3, 13 \rangle$ for γ^{22}. Thus, $\gamma^{22}(x)$ is the composition of the linear function $\lambda^{22}(x) = x_5 \oplus x_6$ and the corrective action of $\delta^{22} = (1101)$. Table $W_{\beta 2}$ presents the arithmetic sum $\gamma^{21}(x) + \gamma^{22}(x)$, and in the table β_2, Fig. 1.12, binomial function $\beta_2(x) = \gamma^{21}(x) \vee \gamma^{22}(x)$ is presented. The distribution of constituents x^1 and x^0 of the function $f(x)$ according to weight values $W_{\beta 2}$ is shown in Table 1.2d.



1	–	1	–	–	–	1	–
0	–	1	–	0	–	0	–
–	0	–	0	–	1	–	–
–	–	–	–	–	0	–	0
–	0	–	0	–	0	–	0
–	1	–	0	–	–	–	–
1	–	–	–	1	–	0	–
–	–	1	–	0	–	1	–

g$_2$

0	19.33	17.33	9.33	12.33	20.00	9.00	9.33
4.50	26.50	9.50	22.17	16.83	***27.17***	7.83	18.83
27.17	16.83	18.83	7.83	26.50	4.50	22.17	9.50
20.00	12.33	9.33	9.00	19.33	0	9.33	17.33
13.50	22.17	11.83	13.17	11.17	16.50	15.17	4.83
21.33	19.33	15.67	14.33	17.33	13.67	15.67	12.67
13.67	17.33	12.67	15.67	19.33	21.33	14.33	15.67
16.50	11.17	4.83	15.17	22.17	13.50	13.17	11.83

D$_{g2}$

8	2	14	9	2	14	0	13
12	1	5	9	0	***13***	9	5
13	0	5	9	1	12	9	5
14	2	13	0	2	8	9	14
9	4	8	6	13	8	4	8
11	7	1	12	15	11	5	0
11	15	0	5	7	11	12	1
8	13	8	4	4	9	6	8

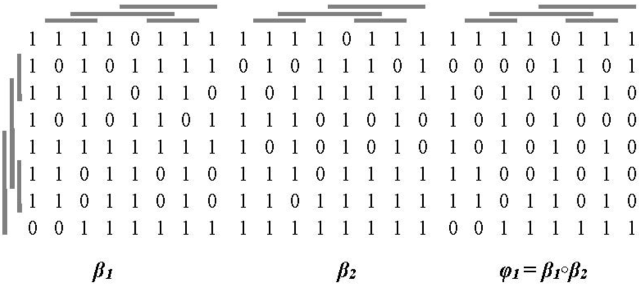
Δ$_{g2}$

1	1	1	1	0	2	1	1
0	1	0	1	1	1	0	1
2	0	2	1	2	1	2	1
1	1	1	0	2	0	1	0
1	0	1	0	1	0	1	0
2	1	2	0	2	1	2	1
1	2	0	1	1	2	1	1
1	1	1	2	1	1	1	2

W$_{\beta 2}$

Fig. 1.11. Representation by Karnaugh Tables of the Function g$_2$, its Spectral Density D$_{g2}$, Correction Δ$_{g2}$ and the Sum of the Binomial Filter W$_{\beta 2}$.

The function of binomial hyperfilter β, defined by conjunction $\beta(x) = \beta_1(x) \& \beta_2(x)$, is shown in table $\beta = \beta_1 \circ \beta_2$. In Table 1.2e indicated. The distribution of constituents x^1 and x^0 of the original function $f(x)$ in areas $X^0 = \{x : \beta(x) = 0\}$ and $X^1 = \{x : \beta(x) = 1\}$.



1	1	1	1	0	1	1	1
1	0	1	0	1	1	1	1
1	1	1	1	0	1	1	0
1	0	1	0	1	1	0	1
1	1	1	1	1	1	1	1
1	1	0	1	1	0	1	0
1	1	0	1	1	0	1	0
0	0	1	1	1	1	1	1

β_1

1	1	1	1	0	1	1	1
0	1	0	1	1	1	0	1
1	0	1	1	1	1	1	1
1	1	1	0	1	0	1	0
1	0	1	0	1	0	1	0
1	1	1	0	1	1	1	1
1	1	0	1	1	1	1	1
1	1	1	1	1	1	1	1

β_2

1	1	1	1	0	1	1	1
0	0	0	0	1	1	0	1
1	0	1	1	0	1	1	0
1	0	1	0	1	0	0	0
1	0	1	0	1	0	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	0
0	0	1	1	1	1	1	1

$\varphi_1 = \beta_1 \circ \beta_2$

Fig. 1.12. Representation by Tables of Binomial Hyperfilters β1, β2, φ1.

Trim $\theta_{\phi 1} = \log_2(k_1 / k_0)$ of function fragment $f(x)$ in area X^1, approximated by filter ϕ_1, amounts to value $\theta_{\phi 1} = \log_2(29/8) = 1.858.$ For comparison, it should be noted that the trim of the original function $f(x)$ has the value $\theta_f = -0.0902.$

1.4. Building a Structure for Cascade Connection of Modules with Bipolar Neurons

Realization of the given LF $f(x)$ synthesized by a neuron is achieved, if $w(x^1) \geq p$ and $w(x^0) < p,$ where x^1, x^0 are unit and zero constituents $x^1 \in \{x : f(x) = 1\}$, $x^0 \in \{x : f(x) = 0\}$. If this condition is not met, then the process of synthesis shall be continued by adding inputs to the neuron and also the corresponding filters. Therefore, in the process of building a neurostructure, it is necessary to analyze to analyze the distribution of constituents in matrix W.

Let in the process of neuron's synthesis m inputs with equal weights $a_i = 1,\ i = 1 \div m$ and corresponding binomial hyper linearized filters (BF) $\phi_i(x) = \beta_{i1} \cdot \beta_{i2} \cdot \ldots \cdot \beta_{ik},\ i = 1 \div m,$ be defined. As a result, there appears a certain distribution of unit x^1 and zero x^0 constituents of the implemented function $f(x)$ according to the levels of excitation $w = 0 \div m$ of the neuron. The following $(m+1)$ additional input at BF and $a_{m+1} = 1$ are intended to increase by one level $w(x^1)$ as regards to the largest number k_1 of unit constituents x^1, and allow such an increase to a minority k_0 of zero constituents x^0. But to implement such a redistribution constantly and efficiently by adding inputs to ensure a great difference of $k_1 - k_0$, is complicated due to the presence of a large number of anti-self-dual pairs $f(x^1) = f(x^1 \oplus \omega)$ and $f(x^0) = f(x^0 \oplus \omega)$. Ultimately, for this reason, a simple synthesis based only on the increase of the number of neuron inputs with binomial filters, leads to high costs relative to the total number of hyperfilters $\gamma(x)$.

In connection with this, it is advisable to carry out synthesis in stages, operating with individual levels of neuron w excitation. In this case, it is necessary to filter out the emerging monochrome fragments of definition X domain, thus, simplifying the task of synthesis of subsequent binomial filters. In the distribution of constituents, formed under the influence of signal sat m inputs of the neuron, the obvious convenient objects for analysis are polar fragments of the domain of definition $X^0 = \{x : w(x) = 0\}$ and $X^m = \{x : w(x) = m\}$, of the corresponding levels of excitation $w = 0$ and $w = m.$

The next $(m+1)$ input of the neuron with the weight a_{m+1} and BF $\phi_{m+1}(x)$ for the majority of unit constituents $x^1 \in X^0$ and the majority $x^1 \in X^m$ must ensure the level of excitation increases up to values $w(x^1) = 0 + a_{m+1}$ and $w(x^1) = m + a_{m+1}$ correspondingly, wherein, for the majority of constituents of zero $x^0 \in X^0$ and the majority $x^0 \in X^m$ the corresponding levels must remain unchanged $w(x^0) = 0$ and $w(x^0) = m.$ As a result of adding $(m+1)$ input, the redistribution of constituents x^1, x^0 of the function $f(x)$ along all fragments where $X^i = \{x : w(x) = i\},\ i = 0 \div (m+1).$ But at a certain next stage, when the number of inputs of neuron is $t > m,$ the fragments X^0, X^t will become monochrome. Fragment X^0 will not contain unit constituents $x^1,$ and fragment X^t is zero constituents x^0 : $\forall x^1 \{x^1 \notin X^0\}, \forall x^0 \{x^0 \notin X^t\}.$

Thus, it can be considered that, for example, bipolar neuron with t inputs, weights $a_i = 1,$ binomial filters at the inputs $\phi_i,\ i - 1 \div t,$ having the value of thresholds $p_1 = 1$ and $p_2 = t,$ ensures error-free implementation of the specified LF $f(x)$ on fragments X^0 and X^t. Both in software and hardware (Fig. 1.13) implementation, at the next level of neurostructure synthesis, it is permitted to operate with the derivative of a partial function

$$h(x)=\begin{cases} f(x), \ if \ x\in X_h \\ "-", \ \ if \ x\notin X_h \end{cases}, \qquad (1.16)$$

where $X_h = X/(X^0 \cup X^t)$. In this case, for $h(x)$ a similar choice of a bipolar neuron and the corresponding filters $\phi_i,\ i=1\div q$ can also be made. It is feasible using the same algorithm of highlighting polar monochrome fragments X_h^0, X_h^q and defining the next LF

$$h'(x)=\begin{cases} h(x), \ if \ x\in X'_h \\ "-", \ \ if \ x\notin X'_h \end{cases}, \qquad (1.17)$$

where $X'_h = X_h/(X_h^0 \cup X_h^q)$. It is clear that the sequence of reductions of the domains of partial LF functions definition is finite, $X \supset X_h \supset X'_h \supset \ldots \supset X_h^{(r)} \supset \varnothing$, and such iteration of actions leads to the creation of a cascade structure from modules containing a bipolar neuron with binomial filters. The cascade of such modules accurately implements the original function. And the number of neuron inputs and the corresponding number of filters in each subsequent module of the cascade will be less than the corresponding numbers in the structure of the previous module.

In the algorithm of every module's synthesis, spectral calculations of the first filter ϕ_1 are made according to the rule (1) in relation to the partial LF, corresponding to the module. But in order to calculate ϕ_2 and subsequent filters ϕ_{i+1} it is necessary to deal with two partial functions

$$h^0(x) = {}^{h(x)}\!\big/_{x\in X^0},\ h^i(x) = {}^{h(x)}\!\big/_{x\in X^i},\ i=2\div t \qquad (1.18)$$

Their formal unification into one function $\hat{h}(x) = h^0(x) + h^i(x)$ will not provide an efficient spectral analysis because of differing trims of the functions h^0 and h^i. The number of constituents x^1 as part of h^0, that is, constituent $x^1 \in X^0$, will be less or significantly less than the corresponding number x^1 in part h^i and the number of constituent x^0 in part h^0. That is why, in order to achieve monochrome of fragments X^0 and X^i "weightiness" of the constituents x^1 of the first part h^0, $x^1 \in X^0$ is higher than "weightiness" constituents $x^1 \in X^i$ and $x^0 \in X^0$.

The same remark also refers to constituents $x^0 \in X^i$ and related to them $x^0 \in X^0$, $x^1 \in X^i$. Consequently, in order to ensure the efficiency of the spectral calculations of filters ϕ_{i+1}, $i = 1 \div (t-1)$, it is necessary to operate such functions as

$$h^{\oplus}(x) = h^0(x) + \overline{h}^i(x), \tag{1.19}$$

$$\overline{h}^i(x) = \begin{cases} 0, \ if \ h^i(x) = 1 \\ 1, \ if \ h^i(x) = 0 \\ "-", \ if \ h^i(x) = "-" \end{cases}, \tag{1.20}$$

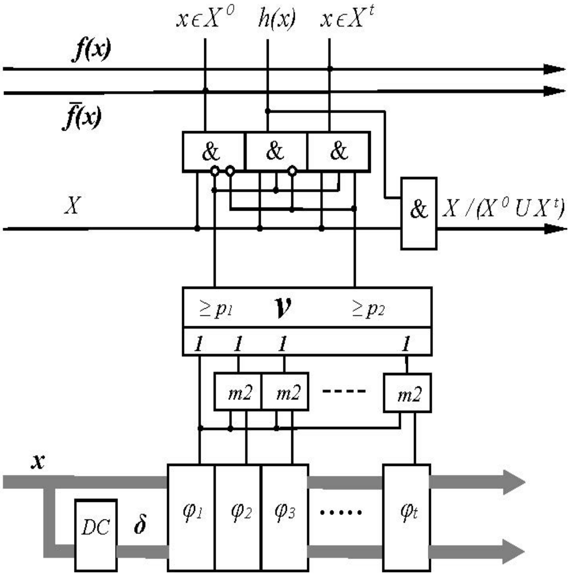


Fig. 1.13. Bipolar Neuron Module with Binomial Hyperfilters at the Inputs.

which unite “weighty“ constituents belonging to polar fragments X^0 and X^i into constituents x^1 of function $h^{\oplus}(x)$. It is clear that after getting the result, filter ϕ_{i+1}, in order to form the correct redistribution of the constituents on the fragments X^0, X^{i+1}, it is necessary to save the action of filter ϕ_{i+1} on fragment X^0 and invert its action on fragment X^i. Taking into consideration the action of the first filter, –

$\phi_1(x)=0,\ if\ x\in X^0$ and $\phi_1(x)=1,\ if\ x\in X^i$, it is necessary to give a signal $\phi_1(x)\oplus\phi_{i+1}(x)$ on (i+1) input of the neuron. The corresponding hardware solution is presented in Fig. 1.13.

Let us illustrate the described rules for the synthesis of the cascade structure, using the example of function $f(x)$ given above. After the previously formed binomial filter ϕ_1 of multiplicity 2, $\phi_1=\beta_1\beta_2$, it is necessary to start calculating the next filter ϕ_2. Having this aim in view, function $h^{\oplus}$ is determined, which according to Table 1.2e must contain 10 constituents x^1 and 54 constituents x^0. In the order described above, taking into account the selected tangential dispersion of space X the spectral density D_h is calculated and also the table of corrective effect δ_h of function $h^{\oplus}$. The maximum element is selected from D_h and the corresponding to it corrective effect is selected from table δ_h. Thus, a pair $\langle 41,11\rangle$ is found and this pair defines the first hyperfilter γ^{11} of the first binomial β_1 of the second filter ϕ_2. In the codes of the spectral argument ω and correction δ, a vector pair $\langle\omega,\delta\rangle=\langle(111000),(1101)\rangle$, corresponds to it and it defines a linear function $x_1\oplus x_2\oplus x_3$ with connection of decoder outputs to it with the numbers "0","2","3". The result of the action of hyperfilter γ^{11} on the distribution of the constituents of function $h^{\oplus}$ is presented in the Table 1.3a. As the inverse region X'' of filter γ^{11} does not contain any constituents x^1 of function $h^{\oplus}$ (column "0" contains only 28 constituents x^0), so, it means that in filter ϕ_2 the first binomial $\beta_1=\gamma^{11}$. Consequently, in order to define binomial β_2, it is necessary to analyse partial function $h_\gamma^*=\gamma^{11}\cdot h^{\oplus}$, which contains 10 constituents x^1 and 26 constituents x^0 of function $h^{\oplus}$. The corresponding calculations of the spectral density and corrective actions define the formative pair $\langle 30,10\rangle$ for the first hyperfilter γ^{21} of binomial β_2 that defines vectors $\langle\omega,\delta\rangle=\langle(010111),(1010)\rangle$ and a linear function $\lambda_{21}=x_2\oplus x_4\oplus x_5\oplus x_6$, that is corrected by output signals 1 and 3 of decoder *DC*. The distribution of constituents for a partial function h_γ^* under filter γ^{21} action is presented in the Table 1.3b. One constituent x^1 and 17 constituents x^0 of this function ends up in the area X'' of filter γ^{21}, determining a partial function $\overline{\gamma^{21}} \cdot \gamma^{11} \cdot h^{\oplus}$. The corresponding spectral calculations of the density and correction allow to single out the second hyperfilter γ^{22} of binomial β_2. A pair $\langle 4, 5 \rangle$ or $\langle \omega, \delta \rangle = \langle (000010), (0101) \rangle$, $\lambda_{22}(x) = x_5$ are defined and the correction with outputs 0 and 2 of decoder DC. The action of filter γ^{22} on the product of $\overline{\gamma^{21}} \cdot \gamma^{11} \cdot h^{\oplus}$ is presented in Table 1.3c.

Table 1.3. Distribution of Constituents According to Hyperfilter Signal.

γ^{11}	*0*	*1*
k_1	—	*10*
k_0	*28*	*26*

a)

γ^{21}	*0*	*1*
k_1	*1*	*9*
k_0	*17*	*9*

b)

γ^{22}	*0*	*1*
k_1	—	*1*
k_0	*13*	*4*

c)

φ_2	*0*	*1*
k_1	—	*10*
k_0	*41*	*13*

d)

w	*0*	*1*	*2*
k_1	—	*2*	*29*
k_0	*19*	*14*	—

e)

Binomial filter $\phi_2 = \beta_1 \cdot \beta_2 = \gamma^{11} \cdot (\gamma^{21} \vee \gamma^{22})$. Its action on function $h^{\oplus}$ is presented in Table 1.3d. The function of the signal on the second input of the neuron is formed by the inverse dual transformation $\phi_1 \oplus \phi_2$. Distribution of constituents of function $f(x)$ according to the levels of excitation w, which are formed under the influence of the first two inputs of the neuron is shown in Table.1.3e. For this distribution, the condition for the formation of the first module of cascade, $\forall x^1 \left\{ x^1 \notin X^0 \right\}$, $\forall x^0 \left\{ x^0 \notin X^t = X^2 \right\}$ is satisfied.

The second module is intended to implement partial function $h(x)$, defined on fragment

$$X^1 = \left\{ x : w(x) = 1 \right\},\ h(x) = \begin{cases} f(x), & \textit{if } x \in X^1 \\ "-", & \textit{if } x \notin X^1 \end{cases} \qquad (1.21)$$

The function contains 9 constituents x^1 and 14 constituents x^0. The spectral analysis of density D_h and corrective matrix δ_h identifies the pair $\langle 14, 15 \rangle$ in accordance with the first hyperfilter γ^{11} binomial β_1 in filter ϕ_1. Next, vectors $\langle \omega, \delta \rangle = \langle (001111), (1111) \rangle$, linear function $\lambda_{11} = x_3 \oplus x_4 \oplus x_5 \oplus x_6$ are formed, and the correction on the outputs "0","1","2","3" of the decoder DC is performed. The distribution of constituents under the influence of γ^{11} is presented in Table 1.4a.

Table 1.4. Distribution of Constituents according to Hyperfilter Signal Values.

γ^{11}	***0***	***1***
k_1	*2*	*7*
k_0	*11*	*3*

a)

γ^{12}	***0***	***1***
k_1	—	*2*
k_0	*8*	*3*

b)

γ^{Σ}	***0***	***1***	***2***
k_1	—	*6*	*3*
k_0	*8*	*5*	*1*

c)

β_1	***0***	***1***
k_1	—	*9*
k_0	*8*	*6*

d)

γ^{21}	***0***	***1***
k_1	*2*	*7*
k_0	*6*	—

e)

Fragment $X'' = \{x : (x \in X^1) \,\&\, (\gamma^{11}(x) = 0)\}$ contains 2 constituents x^1 and 11 constituents x^0, that is why, further on it is necessary to analyze function $h_\gamma = \overline{\gamma^{11}} \cdot h$ and calculate filter γ^{12}. As a result, a pair $\langle 9, 0 \rangle$, vector analogue $\langle \omega, \delta \rangle = \langle (001000), (0000) \rangle$, linear function $\lambda_{12} = x_3$ and no correction from DC outputs are determined. The effect of filter γ^{12} on function h_γ is presented in Table 1.4b. The action on h_γ of the arithmetical sum of filter $\gamma^{\Sigma} = \gamma^{11} + \gamma^{12}$ signals is shown in Table 1.4c. The action of the first binomial $\beta_1 = \gamma^{11} \vee \gamma^{12}$ is presented in Table 1.4d. Next, to determine binomial β_2, the partial function $h_\beta = \beta_1 \cdot h$, of the number of ones and zeros of the constituents is presented in column “1”of the Table 1.4d. Spectral analysis of function h_β reveals the first hyperfilter γ^{21} of binomial β_2, to which a pair $\langle 64, 9 \rangle$ corresponds and also a pair of vectors $\langle \omega, \delta \rangle = \langle (100100), (1001) \rangle$, a linear function $\lambda_{21} = x_1 \oplus x_4$ and the correction from the outputs "0" and "3" of the decoder. The action of γ^{21} on function h_β is indicated in Table 1.4e. The second hyperfilter γ^{22} of binomial β_2 is defined in accordance with the product $\overline{\gamma^{21}} \cdot h_\beta$. A pair $\langle 37, 0 \rangle$, a couple of vectors $\langle \omega, \delta \rangle = \langle (110110), (0000) \rangle$, a function $\lambda_{21} = x_1 \oplus x_2 \oplus x_4 \oplus x_5$ and lack of correction from decoder outputs correspond to it. The action of an arithmetic sum of hyperfilters $\gamma^{\Sigma} = \gamma^{21} + \gamma^{22}$ on partial function h_{β} is indicated in Table 1.5 on the left, and the action of binomial filter $\phi_1 = \beta_1 \cdot \beta_2 = (\gamma^{11} \vee \gamma^{12}) \cdot (\gamma^{21} \vee \gamma^{22})$ of the second module on module h function is shown in Table 1.5 on the right.

Table 1.5. Distribution of Constituents According to Filters of the Second Module.

γ^{Σ}	*0*	*1*	*2*		φ_1	*0*	*1*
k_1	—	5	4		k_1	—	9
k_0	6	—	—		k_0	14	—

Thus, the specified LF $f(x)$ is implemented by cascading connection of two modules with binomial filters of multiplicity 2. Due to the small dimension of the function in the second module, only one binomial filter was required. The implementation diagram is shown in Fig. 1.14 with a slight simplification.

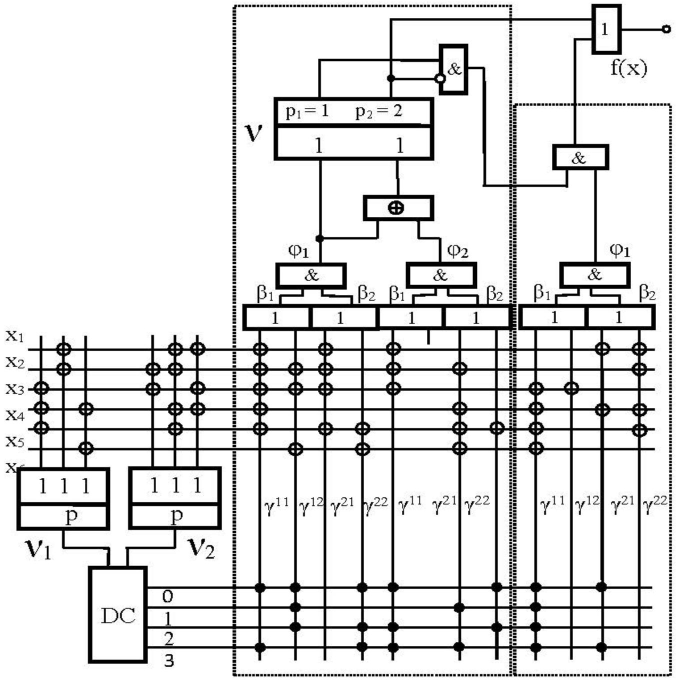


Fig. 1.14. The Structural Scheme of a Given Function *f*(*x*) Realization by the Cascade Connection of Modules.

1.5. Cascade Neural Structures that Implement High-dimension Functions

With the increase of the dimensions of the implemented function, the costs of forming linear signal transformations at the inputs of the decoder are considerably lower the corresponding total costs at the inputs of all neurons. Therefore, software and technical implementation costs are determined, in general, by the total number G of hyperfilters $\gamma(x)$, involved in cascade modules. The entire set of hyperfilters is structured into two matrices. The first is matrix of linear transformations σ from adders modulo 2, and the second is matrix δ of connecting the decoder to the inputs forming linear functions. Matrix δ is similar to matrix "OR" in PAL. Corresponding dimensions of matrices are expressed by the numbers $G \times n$ and $G \times K$, where n is dimension of the implemented function, K is the number of decoder outputs. An alternative to such a realization is to represent LF as a matrix of RAM that has dimension 2^n. It is obvious that basing on general positions, it is advisable to estimate costs of a cascade structure by relative value

$$\varepsilon = \frac{G \cdot (n+K)}{2^n} \tag{1.22}$$

In the given example with function 6 of variables, there were needed $G = 11$ hyperfilters and $K = 4$ outputs of the decoder in order to solve the problem. Therefore, costs are expressed by the value $\varepsilon = 1.71875$. In the work [3] it is pointed out that as LF dimension increases, the costs of linearisation grow linearly. This statement is true if the number of linearized inputs of the structure is limited. Nevertheless, there is reason to hope that the use of dual function transformations at polar fragments X^0, X^t of areas of definition and sequential exclusion of these areas from the operations of synthesis will prevent number G from increasing too much, and the cost index ε will decrease.

In connection with this, it is necessary to pay attention to the multiplicity of applied binomial hyperfilters $\phi(x)$. Let synthesis of a certain module of a cascade be oriented on the partial LF $h(x)$. The numbers of single and zero constituents k_1^h and k_0^h, of function h determine its different $\theta^h = \log_2(k_1^h / k_0^h)$. Filter $\phi_m(x)$ with multiplicity k contains $2k$ hyperfilters $\gamma : \phi_m = \beta_1\beta_2\ldots\beta_k = (\gamma^{11} \vee \gamma^{12})(\gamma^{21} \vee \gamma^{22})\ldots(\gamma^{k1} \vee \gamma^{k2})$ and approximates fragment $X^{\phi} = \{x : \phi(x) = 1\}$ in the domain of definition of function *h*. On fragment X^{ϕ} of function *h*, there is trim $\theta^{h\phi} = \log_2(k_1^{h\phi} / k_0^{h\phi})$, that is defined by the numbers of its unit and zero constituents $k_1^{h\phi} = \left|\{x^1\}\right| : x^1 \in X^{\phi}$, $k_0^{h\phi} = \left|\{x^0\}\right| : x^0 \in X^{\phi}$. The process of filter ϕ_m synthesis can be continued by adding a new binomial β_{k+1} and receiving filter $\phi_m^{+} = \beta_1 \beta_2 \ldots \beta_{k+1}$ of multiplicity $(k+1)$. On the corresponding fragment $X^{\varphi+}$ function h has a trim $\theta^{h\phi+}$, which, naturally, becomes higher, $\Delta\theta^{\phi} = \theta^{h\phi+} - \theta^{h\phi} > 0$. But, if the increase $\Delta\theta^{\phi}$ is small, then the efficiency of filter ϕ_m^{+} application in relation to ϕ_m gets lower. It seems more rational to limit oneself to filter ϕ_m and move on in this module to the synthesis of the next filter ϕ_{m+1}. It is natural to indicate the criteria for choosing the rational multiplicity of the binomial filter.

In this research, the synthesis of binomial filter ϕ_{m+1} was completed when logarithmic conditions $\theta^{h\phi} > \theta^{h} - z \cdot \left|\theta^{m}\right|$ were met and where θ^{m} is a trim of function $h^{\oplus}$ was defined on the polar fragments X^0, X^m, z is a proportionality coefficient, taking into account the dimension and trim of the original function $f(x)$.

The effectiveness of the method for constructing cascade neural structures was checked on the basis of the results of the synthesis of circuits for chaotic LF fully or partially determined. Single and zero values were specified in the domain of these function definition with a uniform density of probability distribution. In Table 1.6 one can see the results of the calculations for LF with dimension $n = 16$.

The following parameters are indicated in the table: χ is the relative power parameter of the domain of definition X, $\chi = |X|/2^n$, k_1, k_0 are the numbers of one and zero values of LF, m_4, m_8 are the numbers of modules in cascades with $K = 4,\ 8$ decoder outputs, G_4, G_8 are the numbers of hyperfilters γ in cascades with $K = 4,\ 8$ decoder outputs, ε_4, ε_8 are the matrix cost indices in cascades with $K = 4,\ 8$ decoder outputs.

Signals on the inputs of the decoder were formed by two, and in case when $K = 4$, by three, when $K = 8$, linearized neurons with threshold activation. Structural vectors of neurons $A_1 = A_2 = (4; 2, 1, 1, 1, 1, 1, 1, 1)$ for $K = 4$ and vectors $A_1 = (3; 2, 1, 1, 1, 1, 1)$, $A_2 = A_3 = (3; 1, 1, 1, 1, 1)$ for $K = 8$.

Assessment of the influence of the monotonicity degree for LF on the matrix cost index ε was carried out using the following example. A threshold function $g(x)$ with $n = 16$ variables that has the value of input weights $a_i = 1,\ i = 1 \div 16$ and the threshold value $p = 9$ was chosen and that insured rank $k_1 = 26333,$ sufficient for analysis efficiency. Threshold functions are completely monotonous. In order to reduce the degree of monotonicity, the original function $g(x)$ was summed up with completely defined and chaotic LF $\rho(x)$ of various ranks ρ: $g(x) \oplus \rho(x)$. Table 1.7 shows the results of the study. The following parameters have been specified: $\rho\%$ is a relative value of the function's rank $\rho(x)$, $\rho\% \approx k_1 / 2^{n-1}$, k_ρ is a rank, the number of unit values of the chaotic function $\rho(x)$, $k_{g \sim \rho}$ is a rank, the number of unit values of the sum $g(x) \oplus \rho(x)$, G_8 is the number of hyperfilters γ in the cascade structure.

Table 1.6. Parameters of Cascade Structures for Chaotically Defined Logical Functions with Dimension n = 16.

χ	**1.0**	**0.9**	**0.8**	**0.7**	**0.6**	**0.5**	**0.4**	**0.3**	**0.2**	**0.1**	**0.05**
k₁	32777	29465	26209	22958	19541	16360	13143	9777	6514	3296	1607
k₀	32759	29462	26212	22950	19550	16360	13137	9778	6512	3297	1606
m₄	61	59	57	53	49	46	41	35	29	22	16
G₄	4591	4168	3718	3276	2772	2342	1880	1419	955	495	250
ε₄	**1.4001**	**1.2710**	**1.1337**	**0.9988**	**0.8450**	**0.7137**	**0.5728**	**0.4321**	**0.2905**	**0.1501**	**0.0753**
m₈	57	54	51	48	44	41	37	32	26	20	14
G₈	3779	3382	3027	2654	2271	1921	1546	1163	783	407	211
ε₈	**1.3820**	**1.2366**	**1.1066**	**0.9700**	**0.8297**	**0.7015**	**0.5642**	**0.4239**	**0.2848**	**0.1471**	**0.0753**

Table 1.7. Influence of the Level of Monotonicity of a Function on the Parameters of the Cascade Structure.

ρ %	**50**	**40**	**30**	**20**	**13**	**7**	**4**	**2**	**0**
k_{ρ}	32678	25991	19677	13059	8462	4640	2599	1230	0
$k_{g\sim}$	32695	31312	30072	28942	27991	27241	26908	26575	26333
${}_{\rho}G_8$	3775	3750	3600	3306	2891	2349	1900	1509	825
ε	**1.381**	**1.371**	**1.316**	**1.209**	**1.057**	**0.858**	**0.694**	**0.551**	**0.300**

The signals on the decoders inputs have been generated by three linearized neurons similarly to the previous study.

It becomes clear from the tables given above, that cascade modular structures are quite applicable for the tasks of large dimension and complexity.

1.6. Conclusion

Functionality of binomial hyper linearized filters allows to eliminate the adjustment of neuron inputs weights. And indeed, the efficiency of the choice in argument $x = (x_1, x_2, \ldots, x_n)$ of function $f(x)$ of variable x_i, $i = 1 \div n$, corresponding to the signal on some neuron input, is characterized by absolute value $|s(e_i)|$ of spectral coefficient $s(e_i)$, where e_i is a unit vector. For arbitrary LF that are chaotic in individual fragments of the definition domain, and do not possess complete monotony and, that is why are not neural functions, it is characteristic to have the ratio of spectral coefficients $|s(\omega)| > |s(e_i)|$ for a number of $\omega \notin \{0, e_1, e_2, \ldots, e_n\}$. In addition, hyper linearization combined with the tangent bundle of the domain of the definition of function X, allows to operate with coefficient $d(\omega)$ of the total spectral density D, determined by the tangent bundle. This insures even greater dominance, since the relation $d(\omega) \geq |s(\omega)|$ is valid. As a result, the efficiency of setting the neuron input weights is significantly inferior in relation to the efficiency of binomial filter setting, which is carried out by choosing a spectral argument ω.

In case of a software implementation of a neural structure, attention should be paid to the analogy between number a_i code, weight of i is the input of the neuron into the network, argument ω code that determines linear function λ_i on i is the input of the neuron in the cascade module. All arguments ω can be combined into a general matrix of dimension $G \times n$ and linearization of the entire structure can be performed through one product $\Gamma \otimes x$ (*mod2*).

In the studies carried out, a class of chaotically defined LF has been analyzed, and which has not been considered and essentially cannot be considered in the traditional methods of neural networks synthesis. The results shown in Table 1.7 indicate that when moving to the analysis of other classes of functions, the efficiency of cascade neurostructures application will increase. Besides that, spectral algorithms are well adapted to the synthesis of LF of arbitrary classes. Matrix costs can further be reduced.

Matrix operations are used in the algorithm of cascade neurostructure synthesis. As a result, matrix circuits and a modular structure are formed. Cascade structure of neuromodules is easy to analyze. If there is a need for hardware implementation, then there are additional simplifications. For example, as the task of each module is to isolate polar fragments corresponding to the minimum and maximum levels of neuron's excitation, then a bipolar two-threshold neuron can be replaced by two logic elements – element NOR and element AND with inputs calculated for the neuron module.

References

[1]. A. N. Sychev, Hyperneurons. Spectral synthesis of neurostructure modules, *Journal of International Scientific Publications: Materials, Methods & Technologies*, Vol. 16, 2022, pp. 225-232.

[2]. A. N. Sychev, Binomial hyperfilters. The application of the spectral method in the neural structures, *Journal of International Scientific Publications: Materials, Methods & Technologies*, Vol. 17, 2023, pp. 169-177.

[3]. M. G. Karpovski, E. S. Moskalev, Spectral Methods of Analysis and Synthesis of Discrete Devices, *Energy*, 1973 (in Russian).

[4]. A. N. Sychev, The effect of presynaptic inhibition in switching neurostructures, *Automatic Control and Computer Sciences*, Vol. 51, Issue 6, 2016, pp. 453-462.

2.

Control of Nonlinear Systems with Chaotic Dynamics

Vladimir Nikolaevich Shashikhin

2.1. Problem Statement

2.1.1. Introduction

Chaotic modes can occur in many nonlinear dynamical systems. For some systems, chaotic modes are harmful as they lead to noises in communication systems or to vibrations of various structures, for others they are useful, for example, for cryptographic systems. Therefore, for some systems, it is necessary to suppress the chaos and stabilize the dynamic system, for others – to create chaos or strengthen it. As a result, one of the urgent tasks of the theory of dynamic chaos is the development of methods for controlling chaos [1].

The stabilization of a dynamic system with chaotic behavior means finding control actions that make it possible to bring the system from a chaotic mode to a regular mode.

To stabilize chaotic systems, the following methods are historically the first and most actively developed at present. The Ott-Grebogi-Yorke method (OGY method) [2] is designed to stabilize an unstable periodic trajectory embedded in the attractor of a chaotic system by small perturbations of the variable parameter of the system. These perturbations are found using a discrete system model based on the linearization of the Poincaré mapping.

Vladimir Nikolaevich Shashikhin
Peter the Great St. Petersburg Polytechnic University, Russia

The Pyragas method [3] provides stabilization of the periodic orbit of a nonlinear system by constructing a phase vector feedback with a lagging argument. The Pyragas method is sensitive to the choice of lag time depending on the cycle period, which is generally unknown in chaotic systems, so the required convergence can rarely be achieved.

The Magnitsky method [4] is designed to localize and stabilize unstable singular points and periodic solutions of chaotic systems by introducing coordinate-parametric feedback in an expanded space. The advantages of the method include the fact that it makes it possible to find fixed points or periodic trajectories of a chaotic system and does not require information about the magnitude of the period and the position of the desired unstable cycle in the phase space.

Traditional approaches and methods of automatic control are also used to control chaotic dynamics. For example, in [5], the stabilization of the chaotic system is carried out using the deviation feedback, and the Routh-Hurwitz criterion is used to select the regulator coefficients.

This work is devoted to solving the problem of stabilization of nonlinear systems with chaotic dynamics. The synthesis procedure is based on the formation of the desired spectrum of Lyapunov characteristic exponents by introducing feedback on the phase vector. The feedback coefficients are determined based on the solution of the Sylvester matrix equation.

2.1.2. Mathematical Model of a Nonlinear System

Let the disturbed motions of a nonlinear dynamic object be described by an autonomous vector differential equation

$$\dot{x}(t) = F(x(t)) + Bu(t),\ x(0) = x_0, \tag{2.1}$$

where $x(t) \in \mathrm{R}^n$ is the state vector, $u(t) \in \mathrm{R}^m$ is the control vector, $m \le n$, $F(x(t)) = \left(f_i(x(t))\right)_{i=1}^{n}$ is the vector function that satisfies the conditions for the existence of solutions of equation (2.1), $f_i(x(t))$ is the real functions that are defined and continuous over the set of arguments in a domain

$$\Omega = \left\{(x,u)\ \middle|\ \|x\| + \|u\| < \wp,\ \wp = \mathrm{const} > 0\right\} \subset R^n \otimes R^m .$$

The vector function F has a given smoothness class by the vector argument $x(t)$ – $F \in \square_{x}^{r}$ or satisfies the Lipschitz condition

$$\|F(x') - F(x'')\| \leq k\|x' - x''\|,\ k > 0\,.$$

In addition, the vector function F :

- has instability in relation to the setting of initial conditions, that is, there is a value δ, such that for some point $x \in \mathrm{R}^n$ and $\varepsilon > 0$ there is a point $y \in \mathrm{R}^n$ for which the distance

$$\mathrm{dist}[x(t_0), y(t_0)] < \varepsilon \text{ , and } \mathrm{dist}[x(t), y(t)] \geq \delta \ \text{ for } \ t > t_0\,;$$

- is topologically transitive, that is, for any two open sets N, M , there is such l that

$$F^l(N) \bigcap M \neq \varnothing\,;$$

- has the element of regularity or otherwise density of periodic trajectories (in any vicinity of any point of phase space there is at least one and, therefore, infinitely many periodic trajectories.

For the trajectories $x(t, x_0)$ of system (2.1), one of three possibilities can be fulfilled:

- either the trajectory $x(t, x_0) = x^0$ is a point of rest or a state of equilibrium;

- either the trajectory $x(t, x_0)$ corresponds to a periodic solution, in this case, there is such a number $T > 0$ that

$$x(t + T, x_0) \equiv x(t, x_0)\,,$$

and the trajectory is called periodic, or closed;

- either for any

$$t_1 \neq t_2 \ \ x(t_1, x_0) \neq x(t_2, x_0)\,.$$

Suppose that the state vector of system (2.1) is large enough. In this case, it can be represented as a set of subsystems. The state of the i -th isolated (non-interacting) subsystem is determined by the expression

$$\dot{x}_i = g_i(t,x_i), \quad x_i(0) = x_{i0}, \quad g_i(t,0) \equiv 0, \quad i = \overline{1,N}. \tag{2.2}$$

Here $x_i \in \mathrm{R}^{n_i}$ is the state vector of the i-th subsystem, $\sum\limits_{i=1}^{N} n_i = n$, $g_i(t,x_i)$: $\mathrm{R} \times \mathrm{R}^{n_i} \to \mathrm{R}^{n_i}$ is the vector functions that determine the state of isolated subsystems; N is the number of subsystems in the system. The functions

$$h_i(t,x): \ \mathrm{R} \times \mathrm{R}^{n} \to \mathrm{R}^{n_i},$$

equal to

$$h_i(t,x) = f_i(t,x) - g_i(t,x_i), \quad i = \overline{1,N}, \tag{2.3}$$

describe the relationship of the i-th subsystem with other subsystems.

The behavior of the i-th interacting subsystem can be represented by the equation

$$\dot{x}_i = g_i(t,x_i) + h_i(t,x), \ \ i = \overline{1,N} \ . \tag{2.4}$$

Equation (2.3) describes the relationships between isolated subsystems (2.2), and equation (2.4) – the behavior of a large-scale system (2.1), represented in the form of interacting subsystems.

2.1.3. Lyapunov Characteristic Exponents

One of the features of irregular modes is the instability of trajectories belonging to a chaotic (strange) attractor. The quantitative measure of this instability is the characteristic exponents, originally introduced by Lyapunov [6]. Formally, the characteristic Lyapunov exponent is introduced as follows: the characteristic exponent of a function $z(t)$ is a number (or symbol $\pm\infty$) defined as follows

$$\lambda(z) \equiv \overline{\lim_{t\to\infty}}\left(t^{-1} \ln\left\|z(t)\right\|\right).$$

The Lyapunov characteristic exponent of the function $z(t)$ is the result of comparing the growth rate of the function $z(t)$ at $t \to \infty$ with an exponent $\exp\{\chi t\}$, for which the characteristic exponent is equal to χ. Among the entire set of Lyapunov characteristic exponents, the largest (senior) exponent $\chi_1 = \chi_{\max}$ is the most important. If $\chi_1 < 0$, then the trajectory of the nonlinear system is asymptotically stable; if $\chi_1 > 0$ — is unstable. The set of characteristic exponents, sorted in descending order $\chi_1 \geq \chi_2 \geq \ldots \geq \chi_n$, is called the Lyapunov spectrum of a nonlinear dynamical system. In n-dimensional systems, the signature of the Lyapunov spectrum (signs of the characteristic exponents) can take the following form:

$$\underbrace{(-,-,-,\ldots,-,-,-)}_{n}, \tag{2.5a}$$

$$(\,0,\underbrace{-,-,\ldots,-,-,-}_{n-1}\,), \tag{2.5b}$$

$$(\underbrace{+,\ldots,+}_{s},0,-,\ldots,-), \tag{2.5c}$$

where equation (2.5a) determines the state of equilibrium, (2.5b) is the limit cycle, and (2.5c) is a strange attractor.

2.1.4. The Task of Chaos Stabilization

The problem of chaos stabilization consists in transforming the irregular mode of system (2.1), which is characterized by Lyapunov spectrum (2.5c), into a regular mode with a spectrum of characteristic exponents (2.5a) or (2.5b), that is, to provide an attractor in the form of a singular point or limit cycle.

To solve the stabilization problem, we will look for control in the form of feedback over the phase vector of the nonlinear system (2.1)

$$u(t) = Lx(t),\ \ L \in \mathrm{R}^{m\times n}, \tag{2.6}$$

which will provide in a closed system

$$\dot{x}(t) = F(x(t), Lx(t)), \quad x(0) = x_0 \tag{2.7}$$

a spectrum of Lyapunov characteristic exponents

$$\sigma(F) = \{\chi_i(F),\ i = \overline{1,n}\},$$

which is equal to the desired (required) spectrum

$$\sigma(G) = \{\chi_i(G),\ i = \overline{1,n}\}. \tag{2.8}$$

The desired spectrum (2.8) is determined by the required character of the regular motion of system (2.1).

To reduce the computational costs of synthesis, the control of a nonlinear system (2.1) must be implemented in the form of a controller (2.6) with a decentralized structure

$$\begin{gathered} u_i(x_i) = -L_{ii}x_i,\ i = \overline{1,N} \Leftrightarrow \\ \Leftrightarrow u(x) = -L_D x,\ L_D = \text{blockdiag}\{L_{ii}\}_{i=1}^{N}. \end{gathered} \tag{2.9}$$

The decentralized regulator determines a set of local regulators (2.9), each of which implements feedback on the phase vector of the corresponding subsystem (2.4).

Thus, the problem of stabilization is reduced to the suppression of chaotic oscillations by bringing them to regular oscillations (stabilization of the limit cycle), or complete suppression of oscillations (stabilization of a singular point) by introducing feedback.

2.2. Feedback Synthesis

2.2.1. Topological Equivalence

The solution to the problem of forming the necessary spectrum of Lyapunov characteristic exponents is based on the use of their dependence on the eigenvalues of the Jacobian matrix. A change in the eigenvalues of the Jacobian matrix entails a change in the Lyapunov characteristic exponents of the nonlinear system. The desired eigenvalues of the Jacobian matrix can be assigned, for example, using the modal control synthesis based on the solution of the Sylvester matrix algebraic equation.

The validity of this approach is based on the theorem on the structural stability (roughness) of nonlinear dynamical systems, formulated in [7], and determining the essence of the topological equivalence of a nonlinear system and a hyperbolic linearized model [8, 9]. It follows from the theorems given in these works. If a linearized system is hyperbolic (has no purely imaginary eigenvalues), then the nonlinear system has stable or unstable manifolds, which are smooth analogs of stable or unstable spaces of the linearized system. Otherwise, the nonlinear system and the linearized system have the same number of singular points and limit cycles.

2.2.2. Synthesis of Feedback for a Nonlinear System

The feedback synthesis algorithm for a nonlinear large-scale system (2.1) includes the following steps [10].

When controlling the spectrum of Lyapunov characteristic exponents, the eigenvalues of Jacobian matrices $J(x)$ are used, which are calculated at various points of the trajectory of the nonlinear system. To do this, the phase space of the nonlinear system is divided into small cells. For each cell, the eigenvalues of the Jacobian matrix of the closed system $\tilde{J}(x)$ are selected in accordance with the problem (2.5a) or (2.5b) being solved.

Next, the feedback coefficient for each cell is calculated. The feedback coefficient of a nonlinear system is determined taking into account the probability of visiting the trajectory of the cells of the phase space of the system (invariant measure of the dynamic system). To calculate the probability of a trajectory visiting a nonlinear system of cells in a phase space p_i, the phase space is divided into small cells C_i, and the solution $x(t, x_0)$ of the dynamic system is found for a sufficiently long time interval. Then, for each cell, the number of solution points that fall into it is determined, and the probability of the trajectory falling into this cell is found:

$$p_i = \frac{N_i}{N},$$

where N_i is the number of points in the subset C_i, N is the total number of points. The size of the cells is selected as follows:

$$h_j = \frac{1}{S(T) - S(T_0)} \sum_{k=S(T_0)}^{S(T-1)} \left| x_j(k+1) - x_j(k) \right|, \ j = \overrightarrow{1,4},$$

where T_0 is the time of the beginning of the calculation of the trajectory of the nonlinear system, such that the transient process has already been completed, T is the time of the end of the calculation of the trajectory, $S(t)$ is the step number corresponding to the current time t. That is, the length of the cell side h_j, parallel to the phase coordinate x_j, is chosen to be equal to the time average of the difference between the coordinates x_j of the next and previous points.

The required eigenvalues of the Jacobian matrix $\tilde{J}(x_i)$ corresponding to the center of each cell are selected by the following formula:

$$\overline{v}\left(\tilde{J}(x_i)\right) = v\left(J(x_i)\right) + \alpha \operatorname{Re}\left(v\left(J(x_i)\right)\right) + \beta \operatorname{Im}\left(v\left(J(x_i)\right)\right), \qquad (2.10)$$

where $v\left(J(x_i)\right)$ are the eigenvalues of the Jacobian matrix of the original nonlinear system, calculated in the center x_i of the cell C_i, α is a coefficient that affects the shift of the eigenvalues along the real axis; β is a coefficient that affects the shift of the eigenvalues along the imaginary axis. When solving the stabilization problem, one needs to reduce the Jacobian eigenvalues by choosing $\alpha < 0$ and $\beta < 0$.

Next, for each cell, we find the feedback coefficients $L_i, i = \overrightarrow{1,N}$ that will provide the specified characteristic indicators in a closed nonlinear system. To calculate them, it is proposed to use the method based on the solution of the Sylvester matrix equation. A detailed description of the method is given in Section 3. Having calculated the feedback coefficients for each cell, we find the feedback coefficient for a nonlinear system by the formula

$$L = \sum_{i=1}^{N} L_i p_i \qquad (2.11)$$

and determine the Lyapunov characteristic exponents of the nonlinear system (2.1), closed by the control $u = Lx$.

To set the required value of the Lyapunov characteristic exponent in the system, the above calculations are performed for several sets of coefficients α and β, and then the feedback coefficient L^* is determined to provide the desired value of the Lyapunov characteristic exponent, according to the criterion

$$L^* = \arg\min \left\{ \varphi = \left(1 - \operatorname{sign}\left(\varepsilon - \left|\chi_{1i} - \chi_1^*\right|\right)\right) \times \max_{j=\overline{1,r}} \left\|L_j\right\| + \left\|L_i\right\| \,\middle|\, \varphi < 2\max_{j=\overline{1,r}} \left\|L_j\right\| \right\}, \qquad (2.12)$$

where $i \in \overline{1,r}$; r is the number of coefficients α and β, χ_1^* is the desired senior Lyapunov characteristic exponent of the nonlinear system; χ_{1i} is the senior Lyapunov characteristic exponent of the nonlinear system with $u = L_i x$.

To stabilize the system (if the desired attractor is a fixed point), the characteristic Lyapunov exponent of a closed system must be less than a given negative number $\chi_{1i} < \overline{\chi_{1i}} < 0$. The selection criterion L^* in this case has the form:

$$L^* = \arg\min \left\{ \varphi = \left(1 - \operatorname{sign}\left(\overline{\chi_{1i}} - \chi_{1i}\right)\right) \times \max_{j=\overline{1,r}} \left\|L_j\right\| + \left\|L_i\right\| \;\; \varphi < 2\max_{j=\overline{1,r}} \left\|L_j\right\| \right\}. \qquad (2.13)$$

When stabilizing the system (if the desired attractor is a limit cycle), the criterion (2.12) is used, in which the desired senior Lyapunov characteristic exponent $\chi_1^* = 0$ is used.

2.2.3. Synthesis of Feedback for a Linearized System

2.2.3.1. System Linearization

Let equation (2.1) describe the deviations of the phase coordinates of a nonlinear object in the vicinity of particular solutions x^S corresponding to control actions u^S. Using the Taylor formula under the assumption that the components of the function $F(x(t)) = (\phi_i(x(t))_{i=1}^n$ are differentiable in a neighborhood $\xi^S = (x^S, u^S)$, we transform equation (2.1) to the quasilinear form

$$\dot{x}(t) = A(\xi^S)x(t) + Bu(t) + f(\xi^S), \quad x(0) = x_0 . \tag{2.14}$$

In equations (2.14), the coefficient $A(\xi^S)$ is calculated at a point ξ^S by the formula

$$A(\xi^s) = \begin{bmatrix} \partial\varphi_1 / \partial x_1 & \dots & \partial\varphi_1 / \partial x_n \\ \dots & \dots & \dots \\ \partial\varphi_n / \partial x_1 & \dots & \partial\varphi_n / \partial x_n \end{bmatrix}_{\substack{x(t)=x^S \\ u(t)=u^S}} . \tag{2.15}$$

Suppose for all values

$$\begin{gathered} \xi^S \in S(x^S, u^S, \rho) = \{(x^S, u^S) : \\ : \ \left\| x - x^S \right\| + \left\| u - u^S \right\| \le \rho, \rho > 0\} \subset \mathrm{R}^n \otimes \mathrm{R}^m \ \ \} \end{gathered}$$

the following evaluations are being completed

$$\left\| f(\xi^S) \right\| \le q \left\| \xi \right\| . \tag{2.16}$$

If the Jacobian matrix is calculated by equation (2.15) and condition (2.16) is satisfied, then equation (2.14) takes the form of a linearized system (or equations in variations)

$$\dot{y}(t) = Ay(t) + Bu(t) . \tag{2.17}$$

System (2.17) can be used to design a control that stabilizes system (2.1) in the vicinity of a particular solution. The real parts of the eigenvalues of the Jacobian matrix determine the behavior of the trajectories of the original nonlinear system.

2.2.3.2. Synthesis of Centralized Control

The problem of positioning the poles of the system is considered, in which the determination of the controller parameters is reduced to solving the matrix Sylvester equation. For system (2.17), it is necessary to find a stabilizing controller in the form of feedback on the state vector

$$u(y(t)) = -Ly(t) \tag{2.18}$$

such that the spectrum of the closed system

$$\dot{y}(t) = (A - BL)y(t) = A_y y(t) \tag{2.19}$$

coincides with the prescribed spectrum, given by the sequence $\mu = \{\mu_1, \ldots, \mu_n\}$

$$\rho(A_y) = \rho(-W) . \tag{2.20}$$

Here $W = \text{diag}\,(\mu_i)_{i=1}^{n} \in \mathrm{R}^{n\times n}$ is a matrix with numbers μ_i on its main diagonal.

The problem of finding the matrix L that determines the "depth" of the feedback from the full state vector is reduced to solving the Sylvester matrix equation

$$AP + PF = BG \tag{2.21}$$

with respect to a matrix $P \in \mathrm{R}^{n\times n}$ with an arbitrary matrix $G \in \mathrm{R}^{m\times n}$ and solving the matrix equation

$$LP = G,\ L = GP^{-1} . \tag{2.22}$$

For dynamical system (2.19), the conditions for the existence of a solution to the pole placement problem and the method for synthesizing a stabilizing control are contained in the theorem given in [10]. The parameters of the controller (2.18), ensuring the fulfilment of condition (2.20) in the closed-loop system (2.19), are determined from relation (2.22), where the matrix P is the solution to Sylvester equation (2.21). The matrix $A \in \mathrm{R}^{n\times n}$ is the Jacobian matrix of a nonlinear system.

2.2.3.3. Synthesis of Decentralized Control

Let us represent the matrix $A \in \mathrm{R}^{n\times n}$ – the matrix of parameters of system (2.17) as a sum

$$A = A_D + A_O , \tag{2.23}$$

where $A_D = \text{blockdiag}\{A_{ii}\}_1^N$ is the block diagonal matrix, the elements of which characterize the parameters of isolated subsystems;

$A_O = \text{block}\left\{A_{ij}\right\}_{i,j=1}^{N}$, $A_{ij} \neq 0$, $i \neq j$ is the block nondiagonal matrix, each block A_{ij} of which determines the intensity of the effects of the j-th subsystem on the i-th subsystem; $B = \text{blockdiag}\{B_{ii}\}_1^N \in \mathrm{R}^{n\times m}$ is the block diagonal input matrix.

Using structural decomposition, system (2.17) can be represented as a set of interacting subsystems

$$\dot{x}_i = A_{ii}x_i + B_{ii}u_i + h_i, \quad x_i(0) = x_{i0},$$
$$h_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} A_{ij}x_j, \tag{2.24}$$

where $x_i \in \mathrm{R}^{n_i}$ is the state vector of the i-th subsystem, $\sum_{i=1}^{N} n_i = n$ $u_i \in \mathrm{R}^{m_i}$ is the vector of control actions of the i-th subsystem, $h_i : \mathrm{R}^n \to \mathrm{R}^{n_i}$ is a vector function characterizing the influence on the i-th subsystem of all other subsystems; $B_{ii} \in \mathrm{R}^{n_i\times m_i}$ is the matrix of controls of the i-th subsystem.

If the matrices G and W with a structure similar to the matrix A

$$G = G_D + G_O \text{ and } W = W_D + W_O,$$

where

$G_D = \text{blockdiag}\left\{G_{ii}\right\}_{i=1}^{N}$, $G_O = \text{block}\left\{G_{ij}\right\}_{i,j=1}^{N}$ $W_D = \text{blockdiag}\left\{W_{ii}\right\}_{i=1}^{N}$, $W_O = \text{block}\left\{W_{ij}\right\}_{i,j=1}^{N}$, then Sylvester equation (2.21) takes the form

$$\left(A_D + A_O\right)P + P\left(W_D + W_O\right) = B\left(G_D + G_O\right),$$

which is equivalent to two equations: the equation for diagonal blocks

$$A_D P + PW_D = BG_D \tag{2.25}$$

and the equation for nondiagonal blocks

$$A_O P + PW_O = BG_O.$$

With a diagonal structure of block matrices A_D, W_D, B and G_D included in equation (2.25), it is equivalent to the Sylvester equations N

$$A_{ii}P_{ii} + P_{ii}W_{ii} = B_{ii}G_{ii},\ i = \overline{1,N}\,, \tag{2.26}$$

which correspond to isolated subsystems. Under these conditions, equation (2.22) takes the diagonal form

$$LP = G_D \Leftrightarrow \left(L_{ii}P_{ii} = G_{ii},\ i = \overline{1,N}\right),$$

and the regulator (2.18) has the desired decentralized structure [11].

Decentralized control ensures the equality of the closed-loop system spectrum to the spectrum of the reference matrix: $\rho\left(A_y\right) = \rho\left(-W\right)$ and reducing computational costs by decomposing the Sylvester equation of dimension n into N equations of dimension n_i ($n_i << n$), corresponding to the subsystems, and implementing local controllers in the form of feedback on the phase vector of the subsystems.

2.3. Study of the System of Synchronous Generators

2.3.1. Model of a Three-machine System

The proposed method for the synthesis of control for a nonlinear large-scale system is considered by the example of control of chaotic oscillations arising in the operation of an electric power system, presented in the form of a system of three interconnected synchronous generators.

To analyze the chaotic modes of the electric power system, the classical model of a synchronous generator is used, which allows for a qualitative and quantitative analysis indicating the irregular nature of the deviation of the rotor angle and frequency.

The equations of the mathematical model of the three-machine electric power system, which has unequal inertia of the rotors of the generators included in it, have the form [12]:

$$\begin{aligned}&\frac{d\delta_1}{dt}=\omega_1,\\&\frac{d\omega_1}{dt}=-B_1\sin(1+\frac{1}{\sqrt{2}})\delta_1+\frac{1}{\sqrt{2}}\delta_3)-C_{13}\sin(\delta_1-\delta_3)+P_1,\end{aligned}\qquad(2.27a)$$

$$\begin{aligned}&\frac{d\delta_2}{dt}=\omega_2,\\&\frac{d\omega_2}{dt}=-B_2\sin((1+\frac{1}{\sqrt{2}})\delta_2+\frac{1}{\sqrt{2}}\delta_3)-C_{21}\sin(\delta_2-\delta_1)+P_2,\end{aligned}\qquad(2.27b)$$

$$\begin{aligned}&\frac{d\delta_3}{dt}=\omega_3,\\&\frac{d\omega_3}{dt}=-B_3\sin((1+\frac{1}{\sqrt{2}})\delta_1+\frac{1}{\sqrt{2}}\delta_3)-C_{31}\sin(\delta_3-\delta_1)+P_3,\end{aligned}\qquad(2.27c)$$

where $\delta_1, \delta_2, \delta_3$ are the deviations of the angle of rotation of the rotor of the generator relative to the synchronously rotating axis; $\omega_1, \omega_2, \omega_3$ are the deviation of the angular frequency; $P_{c13}, P_{c21}, P_{c31}$ are the synchronizing power between generators; P_1, P_2, P_3 are the change in the power supplied to the network by generators; $\varepsilon_{01}, \varepsilon_{02}, \varepsilon_{03}$ are the initial values of the power supplied to the network by the generators in the event of a network disturbance.

The studies of the multimachine system were carried out at the following values of the model parameters

$$\begin{aligned}&B_1=\frac{P_1}{T_{j1}}=1, C_{13}=\frac{P_{c13}}{T_{j1}}=0.1, P_1=\frac{\varepsilon_{01}}{T_{j1}}=0.4,\\&B_2=\frac{P_2}{T_{j2}}=1, C_{21}=\frac{P_{c21}}{T_{j2}}=0.1, P_2=\frac{\varepsilon_{02}}{T_{j2}}=0.4,\\&B_3=\frac{P_3}{T_{j3}}=1, C_{31}=\frac{P_{c31}}{T_{j3}}=0.1, P_3=\frac{\varepsilon_{03}}{T_{j3}}=0.3\,.\end{aligned}$$

Introducing the phase vector (2.28),

$$\begin{aligned}&x(t)=(x_1(t)=\delta_1, x_2(t)=\omega_1, x_3(t)=\delta_2,\\&x_4(t)=\omega_2, x_5(t)=\delta_3, x_6(t)=\omega_3)^T\in\mathrm{R}^6\end{aligned},$$

system (2.27) can be written as

$$\dot{x}(t) = F(x(t)).$$

2.3.2. Research of Processes in the System

2.3.2.1. Uncontrolled System

The study of system (2.27) for the presence of chaotic oscillations was carried out under the initial conditions

$\delta_1(0) = 0.6;\ \omega_1 = 0.3;\ \delta_2(0) = 0.6;\ \omega_2 = 0.3;\ \delta_3(0) = 0.6;\ \omega_3 = 0.3$,
and the singular point of system (2.27) has coordinates:

$$x_0 = (-10.1818;\quad 0;\quad -6.5625;\quad 0;\quad 1.8609;\quad 0)^T .$$

For the indicated values of the parameters and initial conditions, the Lyapunov characteristic exponents of system (2.27) are:

$$\begin{aligned} \lambda_1 &= 0.0036; & \lambda_4 &= -0.0054;\\ \lambda_2 &= 0.0027; & \lambda_5 &= -1.0456;\\ \lambda_3 &= 0.0012; & \lambda_6 &= -3.1895; \end{aligned}$$

Fig. 2.1 shows the projection of the phase portrait of system (2.27) onto the plane $x_3 = \delta_2$ and $x_4 = \omega_2$.

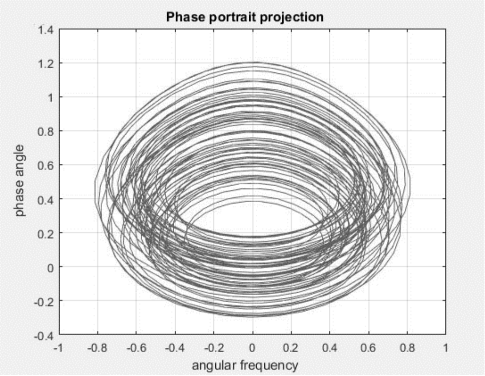


Fig. 2.1. Projection of the phase portrait of the system onto the plane $x_3 = \delta_2$ and $x_4 = \omega_2$.

The spectrum contains positive Lyapunov characteristic exponents, there is therefore a chaotic regime in system (2.27). In addition, Fig. 2.1 shows that the projection of the trajectory of the system in the phase space is a strange attractor, which also indicates an irregular regime.

2.3.2.2. System with Centralized Control

Let us introduce into the system the control of the frequency of each generator, then the matrix *B* is equal to

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}^T ,$$

and the equations of system (2.27) with centralized control (2.18) take the form

$$\dot{x}(t) = F(x(t)) + BLx(t). \tag{2.28}$$

The Jacobian matrix of system (2.28) has the form:

$$J = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \partial f_2/\partial x_1 & 0 & 0 & 0 & \partial f_2/\partial x_5 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \partial f_4/\partial x_1 & 0 & \partial f_4/\partial x_3 & 0 & \partial f_4/\partial x_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \partial f_6/\partial x_1 & 0 & 0 & 0 & \partial f_6/\partial x_5 & 0 \end{bmatrix},$$

where

$$\frac{\partial f_2}{\partial x_1} = -\frac{\cos(\delta_1 - \delta_3)}{10} - (\frac{1}{\sqrt{2}} + 1)\cos(\delta_1(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}),$$

$$\frac{\partial f_2}{\partial x_5} = \frac{\cos(\delta_1 - \delta_3)}{10} - \frac{\sqrt{2}}{2}\cos(\delta_1(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}),$$

$$\frac{\partial f_4}{\partial x_1} = \frac{\cos(\delta_1 - \delta_2)}{10},$$

$$\frac{\partial f_4}{\partial x_3} = -\frac{\cos(\delta_1 - \delta_2)}{10} - (\frac{1}{\sqrt{2}} + 1)\cos(\delta_2(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}),$$

$$\frac{\partial f_4}{\partial x_5} = -\frac{\sqrt{2}}{2}\cos(\delta_2(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}),$$

$$\frac{\partial f_6}{\partial x_1} = \frac{\cos(\delta_1 - \delta_3)}{10} - (\frac{1}{\sqrt{2}} + 1)\cos(\delta_1(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}),$$

$$\frac{\partial f_6}{\partial x_5} = -\frac{\cos(\delta_1 - \delta_3)}{10} - \frac{\sqrt{2}}{2}\cos(\delta_1(\frac{1}{\sqrt{2}} + 1) + \frac{\delta_3}{\sqrt{2}}).$$

The feedback coefficient calculated by the method of synthesis of the centralized controller taking into account (2.16) and (2.17) is equal to

$$L = \left(-5.8045; \quad -9.0067; \quad -7.1735\right)^T .$$

The spectrum of Lyapunov characteristic exponents has the form:

$$\lambda_1 = 0, \lambda_2 = -4.5682, \lambda_3 = -5.2761, .$$
$$\lambda_4 = -7.5076, \lambda_5 = -10.2082, \lambda_6 = -15.8423$$

The senior characteristic exponent is zero, the remaining characteristic exponents are negative; this indicates that the system is brought to regular movement – the limit cycle.

Fig. 2.2 shows the projection of the phase portrait of the system with centralized control on the coordinate plane $x_3 = \delta_2$ and $x_4 = \omega_2$.

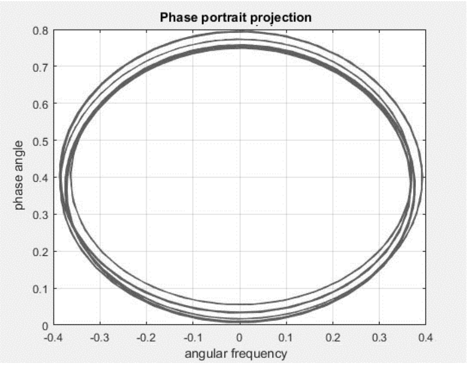


Fig. 2.2. Projection of the phase portrait of the system onto the plane $x_3 = \delta_2$ and $x_4 = \omega_2$

The phase portrait of the system (2.28) with centralized control also indicates regular movement in the form of a closed cycle.

2.3.2.3. System with Decentralized Control

Let us decompose system (2.28) into subsystems that correspond to the equations of one generator with phase coordinates – deviation of the rotor angle of rotation and deviation of the generator frequency. The mathematical model of subsystem (2.24), in this case, is, for example, equation (2.27a). That is, we have three subsystems of dimension two.

Jacobian matrices for each of the subsystems:

$$J_{11} = A_{11} = \begin{bmatrix} 0 & 1 \\ \frac{\partial f_2}{\partial x_1} & 0 \end{bmatrix}; J_{22} = A_{22} = \begin{bmatrix} 0 & 1 \\ \frac{\partial f_4}{\partial x_3} & 0 \end{bmatrix}; J_{33} = A_{33} = \begin{bmatrix} 0 & 1 \\ \frac{\partial f_6}{\partial x_5} & 0 \end{bmatrix}.$$

Formulas for calculating partial derivatives $\partial f_j / \partial x_k$, $j = 2, 4, 6$, $k = 1, 3, 5$ are given in the previous paragraph. The Jacobian matrices for each of the subsystems are the diagonal blocks of the Jacobian matrix for the system as a whole.

For each of the subsystems, the feedback coefficient is calculated using the decentralized control synthesis technique

$$L_{11} = -1.8620, \quad L_{22} = -0.7354, \quad L_{33} = -2.7388.$$

Lyapunov characteristic exponents in a system closed by a decentralized controller are equal to

$$\begin{array}{l} \lambda_1 = 0 \quad \lambda_2 = 0 \quad \lambda_3 = -0.0896 \quad \lambda_4 = -1.0628 \\ \lambda_5 = -3.9880 \quad \lambda_6 = -6.8304 \end{array}.$$

Fig. 2.3 shows the projection of the phase portrait of a nonlinear system with decentralized control on a plane $x_3 = \delta_2$ and $x_4 = \omega_2$.

The spectrum of Lyapunov characteristic exponents and the projection of the phase portrait of a system closed by decentralized control indicate the presence of a regular regime in the form of a cycle.

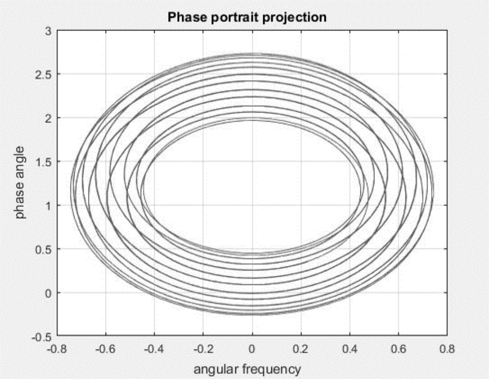


Fig. 2.3. Projection of the phase portrait plane $x_3 = \delta_2$ and $x_4 = \omega_2$

2.4. Conclusion

The method of synthesis of stabilization control for nonlinear systems with chaotic dynamics is proposed. The parameters of the regulator providing the required spectrum of Lyapunov characteristic exponents are determined by the solution of the Sylvester matrix equation and an invariant measure calculated on the trajectories of a nonlinear system.

A technique for the synthesis of control for suppressing chaotic oscillations in a nonlinear large-scale system using phase vector feedback is presented. The feedback coefficient providing a given spectrum of Lyapunov characteristic exponents is calculated by the modal control method based on the solution of the matrix algebraic Sylvester equation extended to nonlinear large-scale systems with chaotic dynamics.

The article considers the use of the proposed method for the synthesis of decentralized control by the example of a system consisting of three synchronous generators. The results of the study confirmed the suppression of chaotic oscillations and the provision of a regular mode in a closed system due to the formation of a spectrum with zero and negative Lyapunov characteristic exponents.

The advantage of the decentralized control is the reduction of computational costs for the synthesis and implementation of control systems for large-scale systems. The synthesized feedback provides suppression of chaotic oscillations not in a small region of the phase space, but in the region of existence of solutions to the equations of the dynamics of a nonlinear system.

References

[1]. B. R. Andriyevsky, A. L. Fradkov, Control of chaos: methods and applications. I. Methods, *Automation and Telemechanics*, Number 5, 2005, pp. 3-45 (in Russian).

[2]. E. Ott, C. Grebogi, G. Yorke, Controlling chaos, *Physical Review Letter*, Vol. 64, 1990, pp. 1196-1199.

[3]. K. Pyragas, Continuous control of chaos by self-controlling feedback, *Physical Letters A*., Vol. 170, 1992, pp. 421-428.

[4]. N. A. Magnitsky, S. V. Sidorov, New Methods of Chaotic Dynamics, *Editorial URSS*, Moscow, 2004.

[5]. R. Yamapi, J. B. Chabi Orou, Harmonic oscillations, stability and chaos control in a non-linear electromechanical system, *Journal of Sound and Vibration*, Vol. 259, 2003, pp. 1253-1264.

[6]. A. M. Lyapunov, The general problem of stability of motion, *Gostekhizdat*, Moscow, 1950.

[7]. A. A. Andronov, L. S. Pontryagin, Rough Systems, *DAN USSR*, Vol. 14, 1937, pp. 247-250.

[8]. D. G. Grobman, Homeomorphism of systems of differential equations, *DAN USSR*, Vol. 128, 1959, pp. 880-881.

[9]. R. O. Omorov, Method of topological roughness of dynamical systems, *Materials Science,* Vol. 24, 2017, pp. 77-83.

[10]. V. N. Shashihin, S. V. Budnik, Synthesis of control for nonlinear systems, *Automatic Control and Computer Sciences*, Vol. 53, 2019, pp. 97-106.

[11]. V. N. Kozlov, V. N. Shashikhin, Synthesis of decentralized robust stabilizing control for the systems with parametric perturbations, *Computing, Telecommunications and Control*, Vol. 13, 2020, pp. 49-60.

[12]. D. V. Ryseev, V. K. Fedorov, Modern problems of dynamics of nonlinear power systems: electromechanical resonance, entropy, deterministic chaos. *Omsk: KAN Polygraphic Center*, 2012.

3.

Ontology-driven Generation of Interoperable Field Device Capabilities

Victor Chavez and Jörg Wollert

3.1. Introduction

Seamless integration of multiple field devices for industrial applications today requires not only a common communication interface but also an intelligent description of the capabilities they provide [1]. As field devices are not standardized by one single organization, there is a heterogeneity in description models and their semantics. Within the framework of the German government's Industry 4.0 (I4.0) strategy [2], the provision of an interoperable interface model for industrial assets aims to solve this type of challenge [3].

In specific, the Asset Administration Shell (AAS) [4], is a standard promoted for the exchange of static and dynamic information in the so-called I4.0 language [5]. The AAS facilitates the digital representation of an asset (e.g., a machine, sensor, or an assembly line) in the form of submodels that can represent specific domain information. Another standard with a similar approach is developed by the OPC Foundation, a well-known organization recognized for its OPC Unified Architecture (OPC UA) [6], a machine-to-machine communication data model standard. Recently, the OPC Foundation has introduced the OPC UA Field Exchange standard (OPC UA FX) [7]. OPC UA FX facilitates interoperable data exchange at the field level. Its functional data model includes identification properties, input data, configuration data, output data, and diagnostic data.

Victor Chavez
Aachen University of Applied Sciences, Institute for Applied Automation and Mechatronics, Aachen, Germany

These metadata models reduce the interoperability issues between devices as they provide the means to create a common semantic model for Industry 4.0 applications. However, the main challenge has now become to define abstract models that can be exchanged and reused among different organizations, groups, or internally in a company. The development of an interoperable semantic model is not an easy task. In the field device context, multiple standards share a similar semantic model for process data exchange and interpretation but cannot be integrated without manual mapping or alignment [8]. A means to integrate different standardized semantics (e.g., IO-Link, CAN Open, EtherCAT, PROFINET) remains elusive because it requires a generalization that can be reused regardless of the implementation.

The challenge of interoperable semantic models is not new, and organizations such as the World Wide Web Consortium (W3C) have been working on standards to bridge this gap. In particular, the Web Ontology Language (OWL) and the Resource Description Framework (RDF) are used to develop interoperable semantic models for the Web. A common approach for the Internet of Things is to define services that allow other entities to understand how to interpret and exchange information [9, 10]. A similar approach from an I4.0 perspective is to define functions that a given resource can implement to produce an effect in the real or physical world. This is also referred to as capabilities [11].

In this chapter, we present a methodology for generating generic capabilities for field devices using an ontology-based approach. Section 3.2 outlines the ontology-based framework employed to represent the semantics of field devices. Section 3.3 delves into the concept of generic field device capabilities, analyzing their relationship to these semantic representations. Section 3.4 focuses on the application of SWRL inference rules to derive these generic capabilities. Section 3.5 introduces a software tool developed to automate the generation of these capabilities and facilitate the integration of SWRL rules. Finally, Section 3.6 concludes with a discussion of the results and potential directions for future research.

3.2. Field Device Semantics

Field devices are standardized by industrial consortiums and organizations. In general, some features are similar regardless of the specific implementation. Based on these features, it is possible to generalize field device concepts that apply to all standards which allows reusability and extensibility. In this section, we will cover the conceptual model of field devices and provide a generalization that can be reused and extended independently of implementation.

One of the earliest standardization efforts was the Modbus protocol by Modicon (now Schneider Electric) in 1979. Originally designed to communicate with their Programmable Logic Controllers (PLCs) and devices, it became a popular protocol worldwide due to its simplicity based on the concept of registers and coils. As processing power increased and electronics manufacturing costs decreased, more advanced physical layers and stack-based protocols emerged over the next decades. Some examples of current field device protocols include CAN Open, IO-Link, PROFINET, and EtherCAT.

Even though these protocols were designed with different use cases in mind, they exhibit similar features. Most of these are based on a controller-device model, where the controller has a connection with one or multiple field devices and provides an interface for the exchange of process data in real-time, and an interface for diagnostics and device management (see Fig. 3.1). From the point-of-view of a user, this requires an engineering tool for the management of the controller and the field devices. To facilitate this task, typically field devices provide a description file that can be downloaded from the manufacturer or an online database. This description file is then loaded to the engineering tool to set up the controller and exchange data with other systems (e.g. a PLC).

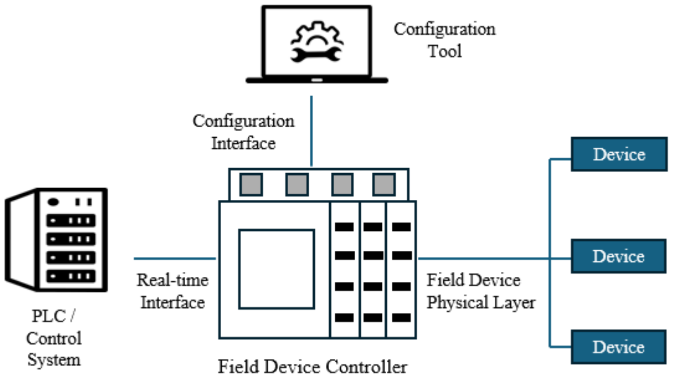


Fig. 3.1. Overview of a Controller-device Model for field devices.

Device descriptions are machine-interpretable files that define static features of a field of devices within the context of a specific standard. The most basic information included in these files is a unique identification (e.g. vendor number, device number, or serial number). For their integration with configuration tools, and real-time controllers, metadata is included with information on how to exchange information over a specific protocol or interface. Additionally, some standards define interoperable features that are independent of the manufacturer in the form of device profiles [12].

Because of the divergence in standards, the semantic data derived from device description files can only be used within a limited scope. In the work of [13] a comparative analysis has been done between IO-Link and CAN Open to provide a generalization of field devices based on their application layer. The authors propose a semantic model that encompasses device description files, their relation to application data, and units of information. A more comprehensive comparative analysis is presented in Table 3.1 concerning the management of application data. The application data can be divided into two categories: asynchronous and synchronous. Asynchronous application data includes parameter exchange, device configuration, and non-real-time data. In contrast, synchronous data refers to real-time process information from sensors or control commands to actuators. The exchange of this application data is facilitated by a specific access mechanism, typically involving an identification or relationship number. The specific application data contained for a field device is serialized in the supported device description language and then processed by either a configuration tool or the application software such as a PLC or control system connected to the field device controller (see Fig. 3.2).

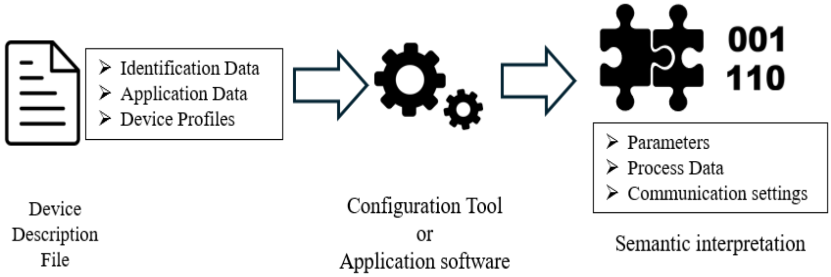


Fig. 3.2. Device description File semantic interpretation process.

Table 3.1. Comparison of Field Device Application Data semantics.

Standard	Application Data	Access Mechanism	Device Description File
IO-Link	**Asynchronous** • Index Service Data Unit **Synchronous** • Process Data Input. • Process Data Output	• Index • Subindex	IO Device Description
EtherCAT	**Asynchronous** • Service Data Object **Synchronous** • Transmit Process Data Object • Receive Process Data Object	• Address • Index	EtherCAT SubDevice Information
CAN Open		• Index	Electronic Data Sheet
PROFINET	**Asynchronous** • Record Data Communication Relationship **Synchronous** • IO-Data Communication Relationship	• Application Process Identifier • Slot number • Subslot number • Index	General Station Description Markup Language

Although device description files provide enough information for an application to process its semantics, the interoperability with other standards is not straightforward. While the generic I4.0 interfaces are defined in standards such as OPC UA FX or AAS, the semantic alignment and mapping between specific application layers are not (see Fig. 3.3). To bridge this gap the authors in [14] propose the Industry 4.0 Field Device Ontology (I40FD). The I40FD ontology enables a generalization of field device concepts such as application data, device profiles, device description files, and their relation to specific protocols.

The conceptual model of the I40FD ontology consists of the relationship of field devices to a device description file. The device description file has a relation to Application Data Objects (ADO), which contain different types of application data, as shown in Table 1. An ADO in turn consists of Application Data Elements (ADE), which are the basic data points from a field device derived from standardized data types (e.g., string, integer, float). In addition, multiple ADOs can be grouped as a part of a device profile or categorized as Process Data Object (PDO) for a synchronous exchange of data. A simplified conceptual model of these relationships is shown in Fig. 3.4.

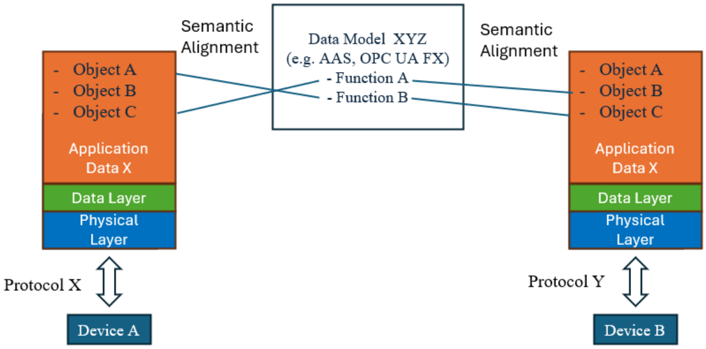


Fig. 3.3. Semantic alignment required to use an I40 generic data model.

Based on these generic concepts, the semantics of generic I4.0 data models (e.g. OPC UA FX, AAS) and specific ADOs can be matched over generic capabilities that describe their functionality (see Fig. 3.5). This automatic alignment reduces the effort of defining an individual mapping to each ADO and instead is done over a generic capability model that is independent of implementation. In the next section, the definition of generic capabilities and their relation to ADOs is explained.

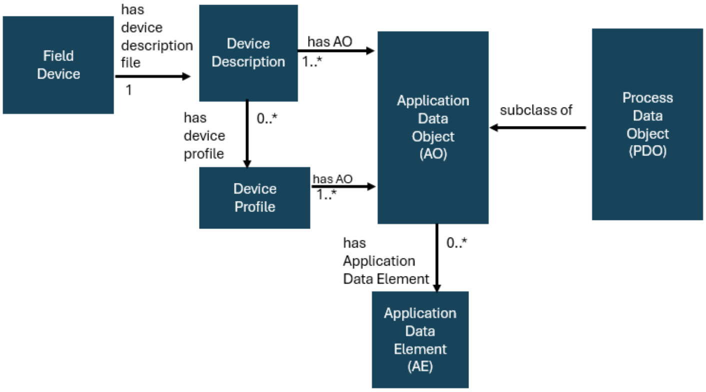

Field Device
has device description file
1
Device Description
has AO
1..*
has device profile
0..*
Device Profile
has AO
1..*
Application Data Object (AO)
subclass of
Process Data Object (PDO)
has Application Data Element
0..*
Application Data Element (AE)

Fig. 3.4. Simplified conceptual model of the I40FD Ontology.

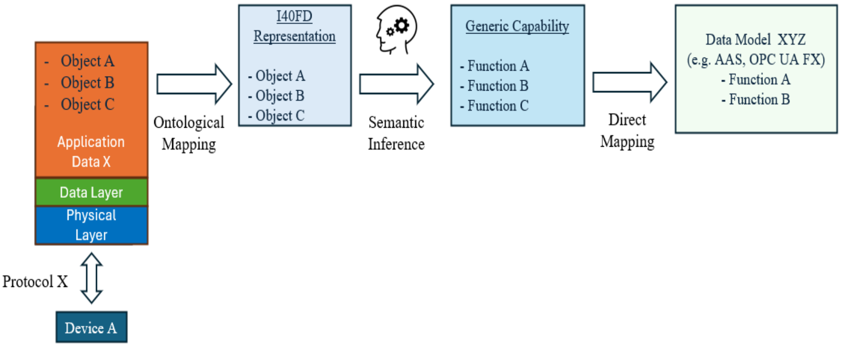


Fig. 3.5. Direct mapping of I40FD ADOs to generic capabilities.

3.3. Generic Field Device Capabilities

Generic capabilities refer to functions that a sensor or actuator can implement to achieve an effect on the physical world. In the literature, the implementation of the functionality of a capability is referred to as a skill [15]. Concrete examples of field device capabilities can be the measurement of a quantity type (e.g., pressure, distance) or the control of a process (e.g., motor speed, valve control). The implementation (i.e., skill) can be a simple analog interface (e.g., 4-20 mA signal) or more complex via the application layer of a communication protocol.

Academic research has shown a focus on a higher level of abstraction of capabilities for manufacturing processes [15-18]. However, to the best of our knowledge, there is no agreed-upon definition of field device capabilities. Using the generic concepts of the I40FD ontology as a reference, generic capabilities for field devices can be derived from the ADOs. Each ADE of an ADO can be associated with process data that either sense a physical process or an actuation that produces work (e.g., displacement, torque). The unit of measurement and the access type of the ADE (i.e., write or read) associated with the process data can be used to infer whether the field device has a sensing or actuating capability (see Fig. 3.6).

The field device capability can then be associated with a generic sensor or actuator definition based on a standardized data dictionary. The generic device type is related to a quantity kind and one or more dictionary definitions (see Fig. 3.7). For example, consider a generic sensor designed to measure electric current. This device would fall under the ECLASS classification of an amperemeter, assigned the International Resource Data Identifier (IRDI) 0173-1#01-AGZ078#020.

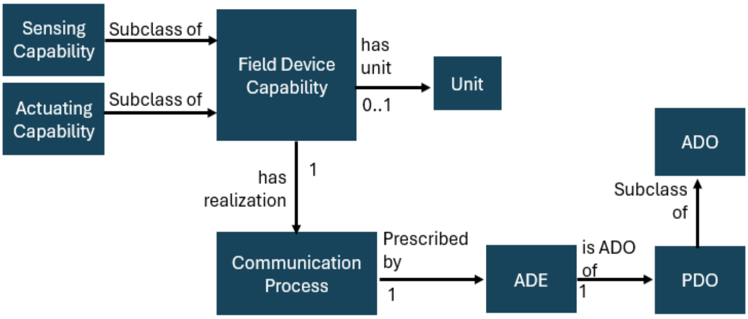


Fig. 3.6. Conceptual model for generic capabilities.

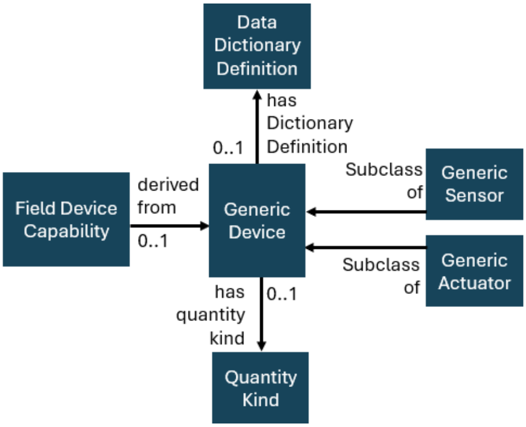


Fig. 3.7. Relationship between a field device capability and a generic device definition.

The relationship between the ADOs, generic capabilities, and standardized data dictionaries enriches the semantic information of the application layer and enables an interoperable description with other systems. The capabilities are not tied to a specific implementation and can be used for example to develop a capability-matching framework [19] or discover available sensing or actuating capabilities for a production process. The semantic model can be used to generate instructions for real-time communication with a field device over a neutral data interface (e.g. OPC UA FX or AAS).

3.4. SWRL Inference Rules

The main benefit of an ontology-based methodology is that it permits the mapping of field device semantics into a neutral format, consequently facilitating the process of knowledge inference. An inference engine takes as input semantic field device instances (individuals) based on the I40FD ontology and outputs the generic capabilities and their relation to a specific implementation.

The OWL syntax allows the definition of axioms that can be used as input for an inference engine. However, not all OWL axioms are able to express the semantics for field devices. To achieve interoperability, new individuals must be generated and associated with specific individuals (i.e., field devices). This is not possible with only the OWL syntax. To enable more comprehensive inferences, we rely on SWRL rules.

An SWRL rule consists of two components: an antecedent (body) and a consequent (head). If the conditions of the antecedent are true, then the conditions of the consequent are also true. An antecedent and a consequent are formed of zero or more atoms. The atoms in these rules may assume one of four forms: C(x), P(x, y), SameAs(x, y), or DifferentFrom(x, y). In each case, C represents an OWL description, P is an OWL property, and x and y are either variables, OWL individuals or OWL data values.

In the following subsections, an explanation of the most important rules that enable the inference of generic capabilities is provided. The graphical representation is done with a modified adaption of the Aided OWL Notation project [20]. For a detailed notation of the rules please consult the documentation of the I40FD ontology [14].

3.4.1. Fundamental Rules

AOs associated with device profiles are also part of the description file that includes this profile (see Fig. 3.8). The device profiles of field devices are typically associated with an identification number. For example, the CAN Open protocol defines a Service Data Object with the index number 4096 for device profiles. A device profile with a specific protocol is associated with a profile identifier. As device descriptions typically provide the encoded device profile value, the actual device profile can be associated with this value (see Fig. 3.9).

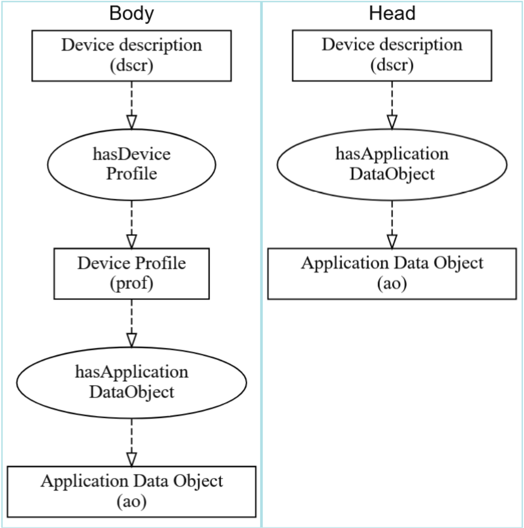

Body
Device description (dscr)
hasDevice Profile
Device Profile (prof)
hasApplication DataObject
Application Data Object (ao)
Head
Device description (dscr)
hasApplication DataObject
Application Data Object (ao)

Fig. 3.8. SWRL inference of AOs from a device profile to a device description.

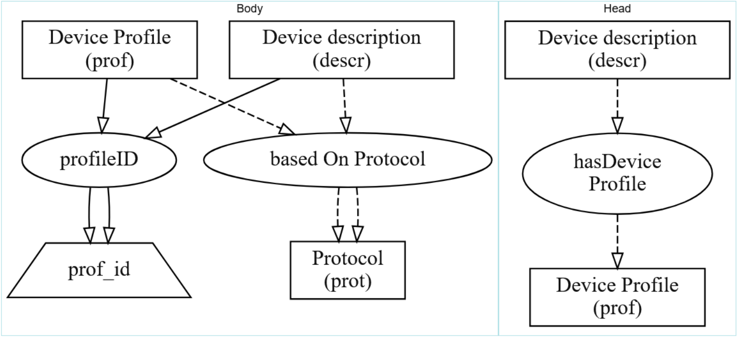


Fig. 3.9. SWRL inference of device profiles from an encoded profile identification.

A field device contains process data with data points that include an encoded unit of measurement. The encoded value is protocol-specific but can be mapped to a physical unit of measurement. The associated unit and the encoded value are instantiated from a semantic vocabulary for that protocol. From the perspective of a device description, only this encoded value is required to infer the respective mapped unit (see Fig. 3.10). The device description can have one or more unit descriptions associated with an ADE that are mapped to the specific protocol encoding.

For example, in IO-Link, an encoded unit with value 1137 can be mapped to the unit of pressure bar. If an IO-Link field device with an ADE is associated with the coded value 1137, we can conclude that this value is mapped to the unit pressure bar.

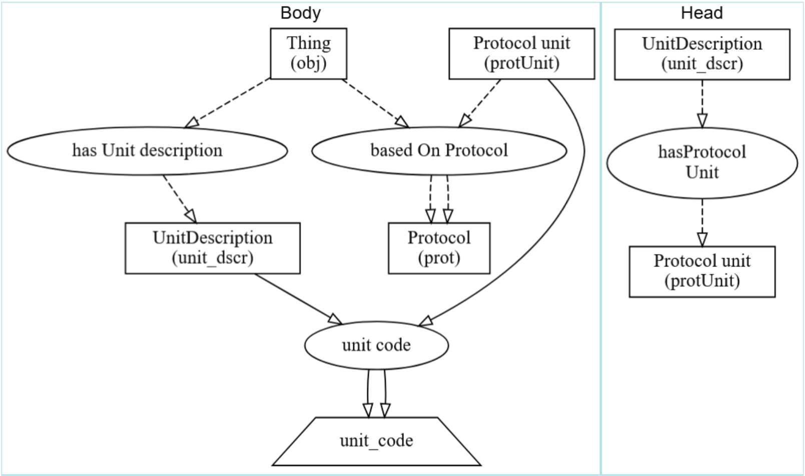


Fig. 3.10. SWRL inference of an encoded unit code to a unit of measurement.

3.4.2. Generic Capability Rules

The rules for the inference of generic capabilities are divided into two categories: the first pertains to sensing capabilities, and the second to actuating capabilities. A generic capability for a field device is inferred based on any ADE that is part of the device description. If an ADE has a unit of measurement associated with a unit description, it can be inferred that the data point from the ADE can be used to either sense or actuate a process. The main differentiation for this is based on the access type. If the ADE has an access type of read, this indicates that the field device has a sensing capability. Conversely, if the ADE can be written to, it has an actuating capability.

The inferred capability is then said to have a realization through a communication process. The communication process consists of access through the specific protocol application layer. The implementation of the capability (i.e., skill) is mapped to an ADE. Fig. 3.11 illustrates the inference of a sensing capability.

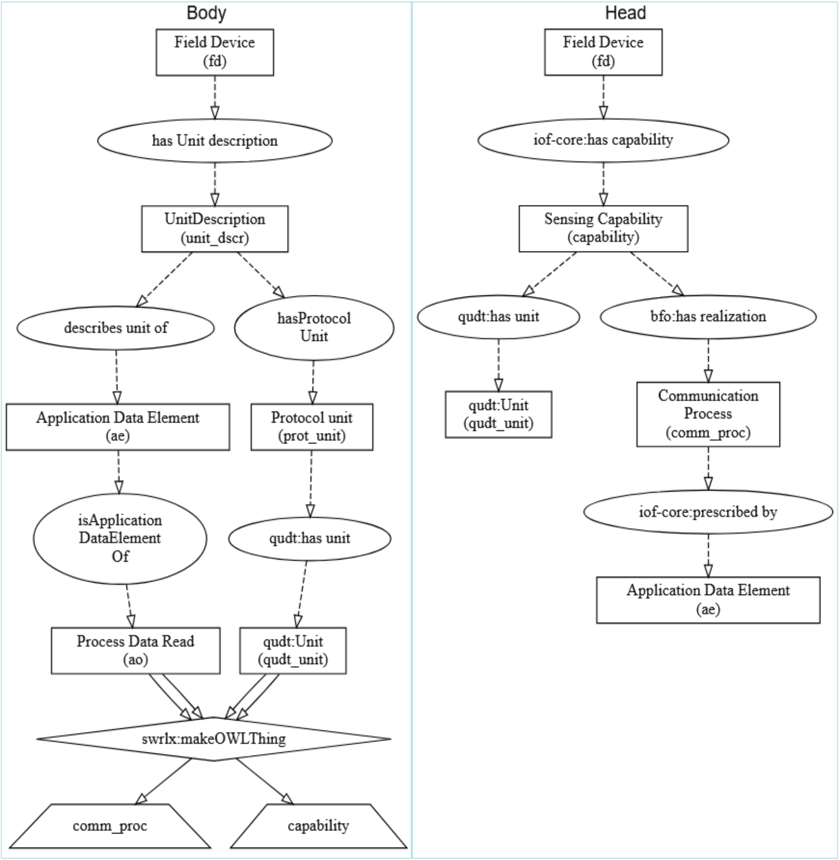


Fig. 3.11. SWRL inference of a field device sensing capability.

The relationship between a generic capability and a generic device type is established through the quantity type that the unit of measure has. A mapping between different data dictionary definitions (e.g. ECLASS, IEC 61360, IEC 61987) and quantity types is used to define an inference with generic definitions. Fig. 3.12 shows an SWRL rule inference for a generic sensor. In this example, a generic sensor consists of one or more data dictionary definitions, and each definition has an IRDI, a preferred name, and a URL.

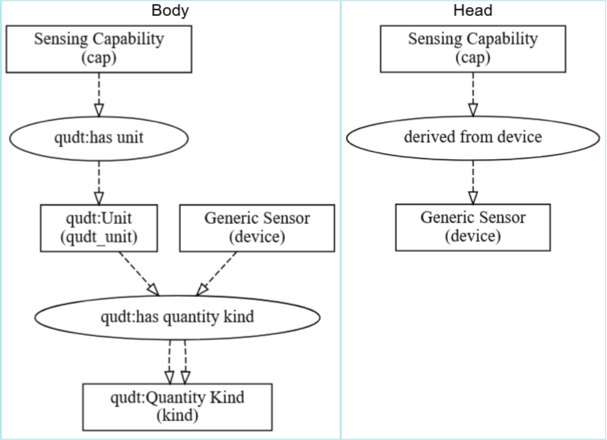


Fig. 3.12. SWRL inference of a generic device from a data dictionary.

3.5. I40FD Inference Engine

In combination with the SWRL rules and the I40FD ontology, an inference engine can process field device instances to automatically infer the generic capabilities. To automate this process, we have developed a software to load the semantics of the I40FD ontology, and process inference requests over an HTTP interface (see Fig. 3.13).

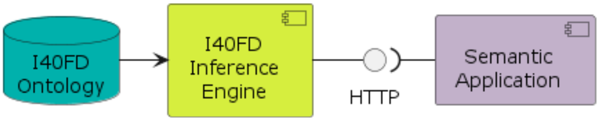


Fig. 3.13. I40FD Inference engine overview.

Since most well-known and maintained software libraries for ontologies and SWRL reasoners are implemented in Java, we decided to keep the development in Java. The SWRL reasoner used for this implementation is based on the SWRL API Drools engine from Stanford University [21]. This engine provides the necessary SWRL built-in atoms to generate new OWL individuals and process the I40FD ontology axioms. The management of OWL axioms has been done with the OWL API project [22].

The I40FD engine takes as input an RDF Turtle representation of a field device and its device description metadata. The axioms are then loaded into the inference engine and the SWRL Drools reasoner is executed. The result of the inference is filtered to include only relevant axioms that contain information about the generic capabilities. The result is then serialized in RDF Turtle format and sent back to the client. Fig. 3.14 shows a sequence diagram illustrating this process.

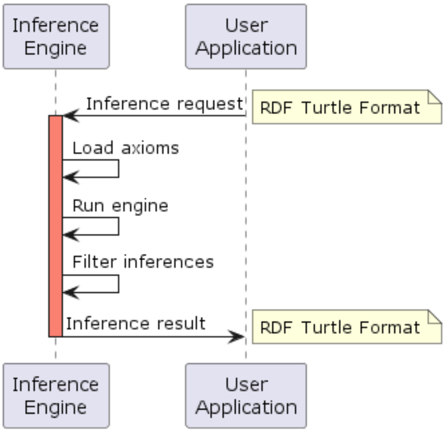


Fig. 3.14. Sequence diagram of the execution of an inference of the I40FD ontology.

The objective of the I40FD engine is to facilitate the automation of semantic inferences and the generation of field device capabilities. A client does not need to load the I40FD ontology or the reasoner; rather, only a field device with the I40FD semantic model in an RDF format needs to be defined. This software is provided with an open-source license and can be found in the git repository of our research institute [23].

3.6. Conclusions and Future Work

This chapter has presented an ontology-based approach to generate generic field device capabilities. Based on the I40FD ontology, we have delved into the semantics of field device semantics, their application layer, and their relation to device description files and units of measurement.

The objective of the proposed work is to reduce the heterogeneity of device models by reducing the efforts to map or align field device protocols to present-day standards for semantic data models, such as OPC UA FX and AAS. Rather than developing an entirely new protocol mapping standard, the existing information from field devices is reused to generalize concepts from the application layer and associate them with specific protocol implementations.

A software-based solution is presented to load the semantics of the I40FD ontology and facilitate the instantiation of field devices through an HTTP interface. A client can parse the information from the device description of a device and obtain its field device capabilities and associated realization through an application layer object.

Currently, the I40FD ontology supports the semantic description of field devices according to the IO-Link and CAN Open standards. Future work will focus on integrating more protocols and automate the semantic mapping to the I40FD ontology. This in turn, will enable advanced use cases such as the automatic matching of field devices based on capabilities, automatic parsing of process data and exchange with other systems (e.g. OPC UA), and offer an interoperable field device interface.

Acknowledgements

The authors would like to thank IFM electronic for the financial support of this research project.

References

[1]. R. Huber, A. M. Oberländer, U. Faisst, M. Röglinger, Disentangling capabilities for Industry 4.0 – an information systems capability perspective, *Information Systems Frontiers*, Vol. 26, 2024, pp. 1667-1695.

[2]. Federal Ministry for Economic Affairs and Climate Action – BMWK, Platform Industrie 4.0, https://www.plattform-i40.de/IP/Navigation/EN/Home/home.html

[3]. M. Iñigo, A. Porto, B. Kremer, F. Larrinaga, J. Cuenca, A. Perez, Towards an Asset Administration Shell scenario: a use case for interoperability and standardization in Industry 4.0, in *Proceedings of the IEEE/IFIP Network Operations and Management Symposium (NOMS'20)*, Hungary, 2020, pp. 1–6.

[4]. Platform Industrie 4.0, Details of the Asset Administration Shell Part 1 – The Exchange of Information Between Partners in the Value Chain of Industrie 4.0, Version 3.0RC02, *Federal Ministry of Education and Research*, 2022.

[5]. VDI/VDE-GMA, VDI/VDE 2193 Part 2 Language for I4.0 Components Interaction Protocol for Bidding Procedures, *Beuth Verlag GmbH*, 2020.

[6]. OPC 10000-1 – Part 1: Overview and Concepts, *OPC UA Foundation*, 2017.

[7]. OPC 10000-80 UAFX: Part 80: Overview and Concepts, *OPC UA Foundation*, 2022.

[8]. J. Nilsson, F. Sandin, Semantic interoperability in Industry 4.0: survey of recent developments and outlook, in *Proceedings of the IEEE 16th International Conference on Industrial Informatics (INDIN'18)*, Jul. 2018, pp. 127-132.

[9]. S. S. Albouq, A. A. A. Sen, N. Almashf, M. Yamin, A. Alshanqiti, N. M. Bahbouh, A survey of interoperability challenges and solutions for dealing with them in IoT environment, *IEEE Access*, Vol. 10, 2022, pp. 36416-36428.

[10]. H. Rahman, M. I. Hussain, A comprehensive survey on semantic interoperability for Internet of Things: State-of-the-art and research challenges, *Transactions on Emerging Telecommunications Technologies*, Vol. 31, Issue 12, 2020, e3902.

[11]. A. Bayha, J. Bock, B. Boss, C. Diedrich, Describing Capabilities of Industrie 4.0 Component, https://www.bmwi.de

[12]. SWRL: A Semantic Web Rule Language Combining OWL and RuleML, https://www.w3.org/Submission/SWRL

[13]. ECLASS e.V., ECLASS Standard, https://eclass.eu/en/eclass-standard

[14]. IEC TR 62390:2005 Common Automation Device – Profile Guideline, https://webstore.iec.ch/publication/6970

[15]. V. Chavez, J. Wollert, An Industry 4.0 ontology-based architecture for interoperability at the field level, in *Proceedings of the 4th IFSA Winter Conference on Automation, Robotics & Communications for Industry 4.0 / 5.0 (ARCI'24)*, 2024, pp. 319-321.

[16]. V. Chavez, J. Wollert, Development of an Industry 4.0 ontology to enable semantic interoperability at the field level, *Sensors & Transducers*, Vol. 265, 2024, pp. 139-147.

[17]. R. Froschauer, A. Kocher, K. Meixner, S. Schmitt, F. Spitzer, Capabilities and skills in manufacturing: a survey over the last decade of ETFA, in *Proceedings of the IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA'22)*, 2022, pp. 1-8.

[18]. X. Ye, J. Jiang, C. Lee, N. Kim, M. Yu, S. H. Hong, Toward the plug-and-produce capability for Industry 4.0: An asset administration shell approach, *IEEE Industrial Electronics Magazine*, Vol. 14, Issue 4, 2020, pp. 146-157.

[19]. L. Alonso, L. Barja, B. Lodeiro, E. Xanthakis, R. Broechler, Asset administration shell modelling and implementation enabling plug and

produce capabilities for modular production, in Flexible Automation and Intelligent Manufacturing: Establishing Bridges for More Sustainable Manufacturing Systems, *Springer*, 2024, pp. 200-207.

[20]. Y. Huang, S. Dhouib, J. Malenfant, AAS capability-based operation and engineering of flexible production lines, in *Proceedings of the 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA'21)*, 2021, pp. 01-04.

[21]. E. Järvenpää, N. Siltala, O. Hylli, H. Nylund, M. Lanz, Semantic rules for capability matchmaking in the context of manufacturing system design and reconfiguration, *International Journal of Computer Integrated Manufacturing*, Vol. 36, Issue 1, 2023, pp. 128-154.

[22]. V. Chavez, Aided OWL Notation Standalone, https://github.com/vChavezB/aowln-sa

[23]. The Board of Trustees of Leland Stanford Junior University, SWRLAPI Drools Engine, https://github.com/protegeproject/swrlapi-drools-engine

[24]. M. Horridge, S. Bechhofer, The OWL API: A Java API for OWL ontologies, *Semantic Web Journal*, Vol. 2, Issue 1, 2011, pp. 11-21.

[25]. V. Chavez, I40FD Inference Engine, https://git.fh-aachen.de/iaam/semantics/i40fd-inference-engine

4.

Cybersecurity Monitoring in Vital Utilities Infrastructure: Integrating Specialized Open-source Intelligence Tools

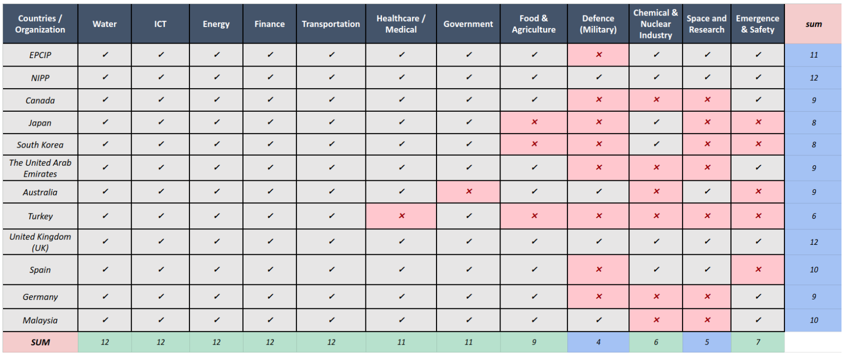
Mert İlhan Ecevit, Furkan Çolhak, Reiner Creutzburg and Hasan Dağ

4.1. Introduction

The rapid digitalization of vital utility infrastructures – such as electricity grids, water supply systems, and natural gas networks – has significantly transformed their operation, offering enhanced efficiency, connectivity, and automation. However, this shift has also introduced substantial cybersecurity vulnerabilities, making these infrastructures prime targets for cyber-attacks. This chapter aims to explore the critical role of cybersecurity in protecting these essential services and to examine how integrating specialized Open-Source Intelligence (OSINT) tools can enhance cybersecurity monitoring within the utility sector.

Critical Infrastructure (CI) is not uniformly defined across different countries and organizations, leading to varying interpretations of what constitutes essential services [1]. Fig. 4.1 illustrates this variability, showing that while sectors like Water, ICT, and Energy are universally recognized as critical, others such as the Chemical & Nuclear Industry, Defence, and Space and Research are deemed critical by only a subset of countries [2]. This divergence in definitions reflects the unique geopolitical, economic, and cultural contexts of each country or organization, influencing which sectors they prioritize for protection [3].

Mert İlhan Ecevit
CCIP, Center for Cyber Security and Critical Infrastructure Protection, Kadir Has University Istanbul, Turkey



Countries / Organization	Water	ICT	Energy	Finance	Transportation	Healthcare / Medical	Government	Food & Agriculture	Defence (Military)	Chemical & Nuclear Industry	Space and Research	Emergence & Safety	sum
EPCIP	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	11
NIPP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
Canada	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓	9
Japan	✓	✓	✓	✓	✓	✓	✓	×	×	✓	×	×	8
South Korea	✓	✓	✓	✓	✓	✓	✓	×	×	✓	×	×	8
The United Arab Emirates	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓	9
Australia	✓	✓	✓	✓	✓	✓	×	✓	✓	×	✓	×	9
Turkey	✓	✓	✓	✓	✓	×	✓	×	×	×	×	×	6
United Kingdom (UK)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	12
Spain	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	×	10
Germany	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓	9
Malaysia	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	10
SUM	12	12	12	12	12	11	11	9	4	6	5	7	

Fig. 4.1. Classifications of Critical Infrastructure by various countries and organizations.

For instance, nations with significant reliance on nuclear energy may classify the Chemical & Nuclear Industry as critical, while others without such reliance may not. Similarly, the emphasis on sectors like Defence and Space and Research varies widely depending on a nation's strategic priorities and security concerns. This variability underscores the importance of tailoring cybersecurity strategies to each country or organisation's specific CI definitions and priorities, ensuring that the most relevant sectors receive the protection they need.

Electricity Grids are among the most critical components of modern infrastructure. These systems generate, transmit, and distribute electricity across vast networks, ensuring a stable and continuous power supply to homes, businesses, and essential services. Electricity grids rely heavily on Supervisory Control and Data Acquisition (SCADA) systems, which monitor and control the flow of electricity across these networks. While essential for operational efficiency, these SCADA systems also represent a significant vulnerability. Cyber-attacks targeting these systems can result in widespread blackouts, disrupting residential and commercial activities and critical services like hospitals and emergency response operations. The cascading effects of power outages on other sectors highlight the crucial role that electricity grids play in maintaining national stability.

Water Supply Systems are equally vital, encompassing the infrastructure involved in sourcing, treating, and distributing water. These systems are essential for public health and safety, ensuring that populations can access clean and safe water. The integration of Industrial Control

Systems (ICS) in water utilities allows for the automated monitoring and regulation of water quality and distribution. However, these systems are also susceptible to cyber threats. The public's health is put at risk when a successful cyber attack on a water supply system contaminates drinking water or stops the water's flow. The potential for physical harm from cyber threats to water distribution infrastructure highlights the need for robust cybersecurity protections.

Natural Gas Networks are another critical component of utility infrastructure, responsible for transporting and distributing natural gas from production facilities to consumers. This infrastructure includes an extensive network of pipelines, storage facilities, and distribution hubs. The operation of these networks relies on sophisticated control systems that manage the flow and pressure of natural gas. A cyber-attack on these control systems results in dangerous situations, such as leaks or explosions, which have catastrophic physical and economic impacts. Disruptions in natural gas supply can also have broader economic implications, affecting industries that rely on a steady supply of energy.

Table 4.1 illustrates the importance of robust security measures across diverse industries by contrasting the essential elements, vulnerabilities, and possible effects of cyberattacks on different utility infrastructures.

Table 4.1. Comparison of Utility Infrastructures.

Infrastructure Type	Key Components	Potential Impact of Cyber-Attacks
Electricity Grids	SCADA Systems, Transmission Lines	Blackouts, Disruption of Dependent Critical Services
Water Supply Systems	Water Treatment Plants, Distribution Control Systems	Public Health Risks, Disruption of Water Supply, Contamination Risk
Natural Gas Networks	Pipelines, Control Systems	Leaks, Explosions, Economic Impact, Supply Disruptions

Cybersecurity in utility infrastructures is not merely a technical issue but a matter of national security. The uninterrupted operation of these services is crucial for public safety, economic stability, and national defence. The convergence of information technology (IT) and operational technology (OT) in power distribution networks has exponentially increased the complexity of security challenges, making traditional security measures inadequate to address growing threats [5]. Furthermore, integrating IoT devices into these systems introduces new vulnerabilities that require advanced solutions [6, 7]. This chapter highlights the importance of a robust cybersecurity framework specifically tailored to the needs of the utility sector, where the stakes are exceptionally high.

The increasing reliance on digital technologies has brought about a paradigm shift in how critical infrastructures function. This reliance allows for real-time monitoring, remote management, and predictive maintenance but also opens the door to various cyber threats that can disrupt operations, compromise data integrity, and even endanger lives. Integrating IoT devices and smart technologies in utility systems further complicates the security landscape by introducing new attack vectors that traditional security measures may not adequately address [6, 8].

Cyber-attacks on utility infrastructures can have far-reaching consequences. Cyber-attacks on utility infrastructures have far-reaching consequences. A successful attack on an electricity grid leads to widespread blackouts, disrupting residential, commercial, and critical services like hospitals and emergency response systems. Similarly, an attack on a water supply system compromises water quality, posing severe health risks to the population. In the case of natural gas networks, a cyber breach leads to explosions or leaks with potentially catastrophic outcomes. These scenarios underscore the need for advanced cybersecurity measures beyond conventional approaches [8, 9]. As highlighted in the literature, the resilience of smart grids against cyber-physical attacks is paramount, given their critical role in maintaining electricity supply and the cascading effects of disruptions on other sectors [8, 10].

The interconnectedness of these infrastructures, especially with energy grids acting as a backbone for other services, amplifies the potential impact of cyber-attacks. This interconnected nature makes it crucial to adopt a holistic approach to cybersecurity that considers the dependencies across different sectors [11]. Blockchain and smart contract technologies offer promising approaches to enhancing the security and resilience of these systems against such attacks, as they can provide decentralized and tamper-proof mechanisms for managing transactions and automating responses to detected anomalies [9, 12].

This chapter focuses on the cybersecurity challenges three key sectors face within vital utility infrastructure: electricity, water, and natural gas. These sectors are chosen for their critical role in sustaining daily life and interdependence with other essential services. The scope of the chapter includes an analysis of the current cyber threat landscape, the vulnerabilities specific to each of these sectors, and the potential of OSINT tools to enhance threat detection and mitigation efforts.

The electricity sector is particularly vulnerable due to the increasing integration of IoT devices and smart technologies, which expand the attack surface for cyber threats [13]. In the water sector, cybersecurity challenges involve the risk of tampering with water treatment processes, leading to contamination and public health crises. The natural gas sector faces the threat of cyber-attacks that disrupt supply chains or cause physical damage to infrastructure [14, 15]. Moreover, the interconnectedness of these infrastructures amplifies the impact of cyber-attacks, necessitating a comprehensive and coordinated approach to cybersecurity [11].

By focusing on these sectors, this chapter aims to provide a comprehensive overview of the cybersecurity challenges within vital utility infrastructure and to explore how OSINT tools can be leveraged to address these challenges effectively. The integration of OSINT tools into cybersecurity strategies offers a proactive approach to threat detection, enabling utilities to anticipate and mitigate cyber threats before they can cause significant harm.

In the following sections, we will delve deeper into the cyber threat landscape, examine the role of OSINT tools in enhancing cybersecurity monitoring, and provide practical recommendations for integrating these tools into the cybersecurity frameworks of utility sectors. This investigation aims to support the continuous endeavours to safeguard vital utility infrastructure from the constantly changing threat environment and guarantee these essential services' continuous dependability and security.

4.2. Understanding the Cybersecurity Landscape

To fully understand the complexities of securing critical infrastructure, it is first essential to explore the evolving cybersecurity landscape and the specific threats these vital systems face.

The rapid digitalization of critical infrastructure has created a dual-edged sword: while it has revolutionized the efficiency and interconnectedness of sectors such as electricity, water, and natural gas, it has also introduced a myriad of vulnerabilities that cyber adversaries are keen to exploit. As we delve into the cybersecurity landscape, it is crucial to understand the nature and evolution of the threats that confront these vital infrastructures.

4.2.1. Cyber Threats Overview

The cybersecurity landscape for critical infrastructure is fraught with diverse and increasingly sophisticated threats. Understanding these threats is essential for developing robust defence mechanisms. Critical infrastructures are susceptible to a range of cyber-attacks, with malware, phishing, DDoS, and APTs being the most prevalent. These attacks can devastate essential services, mainly when directed at Industrial Control Systems (ICS) [16]. Although other significant threats, such as Man-in-the-Middle (MitM) attacks and zero-day exploits, also pose considerable risks, this study concentrates on the four primary threats due to their widespread incidence and their unique challenges in securing ICS environments.

To improve the presentation of these threats, Fig. 4.2 illustrates the various cyber threats that ICSs are exposed to, underscoring the critical need for robust cybersecurity measures across all potential attack vectors, and Table 4.2 provides an overview of the key types of cyber threats, their characteristics, and examples of real-world incidents, illustrating the diverse challenges faced by critical infrastructures.

Malware is a broad category encompassing any software designed to disrupt, damage, or gain unauthorized access to computer systems. In the context of critical infrastructure, malware can be particularly insidious, targeting the industrial control systems (ICS) that manage essential utilities. The Stuxnet worm serves as a stark example, targeting Iran's nuclear facilities and revealing how malware can cause real-world destruction through cyber means [5, 7]. Similarly, the 2015 Ukraine Power Grid Attack highlighted the catastrophic effects malware can have on a country-wide level [22]. Hackers employed BlackEnergy malware to gain access to the Ukrainian power grid, leading to a temporary blackout that affected 225000 people. The attack demonstrated not only the vulnerabilities of power systems but also the potential for cyber-attacks to disrupt essential services on a massive scale [8, 13].

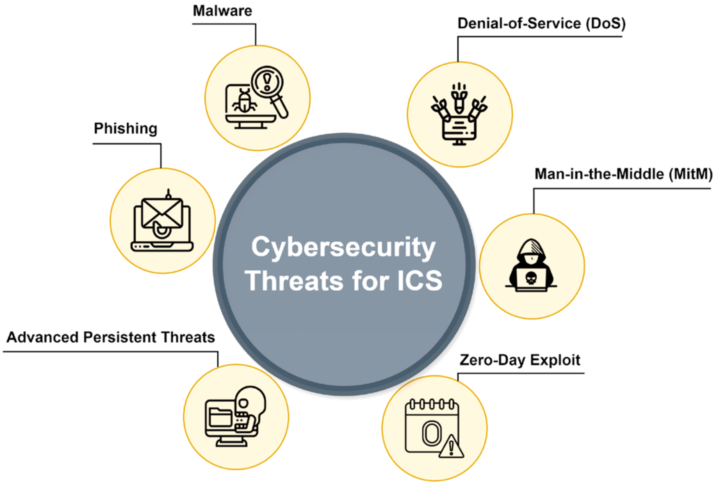


Fig. 4.2. Cyber threats for Industrial Control Systems.

Table 4.2. Overview of Cyber Threats.

Threat Type	Characteristics	Examples
Malware	Disrupts, damages, or gains unauthorized access to systems	Stuxnet [17], BlackEnergy [18]
Phishing	Deceives individuals into divulging sensitive information	Colonial Pipeline Ransomware Attack [19]
Distributed Denial of Service (DDoS)	Overwhelms systems with internet traffic, rendering them inaccessible	New Zealand Stock Exchange Attack [20]
Advanced Persistent Threats (APTs)	Stealthy, long-term infiltration for espionage or sabotage	SolarWinds Orion Hack [21]

Phishing is another prevalent threat, often serving as the entry point for more extensive cyber-attacks. Phishing attacks entail tricking people into disclosing private information, like login passwords, by using phony emails or websites that seem authentic. The Colonial Pipeline Ransomware Attack in 2021, for instance, began with a compromised VPN account, likely obtained through phishing. Using this access, the attackers were able to install ransomware, which caused the pipeline to be temporarily shut down and resulted in major fuel shortages throughout the Eastern United States [23, 24].

Attacks known as Distributed Denial of Service (DDoS) aim to block legitimate users from accessing systems, servers, or networks by flooding them with internet traffic. DDoS attacks are especially dangerous for utilities since their internet systems need to be continuously accessed for control and monitoring. These attacks can be used as a diversionary tactic, masking other malicious activities, or as a direct attempt to disrupt service availability. The 2020 attack on New Zealand's stock exchange is a recent example of how DDoS attacks can cripple critical online services [15, 23].

One of the biggest threats to vital infrastructure is the Advanced Persistent Threat (APT). APTs are distinguished from other attack types by their stealth and persistence, as opposed to their potential for speed and overtness. These threats frequently come from well-resourced state-sponsored organizations that want to penetrate and stay inside a system for a long time. Usually, espionage, sabotage, or the retrieval of confidential information are the objectives. One of the best examples of an APT is the 2020 discovery of the SolarWinds Orion software exploit. Malicious code was included by hackers in Orion platform software upgrades, which are utilized by thousands of companies, including US government institutions. This breach allowed the attackers to have prolonged access to highly sensitive networks, demonstrating the far-reaching implications of APTs [25, 26].

4.2.2. Evolution of Threats

The nature of cyber threats is continuously evolving, driven by advancements in technology and changes in attacker motivations [27]. These developments have significantly impacted the cybersecurity landscape, requiring organizations to adapt their defences accordingly. Fig. 4.3 illustrates the timeline of major cyber threats, from the rise of early malware in 1988 to the anticipated challenges posed by quantum computing in 2025. One of the most significant developments in recent years is the rise of AI-driven attacks.

Cybercriminals are using artificial intelligence and machine learning to create increasingly advanced malware and phishing schemes that can adapt and avoid detection by conventional security measures. For example, AI-driven malware is much harder to counter because it may adapt its behaviour to escape detection by learning from its surroundings [28, 29].

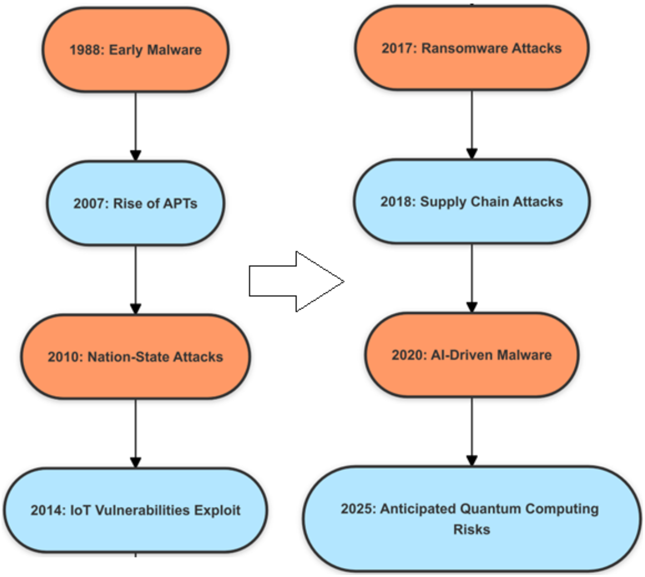

1988: Early Malware
2007: Rise of APTs
2010: Nation-State Attacks
2014: IoT Vulnerabilities Exploit
2017: Ransomware Attacks
2018: Supply Chain Attacks
2020: AI-Driven Malware
2025: Anticipated Quantum Computing Risks

Fig. 4.3. Timeline of Threat Evolution.

The use of new technologies, such as 5G networks and the Internet of Things (IoT), is another crucial area of concern. The number of possible entry points for attackers has skyrocketed as a result of the integration of IoT devices into critical infrastructure. These gadgets are susceptible to hacking because they frequently lack strong security safeguards. An IoT device that has been compromised can act as a gateway to the larger network, allowing attackers to carry out more sophisticated assaults. The vulnerabilities associated with insecure IoT devices are demonstrated by the 2016 Mirai botnet assault, which used IoT devices to conduct one of the most significant DDoS attacks in history [30-32].

Similarly, the rollout of 5G technology, while offering enhanced connectivity and speed, also introduces new vulnerabilities. The increased bandwidth and lower latency of 5G networks facilitate the deployment of IoT devices and cloud-based services in critical infrastructure, but they also expand the attack surface. Cyber adversaries are already exploring ways to exploit these vulnerabilities, potentially leading to more complex and large-scale attacks [15, 33, 34].

Cyber threats have also evolved as a result of geopolitical unrest and the escalation of cyberwarfare. Critical infrastructure is being targeted by state-sponsored entities more frequently in an effort to achieve strategic goals or apply political pressure. Because these attacks frequently have greater resources and sophistication, it is more difficult to fight against them. An increasing trend of cyber warfare directed at critical infrastructure is demonstrated by the attacks on Ukraine's power grid in 2015 and the NotPetya attack in 2017, both of which had an impact on the entire world despite targeting Ukraine [35, 36].

Attacks on the supply chain have recently become an imminent danger. Attackers can compromise hardware or software makers to introduce dangerous components into commonly used products, as demonstrated by the SolarWinds hack. Once several organizations receive these infected products, a widespread vulnerability that can be widely exploited is created. This kind of attack emphasizes how crucial it is to safeguard the immediate systems and the larger ecosystem that supports critical infrastructure [37-39].

In conclusion, a wide range of threats that are constantly developing in complexity and sophistication characterize the cybersecurity landscape for critical utility infrastructure. The need for strong, flexible cybersecurity solutions has never been higher, from more sophisticated threats like AI-driven attacks and the exploitation of IoT and 5G technologies to more conventional dangers like malware and phishing. Developing solutions to safeguard the vital systems that support our civilization requires understanding these dangers and how they have evolved [9, 40, 41].

By comprehensively understanding the cybersecurity landscape, stakeholders in the utility sector can better anticipate and mitigate the risks associated with these evolving threats, ensuring the resilience and reliability of essential services.

4.3. Fundamentals of Cybersecurity Monitoring

In the realm of critical infrastructure, where the stakes include national security, public safety, and economic stability, the importance of effective cybersecurity monitoring cannot be overstated. Cybersecurity monitoring forms the backbone of a robust defence strategy, enabling organizations to detect, respond to, and mitigate cyber threats in real-time. This section delves into the fundamental concepts of cybersecurity monitoring, explores the criticality of protecting vital infrastructure, and examines the tools and techniques that underpin these efforts.

4.3.1. Key Concepts

Threat detection, incident response, and security posture assessment are the three main ideas that underpin cybersecurity monitoring and serve as the cornerstone of a robust security strategy. Fig. 4.4 illustrates the interrelationship between these key concepts, demonstrating how they work together to form a comprehensive cybersecurity monitoring framework.

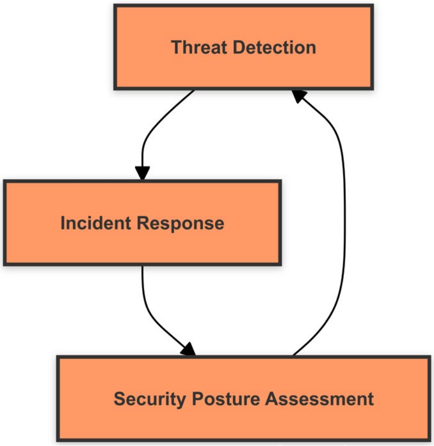


Fig. 4.4. Interrelationship between Key Concepts.

The practice of locating possible security risks within a system or network is known as threat detection. This entails keeping an eye on user activity, system logs, and network traffic continuously in order to spot any irregularities that points to a security breach. To effectively identify existing risks, such malware and phishing efforts, as well as developing dangers that are not yet be listed in threat databases, a combination of automated tools and human skills is needed. The capacity to identify threats quickly is crucial when it comes to vital infrastructure, as even little disruptions can have serious repercussions [5, 12, 13]. Incident Response refers to the actions taken once a threat is detected. Reducing the impact of a security incident, containing the threat, and promptly returning to regular operations are the objectives of incident response. To guarantee that all stakeholders are informed and coordinated during a crisis, a well-structured incident response plan comprises set procedures for various incident types, defined roles and duties for team members, and communication protocols. Collaboration between IT security teams, operational technology (OT) teams, and external entities, including government agencies and law enforcement, is common in critical infrastructure incident response [9, 37, 42].

Security Posture Assessment is the ongoing evaluation of an organization's security measures to determine their effectiveness in protecting against cyber threats. This assessment involves identifying vulnerabilities, assessing the potential impact of different types of attacks, and ensuring that security controls are aligned with the latest threat landscape.

Table 4.3 outlines the key concepts of cybersecurity monitoring, which are fundamental to maintaining the integrity of critical operations within utility infrastructures.

Table 4.3. Key Concepts of Cybersecurity Monitoring.

Concept	**Definition**	**Importance in Critical Infrastructure**
Threat Detection	Identifying potential security threats within a network or system	Crucial for early identification of cyber-attacks
Incident Response	Actions taken once a threat is detected to minimize impact	Ensures quick containment and restoration of operations
Security Posture Assessment	Ongoing evaluation of an organization's security measures	Ensures defences are adequate and adaptable to new risks

Critical infrastructure must undergo a complete security posture review to ensure that defences are both sufficient to counter current threats and adaptable enough to respond to new ones. This process often involves regular compliance audits, penetration tests, and vulnerability scanning in order to identify and address vulnerabilities before they are exploited [10, 24, 43].

These fundamental ideas work together to give businesses the structure for a proactive and adaptable approach to cybersecurity monitoring, allowing them to fend off threats and preserve the integrity of their vital processes.

4.3.2. Critical Infrastructure Security

The term "critical infrastructure" describes the physical and virtual resources and systems that are so essential to a country that their failure or destruction would severely affect public health and safety, national security, economic security, or any combination of these. This covers industries including telecommunications, emergency services, electricity, water, and transportation. Since these infrastructures are essential to how society and the economy operate, their security is of utmost importance [13, 14, 35].

Vulnerabilities in critical infrastructure arise from several factors, including the increasing reliance on digital technologies, the integration of legacy systems with modern networks, and the growing interconnectivity between different infrastructure sectors. For instance, industrial control systems (ICS), which were not created with cybersecurity in mind, are nevertheless used by a lot of utilities. These systems, which regulate everything from water treatment facilities to electricity grids, are frequently linked to business IT networks, opening doors for hackers to compromise systems and cause disruptions. Furthermore, the introduction of new technologies like 5G networks and the Internet of Things (IoT) has increased attack surface and introduced new vulnerabilities that cyber adversaries can take advantage of [5, 28, 40].

The resilience of critical infrastructure to anticipate, withstand, and recover from disruptive events – including cyberattacks – is known as resilience. In order to build resilience, security measures must be strengthened, and infrastructure must be made sure that operations can be swiftly restarted in the event that it is hacked. Strong cybersecurity procedures, redundancy in vital systems, and extensive incident response and recovery plans are all necessary for this. Implementing backup power sources and redundant communication lines, for instance, can assist in guaranteeing that critical services continue to function even in the event of an attack. Likewise, creating and routinely testing incident response procedures can help lessen the impact of an attack and drastically cut down on downtime [40, 44].

Given the essential role that critical infrastructure plays in society, protecting these systems from cyber threats is a top priority. The resilience of these infrastructures is not only a matter of national security but also a key factor in maintaining public trust and ensuring the continuity of essential services.

4.3.3. Existing Monitoring Tools

Organizations rely on a variety of cybersecurity tools and technologies that are made to identify, prevent, and respond to threats in order to monitor and safeguard critical infrastructure. Intrusion Prevention Systems (IPS), Security Information and Event Management (SIEM) systems, and Intrusion Detection Systems (IDS) are some of the most popular.

Tools called intrusion detection systems (IDS) keep an eye on network traffic to look for indications of malicious behaviour or policy infractions. Typically, an intrusion detection system (IDS) detects abnormal network activity by utilizing the behavioural analysis to examine network packets and compare them to a database of known attack signatures. The IDS sends out an alarm when it finds a possible threat, giving security professionals time to look into it and take appropriate action. Critical infrastructure can benefit from an extra layer of protection by having intrusion detection systems (IDS) installed at several locations within a network to monitor traffic entering and departing critical systems [11, 42, 45]. Fig. 4.5 shows how OSINT tools like **Shodan** and **Censys** can enhance IDS by identifying vulnerable, internet-facing devices, while **Nmap** assists in mapping open ports, and **Snort** contributes to deeper packet inspection, thereby improving the overall detection capabilities.

The capabilities of IDSs are enhanced by Intrusion Prevention Systems (IPS), which not only identify threats but also take automatic measures to stop them from succeeding. Malicious traffic can be stopped, shady connections can be closed, and network device configurations can even be changed by an IPS to isolate compromised systems. By preventing attacks before they start, this proactive strategy lessens the chance that vital infrastructure may be harmed. In settings like power grid control systems or emergency communication networks, where quick reaction times are crucial, intrusion prevention systems (IPS) are especially helpful [8, 23, 33]. Fig. 4.6 illustrates how OSINT tools like **OpenVAS** and **Masscan** can be integrated with IPS to provide continuous vulnerability scanning and rapid port detection, while tools like **tcpdump** and **Wireshark** offer detailed packet analysis, which is essential for identifying and mitigating sophisticated threats.

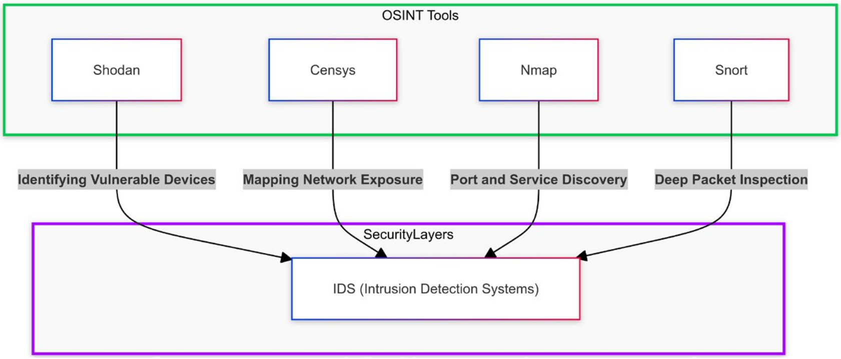


Fig. 4.5. OSINT Tool Functionalities for Intrusion Detection Systems.

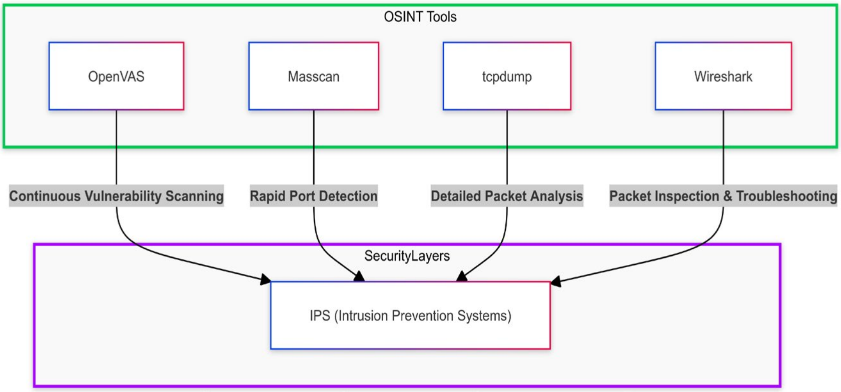

OSINT Tools
OpenVAS
Masscan
tcpdump
Wireshark
Continuous Vulnerability Scanning
Rapid Port Detection
Detailed Packet Analysis
Packet Inspection & Troubleshooting
SecurityLayers
IPS (Intrusion Prevention Systems)

Fig. 4.6. OSINT Tool Functionalities for Intrusion Prevention Systems.

Systems that gather and analyze security data from all areas of an organization's IT environment are called Security Information and Event Management (SIEM) systems. SIEM systems provide a consolidated picture of an organization's security posture by combining logs, alarms, and other data from many security technologies. Security teams can respond more skilfully when SIEM systems correlate information from disparate sources and spot trends that can point to a planned assault. SIEM systems are essential for ensuring that any threats are identified and dealt with before they have a chance to escalate, especially for critical infrastructure, where a security compromise could have dire repercussions [12, 34]. Fig. 4.7 depicts how OSINT tools like **RiskIQ** and **TheHarvester** can enrich SIEM systems with external threat intelligence and domain-specific data, enhancing their ability to identify and respond to threats from a broader context.

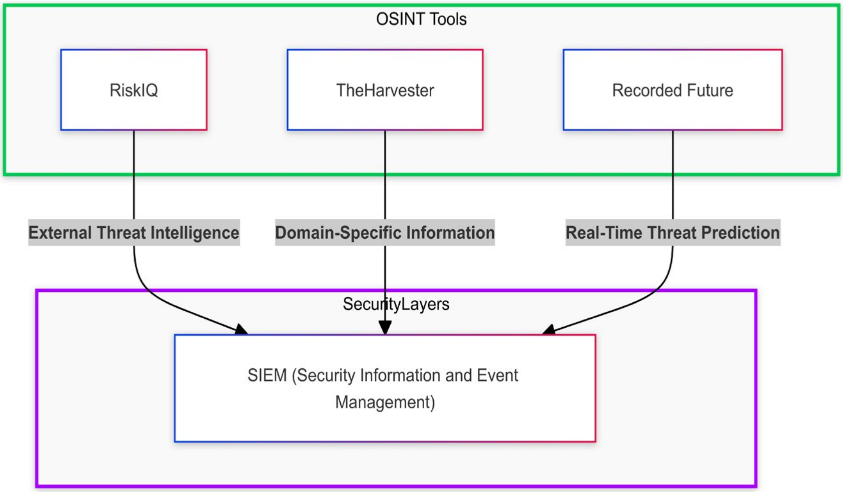

OSINT Tools
RiskIQ
TheHarvester
Recorded Future
External Threat Intelligence
Domain-Specific Information
Real-Time Threat Prediction
SecurityLayers
SIEM (Security Information and Event Management)

Fig. 4.7. OSINT Tool Functionalities for Security Information and Event Management Systems.

The technologies and tools used for cybersecurity monitoring need to change as cyber threats do. Integrating Open-Source Intelligence (OSINT) tools into conventional monitoring frameworks is one of the newer trends in this sector. To offer more context and insight into prospective threats, open-source intelligence (OSINT) programs gather and examine data from publicly accessible sources, including social media, forums, and news websites. Organizations can improve their overall security posture, discover new threats, and obtain a more thorough awareness of the threat landscape by utilizing OSINT [15, 25, 39].

The introduction of OSINT tools represents a natural evolution in cybersecurity monitoring, bridging the gap between traditional, internal-focused security measures and the broader, external intelligence that can be gathered from the public domain. In the following section, we will explore the role of OSINT tools in greater detail and discuss how they can be effectively integrated into cybersecurity strategies for critical infrastructure.

4.4. Role of OSINT Tools in Cybersecurity

4.4.1. Introduction to OSINT

Using information that is readily available to the public, open-source intelligence (OSINT) helps to detect, anticipate, and counteract possible cyberthreats. A vast array of data sources is included in OSINT, such as official documents, social media sites, online forums, news websites, and technical blogs [46-48]. This data is gathered, examined, and turned into intelligence that may be used to take action and improve an organization's security posture.

Within the cybersecurity domain, OSINT serves several critical functions. One of the most vital is proactive threat hunting, where security teams use OSINT tools to identify emerging threats before they manifest into active attacks. By continuously monitoring public sources, organizations can detect early warning signs of impending cyber incidents, such as discussions about new vulnerabilities or the appearance of malicious actors planning an attack [48].

An additional critical component of OSINT is digital footprinting. This entails obtaining comprehensive data regarding the internet presence of a business, encompassing its publicly accessible infrastructure, domain registrations, and exposed services [46, 47]. Cybersecurity experts can spot any weaknesses that attackers may be using by analyzing an organization's digital footprint [47].

Additionally, OSINT is invaluable in defending against social engineering attacks. By analyzing publicly available information, organizations can identify how much sensitive data is accessible and potentially exploitable by malicious actors. This awareness helps in creating strategies to reduce exposure and mitigate the risk of social engineering attacks [46, 48].

4.4.2. Advantages of Using OSINT Tools in Cybersecurity Efforts

The adoption of OSINT tools within cybersecurity frameworks offers several advantages, making them an essential component of a comprehensive defence strategy.

Cost-Effectiveness: Many OSINT tools are open-source or available at low cost, making them accessible to organizations of all sizes. This cost-effectiveness does not compromise their utility; in fact, the open-source nature often leads to continuous improvement and updates driven by a community of users. [46, 48].

Real-Time Threat Intelligence: Real-time information gathering and analysis are possible with OSINT capabilities, giving quick insights into new dangers. This ability shortens the time between threat discovery and mitigation by enabling organizations to react swiftly to possible threats [46, 48].

Comprehensive Coverage: OSINT tools provide extensive coverage spanning numerous data sources. Through the consolidation of information from several platforms, such as social media and deep web forums, OSINT offers a more comprehensive perspective of the threat environment. By using a comprehensive strategy, dangers that are overlooked by monitoring technologies with a narrower focus can be identified [46-48].

With a solid understanding of the role OSINT tools play in cybersecurity, the next step is to explore how these tools can be systematically integrated into the cybersecurity frameworks of critical infrastructures.

4.4.3. Overview of Specialized OSINT Tools for Cybersecurity

To maximize the benefits of OSINT, cybersecurity professionals rely on a suite of specialized tools, each tailored to specific aspects of intelligence gathering and analysis. Below is an overview of some of the most well-known OSINT tools and their applications in enhancing cybersecurity:

The Harvester: This tool is designed to gather OSINT on an organization’s external digital footprint. It collects information such as email addresses, subdomains, IPs, and URLs, making it an excellent resource for mapping out an organization's publicly accessible infrastructure [46].

Shodan: Often referred to as the "search engine for hackers," Shodan scans the internet for connected devices and services, offering insights into potential vulnerabilities. Shodan is particularly useful for identifying internet-facing industrial control systems (ICS), making it a critical tool for securing critical infrastructure [49].

Maltego: Maltego provides a powerful graphical link analysis platform that allows users to visualize relationships between entities based on the data collected. It helps in identifying networks and patterns related to cyber threats, which is crucial for understanding the broader context of a security incident [50].

Censys: Censys continuously analyzes the entire internet, providing information on devices and networks. It serves as a valuable tool for cybersecurity professionals seeking to understand the exposure and vulnerability of their networks [50].

Zoomeye: Similar to Shodan, Zoomeye is a cybersecurity search engine focused on internet-connected devices. It provides detailed information about exposed services and vulnerabilities, making it a vital tool for organizations looking to secure their online assets [51].

Thingful: Internet of Things (IoT) devices worldwide are indexed and located using this search engine. Thingful assists in monitoring IoT devices to make sure they don't serve as entry points for cyberattacks as these devices are progressively integrated into vital infrastructure [52].

RiskIQ: The RiskIQ API offers digital threat management solutions with a focus on identifying threats outside the firewall. It provides insights into external risks, including phishing attempts and malicious domains, which are crucial for protecting organizational assets [53].

Recorded Future: This tool offers threat intelligence by analyzing the internet, including the open web, dark web, and technical sources. It uses machine learning to provide predictive insights, helping organizations anticipate and mitigate threats before they can cause harm [54].

Nmap: Nmap is a network scanning tool that discovers devices and services on a network. It is widely used for vulnerability scanning and is an essential tool for identifying open ports and services that are potentially exploited by attackers [55].

OpenVAS: An open-source framework for vulnerability scanning and management, OpenVAS is a comprehensive tool for identifying security issues in networked systems. It is particularly valuable for organizations looking to maintain a strong security posture without incurring high costs [56].

Masscan: Known for its speed, Masscan is capable of scanning the entire internet in under six minutes. It is a powerful tool for identifying open ports and vulnerable services on a global scale [57].

Wireshark: With the help of the network protocol analyzer Wireshark, users can record and view live network traffic. It is a crucial instrument for identifying possible security breaches and resolving network problems [58].

tcpdump: Tcpdump is a command-line packet analyzer used for network debugging and monitoring. It is highly effective for capturing and analyzing traffic at a granular level, making it a critical tool for network security [59].

PcapXray: This network forensics tool visualizes packet captures and network flows. PcapXray helps in understanding the flow of data through a network, which is essential for identifying potential breaches and securing network architecture [60].

Snort: An open-source network intrusion prevention system, Snort is capable of performing real-time traffic analysis and packet logging. It is widely used for detecting and preventing attacks on networks [61].

Zigbee2MQTT: This tool acts as a bridge for Zigbee devices to MQTT, allowing integration with home automation systems. As IoT devices increasingly become targets for cyber-attacks, tools like Zigbee2MQTT are essential for securing smart home environments [62].

ZMap: A fast network scanner optimized for internet-wide network surveys, ZMap can scan the entire public IPv4 address space. It is invaluable for identifying vulnerable devices and services across the internet [63].

These tools represent a broad spectrum of OSINT capabilities, each playing a unique role in enhancing cybersecurity. When integrated into an organization's security framework, they offer a powerful means to detect, analyze, and respond to cyber threats effectively.

Given the current state of threats, integrating OSINT techniques into cybersecurity procedures is not only a need but also a benefit [34]. These tools give businesses a thorough, up-to-date picture of the hazards, enabling them to take preemptive measures to fend against attacks. The importance of OSINT in guaranteeing the security and resilience of crucial infrastructure will only increase as the cyber threat landscape changes.

4.5. Integration of OSINT Tools into Cybersecurity Strategies for Critical Infrastructures

The integration of Open-Source Intelligence (OSINT) tools into cybersecurity strategies for critical infrastructures is a vital task that requires meticulous planning, precise execution, and continuous management. Critical infrastructures, such as power grids, water supply systems, and transportation networks, are particularly vulnerable to cyber threats due to their importance in maintaining societal functions. OSINT tools, when effectively integrated, can significantly enhance a critical infrastructure's ability to anticipate, detect, and respond to these threats. However, the complexity of this integration in such vital systems necessitates adherence to best practices and a clear understanding of potential challenges specific to critical infrastructures.

4.5.1. Best Practices for OSINT Integration in Critical Infrastructures

Integrating OSINT tools into cybersecurity strategies for critical infrastructures involves several key best practices, each aimed at maximizing the effectiveness of these tools while ensuring alignment with the unique needs and compliance requirements of these sectors [64, 65]. The process of integrating OSINT tools into cybersecurity strategies for critical infrastructures can be effectively visualized in Fig. 4.8.

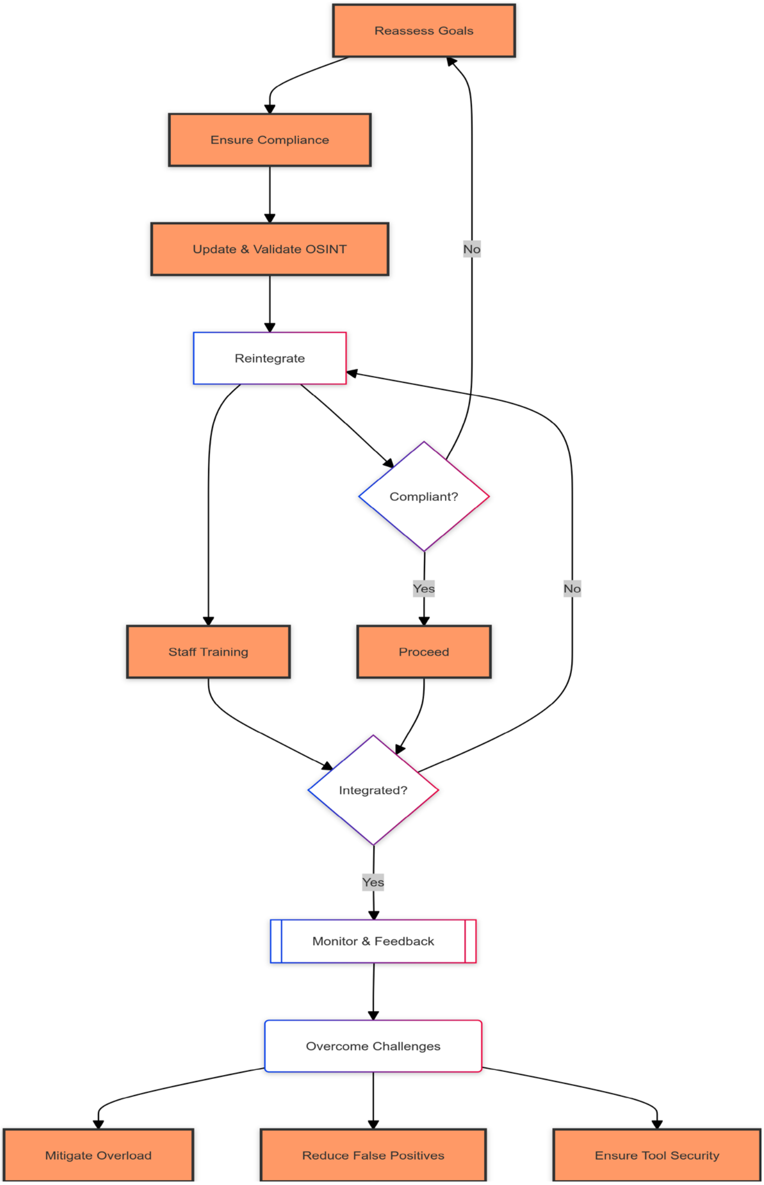


Fig. 4.8. OSINT Integration Flowchart.

Define Clear Objectives: The first step in integrating OSINT tools is to define clear, actionable objectives tailored to the critical infrastructure's unique operational environment. Critical infrastructure organizations must establish what they hope to achieve through the use of OSINT – whether it's enhancing threat detection in SCADA systems, improving incident response in power grids, or gaining insights into vulnerabilities in water supply networks [48, 65-67].

By setting specific goals, these organizations can select the most appropriate OSINT tools and methodologies that align with their critical operations [64,68]. For example, **Shodan** is particularly useful for identifying internet-connected devices and understanding the exposure of SCADA systems to the internet. **Censys** can be used to track and profile devices based on their response to internet scans, which is crucial for vulnerability management in critical infrastructures.

Ensure Compliance with Legal and Ethical Standards: The use of OSINT tools in critical infrastructures must adhere to strict legal and ethical standards to prevent data misuse and protect individual rights. Organizations must ensure compliance with regulations such as GDPR in Europe and NERC CIP in North America [48, 68]. The Berkeley Protocol emphasizes the importance of conducting OSINT activities with respect for human dignity, privacy and implementing safeguards to prevent harm or rights infringements [66]. Tools like Maltego, which provides detailed visualizations of entity relationships, and RiskIQ, which manages digital risk through external threat intelligence, must be used in a manner that aligns with these legal and ethical frameworks.

Moreover, ethical considerations, as outlined in the Berkeley Protocol, require that OSINT activities be transparent, accountable, and purposeful. Organizations should establish guidelines and oversight mechanisms to prevent misuse, ensuring that tools like RiskIQ are employed in ways that respect privacy and ethical standards [66]. In summary, integrating OSINT tools into cybersecurity frameworks provides significant benefits but must be done within a framework that rigorously upholds legal and ethical principles, thereby protecting critical infrastructures while respecting individual rights.

Regularly Update and Validate OSINT Sources: The dynamic nature of cyber threats to critical infrastructures means that the information gathered through OSINT tools can quickly become outdated. To maintain the accuracy and relevance of intelligence, organizations must regularly update and validate their data sources [48, 67]. **Recorded Future** is an essential tool in this regard, as it aggregates real-time data from a variety of sources, including the deep web, ensuring that critical infrastructure organizations are always working with the most current threat intelligence. **Zoomeye**, similar to Shodan, also needs regular updates to its scan results to maintain relevance in tracking the exposure of critical infrastructure devices.

Integrate OSINT with Existing Cybersecurity Tools: Integration of OSINT with current cybersecurity tools and systems, such as Security Information and Event Management (SIEM), Intrusion Detection System (IDS), and Intrusion Prevention System (IPS), is necessary to fully realize the potential of OSINT in critical infrastructures [48, 66, 68]. To provide a holistic security posture, for example, integrating OSINT data with tools like **RiskIQ** or an SIEM platform like **Splunk** helps improve the correlation between external threat intelligence and internal security logs. This information is especially helpful in thwarting phishing attacks targeting workers involved in key infrastructure [64]. By gathering information on subdomains, IP addresses, and email addresses, **TheHarvester** may be connected with SIEMs to offer real-time insights into the organization's vulnerability. This information is especially helpful in thwarting phishing attacks targeting workers involved in key infrastructure [67].

Invest in Staff Training and Development: The effective use of OSINT tools within critical infrastructures requires specialized skills and knowledge. Organizations must invest in training their cybersecurity teams to use these tools proficiently and interpret the data they produce, particularly in the context of critical operations [69]. Tools such as **Wireshark** for network traffic analysis, **Nmap** for network discovery and security auditing, and **Snort** for real-time network intrusion detection can also be used in training programs. Cybersecurity personnel must be adept at using these technologies since they are vital to preserving the security of critical infrastructure systems.

4.5.2. Detailed Framework for OSINT Integration in Critical Infrastructures

A systematic approach to integrating OSINT tools into the cybersecurity frameworks of critical infrastructures can be broken down into four key stages, ensuring that the integration is thorough, strategic, and aligned with the operational objectives of these essential sectors. The following framework for integrating OSINT tools into the cybersecurity strategies of critical infrastructures is designed based on the foundational concepts, best practices, and case studies discussed earlier in this study. This structured approach builds upon the previously detailed information, ensuring that the integration process is comprehensive and aligned with critical infrastructures' unique challenges and requirements.

Stage 1: Development of the OSINT Framework Plan: In this initial stage, critical infrastructure organizations identify the specific systems or infrastructures they wish to analyze using OSINT tools. For example, a water treatment facility uses **Thingful** to map out and monitor IoT devices involved in water quality management, while a power grid uses **Shodan** to assess the exposure of its SCADA systems. This involves defining the scope of the OSINT activities, including the types of data to be collected, the sources to be monitored, and the tools to be employed (Fig. 4.9).

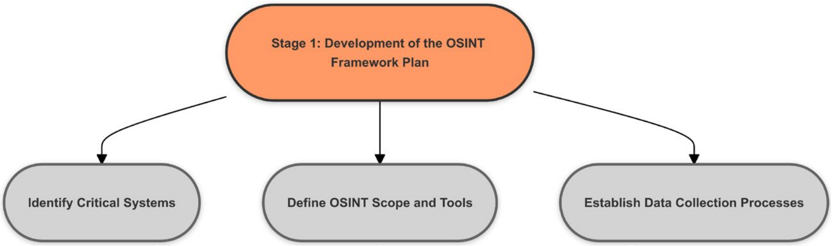

Stage 1: Development of the OSINT Framework Plan
Identify Critical Systems
Define OSINT Scope and Tools
Establish Data Collection Processes

Fig. 4.9. Stage 1: Development of the OSINT Framework Plan

Stage 2: Threat Identification and Analysis: Once the framework is in place, the next step is to identify and analyze potential threats to critical infrastructures. This involves using **Shodan** to scan for vulnerable devices in a power grid's operational technology network, while **TheHarvester** gathers information on domain names and associated email addresses that can be targeted in phishing attacks on water treatment facilities. **Censys** can also play a role in identifying and profiling exposed devices that are being exploited in an attack (Fig. 4.10).

Stage 3: Integration and Automation: Integrating the OSINT tools with the critical infrastructure's current cybersecurity systems is the third stage. This entails automating the correlation of OSINT data with internal network activity using a tool such as **Splunk** to guarantee real-time situational awareness and accelerate the response time to possible threats. In order to gather and examine logs from several sources, including OSINT data, and to detect possible security issues, **Graylog** – additional log management and SIEM tool – can be integrated (Fig. 4.11).

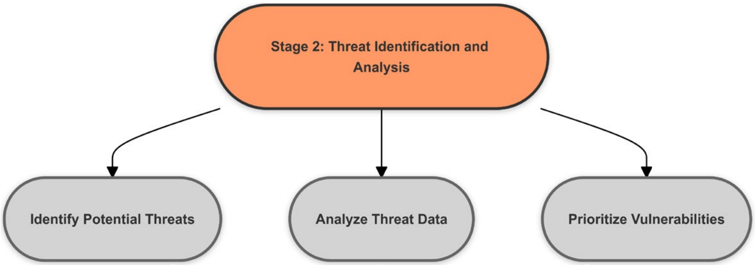

Stage 2: Threat Identification and Analysis
Identify Potential Threats
Analyze Threat Data
Prioritize Vulnerabilities

Fig. 4.10. Stage 2: Threat Identification and Analysis.

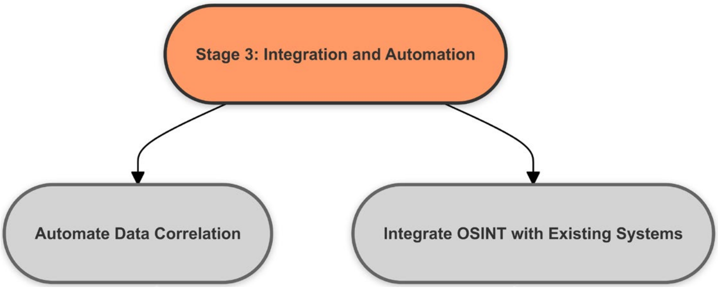

Stage 3: Integration and Automation
Automate Data Correlation
Integrate OSINT with Existing Systems

Fig. 4.11. Stage 3: Integration and Automation.

Stage 4: Continuous Monitoring and Feedback: The final stage focuses on continuous monitoring and refinement of the OSINT integration process in critical infrastructures. Organizations benefit from the use of **Masscan** by regularly scanning their networks for open ports and vulnerabilities, ensuring that any changes in the network environment are quickly identified. Feedback loops should be established to assess the impact of OSINT on the organization's overall cybersecurity posture, using tools like **PcapXray** to visualize network traffic flows and identify potential threats (Fig. 4.12).

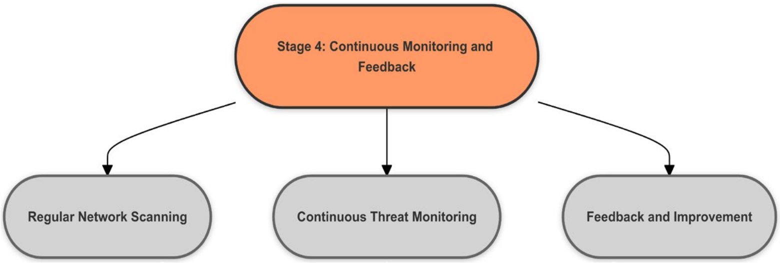

Stage 4: Continuous Monitoring and Feedback
Regular Network Scanning
Continuous Threat Monitoring
Feedback and Improvement

Fig. 4.12. Stage 4: Continuous Monitoring and Feedback.

While the outlined framework offers a robust approach to OSINT integration, it is crucial to recognize and address the potential challenges and pitfalls that can arise during implementation.

4.5.3. Challenges and Pitfalls in OSINT Integration for Critical Infrastructures

Although OSINT tools have many benefits, there are certain obstacles to incorporating them into critical infrastructure cybersecurity plans. To guarantee a successful integration, organizations need to be aware of these potential dangers and implement mitigation techniques.

Challenge 1: Information Overload: The significant amount of data that OSINT systems can produce in critical infrastructures is one of the biggest issues they face. Threat information can be filtered and prioritized with the aid of tools such as **Recorded Future**; however, in order to handle this data efficiently, organizations also need to use machine learning algorithms and advanced analytics. Filtering strategies are crucial to focus on relevant packets from the massive quantity of data that **Wireshark** captures despite its powerful capabilities.

Challenge 2: False Positives: Another frequent problem with OSINT technologies is the incidence of false positives, which can be especially problematic in critical infrastructures. In order to ensure that only legitimate threats are marked for further investigation, tools such as **Snort**, which can combine with OSINT data to detect intrusions, must be carefully calibrated to reduce false positives. In a similar vein, careful configuration of **OpenVAS**, an open-source vulnerability scanner, can lower the proportion of non-critical vulnerabilities that are detected.

Challenge 3: Security of OSINT Tools: Because OSINT tools can be targets of cyberattacks, it is imperative to ensure their security. It's crucial to regularly update programs like **Maltego** and **Shodan**, make sure they can't be exploited, and use strict access controls. Furthermore, by utilizing **OpenVAS** for routine security audits, it is possible to find and fix potential flaws in the tools themselves, preventing the instruments used to safeguard vital infrastructures from turning into liabilities.

To effectively integrate OSINT tools like **Shodan** and **Nmap** into existing utilities' cybersecurity systems, it is essential to outline specific technical steps and address potential challenges. For instance, integrating OSINT with SCADA (Supervisory Control and Data Acquisition) systems requires careful calibration to ensure that real-time data collection does not interfere with operational processes. Utilities can begin by implementing a phased approach, starting with passive monitoring to map out the digital landscape, identifying all internet-facing devices, and categorizing them based on risk. Next, these OSINT tools should be configured to trigger alerts when anomalies, such as unauthorized access attempts or unusual traffic patterns, are detected. The key challenge is ensuring that these alerts are accurate and actionable, minimizing false positives that could overwhelm the system. Solutions include fine-tuning the tools to the specific architecture of the utility's ICS (Industrial Control Systems) and regularly updating the threat intelligence databases these tools rely on.

A practical method of improving an organization's capacity to identify, evaluate, and counter cyber threats is through the incorporation of OSINT tools into cybersecurity plans for critical infrastructures. Organizations can effectively employ open-source intelligence (OSINT) tools to strengthen their security posture by adhering to best practices, which include setting clear objectives, guaranteeing legal compliance, and integrating OSINT with current tools. But it's also critical to be mindful of the difficulties that come with integrating OSINT, such as information overload, false positives, and tool security. Organizations can overcome these obstacles and fully benefit from OSINT in their cybersecurity efforts for critical infrastructures by implementing proper mitigation methods.

4.6. Case Studies of Major Cyber Attacks on Utilities Infrastructure

In recent years, the growing digitalization of critical infrastructure has exposed vital systems to an increasing number of cyber threats. These tools have proven their value in numerous real-world scenarios, such as the following case studies. Notable incidents like the Ukraine Power Grid Attack in 2015 [22], the Colonial Pipeline Ransomware Attack in 2021 [19], and the Oldsmar Water Treatment Facility Incident in 2021 [70] underscore the vulnerabilities inherent in essential services. Fig. 4.13 provides a timeline of these significant cyber-attacks, illustrating the progression and impact of these events over the years. These case studies highlight the methods used by attackers, the consequences of these breaches, and the critical lessons learned in the aftermath. Table 4.4 summarizes significant cyber-attacks on utility infrastructures, detailing the methods used, the impact of these attacks, the lessons learned, and potential OSINT tools that could have been employed to mitigate these incidents.

Table 4.4. Summary of Major Cyber-Attacks and Potential OSINT Tools for Mitigation.

Incident	**Method of Attack**	**Impact**	**Lessons Learned**	**Potential Beneficial Tools**
Ukraine Power Grid Attack (2015)	Phishing, Malware (BlackEnergy), KillDisk, TDoS	Power outage affecting 230000 residents	Importance of cybersecurity in ICS systems	Nmap, Shodan, Tcpdump
Colonial Pipeline Ransomware Attack (2021)	Ransomware (DarkSide), Compromised VPN Account	Temporary shutdown, fuel shortages in the US	Need for multi-factor authentication and backups	Shodan, Nmap, Tcpdump
Oldsmar Water Treatment Facility Incident (2021)	Remote access, Attempted chemical contamination	Attempted increase in sodium hydroxide levels	Critical need for monitoring and securing remote access	Nmap, Tcpdump, Shodan, Maltego

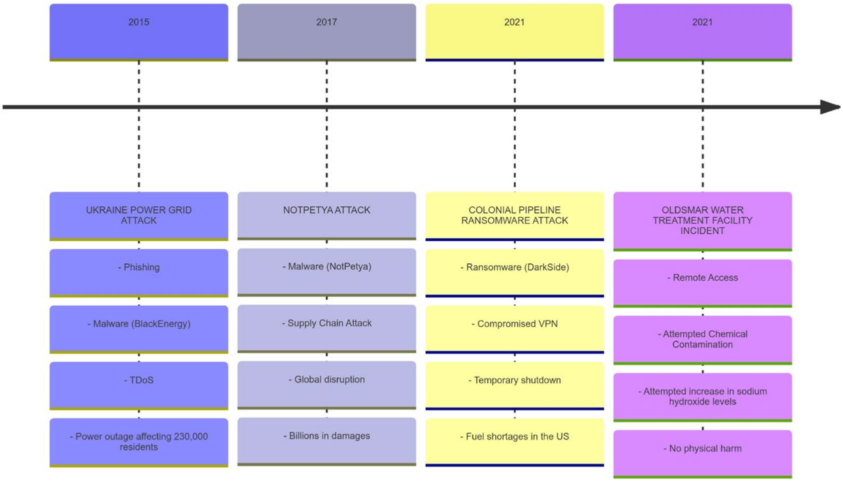

2015
2017
2021
2021
UKRAINE POWER GRID ATTACK
- Phishing
- Malware (BlackEnergy)
- TDoS
- Power outage affecting 230,000 residents
NOTPETYA ATTACK
- Malware (NotPetya)
- Supply Chain Attack
- Global disruption
- Billions in damages
COLONIAL PIPELINE RANSOMWARE ATTACK
- Ransomware (DarkSide)
- Compromised VPN
- Temporary shutdown
- Fuel shortages in the US
OLDSMAR WATER TREATMENT FACILITY INCIDENT
- Remote Access
- Attempted Chemical Contamination
- Attempted increase in sodium hydroxide levels
- No physical harm

Fig. 4.13. Major Cyber Security Incidents Timeline.

The Ukraine Power Grid Attack on December 23, 2015, was the first known cyber attack to create a power outage, which caused some 230000 homes to lose energy for several hours. The attack, credited to the Russian-affiliated Sandworm Group, used phishing emails to spread the BlackEnergy virus, giving users remote access to the industrial control systems that oversee the electrical grid. In addition, the attackers used telephone denial of service (TDoS) attacks to stop impacted consumers from reporting outages and used KillDisk malware to take down systems. Manual intervention was necessary to restore the grid, underscoring the weaknesses in vital infrastructure and the possibility of cyberattacks interfering with vital services.

More recently, on May 7, 2021, the Colonial Pipeline Ransomware Attack showed how susceptible energy infrastructure is to cyber-extortion. The first point of entry for the Russian-speaking ransomware-as-a-service outfit DarkSide was a compromised VPN account without multi-factor authentication. Using a twofold extortion strategy, the attackers encrypted the company's data and demanded a ransom, threatening to reveal the stolen material if the ransom was not paid. Due to the attack, the pipeline was temporarily shut down, which resulted in severe gasoline shortages and price increases throughout the eastern United States. The decryption tool was delayed, even after the corporation paid a $4.4 million ransom, and backups were eventually used for restoration.

The Oldsmar Water Treatment Facility Incident on February 5, 2021, further illustrated the potential dangers posed by cyber threats to public health and safety. In this attack, cybercriminals remotely accessed the facility's control systems and attempted to increase the levels of sodium hydroxide (lye) in the water supply to dangerous levels. Fortunately, the attack was detected and thwarted by a plant operator who noticed the change and reversed it before any harm occurred. Although no physical harm resulted from this attack, it raised awareness of the need for stronger cybersecurity measures in utilities that manage essential services, such as the implementation of intrusion detection systems (IDS) and increased monitoring of remote access protocols.

4.6.1. OSINT Tool Integration: Enhancing Network Security and Threat Detection

In response to these significant incidents, organizations have increasingly turned to Open-Source Intelligence (OSINT) tools to bolster their cybersecurity strategies. Tools such as Nmap, Tcpdump, and Shodan have proven effective in enhancing network monitoring, vulnerability assessment, and real-time threat detection. For example, Nmap was integrated into a financial services company's network monitoring practices to ensure the security of its extensive infrastructure. Regular scans conducted with Nmap identified unnecessary open ports and outdated services, which were then secured or removed [71]. This proactive approach significantly reduced the company's attack surface and provided a comprehensive and current inventory of network assets, strengthening the overall security posture.

Similarly, Tcpdump was deployed by a healthcare provider to continuously monitor network traffic and detect anomalies indicative of cyber threats. The tool enabled real-time analysis of data packets, helping to identify unusual patterns such as traffic spikes or unauthorized access attempts. This capability allowed the healthcare provider to swiftly respond to potential security incidents, blocking unauthorized access and preventing data breaches [72]. Tcpdump's real-time insights also contributed to optimizing network performance and maintaining regulatory compliance.

Shodan, known as the "search engine for hackers," was used by an electricity utility company to scan for internet-connected devices within its network. The tool identified unsecured devices and services that were potentially vulnerable to cyber-attacks. By prioritizing these vulnerabilities for remediation, the company was able to significantly reduce its exposure to potential threats [49]. The integration of Shodan into the company's cybersecurity strategy enhanced its ability to identify and secure vulnerable devices, leading to a more resilient network infrastructure.

4.6.2. Applying OSINT Tools to Case Study Incidents

The integration of OSINT tools such as Nmap, Tcpdump, and Shodan could have significantly enhanced the cybersecurity defences in the aforementioned real-world incidents. For instance, in the Ukraine Power Grid Attack, Nmap could have been used to regularly scan the network for unauthorized devices or unexpected open ports, potentially detecting the presence of the BlackEnergy malware before it could cause harm. Tcpdump can have provided real-time insights into unusual network traffic, signalling an ongoing cyberattack, while Shodan could have identified any internet-facing devices that were vulnerable to exploitation.

In the Colonial Pipeline Ransomware Attack, Shodan could have been employed to scan for vulnerable devices connected to the internet, including those that lacked multi-factor authentication. Nmap could have identified and secured open ports that the attackers have used for initial access. Tcpdump, with its capability to analyze real-time network traffic, could have detected unusual patterns indicative of ransomware deployment, allowing for a quicker response.

Similarly, in the Oldsmar Water Treatment Facility Incident, Tcpdump could have played a crucial role in monitoring network traffic for any unauthorized remote access attempts. Nmap could have been used to conduct regular scans of the facility's network to ensure that no unauthorized devices were connected. Shodan could have provided early warnings by identifying exposed devices that could be targeted by attackers. These examples underscore the importance of integrating OSINT tools into cybersecurity strategies to enhance threat detection and response capabilities, ultimately protecting critical infrastructure from evolving cyber threats.

4.7. Building a Resilient Cybersecurity Monitoring System

The need for a robust cybersecurity monitoring system is growing more and more important as the digital landscape changes, particularly in key utility facilities. By adding Open-Source Intelligence (OSINT) technologies to these systems, their ability to recognize, identify, and react to new threats is improved. The frameworks necessary to develop resilience, the significance of awareness and training in preserving security, upcoming developments that will influence the cybersecurity environment, and the incorporation of OSINT into these frameworks are all covered in this part.

4.7.1. Frameworks for Resilience

Building a resilient cybersecurity monitoring system requires a multi-faceted approach that combines a layered defence strategy, meticulous incident response planning, continuous monitoring, and the implementation of redundancy across critical systems. These components ensure that even if one layer of defence is compromised, others can mitigate the impact, maintaining the security and integrity of vital infrastructure.

Layered Defense: A defence-in-depth approach, also known as layered defence, uses several tiers of security controls across the IT environment of an enterprise. Fig. 4.14 provides a visual representation of the tools applied in a layered security approach, demonstrating how each layer contributes to the overall resilience of the cybersecurity framework. With this strategy, an attack cannot succeed because backup security measures are in place in case the primary one fails. For example, intrusion detection/prevention systems (IDS/IPS), firewalls, endpoint protection, and network segmentation are all eligible to be part of a layered defence in a utility infrastructure. Particular OSINT tools, such as **Shodan** and **Maltego**, can map out possible attack vectors throughout the network and continuously scan internet-facing devices for vulnerabilities.

Incident Response Planning: In order to minimize the harm caused by a cyber attack and quickly return to normal operations, a robust incident response plan is necessary. Clearly defined protocols for identifying, notifying, and handling security problems should be part of this plan.

The incident response plan for utilities, when public safety and service continuity are at risk, needs to be tested and updated on a regular basis using simulated attacks. Real-time threat intelligence from OSINT tools, such as **Recorded Future**, can be used to inform incident response plans and make sure all parties involved are ready to take appropriate action in the case of a genuine danger.

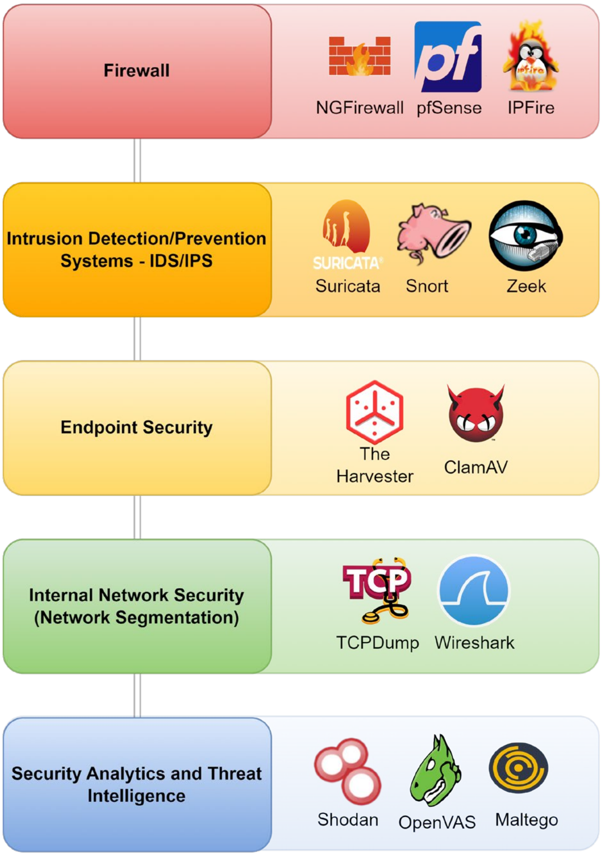

Firewall
NGFirewall
pfSense
IPFire
Intrusion Detection/Prevention Systems - IDS/IPS
Suricata
Snort
Zeek
Endpoint Security
The Harvester
ClamAV
Internal Network Security (Network Segmentation)
TCPDump
Wireshark
Security Analytics and Threat Intelligence
Shodan
OpenVAS
Maltego

Fig. 4.14. OSINT Tools in Layered Security Approach.

Continuous Monitoring: To identify risks in real-time, networks and systems must be continuously monitored. Organizations can continuously scan internet-facing devices for vulnerabilities by utilizing OSINT tools like Censys and Shodan, and they can monitor internal network traffic for anomalies by using tools like Nmap and Masscan.

Constant monitoring makes it possible to see suspicious activity right away, allowing for a quick response that can stop minor problems from becoming significant breaches.

Redundancy: Having backup systems and procedures in place to continue operations even in the event that primary systems fail is known as redundancy. Redundancy for critical infrastructures includes more than just having backup power sources or servers; it also entails making sure that cybersecurity safeguards are replicated throughout the network's tiers. This can entail having duplicated data centres to guarantee service continuity in the event of an attack or several channels of communication for incident reporting. The total resilience of the infrastructure can be increased by routinely scanning and evaluating the exposure of backup systems with OSINT technologies like **Zoomeye** to make sure they stay safe.

Testing and Drills: Regular testing of resilience and incident response strategies is essential for maintaining robust cybersecurity defences in utilities. OSINT tools can be instrumental in these tests, providing real-time data on emerging threats and simulating potential attack scenarios. For example, utilities can use **TheHarvester** to simulate a phishing attack by identifying and targeting publicly exposed employee contact information testing the effectiveness of their email filtering and response protocols. Similarly, simulated DDoS attacks using data from tools like **Shodan** or **GreyNoise** can help utilities assess the resilience of their network infrastructure and identify potential bottlenecks. These drills are most effective when conducted regularly and involve all relevant stakeholders, including IT and OT teams, to ensure a coordinated response.

4.7.2. The Role of Training and Awareness

The human factor is just as important in creating a robust cybersecurity monitoring system as technology solutions. Programs for training and awareness enable staff members to identify and react to possible dangers, promoting a security-conscious culture within the company.

Empowering Employees: Workers are frequently the first to defend against cyberattacks, particularly in industries where information technology (IT) and operational technology (OT) collide. Frequent training sessions covering safe internet practices, the newest cybersecurity risks, and how to utilize OSINT technologies correctly can enable staff members to serve as an extra security layer [73]. For example, knowledge of **Wireshark** and **Tcpdump** is able to help in troubleshooting and discovering network anomalies, and training on **TheHarvester** helps IT workers identify and mitigate phishing attacks.

Fostering a Security-Conscious Culture: A security-conscious culture within an organization ensures that cybersecurity is seen as a collective responsibility rather than just the domain of the IT department. Utilities can lower the risk of human error – frequently the weakest link in cybersecurity defences – by incorporating cybersecurity awareness into routine operations and making it a part of the organizational culture. Initiatives such as regular security briefings, phishing simulations, and the incorporation of cybersecurity topics into all-staff meetings can contribute to this culture [74]. Incorporating OSINT tools like **Maltego** into regular security briefings can provide employees with a visual representation of potential threats, helping them understand the importance of maintaining vigilance.

Staying Updated on Threats: The environment of cybersecurity is always changing as new threats appear in tandem with technological advancements. Organizations must stay informed about the latest developments in cyber threats and defences to maintain resilience. This can be achieved through continuous professional development, cybersecurity bulletin subscriptions, and industry forums participation [75]. Leveraging OSINT tools like **Recorded Future** can also help organizations stay ahead of threats by providing real-time intelligence on potential risks, ensuring that defences are always up-to-date.

4.7.3. Future Trends in Cybersecurity and OSINT Integration

The tactics and resources employed to counteract cyber threats must also change with their evolving nature. Artificial intelligence (AI) and machine learning (ML) developments, the growing convergence of cybersecurity and physical science, and the possible effects of quantum computing on encryption and data security will all influence cybersecurity monitoring in the future.

AI/ML in Threat Detection: By facilitating the creation of increasingly complex threat detection systems, artificial intelligence and machine learning are entirely changing the field of cybersecurity. At previously unheard-of speeds, these technologies are able to scan enormous volumes of data and spot patterns and anomalies that point to a cyber attack [76]. In the context of OSINT, by evaluating data from many sources and anticipating possible attack vectors before they are used, AI/ML can improve the capacity of tools such as **Recorded Future** and **Shodan** to identify emerging threats. In the opposite situation, network data can be gathered using tools like **Nmap**, **Tcpdump**, and **Wireshark** to create threat classification or anomaly detection models.

Integration of Physical and Cybersecurity: The convergence of physical and cybersecurity is becoming increasingly important as more physical infrastructure is connected to the digital world. This integration requires a holistic approach to security that considers both the physical and digital aspects of an organization's operations [77, 78]. For instance, the use of IoT devices in critical infrastructure introduces new vulnerabilities that must be addressed through integrated security strategies that protect both the physical devices and the networks they connect to. OSINT tools like **Thingful** can monitor IoT devices and ensure they do not become entry points for cyber-attacks, providing comprehensive intelligence that covers both physical and cyber domains.

Quantum Computing: Quantum computing has the potential to disrupt current encryption methods, making previously secure systems vulnerable to attack. As quantum computing technology advances, organizations must prepare for the possibility that their encryption protocols are to be rendered obsolete [79-81]. This will require the development of new cryptographic methods and the integration of quantum-resistant algorithms into cybersecurity frameworks. OSINT tools can assist in this transition by providing intelligence on emerging quantum computing threats and helping organizations adapt their defences accordingly. Tools like **RiskIQ** can help monitor the external digital landscape for early signs of quantum-related vulnerabilities, ensuring that organizations are prepared for this next wave of technological advancement.

4.8. Conclusion and Future Work

4.8.1. Summary

In the rapidly evolving landscape of cybersecurity, the integration of Open-Source Intelligence (OSINT) tools into monitoring frameworks has emerged as a critical strategy for enhancing the security of vital utility infrastructures. This chapter has outlined the multifaceted role of OSINT in identifying, predicting, and mitigating cyber threats, particularly in sectors where the stakes are exceptionally high, such as electricity, water, and natural gas. Through detailed exploration of the challenges faced by these infrastructures, as well as the practical applications of specialized OSINT tools, it has become evident that a proactive approach to cybersecurity is not merely advantageous but necessary. This necessity is demonstrated in the following recommendations and case studies, which highlight how OSINT tools have been effectively integrated into cybersecurity strategies.

While there is no denying that the digitalization of utility infrastructures has increased efficiency and connectedness, it has also created a larger and more intricate attack surface. Even if they are still important, traditional security measures are not enough to counter the complex and ever-changing nature of contemporary cyber threats. This gap can be filled by incorporating OSINT technologies like **Shodan**, **Nmap**, and **Tcpdump** into cybersecurity tactics. These tools provide real-time, actionable intelligence that improves an organization's capacity to identify and neutralize threats before they have a substantial impact.

Moreover, this chapter has emphasized the importance of a layered defence strategy, continuous monitoring, and the implementation of redundancy across critical systems. Organizations may improve their ability to detect threats immediately and develop a robust cybersecurity posture that can change with the times by integrating OSINT tools into these frameworks. The necessity for such integration has been further highlighted by the thorough case studies of significant cyberattacks on utility infrastructure, like the Colonial Pipeline Ransomware Attack and the Ukraine Power Grid Attack, which show how these tools can be used to mitigate or even prevent some of the most notable incidents in recent memory.

4.8.2. Call to Action

The successful integration of OSINT tools into cybersecurity strategies requires a coordinated effort among a broad range of stakeholders, each playing a critical role in ensuring the security and resilience of critical infrastructures. These stakeholders include government agencies, private sector organizations, cybersecurity professionals, and academia. By understanding the unique contributions of each group, we can better appreciate the importance of their collaboration in this endeavour.

Government Agencies: Government agencies and regulatory bodies are at the forefront of setting the standards and policies that govern the cybersecurity landscape. Their role is crucial in establishing guidelines for the safe and ethical use of OSINT tools within critical infrastructures. Regulatory frameworks, such as the GDPR in Europe and NERC CIP in North America, provide the foundation for ensuring that OSINT activities comply with legal and ethical standards. By collaborating with industry stakeholders, these agencies can create policies that balance security needs with the protection of individual privacy and civil liberties. It is imperative that these bodies lead the way in fostering a secure and compliant environment for the deployment of OSINT tools.

Private Sector Organizations: The private sector, particularly companies managing critical infrastructure, must prioritize the integration of OSINT tools into their cybersecurity frameworks. These organizations are on the front lines of defending against cyber threats and have a direct stake in the effectiveness of their cybersecurity measures. Investment in cutting-edge technologies, continuous education, and training of cybersecurity professionals are essential steps that private sector organizations must take to enhance their security posture. Additionally, active participation in information-sharing initiatives with government agencies and other industry players will bolster collective defence efforts and ensure that the private sector is well-equipped to respond to emerging threats.

Cybersecurity Professionals: Cybersecurity professionals are the operational backbone of any defence strategy. Their expertise and skills are vital for the effective deployment and utilization of OSINT tools. To maximize the benefits of these tools, cybersecurity professionals must be proficient in their use and continually update their knowledge to keep pace with the evolving threat landscape. This includes understanding both the capabilities and limitations of OSINT tools, as well as integrating them seamlessly with existing security measures. Ongoing professional development and training programs are crucial to maintaining a high level of expertise among cybersecurity personnel.

Academia: Academia and research institutions play a pivotal role in advancing the field of OSINT through research, innovation, and the development of new tools and methodologies. Collaboration between academia and industry can lead to the creation of more sophisticated OSINT tools that address specific challenges faced by critical infrastructure sectors. Academic research can also contribute to the development of best practices for OSINT integration and provide valuable insights into emerging threats and vulnerabilities. By fostering partnerships between academia and the private sector, we can drive innovation and ensure that the next generation of cybersecurity professionals is well-prepared to meet the challenges of the future.

4.8.3. Future Directions

Looking ahead, the future of OSINT in cybersecurity is poised to be shaped by several key trends and developments. These future directions offer promising avenues for enhancing the effectiveness of OSINT tools and their integration into cybersecurity strategies.

Advanced Predictive Analytics: The way risks are recognized and handled may change if artificial intelligence (AI) and machine learning (ML) are applied to OSINT data. Predictive analytics can detect patterns and trends in large datasets to foresee cyber dangers before they manifest, enabling preventive intervention. Subsequent research endeavours in this domain are anticipated to concentrate on enhancing these algorithms to elevate their precision and dependability, rendering them an essential component of cybersecurity surveillance.

Blockchain Technology: The decentralized, tamper-proof data storage and verification provided by blockchain are crucial for improving the integrity of OSINT data. Blockchain provides a more secure foundation for OSINT technologies by guaranteeing that data cannot be changed or erased, lowering the possibility of false information and enhancing confidence in the intelligence obtained. Subsequent studies examine the potential integration of blockchain technology with current OSINT systems, especially in industries where data integrity is critical.

Cross-Sector Collaboration: Because cyber dangers transcend industry boundaries, cross-sector cooperation will be more crucial than ever. Future research should concentrate on creating structures that facilitate cross-sector collaboration and the exchange of best practices and threat intelligence. This entails building centralized platforms for open-source intelligence (OSINT) that compile information from multiple industries and present a more complete picture of the threat environment.

Improving critical infrastructure sectors' cybersecurity posture requires effective cross-sector collaboration. Utilities can share threat intelligence with other industries like energy, water, and transportation by participating in information-sharing systems like the Information Sharing and Analysis Centers (ISACs). Through these platforms, utilities can exchange best practices for managing emerging vulnerabilities and receive early notifications about such dangers. Furthermore, partnerships like the United States' Cybersecurity and Infrastructure Security Agency (CISA) make it easier for public and private sector organizations to work together. In order to test their cybersecurity defences and gain insight from other industries' experiences, utilities can take part in cooperative exercises and simulations. Formal alliances and cooperative agreements can be established to bolster these initiatives and guarantee that everyone is working toward the same cybersecurity goals.

Regulatory Evolution: As the use of OSINT tools becomes more widespread, there will be a growing need for clear and consistent regulatory frameworks that govern their use. Future work in this area should focus on developing regulations that balance the need for security with the protection of individual privacy and civil liberties. This will require ongoing dialogue between regulators, industry leaders, and civil society to ensure that OSINT tools are used ethically and responsibly.

Regulatory frameworks like GDPR in Europe and NERC CIP in North America play a critical role in guiding the use of OSINT tools in critical infrastructure. For instance, utilities in Europe must ensure that their OSINT activities comply with GDPR’s strict data protection requirements, mainly when processing personal data. This has led to the development of compliance strategies that include data minimization, anonymization, and the implementation of strict access controls. In North America, the NERC CIP standards require utilities to maintain robust cybersecurity programs, including the use of OSINT to monitor for potential threats. However, compliance also means ensuring that OSINT data collection does not infringe on regulatory guidelines. Utilities have responded by integrating OSINT tools into their existing compliance frameworks, using these tools to enhance visibility into potential threats while maintaining adherence to regulatory requirements.

Education and Workforce Development: The growing dependence of cybersecurity on OSINT techniques will require staff with the necessary skills to use these tools efficiently. Future initiatives should concentrate on providing chances for continued professional development to cybersecurity professionals and incorporating OSINT training into cybersecurity education programs. By doing this, cybersecurity experts will be prepared to tackle the difficulties posed by the always-changing threat landscape.

Utilities need to make investments in education and workforce development initiatives in order to generate a workforce that can effectively use OSINT tools. In this context, collaborations between academia and business are essential. Programs for cybersecurity training, for instance, created in association with academic institutions, may emphasize the helpful use of OSINT instruments in real-world situations. In order to prepare students for the difficulties they will encounter in the field, these programs may include practical laboratories where they do vulnerability assessments using tools like Nmap and Shodan. Utilities can also take part in programs such as the Cybersecurity Workforce Development Program (CWDP), which provides certification and training in the most recent cybersecurity techniques, including OSINT. Through these programs, cybersecurity professionals are guaranteed to have the knowledge and abilities needed to defend vital infrastructure against changing threats.

4.8.4. Conclusion

The integration of Open-Source Intelligence (OSINT) tools into the cybersecurity frameworks of essential utility infrastructures is a crucial step toward protecting vital industries like natural gas, water, and electricity. The speed at which digitalization is occurring in these fields is matched by the increasing complexity and sophistication of possible cyberattacks. The urgent need for a proactive and flexible approach to cybersecurity has been highlighted by this study, where OSINT tools play a crucial role in improving threat detection, response, and overall resilience.

Thorough analysis and real-world case studies, it is evident that OSINT technologies like as Nmap, Shodan, and Tcpdump are not only helpful but also crucial for locating vulnerabilities, reducing risks, and averting possible cyberattacks. The report also emphasizes the significance of redundancy, ongoing monitoring, and a multi-layered protection plan, all of which are essential for preserving the dependability and integrity of critical infrastructure systems.

In order to properly utilize the potential of OSINT capabilities, the report also urges cooperation between government agencies, businesses, cybersecurity experts, and academic institutions. In order to create and implement robust cybersecurity strategies that can keep up with the changing threat landscape, teamwork is essential.

It is anticipated that developments in blockchain, AI, and machine learning will influence OSINT in cybersecurity in the future. These developments should improve OSINT tools' capabilities and increase their efficacy in safeguarding critical infrastructures from new threats. To further ensure that these tools are used effectively and ethically, cross-sector collaboration, educational campaigns, and the creation of regulatory frameworks will be essential.

In conclusion, the integration of OSINT tools into cybersecurity strategies is a critical step toward building more secure and resilient utility infrastructures. By embracing these tools and fostering a culture of continuous improvement and collaboration, organizations can better protect the essential services that underpin our modern society from the ever-present threat of cyberattacks.

References

[1]. K. Smith, I. D. Wilson, Critical infrastructures: A comparison of definitions, *International Journal of Critical Infrastructures*, Vol. 19, Issue 4, 2023, pp. 323-339.

[2]. D. Clemente, Cyber Security and Global Interdependence: What is Critical?, *Chatham House, Royal Institute of International Affairs*, 2013.

[3]. N. Petrakos, P. Kotzanikolaou, Methodologies and strategies for critical infrastructure protection, in Critical Infrastructure Security and Resilience: Theories, Methods, Tools and Technologies, *Springer*, 2019, pp. 17-33.

[4]. M. I. Ecevit, M. H. Pervez, H. Dag, R. Creutzburg, The Open Source Intelligence (OSINT) in the electricity sector: balancing utility and responsibility, *Electronic Imaging*, Vol. 36, 2024, 318.

[5]. A. Janjić, L. Velimirović, J. Ranitović, Ž. Džunić, Internet of things in power distribution networks – state of the art, in *Proceedings of the 52nd International Scientific Conference on Information, Communication and Energy Systems and Technologies (ICEST'17)*, 2017.

[6]. A. K. Minhaj, S. Khaled, IoT security: Review, blockchain solutions, and open challenges, *Future Generation Computer Systems*, Vol. 82, 2018, pp. 395-411.

[7]. A. Ramamurthy, P. Jain, The Internet of Things in the Power Sector Opportunities in Asia and the Pacific, *The Asian Development Bank*, 2017.

[8]. A. S. Bretas, N. G. Bretas, B. Carvalho, E. Baeyens, P. P. Khargonekar, Smart grids cyber-physical security as a malicious data attack: An innovation approach, *Electr. Power Syst. Res.*, Vol. 149, 2017, pp. 210-219.

[9]. A. Sadu, A. Jindal, G. Lipari, F. Ponci, A. Monti, Resilient design of distribution grid automation system against cyber-physical attacks using blockchain and smart contract, *Blockchain Res. Appl.*, 2021, 100010.

[10]. B. Li, R. Lu, W. Wang, K. R. Choo, Distributed host-based collaborative detection for false data injection attacks in smart grid cyber-physical system, *J. Parallel Distrib. Comput.*, Vol. 103, 2017, pp. 32-41.

[11]. L. Lee, P. Hu, Vulnerability analysis of cascading dynamics in smart grids under load redistribution attacks, *Electr. Power Energy Syst.*, Vol. 111, 2019, pp. 182-190.

[12]. S. Krčo, B. Pokrić, F. Carrez, Designing IoT architecture(s) a European perspective, in *Proceedings of the IEEE World Forum on Internet of Things (WF-IoT'14)*, 2014, pp. 79-84.

[13]. G. Bedi, G. K. Venayagamoorthy, R. Singh, R. R. Brooks, K. C. Wang, Review of Internet of Things (IoT) in electric power and energy systems, *IEEE Internet Things J.*, Vol. 5, Issue 2, 2018, pp. 847-870.

[14]. B. Shakerighadi, A. Anvari-Moghaddam, J. C. Vasquez, J. M. Guerrero, Internet of things for modern energy systems: state-of-the-art, challenges, and open issues, *Energies*, Vol. 11, Issue 5, 2018, 1252.

[15]. H. Jia, C. Shao, D. Liu, C. Singh, Y. Ding, Y. Li, Operating reliability evaluation of power systems with demand-side resources considering cyber malfunctions, *IEEE Access*, Vol. 8, 2020, pp. 87354-87366.

[16]. A. Djenna, S. Harous, D. E. Saidouni, Internet of things meet internet of threats: New concern cyber security issues of critical cyber infrastructure, *Applied Sciences*, Vol. 11, Issue 10, 2021, 4580.

[17]. T. M. Chen, S. Abu-Nimeh, Lessons from Stuxnet, *Computer*, Vol. 44, Issue 4, 2011, pp. 91-93.

[18]. R. Khan, P. Maynard, K. McLaughlin, D. Laverty, S. Sezer, Threat analysis of black energy malware for synchrophasor based real-time control and monitoring in smart grid, in *Proceedings of the 4th International Symposium for ICS SCADA Cyber Security Research*, 2016, pp. 53-63.

[19]. J. Beerman, D. Berent, Z. Falter, S. Bhunia, A review of colonial pipeline ransomware attack, in *Proceedings of the IEEE/ACM 23rd International Symposium on Cluster, Cloud and Internet Computing Workshops (CCGridW'23)*, 2023, pp. 8-15.

[20]. BBC News, New Zealand Stock Exchange Halted by Cyber-Attack, https://www.bbc.com/news/53918580

[21]. R. Alkhadra, J. Abuzaid, M. AlShammari, N. Mohammad, Solar winds hack: In-depth analysis and countermeasures, in *Proceedings of the 12th International Conference on Computing Communication and Networking Technologies (ICCCNT'21)*, 2021, pp. 1-7.

[22]. Analysis of the Cyber Attack on the Ukrainian Power Grid, *Electricity Information Sharing and Analysis Center (E-ISAC)*, 2016.

[23]. C. Alcaraz, P. Najera, R. Roman, J. Lopez, How will city infrastructure and sensors be made smart?, *White Pap*., Vol. 6, Issue 11, 2010, 113.

[24]. Cyber Threats for ICS in Energy in Europe, Object of Research, *Kaspersky ICS CERT*, 2020.

[25]. J. Syed, K. Collins-Thompson, Optimizing search results for a human learning goal, *Information Retrieval Journal*, Vol. 20, 2017, pp. 506-523.

[26]. R. Mallik, H. Kargupta, A sustainable approach for demand prediction in smart grids using a distributed local asynchronous algorithm, in *Proceedings of the Conference on Intelligent Data Understanding (CIDU'11)*, October 19-21, 2011.

[27]. H. Kettani, P. Wainwright, On the top threats to cyber systems, in *Proceedings of the IEEE 2nd international conference on information and computer technologies (ICICT'19)*, pp. 175-179.

[28]. I. Doh, J. Lim, K. Chae, Secure authentication for structured smart grid system, in *Proceedings of the 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, Santa Catarina, Brazil, 2015, pp. 200-204.

[29]. M. Sahabuddin, B. Dutta, M. Hassan, Impact of cyber-attack on isolated power system, in *Proceedings of the* 3rd *Int. Conference Electr. Eng. Inf. Commun. Technol. (iCEEiCT'16)*, 2017, pp. 8-11.

[30]. D. Minoli, J. Kouns, IoT Security (IoTSec) considerations, requirements, and architectures, in *Proceedings of the IEEE Annu. Consum. Commun. Netw. Conference*, 2017, pp. 1006-1007.

[31]. H. Mhaisen, N. Fetais, A. Massoud, Secure smart contract-enabled control of battery energy storage systems against cyber-attacks, *Alexandria Eng. J*., Vol. 58, Issue 4, 2019, pp. 1291-1300.

[32]. S. Hasan, A. Dubey, G. Karsai, X. Koutsoukos, Electrical power and energy systems a game-theoretic approach for power systems defense against dynamic attacks, *Electr. Power Energy Syst*., Vol. 115, 2020, 105432.

[33]. M. Attia, S. Mohammed, H. Sedjelmaci, E. Aglzim, D. Chrenko, An efficient intrusion detection system against cyber-physical attacks in the smart grid, *Comput. Electr. Eng*., Vol. 68, 2018, pp. 499-512.

[34]. M. Stolpe, The internet of things: opportunities and challenges for distributed data analysis, *ACM SIGKDD Explor. Newsl*., Vol. 18, Issue 1, 2016, pp. 15-34.

[35]. K. M. Lässig, J. Kersting, K. Morik, Wind Power Prediction with Machine Learning. Computational Sustainability, *Springer*, 2016.

[36]. S. Snehi, A. Bhandari, Vulnerability retrospection of security solutions for software-defined cyber-physical system against DDoS and IoT-DDoS attacks, *Comput. Sci. Rev.*, Vol. 40, 2021, 100371.

[37]. H. Ge, D. Yue, X. Xie, C. Dou, S. Wang, Security control of cyber-physical system based on switching approach for intermittent denial-of-service jamming attack, *ISA Trans.*, Vol. 104, September 2020, pp. 53-61.

[38]. M. Chen, J. Wan, F. Li, Machine-to-machine communications: architectures, *Standards and Applications*, Vol. 6, Issue 2, 2012, pp. 480-497.

[39]. S. Ghosh, M. H. Ali, Exploring severity ranking of cyber-attacks in modern power grid, in *Proceedings of the IEEE Power & Energy Society General Meeting (PESGM'19)*, Atlanta, GA, USA, 2019, pp. 1-5.

[40]. C. Dong, X. Li, W. Jiang, Y. Mu, J. Zhao, H. Jia, Cyber-physical modelling operator and multimodal vibration in the integrated local vehicle-grid electrical system, *Appl. Energy*, Vol. 286, 2021, 116432.

[41]. K. Sajid, A. Abbas, H. Saleem, Cloud-assisted IoT-based SCADA systems security: a review of the state of the art and future challenges, *IEEE Access*, Vol. 4, 2016, pp. 1375-1384.

[42]. M. Allen, The SAGE Encyclopedia of Communication Research Methods, *SAGE Publications, Inc*, 2017.

[43]. Contributions of Supply and Demand Resources to Required Power System Reliability Services, *Electric Power Research Institute (EPRI)*, 2015.

[44]. P. Eder-Neuhauser, T. Zseby, J. Fabini, G. Vormayr, Cyber attack models for smart grid environments, *Sustainable Energy, Grids and Networks*, Vol. 12, 2017, pp. 10-29.

[45]. M. Ashrafuzzaman, S. Das, Y. Chakhchoukh, S. Shiva, F. T. Sheldon, Computers security detecting stealthy false data injection attacks in the smart grid using ensemble-based machine learning, *Comput. Secure.*, Vol. 97, 2020, 101994.

[46]. J. Pastor-Galindo, P. Nespoli, F. Gómez Mármol, G. Martínez Pérez, The not yet exploited goldmine of OSINT: opportunities, open challenges and future trends, *IEEE Access*, Vol. 8, 2020, pp. 10282-10304.

[47]. D. Govardhan, G. G. S. H. Krishna, V. Charan, S. V. A. Sai, R. R. Chintala, Key challenges and limitations of the OSINT framework in the context of cybersecurity, in *Proceedings of the 2nd International Conference on Edge Computing and Applications (ICECAA'23)*, Namakkal, India, 2023, pp. 236-243.

[48]. J. Rajamäki, S. McMenamin, Utilization and sharing of cyber threat intelligence produced by open-source intelligence, in *Proceedings of the International Conference on Cyber Warfare and Security*, Vol. 19, 2024, pp. 607-611.

[49]. R. Bodenheim, et al., Evaluation of the ability of the Shodan search engine to identify Internet-facing industrial control devices, *International Journal of Critical Infrastructure Protection*, Vol. 7, Issue 2, 2014, pp. 114-123.

[50]. Z. Durumeric, et al., A search engine backed by Internet-wide scanning, in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.

[51]. Cyberspace Search Engine, ZoomEye, www.zoomeye.org

[52]. M. Richardson, R. Bosua, K. Clark, Privacy and the internet of things, *Media and Arts Law Review*, Vol. 21, Issue 3, 2016, pp. 336-351.

[53]. Microsoft Defender Threat Intelligence, Microsoft Security, www.riskiq.com/

[54]. Recorded Future: Securing Our World with Intelligence, Recorded Future: Securing Our World With Intelligence, www.recordedfuture.com/

[55]. G. F. Lyon, Nmap Network Scanning: The Official Nmap Project Guide to Network Discovery and Security Scanning, *Insecure*, 2009.

[56]. Greenbone Openvas, OpenVAS, www.openvas.org/

[57]. R. D. Graham, MASSCAN: Mass IP Port Scanner, https://github.com/robertdavidgraham/masscan

[58]. Wireshark Go Deep, www.wireshark.org/

[59]. Tcpdump Libpcap, www.tcpdump.org/

[60]. PcapXray Design Specification, https://github.com/Srinivas11789/PcapXray

[61]. Network Intrusion Detection Prevention System, Snort, www.snort.org/

[62]. Zigbee2MQTT, www.zigbee2mqtt.io/

[63]. The Zmap Project, zmap.io/

[64]. M. I. Ecevit, M. H. Pervez, H. Dag, R. Creutzburg, The Open Source Intelligence (OSINT) in the electricity sector: balancing utility and responsibility, *Electronic Imaging*, Vol. 36, 2024, pp. 1-13.

[65]. Y. Zhang, R. Frank, N. Warkentin, N. Zakimi, Accessible from the open web: a qualitative analysis of the available open-source information involving cyber security and critical infrastructure, *Journal of Cybersecurity*, Vol. 8, Issue 1, 2022, tyac003.

[66]. The Berkeley Protocol on Digital Open Source Investigations, United Nations Human Rights, https://www.ohchr.org/sites/default/files/2024-01/OHCHR_BerkeleyProtocol.pdf

[67]. M. H. Pervez, M. I. Ecevit, N. Z. Naqvi, R. Creutzburg, H. Dag, Towards better cyber security consciousness: the ease and danger of OSINT Tools in exposing critical infrastructure vulnerabilities, in *Proceedings of the 8th International Conference on Computer Science and Engineering (UBMK'23)*, 2023, pp. 438-443.

[68]. S. Lee, T. Shon, Open source intelligence base cyber threat inspection framework for critical infrastructures, in *Proceedings of the Future Technologies Conference (FTC'16)*, 2016, pp. 1030-1033.

[69]. K. Aaltola, P. Taitto, Utilising experiential and organizational learning theories to improve human performance in cyber training, *Information & Security*, Vol. 43, Issue 2, 2019, pp. 123-133.

[70]. J. Cervini, A. Rubin, L. Watkins, Don't drink the cyber: Extrapolating the possibilities of Oldsmar's water treatment cyberattack, in *Proceedings of*

the International Conference on Cyber Warfare and Security, Vol. 17, Issue 1, pp. 19-25.

[71]. J. Asokan, A. K. Rahuman, B. Suganthi, S. Fairooz, M. S. P. Balaji, V. Elamaran, A case study using companies to examine the Nmap tool's applicability for network security assessment, in *Proceedings of the 12th International Conference on Advanced Computing (IcoAC'23)*, 2023, pp. 1-6.

[72]. P. Radoglou-Grammatikis, P. G. Sarigiannidis, G. Efstathopoulos, T. Lagkas, G. Fragulis, A self-learning approach for detecting intrusions in healthcare systems, in *Proceedings of the IEEE International Conference on Communications (ICC'21)*, 2021, pp. 1-6.

[73]. L. Li, W. He, L. Xu, I. Ash, M. Anwar, X. Yuan, Investigating the impact of cybersecurity policy awareness on employees' cybersecurity behavior, *International Journal of Information Management*, Vol. 45, 2019, pp. 13-24.

[74]. K. Huang, K. Pearlson, For what technology can't fix: Building a model of organizational cybersecurity culture, in *Proceedings of the Hawaii International Conference on System Sciences (HICSS'19)*, 2019, pp. 6398-6407.

[75]. J. Ferdinand, Building organisational cyber resilience: A strategic knowledge-based view of cyber security management, *Journal of Business Continuity Emergency Planning*, Vol. 9, Issue 2, 2015, pp. 185-195.

[76]. B. Geluvaraj, P. M. Satwik, T. A. Ashok Kumar, The future of cybersecurity: Major role of artificial intelligence, machine learning, and deep learning in cyberspace, in *Proceedings of the International Conference on Computer Networks and Communication Technologies (ICCNCT'18)*, 2019, pp. 739-747.

[77]. S. Sridhar, A. Hahn, M. Govindarasu, Cyber-physical system security for the electric power grid, *Proceedings of the IEEE*, Vol. 100, Issue 1, 2011, pp. 210-224.

[77]. Y. Mo, T. H. J. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, B. Sinopoli, Cyber-physical security of a smart grid infrastructure, *Proceedings of the IEEE*, Vol. 100, Issue 1, 2011, pp. 195-209.

[79]. Z. Kirsch, M. Chow, Quantum computing: The risk to existing encryption methods, https://www.cs.tufts.edu/comp/116/archive/fall2015/zkirsch.pdf

[80]. J. K. Cheng, E. M. Lim, Y. Y. Krikorian, D. J. Sklar, V. J. Kong, A survey of encryption standard and potential impact due to quantum computing, in *Proceedings of the IEEE Aerospace Conference*, 2021, pp. 1-10.

[81]. B. Sanders, Quantum cryptography for information-theoretic security: quantum cryptography, in Technological Innovations in Sensing and Detection of Chemical, Biological, Radiological, Nuclear Threats and Ecological Terrorism, *Springer Netherlands*, 2012, pp. 335-343.

5.

Interference and QoS in Machine-to-Machine Transportation Information Exchange Radio Network

Maksim Sidorovich, Yulia Ponomarchuk and Boris Davydov

5.1. Introduction

The data communication using wireless radio channels is the key element in the process of operational train traffic control and management. There are strict requirements to the reliability of data transmission, which is necessary to ensure transport efficiency and safety in harsh and changing conditions, especially when the traffic is intensive. Meeting these requirements is challenging because the railway corridor is usually considered as an area of intense electromagnetic interference. In the meantime, the sources of interference include not only the train power engines, but also elements of the railway's power supply systems. Additionally, trains regularly pass close to the industrial and telecommunication facilities, which also may emit strong interfering signals. All of these lead to the research and development of efficient and reliable solutions that may be implemented and deployed in various environments, including methods and tools to detect the sources of interference, influencing radio communications, as well as techniques to reduce data loss in wireless data transmission channels.

However, there is a lack of research in the specific area of electromagnetic compatibility of railway information systems. The literature survey reveals that insufficient attention has been paid to understanding the nature and sources of interference, their impact on train control systems, and to the development of methods for mitigation

Sidorovich Maksim
Far Eastern State Transport University, Khabarovsk, the Russian Federation

and elimination of interference and jamming. The literature survey in this topic shows that a comprehensive approach to solution of this problem is required, which should include:

- A method of computer modeling of the train-to-train data exchange in interval-based traffic control systems;
- Research of conditions for signal coverage and interference in train-to-train communication systems;
- A method for evaluation of the Bit Error Rate (BER) and its influence on communication traffic under significant interference influence;
- A method for evaluation of the impact of data communication quality on train flow using traffic indicators, as well as Quality of Service (QoS) parameters.

The paper includes three sections. The results of the literature survey are presented and the research problem is stated in the first section. The second section is devoted to the description of the suggested methods for interference and signal propagation modeling. Also, there are the results of computer simulations and comparison of different models for interference evaluation. The last section concludes the paper, summarizes the simulation results and further research directions.

5.2. The Literature Review and Problem Statement

The problem of evaluation of the interference impact to the quality of service in wireless networks, which are based on radio communication, is crucial for the data transmission reliability. There are a lot of papers, devoted to this issue, and they can be associated to the following three topics:

- Detection and classification of interference sources, their description and modeling;

- Detection of dependencies in influence of interference signals in wireless channels;

- Development of methods and tools, which may be used to decrease the impact of interference and jamming to the data transmission reliability and QoS.

It is taken without doubt that the interference and jamming sources may be densely deployed along and near electrified railroads. Therefore, the interference signal level may be quite significant and heterogeneous, when trains and railroad infrastructure communication is observed. The full-scale modeling and research on interference sources is quite a challenging task for a researcher, mainly, because of the movement of signal sources or measuring devices.

Meanwhile, wireless networks, using radio channel, are the sole choice for the deployment of telecommunications between the trains and the control centers. That also means the great significance of radio communications in traffic control systems for transport management and safety. Therefore, computer modeling for security and dispatching systems is the key stage for risk prediction and avoidance, as well as deployment expenses reduction.

Reviewing of papers and other research on interference on Swedish railways, showed, that there are a lot of interference signals registered, which are emitted by train equipment and railways electrical infrastructure [1, 2]. In paper [3], the authors developed a sustainable technique for decrease of interference in GSM-R standard bandwidth, which is important component of railway collision avoidance system (RCAS). The researchers determined that almost of all interference signals were generated by power supply infrastructure of railways. In addition, there were many impacts on train radio communications from cellular networks because of increase of the users' digital traffic due to insufficient carrier frequency spacing. In [4] the researchers described the threats of illegal interventions in traffic control by jamming signals using TETRA and GSM standards in railways applications. This type of problem was considered in [5], where the researchers made the evaluation using the Monte-Carlo method and found impacts of cellular communications like GSM, 3G and LTE to railway's GSM-R.

All the aforementioned issues led to the development of the 'Eccreport 162 Practical Mechanism to Improve the Compatibility Between GSM-R and Public Mobile Networks and Guidance on Practical Coordination' – the report of the European Commission on Electronics and Communications, which calls for actions to ensure the safe interaction between the public GSM networks and GSM-R, reduce mutual interference, and take measures to mitigate the impact of interference on railway radio communications [7]. For hybrid traffic control systems that use Eurobalise, the GSM-R network is fundamental, and the loss of link can lead to a complete halt of movement on a specific section of the railway until the connection is restored, which is critical for ETCS Level 2 and 3 traffic control systems [8, 9].

The research aimed at evaluation of interference influence on the Quality of Service (QoS) in wireless channels used for Vehicle-to-Vehicle (V2V) communication considers the latency and network throughput. This research is especially relevant for challenging terrain surface [10-12]. Adapting these results to railway applications can provide valuable insights for implementing Vehicular Ad-hoc Networks (VANETs) principles in the train management and traffic control. VANETs should provide communication in real time for city transport systems and traffic control for efficient city traffic planning and control [13]. Most papers on VANETs development consider city vehicle transport systems, but there are some about using VANETs in application to the railway transport as a basic technology for radio communications in order to provide additional traffic control services. It is observed that commercial companies in logistics and transport maintenance use these services. In [14], the principles of using the Terrestrial Trunked Radio (TETRA) standard for traffic control systems such as the RCAS and safety systems for different types of transport are considered. It supports direct mode operation (DMO) for direct connection in train-to-train applications like ‘train coupling’ [15].

The simulations of the communication systems allow evaluating their efficiency and reliability before their deployment and exploitation. Therefore, simulation of security and dispatching systems is a key stage for the risk evaluation and prediction, as well as, cost reduction. Before doing that, the literature survey provides the best methods and practices for computer modeling [10-12].

In Russian Railways, the ‘coupled trains’ system is implemented by the Central Station of the network and the central division of locomotives [16, 17]. This system aims to increase the efficiency and railway freight capacity. The main idea is to organize a short interval between the two subsequent trains (master and slave) using the radio channel and automatic velocity control. The implementation of such systems inevitably leads to the increase of radio traffic because a lot of overhead data are required for coordinating position, velocity, and other critical parameters of locomotive and railcars in real time. This includes train-to-train data transmission and between trains and the center control rooms. Notwithstanding, the use of 'coupled trains' allows to increase the efficiency and accelerate cost recovery. The scheme of message exchange within the 'coupled trains' movement is shown in Fig. 5.1.

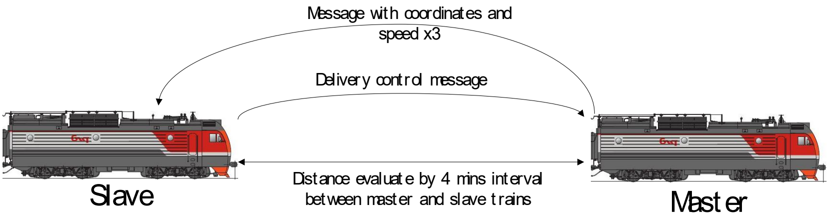

Message with coordinates and speed x3
Delivery control message
Distance evaluate by 4 mins interval between master and slave trains
Slave
Master

Fig. 5.1. The scheme of message exchange within the 'coupled trains' movement.

The previously presented analysis allows defining the research task as to choose the parameters for computer simulation of signal coverage, interference, and other radio channel characteristics in order to evaluate their influence on the BER and the probability of channel blocking using the Erlang B model. In order to solve these tasks, the following steps were undertaken:

- The evaluation of radio channel traffic generated by 'coupled trains';
- The analysis of the impact of impulse interference on train traffic control systems;
- The evaluation of the influence of BER on channel blocking;
- The evaluation of the probability of switching to a secondary analog channel, which reduces the reliability of the train radio communication system;
- The analysis of the methods for simulation of artificial interference from railway infrastructure.

The results of research may be used for estimation of wireless channel parameters and providing efficient and increased freight capacity using 'coupled trains' mode.

5.3. Radio Signal and Interference Modeling

The simulation of data transmission consists of the signal generation at a transmitter, signal propagation and interference in an area and techniques for signal processing at a receiver. Therefore, the computer model consists of:

- Test data generation, frame construction and modulation of the payload signal such as the structure of a DMR frame;

- Simulation of a radiating antenna and other equipment for all channel, an interference source and signal propagation in terrain;

- Description of radio channel with additive Gauss noise and additive interference by Middleton class A type model;

- Evaluation of interference signal impact to QoS, measured by BER.

5.3.1. Impulse Interference Simulation

Impulse interferences may decrease the reliability of data communication in wireless channel and therefore deteriorate parameters of train traffic. There are many contact and switching processes that may generate impulse interference along the railways. In accordance with this research, the analysis of data, gathered in 2020 on railways with power supply system based on alternating current with 25 kV voltage, using the interference control system and diagnostics tools, was conducted. The results of the analysis show that the majority of interference signals (around 38 %) are related to the hardware malfunction and failures of power supply infrastructure. The interference sources are shown in Fig. 5.2.

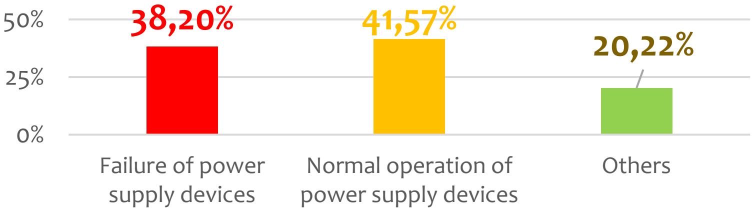


Fig. 5.2. The interference sources.

There are many mathematical and computer models for interference signal generation and propagation [20-22]. Usually, in wireless channel modelling the Middleton classes (A/B/C) models are used, as well as, alpha-stable, or Gauss models. The Middleton class A model, based on statistics and physics, is considered to be universal for EMI modeling and has a lot of implementations in various programming languages [23]. The Middleton class A models are designed in order to analyze the signal propagation, when impulse noise dominates in the radio channel. It is interesting for researchers and allows making accurate models and change parameters in them. The Middleton class A model may give a detailed analysis of impulse interference with special characteristics and special features for data communication in wireless channel.

In order to define the parameters of the Middleton class A model it is necessary to make some specified steps. First of all, to collect the raw data including the amplitude and frequency parameters. In addition, this step contains the duration, shape and amplitude of the noise signal. The data analysis for the important parameters of the Middleton class A model includes computing the mean value of the interference level (A), the standard deviation of the interference level (σ), and the degree of fit of the interference statistics to the Gaussian distribution (m). All these data are required in order to compute the model parameters. The main statistical noise characteristics is the probability density function, ω(ε) – which is defined for model class A as follows [23]:

$$\omega(\varepsilon) = 2e^{-A_A} \sum_{m=0}^{\infty} \frac{A_A^m \, \varepsilon \, e^{-\varepsilon^2/2\sigma_m^2}}{m!\sigma_m^2} \qquad by, \;\; \sigma_m^2 = \frac{\frac{m}{A_A} + \Gamma_A}{1+\Gamma_A} \qquad (5.1)$$

where A_A is the “overlap index”. It is the product of the average number of emission events impinging on the receiver per second and mean duration of a typical interference source emission, and A_A [10-2, 10]; Γ_A is the ratio σ_G^2 / Ω_{2A}, where σ_G^2 is the intensity of the independent Gaussian component, Ω_{2A} is the intensity of the impulse non-Gaussian component, and Γ_A [10-6, 10] in general. The degree of fit of the interference statistics to the Gaussian distribution (m) describes the number of components, which should be considered in interference samples’ generation. In this paper, the parameter m is equal to 10 in accordance with the computing resources.

The samples of different impulse interference signals, generated when current pickup conditions were violated and during arc discharge on train's pantograph occurred, were collected in PhD thesis [24]. The data of amplitude-frequency characteristics of the signals, generated by violation of current pickup conditions on a section of a railway track, which is provided with power supply of alternate current 25 kV and 50 Hz, are presented in Table 5.1.

Table 5.1. The amplitude-frequency spectrum harmonics level of interference signal due to violation of current pickup conditions

Number of Harmonics, n	**1**	**2**	**3**	**4**	**5**	**6**
f, Hz	330	672	1001	1360	1691	2050
L, dB	-23	-26	-31	-32	-41	-51

The algorithm for impulse interference model construction is shown in Fig. 5.3. It allows using the recorded data (i.e., by software-controlled radio), as well as generating synthetic interference signal. The signal-wave envelope is defined by processing the data, which is stored in the device memory. The interference signal is transferred to communication frequency channel, then it is normalized and goes to the block for data processing and synthesis.

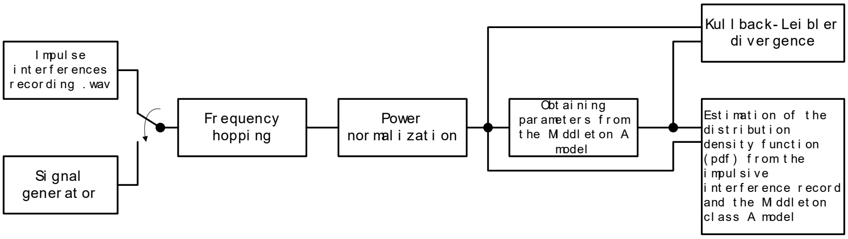

Impulse interferences recording .wav
Signal generator
Frequency hopping
Power normalization
Obtaining parameters from the Middleton A model
Kullback-Leibler divergence
Estimation of the distribution density function (pdf) from the impulsive interference record and the Middleton class A model

Fig. 5.3. Algorithm of the interference model construction.

The model for interference signal generation was constructed in Matlab and consists of generation of the carrier harmonic signal, the input of initial spectral data and uses the Hilbert transform to form the signal envelope. Next, the excessive spikes are smoothed and the signal frequency is shifted to the operating frequency range of 150 MHz. Finally, the white Gauss noise is added to the payload signal with the level, which provides the specified signal-to-noise ratio (SNR), equal to 5.5 dB. The calculation of parameters A and Γ for the model is carried out according to the formulas [25]:

$$A = \frac{9(e_4 - 2e_2^2)^3}{2(e_6 + 12e_2^3 - 9e_2e_4)}, \tag{5.2}$$

$$\Gamma = \frac{2e_2(e_6 + 12e_2^3 - 9e_2e_4)}{3(e_4 - e_2^2)^2}, \tag{5.3}$$

where e_2, e_4, e_6 are the variance, coefficient of kurtosis, and 6th order moment of the signal amplitude.

The analysis of impulse interference with the help of the Middleton class A model allows using the previously obtained experience and data. It allows generating hypotheses and verify them using computer modeling and simulation. In addition, it allows making simulations using SDR devices.

The interference signal, which was generated in the area of connection "contact-wire line – pantograph" under conditions of electrical discharge due to current pickup malfunction and gathered in [24], has the values of parameters: A = 1,11 and Γ = 2,69. The plots of probability density functions (pdf) of real interference signal and its model, obtained by Middleton class A model, are shown in Fig. 5.4.

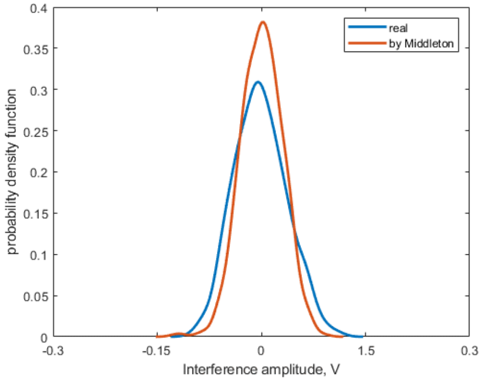


Fig. 5.4. The probability density functions of the real interference signal (cyan line) and its model, obtained by Middleton class A model (red line).

The conformance evaluation of the model and the real interference signal is made using Kullback–Leibler divergence (KL) metrics [26]. The distance computed using KL between the pdfs is equal to 0.153, which means their similarity. The values of mathematical expectation for the real signal and its model are also close, being 0.0053 V and 0.0045 V, respectively.

5.3.2. The Simulation of Signal and Interference Propagation

The Liebler Longley-Rice model is recommended to be used for signal strength prediction in case of rough and uneven terrain. It allows estimating losses of power along signal propagation, regarding many factors such as the distance between transmitter and receiver, frequency, height of antennae, buildings and rough terrain. The model is based on statistical methods for probability estimation for signal power, registered at the receiver [27]. The main equation for the signal propagation is as follows:

$$L = L_{fs} + L_{d} + L_{tr} + L_{ol}, \tag{5.4}$$

where L_{fs} – the free space losses, L_{d} - the diffraction losses, L_{tr} – the tropospheric losses, L_{ol} – the losses due to other causes. This model is based on empirical data and it is widely used for estimation of the cellular networks' coverage.

The signal coverage was simulated using Matlab with carrier frequency equal to 150 MHz. The evaluation of signal level at the receiver of a slave train was conducted, regarding changes in rough terrain. Within computer simulation it was supposed that the trajectories of the trains were synchronized by as they are moving in a "coupled trains" mode.

The simulation of the natural conditions was implemented using the provided input data on the terrain surface, and small changes in the interval between trains (master and slave), which was given around 6 km. This interval is typical for the "coupled trains" mode in traffic control. Also, in the signal coverage model it was taken into account that the transmission power of DMR on railroads is equal to 5 W and sensitivity threshold at the receiver is equal to -113 dBm. The results of signal coverage simulation in "couple trains" mode movement are shown in Fig. 5.5. The top figure demonstrates the transmitted signal level on the terrain. The figure below depicts the level of signal at the receiver.

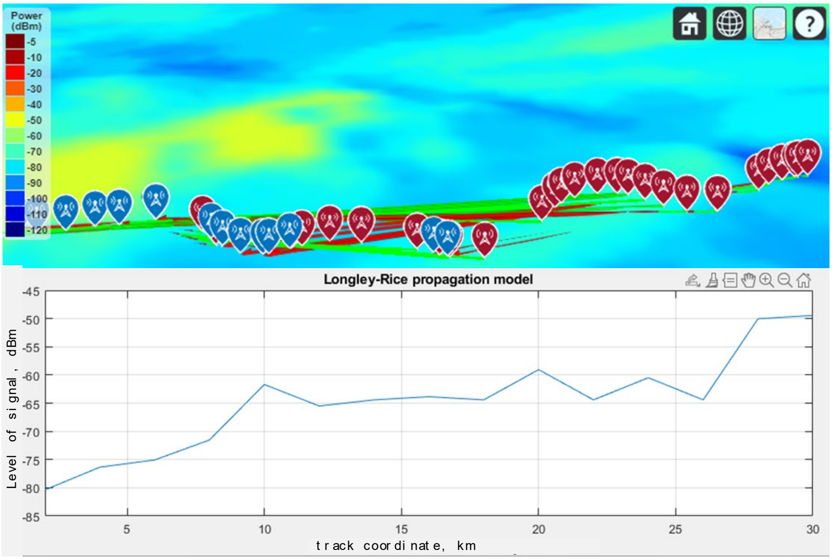


Fig. 5.5. The level of the transmitted (top) and the received (bottom) signal

The simulation results show the possibility to imitate the generation of the payload signal with impulse interference, while modeling the “coupled trains” mode movement on railroads. This allows conducting further computer experiments for the evaluation of influence of traffic modes, radio devices parameters, and other factors to data communication reliability in train-to-train or train-to-infrastructure wireless communications.

5.3.3. Impulse Interference Impact to the Data Communication Reliability

The Bit Error Rate (BER) is the ratio of incorrectly received bits to the total number of transmitted bits in a given period of time. The BER value is used to estimate the influence of impulse interference to data transmission quality. Additionally, the BER is the one of the most widely used indicators of Quality of Service in wireless channel, since it shows the approximate probability of errors in sequences of bits [28].

The simulations of the impact of impulse interference to the BER allows verifying the robustness of a radio system to external effects and designing and implementing measures to improve the reliability of data transmissions. These measures may include the application of filters for noise cancelling, optimal modulation and coding, as well as algorithms for error correction.

As a result of deployment of traffic control systems and the increase of overhead data transmissions in radio networks, more detailed research, considering the evaluation of impact of the impulse additive interference to radio channel accessibility and quality, becomes more urgent. Its results should lead to the development of the efficient, reliable, and energy-efficient communication systems for railway transport system.

The scheme of modeling the data reception over the wireless channel, affected by the impulse interference, is shown in Fig. 5.6. According to the DMR frame structure, the data for transmission was generated randomly. After modulation, the impulse interference, generated by Middleton class A model, was added to the signal. In addition, the white Gaussian noise with signal/noise ratio equal to 18 dB was added afterwards.

Further, the demodulation and decoding at DMR receiver was simulated. Finally, the BER was evaluated and compared with the DMR standard recommendations.

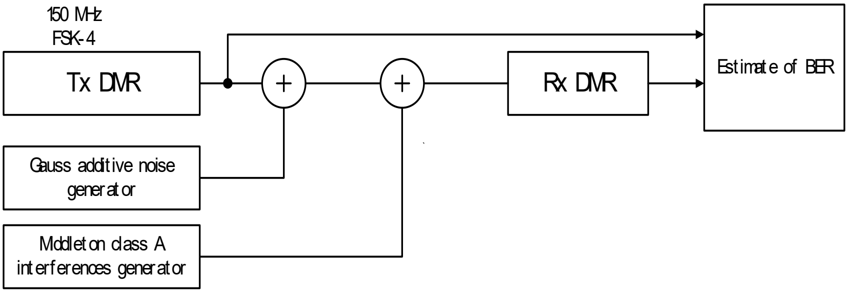

150 MHz
FSK-4
Tx DMR
+
+
Rx DMR
Estimate of BER
Gauss additive noise generator
Middleton class A interferences generator

Fig. 5.6. The scheme for simulation of the received data quality over the wireless channel, affected by the impulse interference

The BER dependence on the signal-to-noise ratio is shown in Fig. 5.7. It should be noted that impulse interference signal strength on railways often has values close to the useful signal, which means that SNR is within limits of 0 to 12 dB.

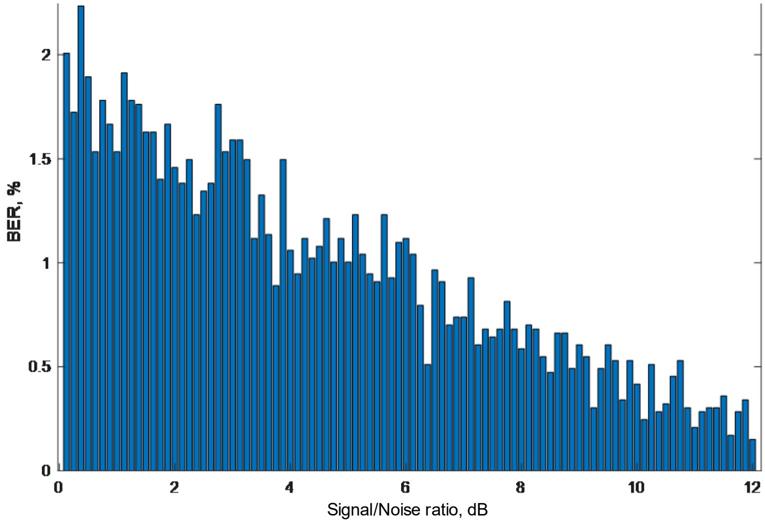


Fig. 5.7. The BER dependence on the SNR

The results, shown in Fig. 5.7, lead to conclusion on the overall level of errors in the useful signal. For railroads when using the DMR, the BER value should be less than 1.5 %.

5.3.4. Modeling Data Transmissions for Train-to-Train Communications

Delays in data transmissions may lead to serious malfunctions and accidents in a transport system, like failures of equipment of various categories of importance. These increase the risks for safety on the railways and reduce railroad traffic capacity. In practice, it means the increase of time delay at stations, breaking the train schedule, and, as a potential consequence, financial loss. In order to optimize the data communication systems and decrease the delays in traffic control systems it is necessary to provide an uninterrupted and efficient operation of railway transport.

Further the influence of BER to the probability of channel blocking may be estimated. The total number of bits in one DMR frame is equal to 264 per 30 ms time interval. The size of a file, needed for the traffic control in "coupled trains" mode, is equal to 168 bits. Therefore, the probability PFE that at least one bit in a bit frame is incorrect can be calculated according to Eq. (5.5), and the probability of successful transmission P_{ST}, given the number of repetitions n, can be computed by Eq. (5.6).

$$p_{FE}{}^{k}=(1-BER)^{k}, p_{FEL}{}^{(n-k)}=BER^{(n-k)} \tag{5.5}$$

$$P_{ST}=\sum_{k=0}^{2}\binom{n}{k}\cdot p_{FE}{}^{k}\cdot p_{FEL}{}^{n-k} \tag{5.6}$$

where n is the total number of bits in the frame; the BER is the percentage of bit errors, %. PFE is the probability of incorrect transmission, where k – the number of allowed errors; PFEL – the probability of error-free transmission, which is equal to 0.9879 according to the calculation. It should be noted that the DMR standard uses Trellis-coding, which is able to correct up to two error bits per frame. Therefore,

$$N_s=N\cdot P_{ST} \tag{5.7}$$

where N is the number of messages, transmitted over the wireless channel in the “coupled trains” mode. The probability PB of channel blocking can be calculated using Eq. (5.8) according to the Erlang’s model B. The Erlang B model is a popular choice in order to determine the parameters of wireless channel for professional radio systems [29].

$$P_B=\frac{A^m/m!}{\sum_{k=0}^{m}\frac{A^k}{k!}} \tag{5.8}$$

Here, the parameter m corresponds to the number of possible reserved channels (m=2). For the “coupled train” mode, the reserved channel is the voice analog channel at 2.15 MHz. However, it is also reserved for voice dispatching. Then, the total traffic intensity A is determined by the relationship:

$$A=\frac{N_s\cdot t_{hold}}{t} \tag{5.9}$$

where hold is the mean time of channel hold equal to the frame duration (30 ms) and t is the time interval for measurements. In our simulated situation, the number of messages during the period of movement over the railroad section is 874, and the number of successful messages is 863. The total estimated intensity of information traffic is estimated as 0.0115 Erl.

The results of calculations show that, if the BER value is 2 %, the probability of channel blocking for DMR with reserved analog channel is equal to 1.3 %. This value means that all the frames may have at least one error bit. Therefore, the increase of throughput of the wireless channel may be detected in the system, because of the necessity for repeated transmissions.

5.4. Experimental Results for Data Transmission Quality Analysis

The experimental analysis of data frames transmission quality in the radio channel within the "coupled trains" mode movement was carried out in a section of a main electrified railroad in the Far Eastern region of Russia. The main purpose of the research was the estimation of the data communication parameters between two trains with short distance between them. The actual distances were within from 5.3 km to 8.3 km. In the experiment the level of the payload signal was registered at the receiver of a slave train, as well as the SNR values. The experimental results are shown in Fig. 5.8, where the levels of the received signal in real environment are compared to the Longley-Rice model.

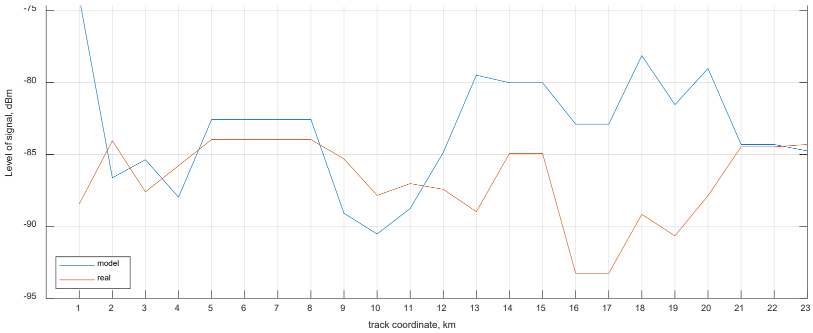


Fig. 5.8. The values of the experimental and model signal levels along the path.

The analysis of the registered measurements leads to the conclusion that the degree of similarity of experimental and simulated data is acceptable. The evaluation of standard deviation of the differences between the experimental and simulated data along the train track is around 1.2 dB. In the experiment conduction process some sections of railroad were found where the SNR values were less than parameters of normal work DMR equipment. This was accompanied by the data frame loss and decrease of the efficient bitrate because of packet repeating. The high level of interference, shown in Fig. 5.9, summarizes the packet loss rates at the section. This figure allows estimating the influence of interference of longer duration, evaluating their impact, and localizing.

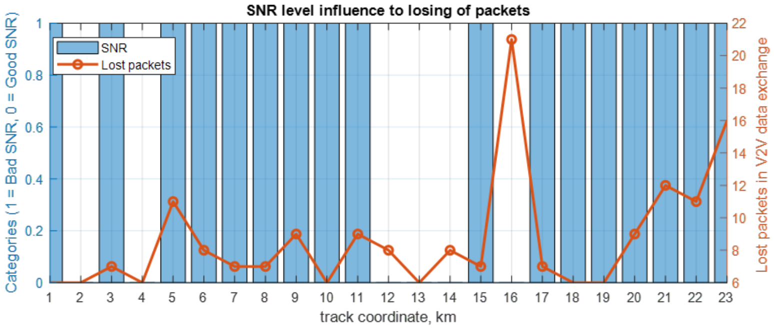


Fig. 5.9. The number of lost packets at high interference level.

The critical losses of frames between the transmitter and receiver match the locations of the decreased useful signal for both the real environment and the simulated one. However, they are different from the permanent zones of radio interference, which were detected previously by a car-laboratory on this segment of the railroad. In addition, it should be noted, that the zone between 16 and 17 kilometres is a zone with current pickup conditions disruptions.

5.5. Conclusions

The simulation results of the DMR radio system and analysis of impulse interference impact to QoS in the corresponding wireless network, which is evaluated using BER, demonstrate the key aspects for providing reliability and efficiency for railways communications. This is even more significant in the context of implementing new types of traffic control like the “coupled trains” mode. The computer simulations of communication standards and protocols in conditions of interference allow understanding and optimizing the parameters of radio systems for special requirements in railway industry. The additive impulse interference can be simulated using Middleton class A model, which is based on the experimental data. This model provides the evaluation of the impact of interference from different sources on QoS parameters of communications in wireless networks.

The analysis of experimental data collected while using “coupled trains” mode movement demonstrated the importance of the measures for interference influence mitigation, since it may cause the delay in the transmissions of control signals and potentially disrupt the transport safety. The research of the features and capabilities of digital communication standards, used in the transport systems, shows that the data transmissions are still susceptible to interference Although such standards were chosen as the future for railway industry, since they may minimize the problems associated with electromagnetic compatibility of trains with the infrastructural objects along the electrified railroads. The presented in this paper model for impulse interference simulations corresponds to the empirical data and allows estimating the signal coverage and the evaluating the impact of impulse interference on the QoS parameters of the wireless communication channel.

The integrated approach to simulation the physical layer of radio communication standards and protocols and to analysis of the impact of additive interference based on BER is an important part of the development and optimization of modern railway communication systems, which is aimed at improving their reliability, safety and economic efficiency.

Acknowledgements

This work is partially supported by the RF Ministry of Transport according to the program of scientific and educational organizations research for 2024. The authors thank the Russian Railways infrastructure directorate of JSC RZD for the statistical data provided.

References

[1]. A. Wisten, S. Niska, J. Ekman, D. Björklöf, J. Delsing, Experimental investigation of EM noise environment surrounding detector systems in Swedish railways, IEE Proceedings - *Electric Power Applications*, Vol. 153, No. 2, 2006, pp. 191–196.

[2]. S. Niska, Measurements and analysis of electromagnetic interferences in the Swedish railway systems, Ph.D. Thesis, *Luleå University of Technology*, Sweden, 2008.

[3]. S. Dudoyer, S. Azzopardi, J.-P. Reynaud, J.-F. Mariscotti, A. Pignari, Susceptibility of the GSM-R transmissions to the railway electromagnetic environment, in Infrastructure Design, Signalling and Security in Railway, IntechOpen, 2012, pp. 503–522.

[4]. S. Lakshminarayana, R. Ma, T. Basar, Signal jamming attacks against communication-based train control: Attack impact and countermeasure, in *Proceedings of the 11th ACM Conference on Security & Privacy in Wireless and Mobile Networks (WiSec'18),* 2018, pp. 160–171.

[5]. M. González-Gonzalo, A. Garrido, J. Batlle, Simulation software tool to evaluate interferences between cellular public networks and GSM-R system, *International Journal of Safety and Security Engineering*, Vol. 6, No. 4, 2016, pp. 720–727.

[6]. European Railway Agency (ERA), ERTMS/ETCS System Requirements Specification, Mandatory Specification SUBSET-026 v3.4.0, January 2015. [Online]. Available: http://www.era.europa.eu/Document-Register/Pages/Set-2-System-Requirements-Specification.aspx

[7]. J. Adin, S. Mendizabal, U. Arrizabalaga, G. Alvarado, G. Solas, J. Rodriguez, Rolling stock emission testing methodology assessment for Balise transmission module system interoperability, *Measurement*, Vol. 77, 2016, pp. 124–131.

[8]. F.-X. Socheleau, J.-M. Passerieux, C. Laot, Acoustic modem performance assessment via stochastic replay of at sea recorded underwater acoustic communication channels, in Proceedings of the *4th UAM Conference: Underwater Acoustic Measurements*, 2011, hal-00609261.

[9]. M. N. Cheptsov, V. E. Sorokin, Calculation of the maximum allowable time without radio communication in train interval regulation systems based on digital radio channel, *Transsiberian Bulletin*, 4, 44, 2020, pp. 74–83. (in Russian) [Online]. Available: https://cyberleninka.ru/article/n/raschet-maksimalno-dopustimogo-vremeni-otsutstviya-radiosvyazi-v-sistemah-intervalnogo-regulirovaniya-dvizheniya-poezdov-na-baze

[10]. J. Felez, M. A. Vaquero-Serrano, Virtual coupling in railways: A comprehensive review, *Machines*, Vol. 11, No. 5, 2023, Article 521.

[11]. H. Duan, D. Li, Y. Tang, Y. Zhang, Y. Liu, Research on virtual coupling train operations based on moving-block and vehicle-to-vehicle communication, *Journal of Physics: Conference Series*, Vol. 1631, No. 1, 2020, Article 012063.

[12]. C. Wille, M. Grünhäuser, Evolution of vehicle to everything (V2X) communication in the railway sector, in *Proceedings of the 2023 International Conference on Intelligent Traffic and Transportation (ICITT'23)*, 2023.

[13]. R. Tomar, M. Prateek, G. H. Sastry, Vehicular Adhoc Network (VANET) – An Introduction, *International Journal of Control Theory and Applications*, Vol. 9, No. 18, 2016, pp. 8883–8888.

[14]. ETSI EN 300 392-2 v3.2.1, Terrestrial Trunked Radio (TETRA); Voice plus Data (V+D); Part 2: Air Interface (AI), European Standard (Telecommunications series), 2007. [Online]. Available: https://www.etsi.org/deliver/etsi_en/300300_300399/30039202/03.02.01_60/en_30039202v030201p.pdf

[15]. C. Lehner, R. García, T. Strang, On the performance of TETRA DMO short data service in railway VANETs, *Wireless Personal Communications,* Vol. 69, No. 4, 2013, pp. 1647–1669.

[16]. E. N. Rozenberg, A. A. Abramov, V. V. Batraev, Interval train movement control, *Railway Transport*, No. 9, 2017, pp. 19–24.

[17]. V. A. Olentsevich, R. Yu. Upyr, A. A. Antipina, Efficiency of implementing interval train movement control using the 'Virtual Coupling' system in the section, *Modern Technologies. System Analysis. Modelling*, 2, 66, 2020, pp. 118–124. (in Russian) [Online]. Available: https://cyberleninka.ru/article/n/effektivnost-vnedreniya-intervalnogo-regulirovaniya-dvizheniya-poezdov-po-sisteme-virtualnaya-stsepka-na-uchastke

[18]. ETSI TS 102 361-1, Digital Mobile Radio Systems; Part 1: DMR Air Interface Protocol, 2017. [Online]. Available: https://www.etsi.org/deliver/etsi_ts/102300_102399/10236101/02.05.01_60/ts_10236101v020501p.pdf

[19]. V. Lantsov, S. Eranosyan, Electromagnetic compatibility of pulse power sources: problems and solutions. Part I, *Power Electronics*, No. 10, 2006, pp. 58–64.

[20]. G. Bedicks, C. E. S. Dantas, F. Sukys, F. Yamada, L. T. M. Raunheitte, C. Akamine, Digital signal disturbed by impulsive noise, *IEEE Transactions on Broadcasting*, Vol. 51, No. 3, 2005, pp. 322–328.

[21]. M. Katayama, T. Yamazato, H. Okada, A mathematical model of noise in narrowband power line communication systems, I*EEE Journal on Selected Areas in Communications*, Vol. 24, No. 7, 2006, pp. 1267–1276.

[22]. G. Ndo, F. Labeau, M. Kassouf, A Markov-Middleton model for bursty impulsive noise: Modelling and receiver design, *IEEE Transactions on Power Delivery*, Vol. 28, No. 4, 2013, pp. 2317–2325.

[23]. D. Middleton, Procedures for determining the properties of the first-order canonical models of Class A and Class B electromagnetic interference, *IEEE Transactions on Electromagnetic Compatibility*, Vol. 21, No. 3, 1979, pp. 190–208.

[24]. M. Gorevoy, Electromagnetic compatibility of the traction power supply system with train radio communication, Ph.D. Thesis, *Moscow State University of Railway Engineering*, Russia, 2011.

[25]. S. M. Zabin, H. V. Poor, Efficient estimation of Class A noise parameters via the EM algorithms, *IEEE Transactions on Information Theory*, Vol. 37, No. 1, 1991, pp. 60–72.

[26]. K. Gulati, M. Nassar, B. L. Evans, Statistical modeling of co-channel interference, in *Proceedings of GLOBECOM 2009 – IEEE Global Telecommunications Conference*, 2009, pp. 1–6.

[27]. P. L. Rice, A. G. Longley, K. A. Norton, A. P. Barsis, Transmission loss predictions for tropospheric communications circuits, Technical Note 101, U.S. Dept. of Commerce NTIA-ITS, Rev. Jan. 1, 1967.

[28]. M. Nassar, K. Gulati, M. R. DeYoung, B. L. Evans, K. R. Tinsley, Mitigating near-field interference in laptop embedded wireless transceivers, *Journal of Signal Processing Systems*, Vol. 63, No. 1, 2011, pp. 1–12.

[29]. B. Baynat, S. S. Elayoubi, F. Fouquet, Extended Erlang-B law for performance evaluation of radio resources sharing in GSM/(E)GPRS Networks, in *Proceedings of the 16th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC' 2005)*, IEEE, 2005, pp. 2277–2281.

6.

On Train Delay Predicting Using a Markov Model

Boris Davydov and Vladimir Chebotarev

6.1. Introduction

Predicting the situation development in a real environment is an important element of the train traffic operational control. The forecast creation is associated with difficulties caused by the uncertainty of the influencing factors that deviate the train's trajectory from the schedule. First of all, these include restrictions due to overloading of the railway infrastructure, as well as technical failures. Another group of factors is related to the uneven passenger and cargo flow nonuniformity. In addition, there are deviations from the normative processes that are caused by errors in personnel work. The quality of railway operations varies randomly throughout the day and over longer time intervals. The prediction of probable schedule disruptions is usually formed by the train manager who carries out operational traffic control on a given railway site by corrective actions developing. At the start of forecasting process, the manager has two types of information. The first type characterizes the current situation in the control area, namely, the spatial train position, the state of the rolling stock and infrastructure. Some of these features are known quite accurately (in particular, the coordinates of trains), while others are known with a high degree of uncertainty (for example, the spatial distribution of precipitation). This type also includes data on the "value" of trains, that is, the priority of their passage through a bottleneck when conflicting operational situation arise. The second type of information using in operational scheduling includes a set of information about situations that had arisen at the control area in previous time periods. These historical data form some integrated images that the

Boris Davydov
Far Eastern State Transport University, Khabarovsk, Russia

decision maker uses as a “base of reference” when developing the operational management decisions. In a similar manner, patterns of the vehicle run process are identified by historical information handling which is implemented in the intelligent personnel support system. Target transportation objectives and previous experience in train traffic managing are accumulated in a set of rules and instructions, which is also used by the decision maker as guiding or restrictive information. As you can see, both fairly specific and fuzzy information are used in the process of predicting. Deterministic modeling of an individual train trajectory and of a group of trains often gives unreliable results, as shown by common practice. It allows us to take the influence of only one factor into account, namely the spatio-temporal dynamics of moving units on a fixed railway network. Process detailing in deterministic models often provides a small gain in solving the delay prediction problem, although it is accompanied by a large amount of computation. A more accurate forecast can be obtained if we take into account the random component of the railway process, which depends in particular on a number of additional factors other than the limited capacity of stations and sections. Therefore, probabilistic modeling of the running process is becoming increasingly widespread [1].

The current situation on the railway site at the time of prognosis created can be described in one of two ways: either based on an analysis of train trajectories [2] or by taking the factors influencing this movement into account [3]. In the first case, the forecast of the scenario development is created by processing current data on the executed train schedule. In this case, regression analysis or Markov chains are used as the main methods for calculating delays predicted [4, 5]. When using the second method of calculating delays, it is necessary to know the indicators and characteristics of the current situation such as infrastructure load, weather conditions, etc. as well as the functional relationship of schedule deviations with the intensity of these factors. The relationship function between each pair of values can be stored in the knowledge base as a previously obtained pattern and can be extracted from the historical data set in real time during the forecasting process. It is this latter technique that is used to predict delays by machine processing of statistical data [6]. The analysis shows that a significant increase of processed historical data volume does not lead to an improvement in the reliability of the tardiness forecast in some cases. Indeed, the use of big data and machine learning methodology allows us to obtain a mean absolute prediction error (MAE) of less than one minute. However, the forecast result cannot be considered reliable if the dispersion area of the random delay value, determined by the RMSE indicator, reaches tens of minutes (see [7], for example). The dispersion may be caused by a large number of influencing factors, their weak correlation and rapid variability. This, in turn, may cause a wide variety of operating situations and, accordingly, a dispersion of train movement characteristics, which negatively affect the results of model training. Apparently, this serves as a limitation in the behavior patterns analysis of a complex transport system and the developing of operational control methods. It is likely that there are traditional models that can achieve predicting reliability comparable to machine models by making the most of a limited amount of statistical data. In our opinion, these models also include Markov chains, which are created for each specific operational situation. The paper [8] is one of the researches developing this direction.

Our studies show that the process of train movement on busy suburban lines can be considered statistically stable enough at observation intervals of no more than three hours. This interval increases to six hours on railways with predominantly freight traffic. It is quite difficult to determine probabilistic characteristics by processing information over such short intervals, since during the observation period up to ten (in the best case, up to twenty) trains pass through the control area. Therefore, it is necessary to solve the problem of effective historical and current data use when a future delay prediction creating based on a very limited amount of operational information.

In this study, the train run process is represented as a sequence of deviations from the schedule upon arrivals and departures from stations. The values of these quantities coming from a given location in real time form a data profile of the current situation. This profile includes marks for all visited trains at a given location in a three-hour interval (or six hours for freight line traffic). After the interval expires, this profile is added to the historical database.

In this paper, the prediction creating process of train run is divided into three stages. At the first stage, the characteristics of the Markov model are determined for each of the historical profiles stored in the databases. This extensive analytical work is carried out before the start of the current dispatching period. In operational work, the need arises to prediction create when schedule violations are detected. Then a three-hour time interval in the past with a train situation that is close to the current scenario is determined. The closeness of the current and the reference situation is recorded as a result of comparing the corresponding profiles. This algorithm based on the k-nearest neighbours method (k-NN method), is used to determine a similar train traffic situation [9]. At the third stage, the problem of the arrival deviation probability distribution is solved using the Markov model of delay evolution process in the realized reference situation. The proposed approach is applicable when adapting to rapid changes in the situation. The prognosis is promptly revised by searching for a new reference situation in the databases when significant deviations from the schedule or train flow congestion on the railway line occur. The algorithm for predicting the arrival deviation distribution at the destination can be simplified if we take into account the correlations between random deviations that determine the train run trajectory. In Section 6.4 of this paper, a study is conducted that allows, in the presence of correlation coefficients close to 0 or ±1, reducing the number of calculated transition matrices used to prediction formulate.

The paper is structured as follows. Section 6.2 presents the literature review on the problem of short-term train traffic predicting. Section 6.3 presents the algorithm for creating a Markov model for the process of schedule deviation evolution. Section 6.4 examines the problem of correlation estimating in a sequence of random deviations representing a train trajectory. Section 6.5 is devoted to the analysis of statistics obtained during the actual operation of commuter trains on a busy railway line and the verification of the results of stochastic traffic modeling. A method for assessing the accuracy of the forecast is also proposed there.

6.2. Literature Review

At an early stage of solving the predicting problem, several approaches for train arrival times calculating were developed. Among them, it is necessary to highlight, first of all, micro-modeling methods [10], analytical methods [11] and data-driving methods [12]. Deterministic micro-models driven by events [2, 13] allow a detailed analysis of the train delay formation process. However, such a delay prediction is not reliable enough, since most of them do not take the stochastic nature of disturbing factors influence within the forecast horizon into account. The direction of stochastic traffic modeling lay down by the works of M. Carey and A. Kwiecinski in the 90s has shown its significant advantages in dispatching problems solving (see, for instance, [14]). Over the past two decades, methods for describing train traffic and probabilistic forecasting of their delays have been intensively developed. The distributions of free running times are given and the process dependencies are computed as L.E. Meester and S. Muns [15] formalized the model presented in the earlier research as a stochastic event graph and a mixture of the corresponding distributions.

T. Büker and B. Seybold [16] modelled delays as random variables, described with suitable distribution functions, and applied analytical methods to compute delay propagation in a mesoscopic graph-based model. A stochastic delay propagation approach based on processed train event recorders data was presented by Medeossi et al. [17]. Delay propagation is computed by means of stochastic blocking times. The method relies on asynchronous simulation of individual train runs based on probability distributions of train motion parameters [18].

Recently, Markov chains have been increasingly used to predict deviations from a schedule. Apparently, one of the first works that uses Markov chain to describe the railway process is the paper [19]. In this research, a model for urban passenger train traffic is created. The method for modelling the uncertainty of train delays based on a Markov stochastic process is described in [20, 21]. The dynamics of a train late arrival over time and space is presented in the form of delay evolution as a time-dependent random variable. A train run is represented as a Markov chain with state transitions in discrete moments that represent arrival and departure events. Probability distribution of an arrival delay in every station change over time in discrete steps as more information about current delays becomes available. The approach based on probability transition matrices can be useful in modeling out-of-control situations on a large railway network [22]. From historical data, it is possible to derive a statistical relationship between delays in adjacent locations and use this result to assess the scale and danger of a failure incident spreading across the network. A number of studies have shown that using a Markov model together with the processing of large data sets can significantly increase the reliability of train delays predicting [5]. The Markov model, like any model, has some limitations, since it assumes that the train's arrival time in the future depends only on the current delay and does not depend on delays in the past. In cases where the dependencies are more complex, a model called Bayesian network is used [23]. Models for predicting vehicle traffic that use time series obtained from statistical data are also common. Thus, in [24] the seasonal autoregressive integrated moving average (SARIMA) model is used to predict the time characteristic of a bus trip. The study compares statistical time series models with learning-based models for short-term traffic predicting. The standard time series representation of vehicle travel times does not account for interval variation, which results in the loss of important information about the traffic process. This is the main drawback when using the SARIMA model to solve the forecasting problem.

Predicting the development of the train run process using regression dependencies is carried out in a number of studies (see, for example, [25]). The general scheme for solving the problem of assessing arrival deviations using a regression model proposed in [12], uses such explanatory variables as the moving historical average of delays, the number of meetings, passages, overtaking, departure headway, etc. to predict a train run times. The study uses correlation analysis to identify explanatory variables that reflect the impact of delay sources and influence travel time. Several approaches are applicable for real-time vehicle traffic predicting. Among the most popular data-driven methods are Kalman filter, artificial neural network, time series analysis (TSA), and k-nearest neighbours (k-NN) [26]. The Kalman algorithm is recursive, using a two-stage process. In the forecasting stage, it generates current estimates of the variables that include uncertainties. In the second stage, when the next observation is received, these estimates are updated using a weighted average. The process of calculation performing does not require a large amount of data.

The problem of historical data on vehicle traffic structuring necessary for the prognosis generating, is considered in a number of works. Some approaches (see [27, 28], for example) classify delay profiles mainly by visual assessment. The K-means clustering method proposed in [29] allows automatic identification of patterns in delay behavior. Profile clustering serves to identify similar (identical) delay sequences in the entire set of historical records. K-means clustering is an iterative process based on identifying the average element (centroid) in each cluster. One of the most effective is the nearest neighbours approach (k-NN approach), which is a nonparametric method and is used in classification and regression analysis problems. The input variables are the k closest training examples in the feature space. The algorithm assigns greater weight to closer neighbours by measuring the distance between current and historical values [30]. The commonly used metric in searching for the best solution is the Euclidean distance. In one of the known studies [9], the modified k-NN method is used in the task of predicting the vehicle travel time at the stage of searching for a situation that is close to the current one. When comparing delay profiles, this work proposes to use a metric defined by the correlation coefficient. It is shown that correlation coefficients can take slow changes (trends) in current and historical profiles into account, and regression equations can correct for the random component. be vector representation of influencing factors set that is used in a data-driven methodology, can serve as a basis for a close historical situation determining. Multiple linear regressions (MLR) are a widely used statistical method in classification problems which uses several explanatory variables (regressors) to predict the outcome variable [31]. Note that data-driven predicting approaches require a large amount of high-quality historical statistics.

To overcome the complexity of the railway transportation process and the difficulty of capturing all the factors causing delays, an effective approach based on big data methodology is proposed in [32]. This approach was made possible by access to a large and comprehensive dataset of rail transport performance and the use of machine learning techniques. Subsequently, a large number of studies have emerged using machine learning techniques to predict vehicle traffic (see [26, 33] for example). In one of the early studies [4], train arrival times are predicted using support vector regression trained and cross-validated on a large-scale historical dataset. The paper focuses on the problem of predicting arrivals at freight railway stations using real-time data. N. Markovic et al. [32] investigate support vector regression models that reflect the relationship between train arrival delays and various characteristics of the railway system such as passenger train category, scheduled arrival time at the station, infrastructure influence, etc. The research mainly focuses on finding dependencies between delay and existing system properties, as well as on understanding the factors influencing train delays. In [34], train delays are predicted using random forest regression, flexible network regression, and an algorithm that estimates situations based on cosine similarity. It was found that machine learning methods reduce the errors in determining the parameters of the regression equation by two times. However, the reliability of forecasting using machine learning algorithms is quite low, since the estimate of the standard deviation of the RMSE modeling result from the true delay characteristic is an order of magnitude higher than the MAE. L. Oneto et al. [35] proposed shallow and deep extreme learning algorithms that take into account the types of working day (weekday or holiday), waiting time, and the running times of all other trains running on the same network section during the day to predict arrival delays. These models rely on characteristics other than historical train delay information (such as station congestion, weather, etc.) and may not be applicable in scenarios where these additional characteristics are not recorded.

The study [36] proposed the train delay prediction model that combines a fully connected neural network (FCNN) and two long short-term memory (LSTM) components to account for operational interactions. These interactions include, first of all, the mutual influences of trains in a dense vehicle flow. The work analyzes the influence of timetable-related factors and separately takes into account the actual sequence of stopping points along the train route and the weather factor. The proposed technique allows increasing the accuracy of forecasting but requires a lot of time for data processing and the use of powerful computing tools. The study [7] proposes a methodology for supporting dispatcher decision-making using hints about changes in the stopping and running times of the target train. The nature of the changes and their magnitude are determined using a predicting model based on ensemble learning (the deep forest, DF method). The paper also proposes a methodology for filling the data gap in the selection of control actions in the railway traffic re-planning process (SMOTE). In [37], a hybrid framework for solving the delay predicting problem, called the context-driven Bayesian network (CDBN), is presented. This framework consists of delay evolution detection model based on the K-means clustering and a train delay prediction model, i.e. a Bayesian network. The paper uses a clustering algorithm to identify the delay evolution patterns and classify the data into different categories based on delay jumps, i.e., the change in delay from one station to the next one.

An analysis of the previous studies results leads us to the conclusion that state-of-the-art approaches based on machine learning have great potential for solving delay predicting problem. However, they require large volumes of statistical data, which is not always accompanied by an increase in the reliability of predicting the situation development and determining rational dispatch decisions. It can be assumed that Markov and Bayesian models of the train traffic process using entire volume of current information will allow achieving the same higher forecast characteristics as those obtained by machine processing approaches. This will ensure a high level of results interpretability the schedule deviations prediction, which is very important in the operational work of dispatch personnel. One possible approach to solving this problem using Markov chain is proposed in this study.

6.3. The Markov Model Creation

We will create the Markov model for a sequence of commuter train delays on Russian Railways. First, let us introduce the necessary notations and concepts. Assume that the railway section of the railway consists of *N* stations. We will use the following notations:

ξ_j is a random variable equal to the deviation from the schedule upon arrival of a train at station j (at the station with number $j,\ j = 2,\ 3,\ \ldots,\ N$),

δ_j is a random variable equal to the deviation from the schedule upon departure of a train from station $j,\ \ j = 1,\ 2,\ \ldots\ ,\ N - 1$.

We will take deviations from the schedule into account starting with departure from station *1* and ending with arrival at station *N*. In other words, we are interested in the sequence of the following random variables

$$\delta_1, \underbrace{\xi_2, \delta_2, \xi_3, \ \ldots\ , \xi_{N-1}, \delta_{N-1}}_{(N-2)\text{ pairs}}, \xi_N. \tag{6.1}$$

Suppose that some interval (a, b) includes all possible values of all random variables (6.1). We divide $(a,\ b)$ into some number K of non-intersecting intervals

$$(a,\ b) = (a_0,\ a_1] \cup (a_1,\ a_2] \cup \ \ldots\ \cup (a_{K-2},\ a_{K-1}] \cup (a_{K-1},\ a_K), \tag{6.2}$$

where $a_0 = a,\ \ a_K = b$. We assign the number j to each interval $\left(a_{j-1},\ a_j\right)$.. Note that the number K and the partition intervals themselves in (6.2) will be chosen based on some specific empirical considerations. We define a set *S* consisting of interval numbers:

$$S = \{1,\ 2,\ \ldots\ ,\ K\}.$$

Given the sequence (6.1), we define the following sequence of 2N - 2 discrete random variables

$$Y_1, \underbrace{X_2, Y_2, X_3, \ \ldots\ ,\ X_{N-1}, Y_{N-1}}_{(N-2)\text{ pairs}}, X_N, \tag{6.3}$$

taking values in the set S according to the rule:

$$\begin{aligned} X_n &= s \in S \;\Leftrightarrow \xi_n \in (a_{s-1}, a_s],\; n = 2, \;\dots,\; N; \\ Y_n &= s \in S \;\Leftrightarrow \delta_n \in (a_{s-1}, a_s],\; n = 1, \;\dots,\; N-1. \end{aligned} \tag{6.4}$$

Note that in (6.4), for the sake of simplicity of the formulation, we have made an inaccuracy: we should have written $(a_{K-1},\, a_K)$ instead of $(a_{K-1},\, a_K]$ in the case $s = K$.

The model assumption is that the sequence (6.3) is a Markov chain, i.e. these random variables satisfy the so-called Markov property: the conditional distribution of each random variable in (6.3) depends on the one immediately preceding variable and does not depend on other preceding random variables, i.e.

$$\mathbf{P}(Y_n = j | X_n = i, Y_{n-1} = y_{n-1}, X_{n-1} = x_{n-1}, \,\dots\, , X_2 = x_2, Y_1 = y_1)$$

$$= \mathbf{P}(Y_n = j | X_n = i) =: p_{ij}(n), \tag{6.5}$$

$2 \le n \le N-1$, and

$$\mathbf{P}(X_{n+1} = j \mid Y_n = i, X_n = x_n, Y_{n-1} = y_{n-1}\,,$$

$$X_{n-1} = x_{n-1}, \,\dots\, , X_2 = x_2,\; Y_1 = y_1) \;=\; \mathbf{P}(X_{n+1} = j | \;Y_n \;=\; i) \;=: q_{ij}(n), \tag{6.6}$$

where $1 \le n \le N-1,\;\; j,\; i,\; x_n,\; y_{n-1},\; x_{n-1},\; y_{n-2},$ $x_{n-2}, \,\dots\, ,\; x_2,\; y_1$ are arbitrary elements from set S, which are called by *states*.

The conditional probabilities (6.5) and (6.6) generate the so-called transition matrices:

$$P(n) = \begin{pmatrix} p_{11}(n) & p_{12}(n) & \cdots & p_{1K}(n) \\ p_{21}(n) & p_{22}(n) & \dots & p_{2K}(n) \\ \vdots & \vdots & \vdots & \vdots \\ p_{K1}(n) & p_{K2}(n) & \cdots & p_{KK}(n) \end{pmatrix} \text{ and}$$

$$Q(n) = \begin{pmatrix} q_{11}(n) & q_{12}(n) & \cdots & q_{1K}(n) \\ q_{21}(n) & q_{22}(n) & \dots & q_{2K}(n) \\ \vdots & \vdots & \vdots & \vdots \\ q_{K1}(n) & q_{K2}(n) & \cdots & q_{KK}(n) \end{pmatrix}.$$

The distribution of the random variable X_n will be denoted in the form of the row vector

$$\vec{p}(n) = \big(p_1(n), p_2(n), \dots, p_K(n)\big), \tag{6.7}$$

where $p_s(n) = \mathbf{P}(X_n = s),\ s \in S$. Similarly, the distribution of the random variable Y_n will be denoted by

$$\vec{q}(n) = \big(q_1(n), q(n), \dots, q_K(n)\big), \tag{6.8}$$

where $q_s(n) = \mathbf{P}(Y_n = s),\ s \in S$. It is allowed to write in (6.7) and (6.8) without separating the vector components with commas. In this case, we will understand the row vectors as single-row matrices. Markov chains are characterized by transition matrices $P(n)$ and $Q(n)$ in the following sense:

1) The distribution of the random variable Y_n is determined by the distribution of the previous random variable X_n according to

$$\vec{q}(n) = \vec{p}(n) \cdot P(n); \tag{6.9}$$

2) The distribution of the random variable X_{n+1}, in turn, is determined by the distribution of the previous random variable Y_n according to

$$\vec{p}(n+1) = \vec{q}(n) \cdot Q(n), \tag{6.10}$$

where ‘' · " denotes matrix multiplication operation.

Let us explain the equality (6.9), taking for simplicity $K = 3$. By the definition of the matrices product, we have

$$\begin{aligned}
\vec{p}(n) \cdot P(n) &= \big(p_1(n)\ \ p_2(n)\ \ p_3(n)\big) \\
&= \left(\sum_{i=1}^{3} p_i(n)p_{i1}(n) \sum_{i=1}^{3} p_i(n)p_{i2}(n) \sum_{i=1}^{3} p_i(n)p_{i3}(n)\right) \\
&= \big(q_1(n)\ \ q_2(n)\ \ q_3(n)\big).
\end{aligned} \tag{6.11}$$

Note that the last equality in (6.11) is a consequence of a fundamental fact of probability theory called the law of total probability. For example,

$$\begin{aligned}
\sum_{i=1}^{3} p_i\,(n)p_{i1}(n) &= \sum_{i=1}^{3} \mathbf{P}(X_n = i)\mathbf{P}(Y_n = 1|X_n = i) \\
&\underset{\substack{\text{by } law\ of\ total \\ probability}}{=} \mathbf{P}(Y_n = 1) \equiv q_1(n).
\end{aligned} \tag{6.12}$$

Similarly, we are convinced of the validity of (6.10):

$$\vec{q}(n)\cdot Q(n) = \big(q_1(n)\ \ q_2(n)\ \ q_3(n)\big)\cdot\begin{pmatrix} q_{11}(n) & q_{12}(n) & q_{13}(n) \\ q_{21}(n) & q_{22}(n) & q_{23}(n) \\ q_{31}(n) & q_{32}(n) & q_{33}(n)\end{pmatrix}$$

$$= \left(\sum_{i=1}^{3} q_i(n)q_{i1}(n) \sum_{i=1}^{3} q_i(n)q_{i2}(n) \sum_{i=1}^{3} q_i(n)q_{i3}(n)\right)$$

$$= \big(p_1(n+1)\ \ p_2(n+1)\ \ p_3(n+1)\big). \qquad (6.13)$$

For the sake of completeness of argumentation, we note that the last equality in (6.13) is obtained in the same way as (6.12), namely, for each fixed $j = 1, 2, 3$

$$\sum_{i=1}^{3} q_i(n)q_{ij}(n) = \sum_{i=1}^{j} \mathbf{P}(Y_n = i)\mathbf{P}(X_{n+1} = j|Y_n = i)$$

$$= \ \mathbf{P}(X_{n+1} = j) \equiv p_j(n+1). \qquad (6.14)$$

Note that in the Markov chain (6.3) the first transition occurs from Y_1 to X_2. This transition occurs according to the formula (6.10) using the matrix $Q(1)$. The next transition, from X_2 to Y_2, occurs according to the formula (6.9) via the matrix $P(2)$ and so on.

Thus, the Markov chain (6.3) is completely determined by the sequence of the following transition matrices:

$$\underbrace{Q(1), P(2), \ldots, Q(N-2), P(N-1), Q(N-1)}_{2N-3}. \qquad (6.15)$$

Along with the matrices (6.15), which can be called single-step transition matrices, we will need the so-called multi-step transition matrices. Let us denote

$$p_{ij}(n,m) = \mathbf{P}(Y_m = j|X_n = i),\ \ 2 \le n \le m \le N-1, \ (6.16)$$

$$q_{ij}(n,m) = \mathbf{P}(X_m = j|Y_n = i),\ \ 1 \le n < m \le N-1. \ (6.17)$$

Let us denote by $P(n,m)$ the matrix (of size $K \times K$) consisting of probabilities (6.16), and by $Q(n,m)$ of probabilities (6.17). Then $P(n,m)$ is a transition matrix from X_n to Y_m, and $Q(n,m)$ is a transition matrix from Y_n to X_m, since, as it is easy to verify, similarly to equalities (6.9) and (6.10), the following equalities holds,

$$\vec{q}(m) = \vec{p}(n) \cdot P(n,m), \tag{6.18}$$

$$\vec{p}(m) = \vec{q}(n) \cdot Q(n,m). \tag{6.19}$$

Note that $P(n,n) = P(n)$ and $Q(n,n+1) = Q(n)$. It is also known from the Markov chains theory that $P(n,m)$ and $Q(n,m)$ coincide with the products of all intermediate one-step transition matrices, i.e.

$$P(n,m) = \underbrace{P(n) \cdot Q(n) \cdot P(n+1) \cdots Q(m-1) \cdot P(m),}_{2(m-n)+1} \tag{6.20}$$

$$Q(n,m) = \underbrace{Q(n) \cdot P(n+1) \cdots Q(m-2) \cdot P(m-1) \cdot Q(m-1)}_{2(m-n)-1} \tag{6.21}$$

Summary of Section 6.3. Formulas (6.9), (6.10), (6.18) and (6.19) define the behavior of the delay distributions (more precisely, the delay states) of trains arriving at stations and departing from them. In addition, equalities (6.20) and (6.21) allow us to find the matrices of multi-step transitions through intermediate matrices of single-step transitions.

6.4. On the Relationship between Transition Matrices and Corresponding Correlation Coefficients

Consider, for example, the matrix *P(n;m),* one of the two types of transition matrices described in Section 6.3. Recall that if A and B are some random events, then it follows from the definition of conditional probability that $\mathbf{P}(AB) = \mathbf{P}(B)\,\mathbf{P}(A|B)$. Consequently,

$$\begin{aligned}\widetilde{p_{\imath\jmath}}(n,m) &:= \mathbf{P}(X_n = i, Y_m = j) = \mathbf{P}(X_n = i)\,\mathbf{P}(X_n = j|\, Y_m = i) \\ &= p_i(n)\;\; p_{ij}(n,m),\end{aligned}$$

i.e., given a known distribution of X_n, the joint distribution of X_n and Y_m is determined by the matrix $P(n,m)$. Knowing the joint distribution of X_n and Y_m allows us to find their covariance:

$$\mathrm{cov}(X_n, Y_m) \equiv \mathbf{E}[\,(X_n - \boldsymbol{E}X_n)(Y_m - \mathbf{E}Y_m)\,]$$

$$= \textstyle\sum_{i=1}^{K} \sum_{j=1}^{K} [(i - \mathbf{E}X_n)(j - \mathbf{E}Y_m)]\, p_i(n)\;\; p_{ij}(n,m), \tag{6.22}$$

where $\mathbf{E}$ means the mathematical expectation. Having calculated the covariance, we obtain the correlation coefficient using the formula

$$r(X_n, Y_m) = \frac{\operatorname{cov}(X_n, Y_m)}{\sqrt{\mathbf{D}X_n \mathbf{D}Y_m}}, \qquad (6.23)$$

where $\mathbf{D}$ is the variance.

Thus, the transition matrix from X_n to Y_m according to the formulas (6.22) and (6.23) either completely determines the numerical value of the correlation coefficient $r(X_n, Y_m)$, or corresponds to the case when $r(X_n, Y_m)$ does not exist.

Let us consider some examples. In them, we will write X instead of X_n, and Y instead of Y_m. We will consider the case $K = 2$ which simplifies the notation but does not limit the generality of the conclusions. We will denote the transition matrix from X to Y by P, i.e. $P = \begin{pmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{pmatrix}$, where

$$p_{ij} = \mathbf{P}(Y = j | X = i), \qquad i = 1, 2, \qquad j = 1, 2.$$

Example 1. Let us give an example of a transition matrix in the case where the correlation coefficient does not exist. We will show that if the transition matrix from X to Y contains a column, all elements of which are equal to 1, then the correlation coefficient $r = r(X, Y)$ does not exist.

Let $p_{12} = p_{22} = 1$ for definiteness. We denote the distributions of X and Y by (p_1, p_2) and (q_1, q_2) respectively. Using (6.18) we obtain the following equality,

$$(q_1\ q_2) = (p_1\ p_2) \cdot P = (p_1\ p_2) \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \underset{\text{since } p_1 + p_2 = 1}{=} (0\ 1),$$

i.e. $q_2 = 1$. This in turn means that Y is in state 2 with probability 1. Simply put $Y = 2$ is a constant. Given that $\mathbf{E}Y = 2$ and using (6.22) we find that

$$\mathrm{cov}(X,Y) = \sum_{i=1}^{2}\sum_{i=1}^{2}[\,(i-\mathbf{E}X)(j-2)\,]\,p_i\,p_{ij}$$
$$= \sum_{i=1}^{2}(i-\mathbf{E}X)\,p_i\left((1-2)\underbrace{p_{i1}}_{0} + 0\cdot p_{12}\right) = 0.$$

Moreover, it is obvious that $\mathbf{D}Y = 0$. Using now (6.23), we get $r = \frac{0}{0}$, i.e. uncertainty.

Example 2. Let us show that if matrix P is identity, i.e. $P = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$, and $\mathbf{D}X \neq 0$, then $r = 1$. Indeed, again due to (6.18),

$$(q_1\ q_2) = (p_1\ p_2)\cdot\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} = (p_1\ p_2)$$

i.e. Y has the same distribution as X. Hence,

$$\mathrm{cov}(X,Y) = \mathbf{E}(X-\mathbf{E}X)^2 = \mathbf{D}X,$$

which, in view of (6.23), implies $r = 1$.

Example 3. Let us show that if matrix P has ones along the second diagonal, i.e. $P = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$, and $\mathbf{D}X \neq 0$, then $r = -1$. Using (6.18) we get the equality

$$(q_1\ q_2) = (p_1\ p_2)\cdot\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} = (p_2\ p_1)$$

i.e.

$$q_j = p_{3-j}, \qquad j = 1, 2.$$

It follows that $Y = 3 - X$. Then $\mathbf{E}Y = 3 - \mathbf{E}X$, $\mathbf{D}Y = \mathbf{D}X$ and

$$\mathrm{cov}(X,Y) = \mathbf{E}[\,(X-\mathbf{E}X)(Y-\mathbf{E}Y)\,] = \mathbf{E}[\,(X-\mathbf{E}X)(-X+\mathbf{E}X)\,] = -\mathbf{D}X.$$

Using (6.23), we find that $r = \frac{-\mathbf{D}X}{\mathbf{D}X} = -1$.

Example 4. Suppose that the transition matrix P consists of identical rows and $\mathbf{D}X \neq 0$. We will show that $r = 0$ in this case. Let

$P = \begin{pmatrix} a_1 & a_2 \\ a_1 & a_2 \end{pmatrix}$, $a_1 > 0,\ a_2 > 0$ and $a_1 + a_2 = 1$. In view of (6.18) and the equality $p_1 + p_2 = 1$,

$$(q_1\ q_2) = (p_1\ \ p_2) \cdot \begin{pmatrix} a_1 & a_2 \\ a_1 & a_2 \end{pmatrix} = (a_1\ a_2). \qquad (6.24)$$

Taking into account that in the case under consideration the equality $p_{ij} = a_j$ holds (regardless of i), we conclude that $\widetilde{p_{\iota J}} = p_i a_j$. Using this we obtain

$$\begin{aligned} \operatorname{cov}(X,Y) &= \sum_{i=1}^{2}\sum_{i=1}^{2} [\,(i - \mathbf{E}X)(j - \mathbf{E}X)]\, p_i\, a_j \\ &= \left(\sum_{i=1}^{2} (i - \mathbf{E}X)p_i\right) \sum_{j=1}^{2} (j - \mathbf{E}Y) \quad \underbrace{a_j}_{=q_j \text{ из } (24)} \\ &= \mathbf{E}(X - \mathbf{E}X) \cdot \mathbf{E}(Y - \mathbf{E}Y) = 0. \end{aligned}$$

It follows from the conditions on a_j that $\mathbf{D}Y \neq 0$. Taking this into account, as a result of (6.23) we obtain $r = 0$.

We conclude Section 6.3 with the following statement.

Proposition 1. *Consider in the sequence (6.3) any three random variables, for example,* X_n, Y_n, X_{n+1}. *Let the number* c *belong to the interval* $[—1, 1]$. *If* $r(X_n, Y_n) = c$, $\quad r(Y_n, X_{n+1}) = \pm 1$, *then*

$$r(X_n,\ X_{n+1}) = \pm c. \qquad (6.25)$$

The equality (6.25) is also true when $r(X_n,\ Y_n) = \pm 1$, *and* $r(Y_n,\ X_{n+1}) = c$**.**

Proof. It is known that if the correlation coefficient between random variables is ± 1, then the random variables are linearly dependent. Let us limit ourselves to the case $r = (Y_n,\ X_{n+1}) = 1$. Linear dependence means the following: there exist two numbers a and b such that

$$X_{n+1} = aY_n + b,$$

where $a = \sqrt{\mathbf{D}X_{n+1}/\mathbf{D}Y_n}$, $b = \mathbf{E}X_{n+1} - a\mathbf{E}Y_n$ (see, for example, [38, p. 307, 308]). Thus,

$$\begin{aligned} r(X_n, X_{n+1}) &= \frac{\mathbf{E}[\,(X_n - \mathbf{E}X_n)(X_{n+1} - \mathbf{E}X_{n+1})\,]}{\sqrt{\mathbf{D}X_n \mathbf{D}X_{n+1}}} \\ &= \frac{a\mathbf{E}[\,(X_n - \mathbf{E}X_n)(Y_n - \mathbf{E}Y_n)\,]}{\sqrt{\mathbf{D}X_n \mathbf{D}X_{n+1}}} \\ &= \frac{\mathbf{E}[\,(X_n - \mathbf{E}X_n)(Y_n - \mathbf{E}Y_n)\,]}{\sqrt{\mathbf{D}X_n \mathbf{D}Y_n}} = r(X_n,\, Y_n) = c. \end{aligned}$$

The equality (6.25) is proved for $r(Y_n,\ X_{n+1}) = 1.$

On the possible application of Proposition 1. Let us assume that

$$r(Y_n,\ X_{n+1}) = 0.9.$$

Then, according to Proposition 1,

$$r(X_n,\ X_{n+1}) \approx r(X_n,\ Y_n).$$

This means that if, for example, $r(X_n,\ Y_n)$ is close to zero, then $r(X_n,\ X_{n+1})$ will also be close to zero.

6.5. An Example of Using the Proposed Markov Model to Create a Forecast

Let us consider a section of suburban railway in the Moscow region, consisting of seven stations and the sections between them.

Divide the time of day from 7 to 19 hours into three time intervals: morning rush hours from 7 to 10 a.m. (morning peak), daylight hours from 10 a.m. to 4 p.m. (off-peak), and evening rush hours from 4 to 7 p.m. (evening peak). The proposed division is due to the fact that each such time interval corresponds to its own intensity of train traffic and passenger flow.

We know the statistics on schedule deviations for 24/04/01 received from the train traffic monitoring system, time period – morning peak, in the form of the sequence

$$\delta_1^*, \xi_2^*, \delta_2^*, \xi_3^*, \delta_3^*, \dots, \xi_6^*, \delta_6^*, \xi_7^* \qquad (6.26)$$

which is a sample analogue of sequence (1) with $N = 7$. To define a sample analogue of the Markov chain (6.3), we choose $K = 3$ and the following partition intervals,

$$(a_0, a_1] = (-\infty, 0.5],\ (a_1, a_2] = (0.5, 1.5],\ (a_2, a_3) = (1.5, \infty) \quad (6.27)$$

Then the set of states S consists of three elements, the interval numbers from (6.27):

$$S = \{1, 2, 3\}.$$

Define the sequence

$$Y_1^*,\ X_2^*,\ Y_2^*,\ X_3^*,\ Y_3^*,\ \dots\ ,\ X_6^*,\ Y_6^*,\ X_7^* \qquad (6.28)$$

which is a sample analogue of sequence (6.3) with $N = 7$, according to the following rule,

$$X_n^* = s \in S \Leftrightarrow \xi_n^* \in (a_{s-1}, a_s], \qquad n = 2,\ \dots\ , 7,$$

$$Y_n^* = s \in S \Leftrightarrow \delta_n^* \in (a_{s-1}, a_s], \qquad n = 1,\ \dots\ , 6. \quad (6.29)$$

Note that in (6.29) the following inaccuracy was intentionally allowed for the sake of simplifying the formulation: in the case of $s = 3$ instead of $(a_2, a_3]$ one should write (a_2, a_3).

The prediction of arrival at the terminal station is represented by the probability distribution of random variable X_7^*, the value of the deviation from the schedule upon arrival. This prediction, according to the proposed method, is determined based on statistical data obtained earlier when performing the same factors namely, day of the week and period of the day. In the example under consideration, the prediction of the schedule deviations during the morning rush hours of Monday, April 8 is determined taking the data obtained in a similar period a week ago into account.

6.5.1. Statistical Data Processing for 24/04/01 (Morning Rush Hours)

6.5.1.1. The Initial Data

In Table 6.1 the numerical values of quantities (6.26) obtained for 24/04/01 are recorded.

Table 6.1. Sample of size 25 from the distribution of the 12-dimensional random vector $(\delta_1, \xi_2, \delta_2, \xi_3, \delta_3, \ldots, \xi_7)$.

№	$\boldsymbol{\delta_1^*}$	$\boldsymbol{\xi_2^*}$	$\boldsymbol{\delta_2^*}$	$\boldsymbol{\xi_3^*}$	$\boldsymbol{\delta_3^*}$	$\boldsymbol{\xi_4^*}$	$\boldsymbol{\delta_4^*}$	$\boldsymbol{\xi_5^*}$	$\boldsymbol{\delta_5^*}$	$\boldsymbol{\xi_6^*}$	$\boldsymbol{\delta_6^*}$	$\boldsymbol{\xi_7^*}$
1	0	0.5	1	—0.5	1.5	0	1	—0.5	1	—0.5	—0.5	—0.5
2	0	0.5	1	1.5	2.5	1	2	—0.5	1	–1.5	1.5	0.5
3	1	1.5	2	0.5	1.5	1	1	—0.5	1	0.5	0.5	0.5
4	0	0.5	1	—0.5	0.5	–1	0	—0.5	0	—0.5	1.5	0.5
5	0	—0.5	1	0.5	1.5	0	2	—0.5	0	–1.5	0.5	0.5
6	0	0.5	1	—0.5	0.5	1	0	—0.5	1	—0.5	0.5	—0.5
7	0	0.5	0	—0.5	0.5	–1	0	—0.5	1	0.5	0.5	—0.5
8	0	0.5	0	—0.5	0.5	–2	0	—0.5	0	–1.5	0.5	—0.5
9	0	0.5	1	—0.5	0.5	–1	0	–1.5	0	–1.5	—0.5	—0.5
10	0	0.5	0	–1.5	0.5	–1	0	–1.5	0	–1.5	0.5	—0.5
11	0	—0.5	1	0.5	2.5	1	3	—0.5	1	—0.5	0.5	—0.5
12	0	0.5	1	0.5	0.5	0	0	0.5	1	—0.5	0.5	—0.5
13	0	0.5	1	—0.5	0.5	–1	0	—0.5	0	–1.5	0.5	—0.5
14	0	0.5	0	—0.5	0.5	0	0	–1.5	0	–1.5	0.5	—0.5
15	6	4.5	6	5.5	6.5	5	7	5.5	6	4.5	4.5	2.5
16	5	3.5	3	2.5	2.5	4	6	2.5	3	1.5	2.5	0.5
17	3	2.5	4	2.5	3.5	2	3	1.5	3	0.5	1.5	—0.5
18	0	0.5	0	—0.5	0.5	–1	1	—0.5	0	–1.5	0.5	–1.5
19	0	1.5	2	1.5	1.5	0	2	0.5	1	0.5	0.5	–1.5
20	1	0.5	1	—0.5	0.5	–1	0	–1.5	0	–1.5	0.5	—0.5
21	0	0.5	0	—0.5	0.5	–1	1	—0.5	0	–1.5	0.5	–1.5
22	0	0.5	0	—0.5	0.5	–1	0	—0.5	0	–1.5	0.5	–1.5
23	0	0.5	0	—0.5	0.5	–1	0	—0.5	0	–1.5	0.5	–1.5
24	0	0.5	1	0.5	0.5	–1	1	—0.5	1	—0.5	0.5	—0.5
25	1	1.5	2	0.5	1.5	0	1	0.5	2	0.5	0.5	0.5

The following Table 6.2 contains the values of (6.28) corresponding to the values of (6.26) from Table 6.1 according to the rule (6.29).

Table 6.2. Sample of size 25 from the distribution of the 12-dimensional random vector $(Y_1, X_2, Y_2, X_3, Y_3, \dots, X_7)$.

№	$\boldsymbol{Y_1^*}$	$\boldsymbol{X_2^*}$	$\boldsymbol{Y_2^*}$	$\boldsymbol{X_3^*}$	$\boldsymbol{Y_3^*}$	$\boldsymbol{X_4^*}$	$\boldsymbol{Y_4^*}$	$\boldsymbol{X_5^*}$	$\boldsymbol{Y_5^*}$	$\boldsymbol{X_6^*}$	$\boldsymbol{Y_6^*}$	$\boldsymbol{X_7^*}$
1	1	1	2	1	2	1	2	1	2	1	1	1
2	1	1	2	2	3	2	3	1	1	1	2	1
3	2	2	3	1	2	2	2	1	2	1	1	1
4	1	1	2	1	1	1	1	1	1	1	2	1
5	1	1	2	1	2	1	3	1	1	1	1	1
6	1	1	2	1	1	2	1	1	2	1	1	1
7	1	1	1	1	1	1	1	1	2	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	2	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1
11	1	1	2	1	3	2	3	1	2	1	1	1
12	1	1	2	1	1	1	1	1	2	1	1	1
13	1	1	2	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1
15	3	3	3	3	3	3	3	3	3	3	3	3
16	3	3	3	3	3	3	3	3	3	2	3	1
17	3	3	3	3	3	3	3	2	3	1	2	1
18	1	1	1	1	1	1	2	1	1	1	1	1
19	1	2	3	2	2	1	3	1	2	1	1	1
20	2	1	2	1	1	1	1	1	1	1	1	1
21	1	1	1	1	1	1	2	1	1	1	1	1
22	1	1	1	1	1	1	1	1	1	1	1	1
23	1	1	1	1	1	1	1	1	1	1	1	1
24	1	1	2	1	1	1	2	1	2	1	1	1
25	2	2	3	1	2	1	2	1	3	1	1	1

Note that pairs of the form $(X_n^*,\ Y_n^*)$ define the states of deviations of trains from the schedule of arrivals at station n and departures from it, and pairs $(Y_n^*,\ X_{n+1}^*)$ define the states of deviations of trains from the schedule on the section between stations n and $n+1$. We must find transition matrices of two types: for stations and for sections, in accordance with the remark made.

6.5.1.2. Transition Matrices

Define transition probabilities:

$$p_{ij}(n) = \mathbf{P}(Y_n^* = j | X_n^* = i) \quad \text{(for stations)}, \qquad n = 2, \dots, 6;$$

$$q_{ij}(n) = \mathbf{P}(X^*_{n+1} = j | Y^*_n = i) \quad \text{(for sections)}, \qquad n = 1, \dots, 6.$$

They form transition matrices:

$$P(n) = \begin{pmatrix} p_{11}(n) & p_{12}(n) & p_{13}(n) \\ p_{21}(n) & p_{22}(n) & p_{23}(n) \\ p_{31}(n) & p_{32}(n) & p_{33}(n) \end{pmatrix} \text{ and } Q(n)$$
$$= \begin{pmatrix} q_{11}(n) & q_{12}(n) & q_{13}(n) \\ q_{21}(n) & q_{22}(n) & q_{23}(n) \\ q_{31}(n) & q_{32}(n) & q_{33}(n) \end{pmatrix}.$$

Note that it would be more correct to use the notations $p^*_{ij}(n)$ and $q^*_{ij}(n)$ instead of $p_{ij}(n)$ and $q_{ij}(n)$, respectively. However, we abandoned this idea to avoid cumbersome notation.

The transition matrices are found using the algorithm described in [39]. The essence of the algorithm is the sequential creation of tables of type 2, 3, and 4 from [39]. However, here, instead of finding tables of type 2 from [39] for each pair of random variables from (6.28), we will use the corresponding pair of columns from the previous Table 6.2.

Now we explain how to find the elements of the matrix $Q(1)$ (the transition matrix for the first run). Let us consider the second and third columns of Table 6.2. The values of Y^*_1 and X^*_2 will be denoted by y_{1k} and x_{2k}, respectively, where $k = 1, 2, \dots, 25$. Then each such k corresponds to a pair (y_{1k}, x_{2k}) (a two-dimensional vector) which is a pair of natural numbers (i, j), where $1 \leq i, j \leq 3$.

We fix a pair (i, j) $(i,\ j = 1, 2, 3)$. Denote by n_{ij} the number of those two-dimensional vectors with coordinates from the second and third columns of Table 6.2 that coincide with (i, j). We also denote $n_i = \sum_{j=1}^{3} n_{ij}$, the number of all the specified vectors with a fixed first coordinate i. Using the introduced notations, we obtain Table 6.3 from Table 6.2.

Further, just as in [20] (see also [39, p. 96]), the frequencies $\frac{n_{ij}}{n_i}$ in the considered problem of finding the elements of the matrix $Q(1)$ will be understood as conditional probabilities $q_{ij}(1)$. To justify this approach, we present the following elementary equalities:

$$\frac{n_{ij}}{n_i} \equiv \frac{n_{ij}/\sum n_i}{n_i/\sum n_i} = \frac{\mathbf{P}(Y_1^* = i, X_2^* = j)}{\mathbf{P}(Y_1^* = i)} = \mathbf{P}(X_2^* = j|Y_1^* = i) \equiv q_{ij}(1).$$

Table 6.3. The quantities n_{ij} corresponding to the second and third columns of Table 6.2.

$\boldsymbol{X_2^* = j}$ $\boldsymbol{Y_1^* = i}$	**1**	**2**	**3**	$\boldsymbol{n_i = \sum_{j=1}^{3} n_{ij}}$
1	18	1	0	19
2	1	2	0	3
3	0	0	3	3

From Table 6.3 we obtain Table 6.4, consisting of conditional probabilities $q_{ij}(1)$, elements of the transition matrix $Q(1)$.

Table 6.4. Conditional probabilities $q_{ij}(1)$.

$\boldsymbol{j}$ $\boldsymbol{i}$	**1**	**2**	**3**
1	$q_{11}(1) = \frac{18}{19}$	$q_{12}(1) = \frac{1}{19}$	$q_{13}(1) = 0$
2	$q_{21}(1) = \frac{1}{3}$	$q_{22}(1) = \frac{2}{3}$	$q_{23}(1) = 0$
3	$q_{31}(1) = 0$	$q_{32}(1) = 0$	$q_{33}(1) = 1$

Thus, we found the transition matrix $Q(1)$ (transition from Y_1^* to X_2^*):

$$Q(1) = \begin{pmatrix} \frac{18}{19} & \frac{1}{19} & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{6.30}$$

The remaining one-step transition matrices corresponding to Table 6.2 are found similarly. Omitting the calculations, we write them out:

$$P(2)=\begin{pmatrix}\frac{8}{18} & \frac{1}{19} & 0\\ 0 & 0 & 1\\ 0 & 0 & 1\end{pmatrix},\ Q(2)=\begin{pmatrix}1 & 0 & 0\\ \frac{10}{11} & \frac{1}{11} & 0\\ \frac{1}{3} & \frac{1}{6} & \frac{1}{2}\end{pmatrix},\ P(3)=\begin{pmatrix}\frac{3}{4} & \frac{1}{5} & \frac{1}{20}\\ 0 & \frac{1}{2} & \frac{1}{2}\\ 0 & 0 & 1\end{pmatrix},$$

$$\begin{gathered}Q(3)=\begin{pmatrix}\frac{14}{15} & \frac{1}{15} & 0\\ \frac{4}{5} & \frac{1}{5} & 0\\ 0 & \frac{2}{5} & \frac{3}{5}\end{pmatrix},P(4)=\begin{pmatrix}\frac{11}{18} & \frac{5}{18} & \frac{1}{9}\\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2}\\ 0 & 0 & 1\end{pmatrix},Q(4)=\begin{pmatrix}1 & 0 & 0\\ 1 & 0 & 0\\ \frac{4}{7} & \frac{1}{7} & \frac{2}{7}\end{pmatrix},\\ P(5)=\begin{pmatrix}\frac{6}{11} & \frac{9}{22} & \frac{1}{22}\\ 0 & 0 & 1\\ 0 & 0 & 1\end{pmatrix},Q(5)=\begin{pmatrix}1 & 0 & 0\\ 1 & 0 & 0\\ \frac{1}{2} & \frac{1}{4} & \frac{1}{4}\end{pmatrix},P(6)=\begin{pmatrix}\frac{20}{23} & \frac{3}{23} & 0\\ 0 & 0 & 1\\ 0 & 0 & 1\end{pmatrix},\end{gathered}\tag{31}$$

$$Q(6)=\begin{pmatrix}1 & 0 & 0\\ 1 & 0 & 0\\ \frac{1}{2} & 0 & \frac{1}{2}\end{pmatrix}.$$

6.5.1.3. Correlation Coefficients

Let us calculate the correlation coefficients between random variables (28).

Find $r(Y_1^*, X_2^*)$. Using the second and third columns of Table 6.2, by means of elementary but rather cumbersome calculations, and taking into account the notation from Subsection 6.5.1.2, we find:

$$\mathbf{E}Y_1^* = \frac{1}{25}\sum_{k=1}^{25} y_{1k} = 1.36, \qquad \mathbf{E}X_2^* = \frac{1}{25}\sum_{k=1}^{25} x_{1k} = 1.36,$$

$$s_{Y_1}^2 = \frac{1}{25}\sum_{k=1}^{25}(y_{1k} - 1.36)^2 = 0.4704, \qquad s_{Y_1} = 0.686,$$

$$s_{X_2}^2 = \frac{1}{25}\sum_{k=1}^{25}(x_{1k} - 1.36)^2 = 0.4704, \qquad s_{X_1} = 0.686,$$

$$\mathrm{cov}(Y_1^*, X_2^*) = \frac{1}{25}\sum_{k=1}^{25}(y_{1k} - 1.36)(x_{2k} - 1.36) = 0.4304,$$

$$r(Y_1^*, X_2^*) = \frac{\operatorname{cov}(Y_1^*, X_2^*)}{s_{y_1} s_{x_2}} = 0.915.$$

The correlation coefficients for the remaining pairs of the random variables from (6.28) are found similarly. Without giving the calculations, we write them out:

$$r(X_2^*, Y_2^*) = 0.762, \qquad r(Y_2^*, X_3^*) = 0.607, \qquad r(X_3^*, Y_3^*) = 0.754,$$

$$r(Y_3^*, X_4^*) = 0.794, \qquad r(X_4^*, Y_4^*) = 0.612, \qquad r(Y_4^*, X_5^*) = 0.5,$$

$$r(X_5^*, Y_5^*) = 0.636, \qquad r(Y_5^*, X_6^*) = 0.43, \qquad r(X_6^*, Y_6^*) = 0.797,$$

$$r(Y_6^*, X_7^*) = 0.584.$$

Calculations of the correlation coefficients between initial random variables (6.26) with the values indicated in Table 6.1 lead to the following results:

$$r(\delta_1^*, \xi_2^*) = 0.936,$$

$$r(\xi_2^*, \delta_2^*) = 0.87, \qquad r(\delta_2^*, \xi_3^*) = 0.922, \qquad r(\xi_3^*, \delta_3^*) = 0.93,$$

$$r(\delta_3^*, \xi_4^*) = 0.875, \qquad r(\xi_4^*, \delta_4^*) = 0.9, \qquad r(\delta_4^*, \xi_5^*) = 0.864,$$

$$r(\xi_5^*, \delta_5^*) = 0.938, \qquad r(\delta_5^*, \xi_6^*) = 0.915, \qquad r(\xi_6^*, \delta_6^*) = 0.768,$$

$$r(\delta_6^*, \xi_7^*) = 0.705.$$

Note that almost all the correlation coefficients of random variables from (6.26) turned out to be closer to 1 than the corresponding coefficients of random variables from (6.28) (except $r(\xi_6^*, \delta_6^*)$), i.e. the correlation dependence between the latter is, as a rule, weaker than that of the original random variables.

6.5.2. Statistical Data Processing for 24/04/08 (Morning Rush Hours)

Table 6.5 contains the numerical values of (6.26) obtained for 04/08/2024. There we also placed the values of (6.28) corresponding to (6.26) according to rule (6.29).

Table 6.5. Two samples (each of size 15) from the distributions of 12-dimensional random vectors $(\delta_1, \xi_2, \delta_2, \xi_3, \delta_3, \dots, \xi_7)$ and $(Y_1, X_2, Y_2, X_3, Y_3, \dots, X_7)$.

№	$\boldsymbol{\delta_1^*}$	$\boldsymbol{\xi_2^*}$	$\boldsymbol{\delta_2^*}$	$\boldsymbol{\xi_3^*}$	$\boldsymbol{\delta_3^*}$	$\boldsymbol{\xi_4^*}$	$\boldsymbol{\delta_5^*}$	$\boldsymbol{\xi_5^*}$	$\boldsymbol{\delta_5^*}$	$\boldsymbol{\xi_6^*}$	$\boldsymbol{\delta_6^*}$	$\boldsymbol{\xi_7^*}$
	$\boldsymbol{Y_1^*}$	$\boldsymbol{X_2^*}$	$\boldsymbol{Y_2^*}$	$\boldsymbol{X_3^*}$	$\boldsymbol{Y_3^*}$	$\boldsymbol{X_4^*}$	$\boldsymbol{Y_4^*}$	$\boldsymbol{X_5^*}$	$\boldsymbol{Y_5^*}$	$\boldsymbol{X_6^*}$	$\boldsymbol{Y_6^*}$	$\boldsymbol{X_7^*}$
1	0	0.5	0	0.5	1.5	0	1	-0.5	0	-0.5	0.5	-1.5
	1	1	1	1	2	1	2	1	1	1	1	1
2	0	0.5	1	0.5	1.5	2	4	3.5	5	2.5	4.5	2.5
	1	1	2	1	2	3	3	3	3	3	3	3
3	0	0.5	0	-0.5	1.5	0	1	-0.5	0	-1.5	-0.5	-0.5
	1	1	1	1	2	1	2	1	1	1	1	1
4	1	1.5	2	2.5	3.5	3	3	1.5	3	1.5	1.5	-0.5
	2	2	3	3	3	3	3	2	3	2	2	1
5	0	1.5	1	0.5	1.5	1	1	0.5	1	-0.5	2.5	1.5
	1	2	2	1	2	2	2	1	2	1	3	2
6	-1	0.5	1	-0.5	0.5	0	1	-0.5	0	-0.5	0.5	0.5
	1	1	2	1	1	1	2	1	1	1	1	1
7	0	0.5	1	-0.5	0.5	-1	1	-0.5	1	-0.5	0.5	-0.5
	1	1	2	1	1	1	2	1	2	1	1	1
8	0	0.5	0	-1.5	0.5	-1	0	-1.5	0	-1.5	-0.5	-0.5
	1	1	1	1	1	1	1	1	1	1	1	1
9	0	0.5	0	-1.5	0.5	-1	1	-0.5	0	-1.5	1.5	0.5
	1	1	1	1	1	1	2	1	1	1	2	1
10	0	0.5	0	-0.5	0.5	0	1	-0.5	1	1.5	2.5	0.5
	1	1	1	1	1	1	2	1	2	2	3	1
11	2	1.5	2	0.5	1.5	0	0	-0.5	0	0.5	1.5	0.5
	3	2	3	1	2	1	1	1	1	1	2	1
12	4	4.5	5	4.5	4.5	3	4	0.5	2	-1.5	-0.5	-0.5
	3	3	3	3	3	3	3	1	3	1	1	1
13	2	0.5	2	1.5	1.5	1	1	0.5	1	-0.5	0.5	-0.5
	3	1	3	2	2	2	2	1	2	1	1	1
14	0	1.5	1	0.5	1.5	0	1	-0.5	1	-0.5	0.5	-0.5
	1	2	2	1	2	1	2	1	2	1	1	1
15	0	0.5	1	-0.5	0.5	-1	0	-0.5	1	-0.5	-0.5	-1.5
	1	1	2	1	1	1	1	1	2	1	1	1

6.5.3. Delay Prediction and Accuracy Assessment

Denote

$$\vec{p}^{(1)}(7) = \left(p_1^{(1)}(7),\ p_2^{(1)}(7),\ p_3^{(1)}(7)\right) \ \text{and} \ \ \vec{p}^{(8)}(7) = \left(p_1^{(8)}(7),\ p_2^{(8)}(7),\ p_3^{(8)}(7)\right)$$

which are the delay distributions upon arrival at the destination (station number 7th) on 24/04/01 and 24/04/08, respectively (morning rush hours).

Based on Table 6.5, we have the initial distribution of the delay, i.e. the distribution of the random variable Y_1^*:

$$\left(q_1^{(8)}(1),\ q_2^{(8)}(1),\ q_3^{(8)}(1)\right) = \left(\tfrac{11}{15},\ \tfrac{1}{15},\ \tfrac{3}{15}\right). \qquad (6.32)$$

Using formula (6.21) and taking into account (6.30) and (6.31), we can find $Q(1,7)$, the transition matrix from Y_1^* to X_7^* for 24/04/01 (morning).

Then, using formula (6.19), by the known initial distribution $\left(q_1^{(8)}(1), q_2^{(8)}(1), q_3^{(8)}(1)\right)$ with the help of the matrix $Q(1,7)$ we find the forecast distribution, let us denote it $\vec{\overline{p}}^{(8)}(7) = (\overline{p}_1^{(8)}(7), \overline{p}_2^{(8)}(7), \overline{p}_3^{(8)}(7))$ which is a kind of approximation to the real distribution of deviations from the schedule upon arrival $\vec{p}^{(8)}(7) = (p_1^{(8)}(7), p_2^{(8)}(7), p_3^{(8)})$. Thus,

$$\vec{\overline{p}}^{(8)}(7) = \left(q_1^{(8)}(1), q_2^{(8)}(1), q_3^{(8)}(1)\right) \cdot Q(1,7). \qquad (6.33)$$

By successively multiplying the matrices (6.30) and (6.31) according to formula (6.21), we find that

$$Q(1,7) = Q(1) \cdot P(2) \cdot Q(2) \cdot P(3) \cdot Q(3) \cdot \ \dots \ \cdot P(6) \cdot Q(6) = \begin{pmatrix} 0.96 & 0 & 0.04 \\ 0.93 & 0 & 0.07 \\ 0.92 & 0 & 0.08 \end{pmatrix}.$$

By virtue of (6.33) and (6.32) then

$$\overline{\vec{p}}^{(8)}(7) = \left(\overline{p}_1^{(8)}(7), \overline{p}_2^{(8)}(7), \overline{p}_3^{(8)}(7)\right) = \left(\frac{11}{15}, \frac{1}{15}, \frac{3}{15}\right) \cdot$$
$$\begin{pmatrix} 0.96 & 0 & 0.04 \\ 0.93 & 0 & 0.07 \\ 0.92 & 0 & 0.08 \end{pmatrix} \qquad = (0.95, 0, 0.05). \qquad (6.34)$$

Using Table 6.5 we find the true distribution (of the random variable X_7^* for 24/04/08 (morning):

$$\left(p_1^{(8)}(7),\ p_2^{(8)}(7),\ p_3^{(8)}(7)\right) = \left(\frac{13}{15}, \frac{1}{15}, \frac{1}{15}\right) = (0.866,\ \ 0.067,\ \ 0.067).(6.35)$$

Note that the transition from ordinary fractions to decimal ones was made in (6.35) for clarity, where the absolute error in recording each coordinate does not exceed 0.001. It is obvious that the discrete distributions (6.35) and (6.34) are close to each other, although they do not coincide. The question arises how to measure this difference, or, in other words, how to measure the forecast error (relative to the chosen metric).

We will use the Kolmogorov metric.

Estimation of prediction error in the Kolmogorov metric. Recall that the Kolmogorov distance (also called the uniform distance) between arbitrary probability distributions U and V is determined by the formula:

$$d_K(U,V) = \sup_{-\infty < x < \infty} |F_U(x) - F_V(x)|,$$

where $F_U(x) = U\{(-\infty, x)\},\ F_V(x) = V\{(-\infty, x)\}$, i.e. $F_U(x)$ and $F_V(x)$ are the distribution functions for distributions U and V . In the example under consideration, the distribution functions for the distributions $U = \overline{\vec{p}}^{(8)}(7)$ and $V = \vec{p}^{(8)}(7)$ are non-decreasing step functions that have jumps at the same points, namely, at points 1, 2, and 3. It is easy to see that the value of

$$\sup_{-\infty < x < \infty} |F_U(x) - F_V(x)|$$

coincides with the maximum value of the modulus of the difference of the functions $F_U(x)$ and $F_V(x)$ at the indicated discontinuity points

$x = 1,\ 2$ and 3. Based on (6.34) and (6.35) we obtain the distribution functions:

$$F_U(x) = \begin{cases} 0 & \text{for } x \le 1, \\ 0.95 & \text{for } 1 < x \le 3, \\ 1 & \text{for } x > 3, \end{cases} \qquad F_V(x) = \begin{cases} 0 & \text{for } x \le 1, \\ 0.866 & \text{for } 1 < x \le 2, \\ 0.933 & \text{for } 2 < x \le 3, \\ 1 & \text{for } x > 3. \end{cases}$$

We have

$$d_K\left(\overset{\rightarrow(8)}{\overline{p}}(7), \overrightarrow{p}^{(8)}(7)\right) = \max\ \{0.95 - 0.866,\ \ 0.95 - 0.933\}$$

$$= \max\{0.084,\ \ 0.017\} = 0.084. \qquad (6.36)$$

Thus, the proposed model in the example under consideration gives a prediction for the distribution of schedule deviations (more exactly, for distribution of the delay states) upon arrival on 04/08/24 at station 7 with an error in terms of d_K not exceeding 0.085.

The simplest approach to delay predicting. This approach is to declare the known distribution of deviations upon arrivals at station 7 for some previous period as predictive. Of course, the choice of such a period is associated with the factors influencing the intensity of both the traffic flow and the passenger flow.

Since the intensity of train traffic and passenger flow on 04/08/.24 (morning) can be considered similar to the intensities of a week ago (Monday, morning rush hour), we will consider the distribution of delays at the terminal station for 24/04/01 as predicted for 24/04/08. We will also estimate the error of such a forecast in the d_K metric.

Using Table 6.1, we find the distribution of delays at the final station for 24/04/01:

$$\left(p_1^{(1)}(7),\ p_2^{(1)}(7),\ p_3^{(1)}(7)\right) = \left(\tfrac{24}{25}, 0, \tfrac{1}{25}\right) = (0.96,\ \ 0,\ \ 0.04). \quad (6.37)$$

Denote by $F_W(x)$ the distribution function for distribution (6.37). Using (6.37), we find that

$$F_W(x) = \begin{cases} 0 & \text{for } x \le 1, \\ 0.96 & \text{for } 1 < x \le 3, \\ 1 & \text{for } x > 3, \end{cases}$$

Calculating the Kolmogorov distance between distributions (6.37) and (6.35), we obtain

$$d_K\left(\vec{p}^{(1)}(7),\ \vec{p}^{(8)}(7)\right) = \max\{0.96 - 0.866,\ \ 0.96 - 0.933\} = 0.094.(38)$$

Thus, the error of the “simplest” forecast in the d_K metric is no more than 0.095. Comparing (6.36) with (6.38), we see that the Markov model proposed in the paper gives a more accurate forecast in the d_K metric, although it should be recognized that the improvement is only 10 %.

6.6. Conclusions

The paper proposes to model the sequence of delays of suburban trains on Russian Railways as a non-stationary Markov chain. To create the prediction for the distribution of arrival deviations at the terminal station, the Markov chain is used for the past period which is closest to the current period in terms of the situation developing performance. The paper proposes using the Kolmogorov metric to estimate the accuracy of the prediction too. A real example of a section of suburban railway in the Moscow region, consisting of seven stations and sections between them, shows that the prediction error is small enough in this metric.

Acknowledgements

This work is partially supported by the RF Ministry of Transport according to the program for research implementation by scientific and educational organizations in 2024. The authors thank the Russian infrastructure manager JSC RZD for the statistical data provided. We would also like to thank Sergey Shiliaev and Vladimir Karpeza, the students of the Far Eastern State Transport University for their assistance in transition matrices calculating.

References

[1]. P. Kecman, F. Corman, L. Meng, Train delay evolution as a stochastic process, in *Proceedings of the 6th International Conference on Railway Operations Modelling and Analysis (RailTokyo' 2015),* 2015, pp. 1-19.

[2]. V. Cacchiani, D. Huisman, M. Kidd, L. Kroon, et al., Overview of recovery models and algorithms for real-time railway rescheduling,

Transportation Research Part B: Methodological, Vol. 63, 2014, pp. 15-37.

[3]. C. W. Palmquist, N. O. E. Olsson, L. W. Hiselius, Some influencing factors for passenger train punctuality in Sweden, *International Journal of Prognostics and Health Management,* Vol. 020, 2017, pp. 1-13.

[4]. W. Barbour, J. C. M. Mori, S. Kuppa, D. B. Work, Prediction of arrival times of freight traffic on US railroads using support vector regression, *Transportation Research Part C: Emerging Technologies,* Vol. 93, 2018, pp. 211-227.

[5]. M. S. Artan, I. Sahin, Exploring patterns of train delay evolution and timetable robustness, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 23, Issue 8, 2023, pp. 11205-11214.

[6]. N. Bešinović, F. Flammini, L. De Donato, R. M. P. Goverde, et al., Artificial intelligence in railway transport: taxonomy, regulations and applications, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 23, Issue 9, 2022, pp. 14011–14024.

[7]. C. Jiang, P. Huang, J. Lessan, L. Fu, C. Wen, Forecasting primary delay recovery of high-speed railway using multiple linear regression, supporting vector machine, artificial neural network, and random forest regression, *Canadian Journal of Civil Engineering,* Vol. 46, Issue 5, 2019, pp. 353–363.

[8]. J. Xu, W. Wang, Z. Gao, H. Luo, Q. Wu, A novel Markov model for near-term railway delay prediction, ArXiv preprint, *arXiv:2205.10682*, 2022.

[9]. J. Jang, Travel-time prediction using K-nearest neighbor method with distance metric of correlation coefficient, *The Open Transportation Journal*, Vol. 13, 2019, pp. 141-150.

[10]. M. Marinov, J. Viegas, A mesoscopic simulation modelling methodology for analyzing and evaluating freight train operations in a rail network, *Simulation Modeling Practice and Theory,* Vol. 19, Issue 1, 2011, pp. 516–539.

[11]. A. A. Assad, Models for rail transportation, *Transportation Research Part A: General,* Vol. 14, Issue 3, 1980, pp. 205–220.

[12]. K. Bonsra, J. Harbolovic, Estimation of run times in a freight rail transportation network, Master's Thesis, *Massachusetts Institute of Technology,* 2012, 51 p.

[13]. F. Corman, A. D'Ariano, A. D. Marra, D. Pacciarelli, M. Sama, Integrating train scheduling and delay management in real-time railway traffic control, *Transportation Research Part E: Logistics and Transportation Review,* Vol. 105, 2017, pp. 213-239.

[14]. M. Carey, A. Kwiecinski, Properties of expected costs and performance measures in stochastic models of scheduled transport*, European Journal of Operational Research,* Vol. 83, 1995, pp. 182–199.

[15]. L. E. Meester, S. Muns, Stochastic delay propagation in railway networks and phase-type distributions, *Transportation Research Part B: Methodological*, Vol. 41, Issue 2, 2007, pp. 218–230.

[16]. T. Büker, B. Seybold, Stochastic modelling of delay propagation in large networks, *Journal of Rail Transport Planning & Management,* Vol. 2, Issues 1-2, 2012, pp. 34–50.

[17]. G. Medeossi, G. Longo, S. de Fabris, A method for using stochastic blocking times to improve timetable planning, *Journal of Rail Transport Planning & Management,* Vol. 1, Issue 1, 2011, pp. 1–13.

[18]. G. Longo, G. Medeossi, A. Nash, Estimating train motion using detailed sensor data, in *Transportation Research Board 91st Annual Meeting*, Washington, 2012, pp. 1–15.

[19]. S. Ozekici, S. Sengor, On a rail transportation model with scheduled services, *Transportation Science,* Vol. 28, Issue 3, 1994, pp. 246–255.

[20]. I. Sahin, Markov chain model for delay distribution in train schedules: assessing the effectiveness of time allowances, *Journal of Rail Transport Planning & Management,* Vol. 7, Issue 3, 2017.

[21]. R. Gaurav, B. Srivastava, Estimating train delays in a large rail network using a zero shot Markov model, in *Proceedings of the 21st IEEE International Conference on Intelligent Transportation Systems (ITSC' 2018),* 2018, pp. 1221–1226.

[22]. M. M. Dekker, R. N. van Lieshout, R. C. Ball, A next step in disruptin management: combining operations research and complexity science, *Public Transport,* Vol. 14, Issue 1, 2021, pp. 5-26.

[23]. J. Lessan, L. Fu, C. Wen, A hybrid Bayesian network model for predicting delays in train operations, *Computers & Industrial Engineering,* Vol. 127, 2019, pp. 1214-1222.

[24]. M. Lippi, M. Bertini, P. Frasconi, Short-term traffic flow forecasting: an experimental comparison of time-series analysis and supervised learning, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 14, Issue 2, 2013, pp. 871–882.

[25]. P. Kecman, R. M. P. Goverde, An online railway traffic prediction model, in *Proceedings of the 5th International Seminar on Railway Operations Modelling and Analysis Rail Copenhagen*, 2013, pp. 1-19.

[26]. K. Y. Tiong, Z. Ma, C. W. Palmqvist, A review of data-driven approaches to predict train delays, *Transportation Research Part C: Emerging Technologies,* Vol. 148, 2023, Article 104027.

[27]. P. Huang, C. Wen, L. Fu, Q. Peng, Z. Li, A hybrid model to improve the train running time prediction ability during high-speed railway disruptions, *Safety Science,* Vol. 122, 2020, Article 104510.

[28]. H. Li, D. Parikh, Q. He, B. Qian, Z. Li, D. Fang, A. Hampapur, Improving rail network velocity: a machine learning approach to predictive maintenance, Transportation Research Part C: Emerging Technologies, Vol. 45, 2014, pp. 17–26.

[29]. F. Cerreto, B. F. Nielsen, O. A. Nielsen, S. S. Harrod, Application of data clustering to railway delay pattern recognition, *Journal of Advanced Transportation,* Vol. 2018, Article ID 6164534.

[30]. P. Hall, B. U. Park, R. J. Samworth, Choice of neighbor order in nearest-neighbor classification, *Annals of Statistics,* Vol. 36, No. 5, 2008, pp. 2135–2152.

[31]. D. J. Beal, Information criteria methods in SAS® for multiple linear regression models, in *Proceedings of the SouthEast SAS Users Group Conference (SESUG),* 2006, Paper SA05, pp. 1–10.

[32]. N. Marković, S. Milinković, K. S. Tikhonov, P. Schonfeld, Analyzing passenger train arrival delays with support vector regression, *Transportation Research Part C: Emerging Technologies,* Vol. 56, 2015, pp. 251–262.

[33]. C. Wen, W. Mou, P. Huang, Z. Li, A predictive model of train delays on a railway line, *Journal of Forecasting,* Vol. 39, Issue 3, 2020, pp. 470–488.

[34]. W. Klumpenhouwer, A. Shalaby, Using delay logs and machine learning to support passenger railway operations, *Transportation Research Record,* Vol. 2676, Issue 9, 2022, pp. 134–147.

[35]. L. Oneto, E. Fumeo, G. Clerico, R. Canepa, et al., Train delay prediction systems: a big data analytics perspective, *Big Data Research,* Vol. 11, 2018, pp. 54–64.

[36]. P. Huang, C. Wen, L. Fu, J. Lessan, C. Jiang, Q. Peng, X. Xu, Modeling train operation as sequences: A study of delay prediction with operation and weather data, *Transportation Research Part E: Logistics and Transportation Review,* Vol. 141, 2020, Article 102022.

[37]. P. Huang, T. Spanninger, F. Corman, Enhancing the understanding of train delays with delay evolution pattern discovery: A clustering and Bayesian network approach, *IEEE Transactions on Intelligent Transportation Systems,* Vol. 23, Issue 9, 2022, pp. 1–16.

[38]. G. Cramer, Mathematical Methods of Statistics, *Mir Publishers,* Moscow, 1976 (in Russian).

[39]. V. I. Chebotarev, B. I. Davydov, K. S. Kablukova, On an approach to assessing the quality of operational control of train traffic, *Science and Technology of Transport,* No. 3, 2023, pp. 66-76 (in Russian).

Advances in Intelligent Systems

Volume 3

Sergey Y. Yurish, Editor

Advances in Intelligent Systems, Vol. 3 presents a diverse collection of cutting-edge research in neural computation, control theory, cybersecurity, and intelligent infrastructure. This volume features novel approaches to neural network synthesis using binomial hyperfilters, advanced methods for controlling nonlinear systems with chaotic dynamics, and ontology-based frameworks for device interoperability.

Each chapter combines theoretical innovation with real-world applications, making this book essential reading for engineers, researchers, and graduate students in AI, control systems, and cyber-physical technologies.

IFSA

ISBN 978-84-09-72041-5

9 788409 720415