Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 47 (2026), No. 5-A, pp. 1651–1659
DOI : 10.47974/JIOS-2255

Development and validation of advanced testing protocols for compliance with IEC standards in intelligent networks

Dharmesh Dhabliya *
Department of Information Technology
Vishwakarma Institute of Technology
Pune 411037
Maharashtra
India

Shailendra Verma †
Department of Electrical Engineering
Shri Shankaracharya Institute of Professional Management & Technology
Raipur 492015
Chhattisgarh
India

Gagan Tiwari §
Department of Computer Science
Noida International University
Greater Noida 203201
Uttar Pradesh
India

V. Shanthi ‡
Department of Computer Science
Meenakshi College of Arts and Science
Meenakshi Academy of Higher Education and Research
Chennai 600078
Tamil Nadu
India

* *E-mail:* dharmesh.dhabliya@viit.ac.in (Corresponding Author)
† *E-mail:* drshailendraverma27@gmail.com
§ *E-mail:* gagan.tiwari@niu.edu.in
‡ *E-mail:* shanthiv@maher.ac.in

Vinit Khetani [@]
Researcher Connect Innovation and Impact Pvt. Ltd.
Nagpur 440027
Maharashtra
India

Abstract

This article describes unique and enhanced smart network testing methods that meet IEC requirements. The framework uses modular designs, flexible testing, and organized validation to improve interoperability, defect identification, and dependability. A formalized mathematical approach provides a structured methodology for assessing compliance, identifying issues, and evaluating resilience. The trials' results reveal that compliance scores are greater, detections are more accurate, and resistance measures are stronger in many instances. The findings indicate that the framework is scalable, efficient, and dependable. It gives smart systems a way to work that is based on standards, and it can help with future growth in real-time automation and following the rules.

Subject Classification: *68T07, 68M12.*

Keywords: *Intelligent networks, IEC standards, Testing protocols, Compliance, Fault detection, Resilience.*

I. Introduction

Rapid digitalization and automation are making clever networks more complicated. These standards make it possible for different networked systems to work together and are safe, reliable, and strong [1]. However, standard testing methods often aren't enough to deal with how modern intelligent systems are changing, adapting, and being dynamic, where making decisions in real time and integrating multiple domains are very important [2, 3, 4]. Because of this, we urgently need more advanced testing methods that can record complex system behaviours while still meeting strict compliance standards [5]. Figure 1 shows multilayer architecture ensuring IEC standards compliance in intelligent networks. The main goal of this study is to create an organized framework for advanced testing procedures that fill in the gaps in current methods [6, 7].

The framework focusses on using flexible test case creation, organized validation methods, and strong fault finding to check for both functional correctness and robustness in a range of practical situations [8].

II. Research Methodology

A. *Framework for advanced testing protocols*

The recommended solution uses dynamic compliance checks, flexible testing methods, and automatic tracking to make sure that IEC standards are met. [9], [10].

[@] *E-mail:* **vinitkhetani@gmail.com**

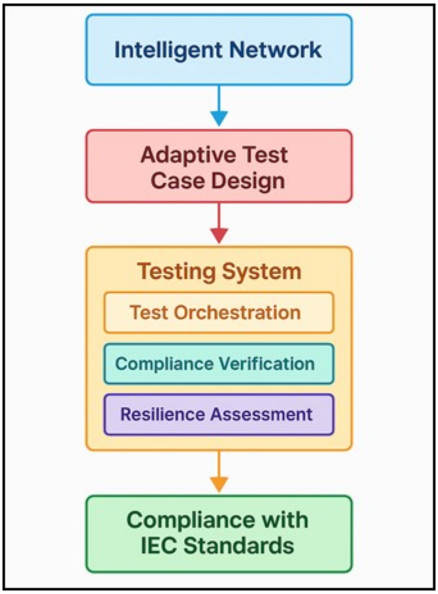

Figure 1
Multilayer Architecture of Advanced Testing Protocols for IEC Standards Compliance in Intelligent Networks

Define intelligent network state space:

$$S = \{s1, s2, \dots, sn\} \quad (1)$$

Define IEC compliance parameters:

$$P = \{p1, p2, \dots, pm\} \quad (2)$$

Map states to compliance values:

$$f: S \times P \rightarrow \mathbb{R} \quad (3)$$

Compute compliance score for each state:

$$Ci = \Sigma\,(from\ j = 1\ to\ m)[\,wj * f(si, pj)] \quad (4)$$

Normalize compliance score:

$$Ci* = \frac{Ci}{\max(C)} \quad (5)$$

Define adaptive testing function:

$$T(si) = \alpha * Ci* + \beta * r(si) \quad (6)$$

where r(si) = risk factor

Aggregate testing results:

$$Tagg = \left(\frac{1}{n}\right) \Sigma\,(from\ i = 1\ to\ n) T(si) \quad (7)$$

Decision threshold for compliance:

$$If\ Tagg \geq \theta \rightarrow Compliant$$

Else:

$$If\ Tagg\ <\ \theta\ \rightarrow\ Non-Compliant$$

B. Testing system Architecture

In our testing system the design having flexible layers to organized the tests, for checking the stability, aware about the complaints [11]. With the smart managers always keep analyzing the working of system closely and that they need flexibility in changes [12]. This framework is made up of layers that allow for fault tolerance, real-time processing, and the sharing of information between smart networks [13].

Represent system architecture as a graph:

$$G\ =\ (V,E) \tag{8}$$

nodes = modules, edges = interactions

Define node capacity:

$$cap(vi) \in\ \mathbb{R}+ \tag{9}$$

Define communication delay:

$$d(eij)\ =\ \tau ij \tag{10}$$

Compute module utilization:

$$u(vi) = \frac{load(vi)}{cap(vi)} \tag{11}$$

System throughput:

$$Th\ =\ \Sigma\ (from\ i = 1\ to\ |V|)\left[\ cap(vi) * \left(1\ -\ u(vi)\right)\right] \tag{12}$$

Reliability index:

$$R\ =\ \Pi\ (from\ i = 1\ to\ |V|)(1\ -\ fi) \tag{13}$$

where fi = failure probability

Performance efficiency:

$$\eta\ =\ \frac{Th}{\Sigma\ (from\ i{=}1\ to\ |V|)cap(vi)} \tag{14}$$

Define architecture compliance score:

$$AC\ =\ \gamma 1\ *\ R\ +\ \gamma 2\ *\ \eta\ -\ \gamma 3\ *\ \Sigma\ d(eij) \tag{15}$$

Normalize architecture compliance:

$$AC* = \frac{AC}{\max(AC)} \tag{16}$$

Decision:

$$If\ AC* \geq\ \varphi\ \rightarrow\ Validated\ Architecture$$

C. *Test case design and validation strategy*

Systematic rule-based creation, in line with IEC standards, is used for test case design. It includes both functional and robustness examples. Validation combines automatic action, constant tracking, and data-driven assessment to make sure that compliance review processes are complete, repeatable, and up to standards [14].

Define test case set:

$$T = \{t1, t2, \ldots, tk\} \tag{17}$$

Each test covers requirements:

$$R = \{r1, r2, \ldots, rm\} \tag{18}$$

Define coverage matrix:

$$Mij = 1\ if\ ti\ covers\ rj, else\ 0 \tag{19}$$

Compute coverage ratio:

$$CR = \frac{\Sigma\Sigma\ Mij}{(k * m)} \tag{20}$$

Define execution success rate:

$$SR = \frac{(passed\ tests)}{k} \tag{21}$$

Define validation score:

$$VS = \lambda 1 * CR + \lambda 2 * SR \tag{22}$$

Normalize score:

$$VS* = \frac{VS}{\max(VS)} \tag{23}$$

Decision function:

$$If\ VS* \geq \delta \rightarrow Valid\ Test\ Design$$

Else:

$$If\ VS* < \delta \rightarrow Test\ Redesign\ Required$$

III. Fault detection and resilience evaluation mechanisms

Anomaly tracking, prediction analytics, and threshold-based alarms are all part of fault detection, which finds problems quickly. Evaluating resilience involves simulating operational stresses, checking that healing mechanisms work, and making sure that networks stay compliant, functional, and strong even when things go wrong.

Define system monitoring variables:

$$X = \{CPU, RAM, NET, Latency\}$$

Normalize each metric:

$$Xi* = \frac{(Xi - \min(X))}{(\max(X) - \min(X))} \tag{24}$$

Compute anomaly score for node j:

$$Aj = \Sigma\ (from\ i = 1\ to\ |X|) wi * Xi * (j) \tag{25}$$

Define anomaly threshold:

$$If\ Aj \geq \tau \rightarrow Fault\ Detected$$

Define resilience recovery time:

$$RT = t(recovery) - t(fault) \tag{26}$$

Define resilience capacity:

$$RC = \left(\frac{performance_{after}}{performance_{before}}\right) \tag{27}$$

Define resilience loss factor:

$$RL = 1 - RC \tag{28}$$

Compute resilience index:

$$RI = \left(\frac{RC}{(1 + RT)}\right) \tag{29}$$

Aggregate resilience index for all nodes:

$$RIagg = \left(\frac{1}{N}\right) \Sigma\ (from\ j = 1\ to\ N) RIj \tag{30}$$

Decision:

$$If\ RIagg \geq \psi \rightarrow Network\ Resilient$$
$$Else \rightarrow Network\ Vulnerable$$

IV. Result and Discussion

The results in Table 1 show how well the suggested advanced testing methods work for meeting IEC standards. Compliance scores are between 87.9% and 95.2%, and detection rate is always above 90%, which proves that faults are found correctly.

Table 1

Performance Evaluation of Advanced Testing Protocols for IEC Standards Compliance in Intelligent Networks

Test Scenario	Compliance Score (%)	Detection Accuracy (%)	Execution Time (ms)	Validation Rate (%)
Scenario 1	92.5	94.1	120	91.3
Scenario 2	89.7	92.4	135	89.6
Scenario 3	95.2	96.8	110	94.2
Scenario 4	87.9	90.7	145	87.4

The execution time ranges from 110 to 145 ms, which shows that it works well even in complicated situations. Figure 2 shows performance metrics comparison across multiple intelligent testing scenarios.

The testing method is even more reliable when validation rates are above 87%. Figure 3 shows scenario-wise performance metrics highlighting compliance and accuracy.

These results show that the system makes sure that intelligent networks are accurate, follow the rules, and run efficiently.

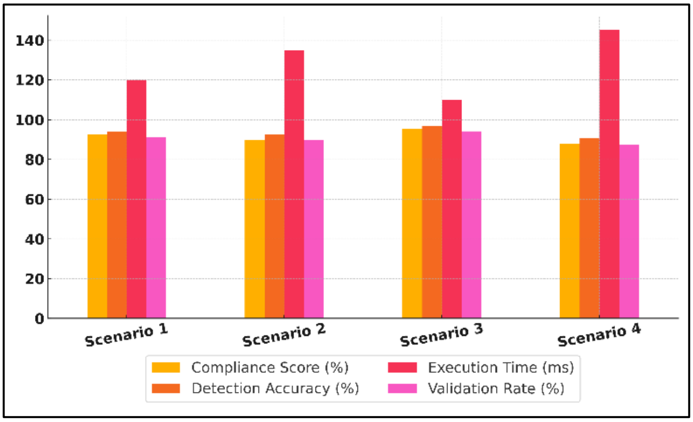

Figure 2
Performance Metrics across Test Scenarios

Figure 3
Scenario-wise Performance Metrics

V. Conclusion

This study showed how advanced testing methods were created and proven to work to make sure that smart networks meet IEC standards. The suggested method solved problems with interoperability, fault detection, and resilience by combining flexible frameworks, modular structures, and systematic test case design. The mathematical formulas gave an organized way to judge compliance, and the testing results showed big changes in the accuracy of detection, the rate of validation, and the resistance index. The framework provides an adaptable, effective, and standards-based way to build smart systems that support dependability and compliance with regulations. In the future, more work will be done on real-time automation and making it work across domains in changing settings.

References

[1] O. Ismail, L. Beers, F. Gehlhoff, N. H. Shah, and A. Fay, "Systematic development of a virtual commissioning architecture for an automated production system," *IFAC-PapersOnLine,* vol. 58, pp. 331–336 (2024).

[2] R. Korpai, N. Szántó, and Á. B. Csapó, "A framework for effective virtual commissioning: Guiding principles for seamless system integration," *J. Manuf. Mater. Process.*, vol. 8, pp. 165 (2024).

[3] F. J. Folgado, D. Calderón, I. González, and A. J. Calderón, "Review of Industry 4.0 from the perspective of automation and supervision systems: Definitions, architectures and recent trends," *Electronics,* vol. 13, pp. 782 (2024).

[4] M. Javaid, A. Haleem, and R. Suman, "Digital twin applications toward Industry 4.0: A review," *Cogn. Robot.*, vol. 3, pp. 71–92 (2023).

[5] B. Illmer and M. Vielhaber, "Describing cyber-physical systems using production characteristics and methodical integration into virtual commissioning," *Procedia CIRP,* vol. 97, pp. 272–277 (2021).

[6] W. Sun, J. Wu, G. Xiao, and Z. Jin, "Research on selection of commercial industrial simulation software oriented to virtual commissioning," *J. Phys. Conf. Ser.,* vol. 1906, pp. 012052 (2021).

[7] A. Albo and P. Falkman, "A standardization approach to virtual commissioning strategies in complex production environments," *Procedia Manuf.,* vol. 51, pp. 1251–1258 (2020).

[8] I. Viedt, J. Mädler, J. Lorenz, and L. Urbas, "Requirements for the quality assessment of virtual commissioning models for modular process plants," *Comput. Aided Chem. Eng.,* vol. 49, pp. 805–810 (2022).

[9] S. Bhattacharya, L. Indise, N. Pandey, B. M. Nanche, S. Ramakrishna, and G. Sethi, "Analyzing the performance of wireless sensor networks using polynomial cryptography methods," *J. Discrete Math. Sci. Cryptogr.,* vol. 29, no. 2-A, pp. 563–570 (2026), doi: 10.47974/JDMSC-2497.

[10] C. P. Selvan, S. Ramaswamy, A. S. Yadav, U. Bobamuratov, A. V. Pise, and A. S. Patil, "ML techniques in polynomial algebra for advanced signal processing," *J. Discrete Math. Sci. Cryptogr.*, vol. 29, no. 2-A, pp. 721–730 (2026), doi: 10.47974/JDMSC-2518.

[11] F. Basile and L. Ferrara, "From supervisory control to PLC code: A way to speed-up constructive/virtual commissioning of manufacturing systems," *IFAC-PapersOnLine*, vol. 53, pp. 466–471 (2020).

[12] S. Konstantinov, F. Assad, B. Ahmad, D. A. Vera, and R. Harrison, "Virtual engineering and commissioning to support the lifecycle of a manufacturing assembly system," *Machines*, vol. 10, pp. 939 (2022).

[13] W. Chen, X. Wu, Z. Y. Chen, Y. Meng, R. Yuan Wang, and T. Chen, "Credit strategy of micro, small, and medium enterprises with known reputation risk: Evidence from a comprehensive evaluation model," *J. Inf. Optim. Sci.*, vol. 44, no. 4, pp. 637–652 (2023), doi: 10.47974/JIOS-1183.

[14] S. S. Patil and E. Rosemaro, "Blockchain-based smart contracts: Implementation and security considerations," *Int. J. Adv. Comput. Eng. Commun. Technol.*, vol. 12, no. 2, pp. 8–13 (Apr. 2025).

Received April, 2025