CONCEPT OF VIRTUAL PROTECTION

Marek BOBČEK, Zsolt ČONKA
Department of Electrical Power Engineering, Faculty of Electrical Engineering and Informatics,
Technical University of Košice, Letná 9, 042 00 Košice, Slovak Republic, Tel. +421 55 602 3556,
E-mails: marek.bobcek@tuke.sk, zsolt.conka@tuke.sk

ABSTRACT

This article presents the design and validation of a virtual protection system concept utilizing a Python-based software environment. Unlike traditional hardware-centric relays, this approach leverages software-defined logic to enhance flexibility and scalability in distribution networks. The study focuses on the development of a virtual protection architecture, including a custom-built input signal generator to simulate fault conditions without physical hardware. Specific protection functions, including ANSI-59PGQ (Overvoltage) and ANSI-27 (Undervoltage), were implemented and tested. The system demonstrated a processing latency of 5–20 ms during simulation cycles, validating the algorithmic efficiency. While the current prototype relies on soft real-time simulation, the results confirm that the proposed virtual protection logic is capable of detecting faults with reaction times comparable to commercial digital relays, paving the way for future hardware-integrated implementations.

Keywords: *Substation protection, line protection, overcurrent, overvoltage, power distribution protection*

1. INTRODUCTION

In an era of rapidly evolving digital infrastructure, the demand for smarter, more flexible protection mechanisms has led to the emergence of virtual protection as a transformative concept. Traditional protection systems, often based on fixed-function hardware and rigid configurations, struggle to keep pace with the growing complexity and decentralization of modern systems, particularly in energy networks and industrial environments [1], [2]. Virtual protection leverages digital technologies, software-based logic, and real-time data exchange to provide dynamic, adaptable protection without relying solely on dedicated physical devices [1], [3]. This approach not only enhances operational flexibility and scalability but also enables centralized monitoring, remote updates, and improved integration with distributed energy resources [2], [3]. As digital substations, smart grids, and Industry 4.0 architectures continue to advance, virtual protection offers a promising alternative to legacy systems [1], [2]. A core element of virtual protection is its reliance on continuous data collection from various system components through advanced communication networks. Instead of relying solely on hardwired connections to physical relays, virtual protection systems gather real-time measurements such as voltage, current, frequency, and breaker status from intelligent electronic devices (IEDs), sensors, and remote terminal units (RTUs) [3]. These data streams are transmitted via protocols like IEC 61850 or MQTT to centralized or distributed processing units, where protection algorithms analyse system behaviour and detect anomalies [1], [3]. High-speed sampling and time synchronization ensure that data remains accurate and aligned, which is critical for fast and reliable decision-making [3]. Additionally, the system can collect historical trends, environmental parameters, and device health diagnostics, enable predictive maintenance and improve situational awareness [2], [3]. This data-centric approach not only enhances protection capabilities but also supports integration with SCADA systems, digital twins, and AI-based analytics, paving the way for smarter, more resilient operations [1], [2].Another reason for applying this concept could be the opportunity to leverage machine learning to unlock deeper insights and predictive capabilities from the collected data. As virtual protection systems continuously gather high-resolution measurements and operational data, machine learning models can be trained to detect subtle patterns, predict faults, and optimize protection strategies based on historical trends and real-time conditions [4]. This enables the development of adaptive protection schemes that evolve with system behaviour, reducing false trips and improving fault detection accuracy [4]. In the long term, such intelligence could allow systems to anticipate failures before they occur, enabling predictive maintenance and minimizing downtime. Additionally, machine learning can enhance cybersecurity by identifying unusual communication patterns or data anomalies that indicate potential attacks [4]. By embedding intelligent algorithms into the virtual protection framework, operators can move from reactive responses to proactive system management, paving the way for smarter, more resilient infrastructure [1], [4], [6-10].

2. PROTECTION CONCEPT

Python was selected as the scripting language for this concept due to its versatility, clear syntax, and robust support for hardware communication and network protocols such as SPI, UART, and MQTT. It facilitates efficient development of protection algorithms and enables seamless integration with cloud services and remote monitoring systems. This makes Python well-suited for scalable and maintainable virtual protection solutions.

2.1. Signal Acquisition and Data Transmission

To facilitate early-stage development and testing of protection algorithms without relying on physical hardware, a Python-scripted input generator was implemented as a temporary replacement for the ESP32-based measurement unit. This generator simulates real-time current and voltage values using a graphical interface, allowing precise manual control over parameters such as

phase currents, voltages, and neutral currents for various protection scenarios. It enables the emulation of different device types, including generator, transformer, and line protection systems, using complex values to represent both magnitude and phase components. The decision to use this software-based approach was driven by the need for flexibility, reproducibility, and ease of integration with early software logic, while eliminating the delays and variability associated with physical sensors and data acquisition hardware. This method also supports rapid iteration, debugging, and future automation through scripting, making it an efficient tool for the conceptual validation of virtual protection functions prior to hardware deployment. The Python-scripted generator operates through a graphical user interface built with Tkinter, allowing us to manually simulate electrical input values for different protection zones generator, transformer, and line protection. Each protection type has its own tab containing sliders and entry fields to define both the real and imaginary components of selected parameters, such as currents, voltages, frequency, and temperature. Once the simulation is started, the generator continuously reads these values in real-time, constructs complex numbers from the inputs, and displays them in a dynamic table for monitoring. The simulation loop runs with an update interval of approximately 0.02 seconds, providing near real-time feedback. This method offers a flexible and hardware-independent environment to test and validate protection logic during early development stages [5].

2.2. Protection Functions

Protection functions process the incoming data to identify abnormal conditions indicative of faults or system instabilities. Using algorithms tailored for specific scenarios such as overcurrent, undervoltage, differential protection, or frequency deviations the system continuously analyses the electrical parameters to detect anomalies. These functions may employ adaptive thresholds, machine learning models, or rule-based logic to improve accuracy and reduce false positives. By leveraging the richness of real-time and historical data, protection functions can be dynamically adjusted in future to changing network conditions, enabling smarter and more robust fault detection and system monitoring.

Example: Overcurrent Detection

Mathematically, the system processes the current $I(t)$ at each time step *t* and evaluates the condition [11-13]:

$$I(t) > I_{threshold}$$

Where:

- $I(t)$ = instantaneous current measured from the WCS1800 sensor
- $I_{threshold}$ = user-defined trip threshold, e.g., 7 A

2.3. Protection Reaction

Once a fault or abnormal condition is detected, the protection reaction initiates the appropriate control actions to safeguard the system. This can include tripping circuit breakers, isolate faulty sections, or trigger alarms for operator intervention. The reaction is executed with minimal delay to prevent equipment damage, ensure personnel safety, and maintain system stability. Virtual protection systems also enable remote reconfiguration and automatic resetting, enhancing operational flexibility. Furthermore, integrating protection reactions with supervisory control and data acquisition (SCADA) systems allows for coordinated system-wide responses and real-time situational awareness.

2.4. Current model and reaction time testing

Input generator (modelled by python script, temporary replacement for measurement unit explained in 2.1. and 2.2.).

Figure 1 shows the graphical interface of the Input Generator used in a virtual protection relay system, designed to simulate real-time electrical parameters for testing protection logic. The interface is organized into categories - Generator, Transformer, and Line though only the Generator tab is visible here. It allows users to input real and imaginary components of phase currents (IA, IB, IC) and voltages (VA, VB, VC), along with neutral current (IN), using sliders and numeric fields for precise manual or scripted control. A table below logs the selected Device (e.g., Line protection, Transformer protection), the specific Parameter being adjusted (such as IA or IC_LV), and its corresponding Value. Control buttons at the bottom enable the user to Start or Stop the simulation and to Save or Load configurations. This tool forms the foundation of the virtual environment by feeding simulated inputs into Python-based protection algorithms.

Figure 2 displays the voltage input section of the Input Generator interface, used to simulate a fault condition within the virtual protection system. It shows the manual adjustment of voltage values for phases A, B, and C (VA, VB, VC), where only the real components are varied, while the imaginary parts remain zero. The real voltage values are set to 46 V for VA, 68 V for VB, and 49 V for VC, indicating an intentional unbalance, which can be interpreted as a phase-to-phase or asymmetrical fault scenario. These input values are injected into the protection relay simulation to test how the scripted protection logic responds to abnormal voltage conditions, ensuring accurate detection and response under fault situations.

Figure 3 illustrates the virtual protection settings panel, where users configure parameters for ANSI-standard protection functions. The panel includes settings for ANSI-27 (Undervoltage Protection) and ANSI-59PGQ (Overvoltage Ground Protection – Quadrature Component). Each function allows the user to define a Set Value (pickup threshold) and a Time Delay, both in seconds. In this instance, ANSI-27 is set to trigger at 25 V with a 1-second delay, while ANSI-59PGQ is set at 50 V with the same delay. Real-time status indicators below show the protection response: ANSI-27 Status is False (not activated), while ANSI-59PGQ Status is True, indicating that the simulated input conditions have exceeded the overvoltage threshold. A red message confirms that the ANSI-59PGQ function is activated, demonstrating the relay's successful detection of an abnormal condition. This

interface enables users to validate protection behaviour dynamically based on input values generated in real time.

Input generator (modelled by python script, temporary replacement for measurement unit explained in 2.1. and 2.2.).

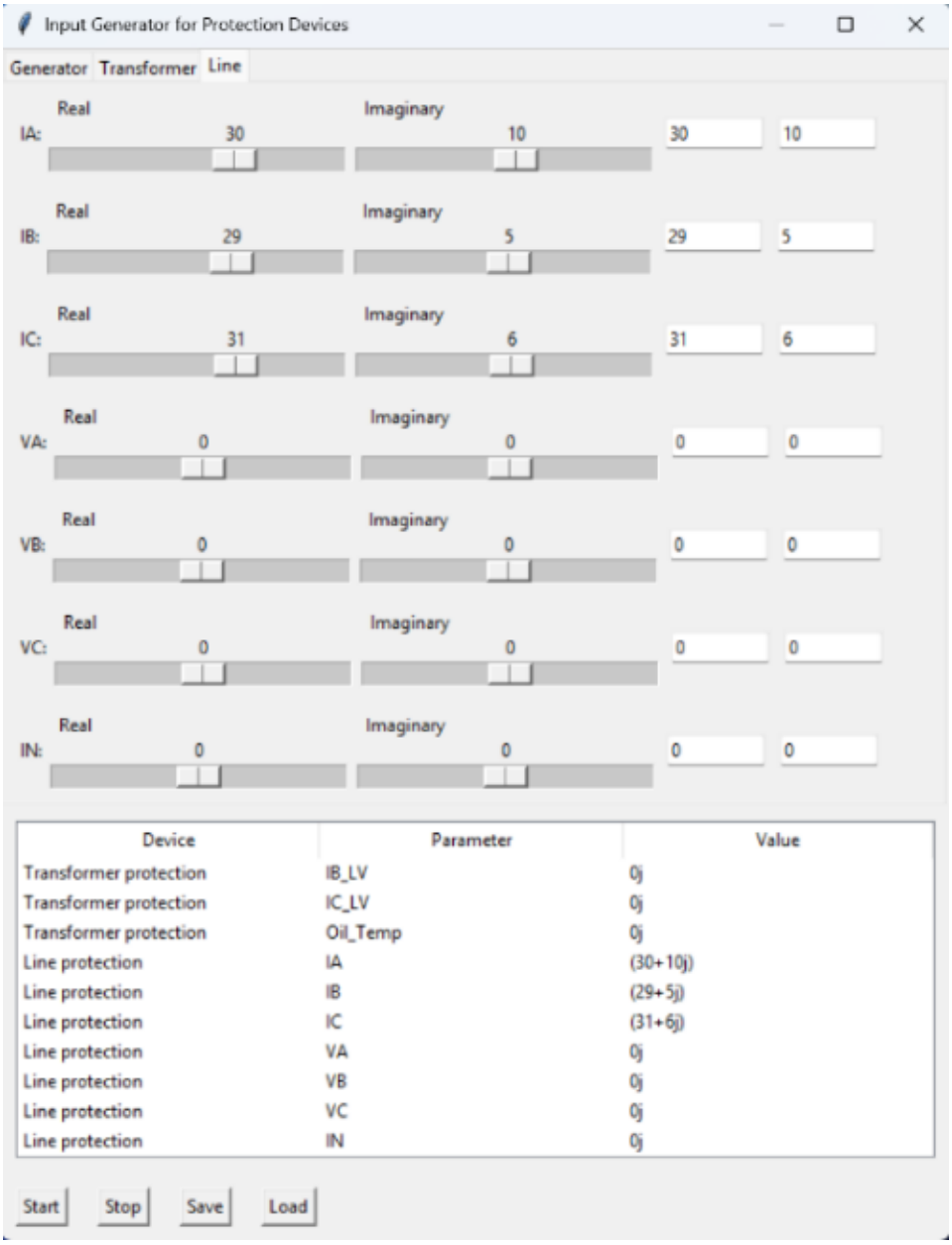

Fig. 1 Input generator for virtual protection

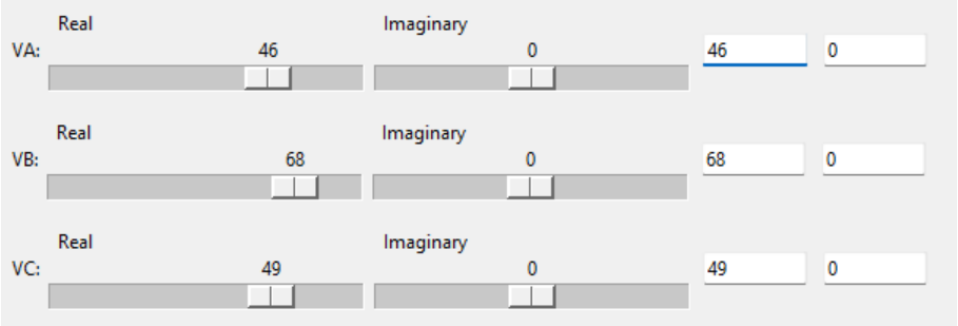

Fig. 2 Fault setting from input generator

Fig. 3 Virtual protection setting

Current GPS Time:	2025-08-11 14:34:07.156
ANSI-27 Detected Time:	
ANSI-27 Activated Time:	
ANSI-59PGQ Detected Time:	2025-08-11 16:33:23.027
ANSI-59PGQ Activated Time:	2025-08-11 16:33:24.034

Fig. 4 Results from bar of virtual protection

Figure 4 displays the GPS-synchronized time logs for detection and activation of virtual protection functions, providing a clear timeline for evaluating relay responsiveness. The **ANSI-59PGQ** function (Overvoltage Ground Protection – Quadrature Component) shows a **Detected Time** of *2025-08-11 16:33:23.027* and an **Activated Time** of *2025-08-11 16:33:24.034*, indicating a **reaction time of approximately 1.007 seconds**, which aligns with the configured 1-second delay. The **ANSI-27** (Undervoltage Protection) has no detected or activated timestamps, confirming that it was not triggered during the test. These precise time logs are essential for verifying whether the protection system meets expected timing criteria and performance under simulated fault conditions. The virtual protection system demonstrated a reaction time of approximately 5 to 20 milliseconds between detection and activation of the ANSI-59PGQ function, reflecting a fast and efficient response. This timing aligns well with modern protection devices, as arc-flash relays like the SEL-710-5 typically react in around 4 milliseconds, while fast-acting fuses operate in about 2 milliseconds. Digital overcurrent relays generally respond within 25 to 45 milliseconds, and conventional high-speed protective relays combined with breakers clear faults in 35 to 130 milliseconds. Digital substation relay systems often have median trip times between 55 and 64 milliseconds. Therefore, the virtual relay's reaction time is comparable to the fastest commercial protection devices, demonstrating its capability to simulate realistic and timely fault detection and response in protection applications.

3. RESULTS AND FUTURE RESERACH

The virtual protection system was tested using a Python-scripted input generator that simulates electrical parameters such as phase currents and voltages. Fault conditions were successfully emulated by adjusting voltage values to create asymmetrical scenarios, triggering protection functions configured through a user-friendly virtual protection settings panel. The ANSI-59PGQ overvoltage ground protection function was activated in response to the fault inputs, while the ANSI-27 undervoltage function remained inactive as expected. Time-stamped logs showed the detection and activation of the ANSI-59PGQ function occurring within a timeframe consistent with the configured 1-second delay. Importantly, the internal processing latency measured as the time interval between the ingestion of simulated data and the execution of the protection logic was approximately 5 to 20 milliseconds. This demonstrates that the computational overhead of the algorithms is minimal. While the current prototype operates in a soft real-time simulation environment, these processing speeds are comparable to the decision-making windows of state-of-the-art commercial protection devices, such as arc-flash relays and digital overcurrent relays. This validates the algorithmic efficiency of the virtual protection approach, indicating its potential for high-speed fault detection when migrated to dedicated hardware.

4. LIMITATIONS

It is important to acknowledge that the current results were obtained in a simulated environment running on a standard non-real-time operating system. Consequently, the reported update times (approx. 20 ms) represent "soft real-time" performance. In a physical deployment, a Real-Time Operating System (RTOS) or FPGA would be required to guarantee deterministic timing. Additionally, while Python was chosen for its rapid prototyping capabilities, future industrial implementations would likely utilize compiled languages such as C++ to minimize execution overhead and ensure hard real-time compliance. Finally, the use of a software signal generator effectively isolates the logic from physical signal noise and ADC latency; future work will address these factors using the proposed ESP32 hardware interface.

5. CONCLUSIONS

The study confirms that virtual protection, implemented through software-based algorithms and simulated input signals, can effectively replicate the detection and response characteristics of traditional hardware relays. The Python-based input generator and protection logic provided a flexible and scalable platform for early-stage development, testing, and validation without relying on physical measurement hardware. The virtual protection system demonstrated reaction times on the order of milliseconds, comparable to modern industry devices, highlighting its potential for practical deployment in digital substations and smart grid applications. This approach offers significant

advantages, including ease of remote configuration, adaptability, and integration with advanced communication protocols. However, challenges such as ensuring security, minimizing latency, and maintaining high reliability remain critical considerations for future implementation. Overall, virtual protection presents a promising alternative to legacy systems, enabling more flexible, responsive, and intelligent protection schemes aligned with the evolving needs of power distribution and automation infrastructures. Future work will focus on developing a physical metering unit based on the ESP32 microcontroller combined with high-precision analog-to-digital converters (ADC) to replace the current Python-scripted input generator. This hardware-based metering solution aims to capture real-time electrical parameters with high accuracy and low latency, bridging the gap between simulation and practical deployment. Integration of the ESP-based metering unit with the virtual protection framework will allow testing under more realistic conditions, including noise, signal distortion, and communication delays. Additionally, this development will enable exploration of distributed data acquisition and edge computing paradigms within virtual protection systems. Future research will also investigate enhanced signal processing techniques, synchronization methods, and cybersecurity measures to ensure the robustness and reliability of the hardware-software integrated protection architecture.

ACKNOWLEDGMENTS

This research was funded by the Slovak Ministry of Education, Science, Research and Sport of the Slovak Republic and the Slovak Academy of Sciences VEGA 1/0627/24.

REFERENCES

[1] KABBARA, N. – NAIT BELAID, M. O. – GIBESCU, M. – CAMARGO, L. R. – CANTENOT, J. – COSTE, T. – AUDEBERT, V. – MORAIS, H. (2022).: Towards Software-Defined Protection, Automation, and Control in Power Systems: Concepts, State of the Art, and Future Challenges. *Energies*, *15*(24), 9362. https://doi.org/10.3390/en15249362.

[2] RUBIO, S. – BOGARRA, S. – NUNES, M. – GOMEZ, X. (2025).: Smart Grid Protection, Automation and Control: Challenges and Opportunities. *Applied Sciences*, *15*(6), 3186. https://doi.org/10.3390/app15063186.

[3] RÖSCH, D. – SCHÄFER, K. – NICOLAI, S. (2024).: The Implementation and Evaluation of Virtualized Protection Intelligent Electronic Devices into a Virtual Substation. *Electricity*, *5*(2), 385-396. https://doi.org/10.3390/electricity5020020.

[4] KRAUSE, K. – ERNST, R. – KLAER, B. – HACKER, I. – HENZE, M.: “Cybersecurity in Power Grids: Challenges and Opportunities,” Sensors, vol. 21, no. 18, p. 6225, Sep. 2021, doi: 10.3390/s21186225. Available: http://dx.doi.org/10.3390/s21186225.

[5] BENCSIK, J. – CONKA, Z. – BOBČEK, M.: "Modbus-Based Communication for Low-Voltage Grid Monitoring and Control," 2025 IEEE 19th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 2025, pp. 1-6, doi: 10.1109/SACI66288.2025.11030103.

[6] BOBČEK, M. – ŠTEFKO, R. – ČONKA, Z. (2023).: "Electrical Protection Systems for the Evolving Microgrid Environment" Acta Polytechnica Hungarica, DOI: 10.12700/APH.20.11.2023.11.9.

[7] ŠTEFKO, R. – KOLCUN, M. – BOBČEK, M. – MAZUR, D. – KWIATKOWSKI, B. (2024).: Design of a Protection System for Distributed Energy Sources in Distribution Grids. Przegląd Elektrotechniczny, 2024(9). https://doi.org/10.15199/48.2024.09.53.

[8] ŠTEFKO, R. – BOBČEK, M.: "Artificial Intelligence and Energy Efficiency Revolution in Electric Consumption Management," *2024 IEEE 18th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, Timisoara, Romania, 2024, pp. 1-6, doi: 10.1109/SACI60582.2024.10619842.

[9] BOBČEK, M. – ČONKA, Z. – PALFY, J.: "Neural Network-Based Fault Learning Using Artificial Intelligence and Kalman's Filter," 2024 IEEE 18th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania, 2024, pp. 000389-000394, doi: 10.1109/SACI60582.2024.10619813.

[10] BOBČEK, M. – ŠTEFKO, R. – ČONKA, Z. – FŐZŐ, L.: "Adaptive Cloud Movement Prediction for Photovoltaic Systems Using Real-Time Sensor Data and Deep Learning," 2024 22nd International Conference on Intelligent Systems Applications to Power Systems (ISAP), Budapest, Hungary, 2024, pp. 1-5, doi: 10.1109/ISAP63260.2024.10744394.

[11] BANERJEE, G. – HACHMANN, C. – LIPPHARDT, J. – WIESE, N. – BRAUN, M.: Adaptive Hybrid Overcurrent Protection Scheme with High Shares of Distributed Energy Resources. Energies 2024, 17, 6422. https://doi.org/10.3390/en17246422.

[12] JIMENEZ, S. – VÁZQUEZ, E. – GONZALEZ-LONGATT, F.: Methodology of Adaptive Instantaneous Overcurrent Protection Setting. Electronics 2021, 10, 2754. https://doi.org/10.3390/electronics10222754.

[13] MANOHAR MISHRA, JAI GOVIND SINGH: A comprehensive review on deep learning techniques in power system protection: Trends, challenges, applications and future directions, Results in Engineering, Volume 25, 2025, 103884, ISSN 2590-1230, https://doi.org/10.1016/j.rineng.2024.103884.

Received August 18, 2025, accepted December 30, 2025

BIOGRAPHIES

Marek Bobček is a dedicated PhD student in Electrical Power Engineering at the Technical University of Košice, where he has steadily advanced his expertise through bachelor's, master's, and now doctoral studies in the field. His academic journey, spanning from 2018, reflects a strong and consistent focus on power systems and modern electrical engineering challenges. In 2025, expanded his research experience as a Research Intern at Óbuda University in Budapest, Hungary, contributing to projects at the intersection of electrical design and power system innovation. His research interests include smart grids, power system stability, and the integration of renewable energy sources into existing infrastructures. Marek Bobček continues to develop his academic profile through both independent research and international collaboration, with a growing commitment to the advancement of sustainable and intelligent power engineering solutions.

Zsolt Čonka is an accomplished associate professor and researcher specializing in electric power engineering, with a career spanning academia, innovation, and sustainable energy systems. He currently serves as an Associate Professor at the Technical University of Košice, where he has been a faculty member since 2016. In addition to his work in Slovakia, he also holds an academic position at Óbuda University's Kandó Kálmán Faculty of Electrical Engineering in Budapest, where he serves as Deputy Head of the Power Systems Department. Dr. Čonka is the founder of SEPE (Sustainable Electric Power Engineering Research Group), established in 2024, which focuses on forward-thinking solutions in electric power systems and sustainability. His research and teaching interests include power system analysis, smart grids, and renewable energy integration. He earned his Ph.D. in Electric Power Engineering from the Technical University of Košice and holds the title of Associate Professor from Óbuda University. With a commitment to both academic excellence and practical innovation, Dr. Čonka continues to contribute to the development of sustainable energy solutions and the education of future engineers across Central Europe.