The EMBRACE Methodology for Human Reliability Analysis: An AI-Assisted Expert Evaluation of Its Support Software

Yochan Kim[a], Young-Suh Kim[b], Marialaura Delvecchio[c], Davide Zanardi[c], Buse Tezçi[c], Francesco Tesauri[c], Sun Yeong Choi[a], Jaewhan Kim[a], and Jinkyun Park[a]

[a] Korea Atomic Energy Research Institute, Daejeon, Republic of Korea, {yochankim, sychoi, jhkim4, kshpjk}@kaeri.re.kr

[b] FNC tech, Yongin, Republic of Korea, dudtj0512@fnctech.com

[c] RE:LAB S.r.l., Reggio Emilia, Italy, {marialaura.delvecchio, davide.zanardi, buse.tezci, francesco.tesauri}@re-lab.it

Abstract: The EMBRACE (Empirical data-Based crew Reliability Assessment and Cognitive Error analysis) method was developed to capture potential cognitive errors in procedure-following activities and failure possibilities due to time constraints. Grounded in an empirical foundation derived from full-scope simulator observations, the method aims to provide highly realistic human reliability assessment (HRA) results. Yet usability challenges remain that may induce human errors during the analysis process itself, derived from the inherent complexity of HRA workflows and the requirement to decompose procedures into primitive tasks. To address this issue, EMBRACE Support Software (ESS) was developed to assist analysts by automating the linkage between procedure sentences and primitive tasks and by visualizing the timeline and procedural flows of given human failure events. This study presents a usability evaluation of the ESS, employing a hybrid approach that combines AI-driven analysis through BANCONE®, an AI-powered Software-as-a-Service (SaaS) platform, with expert-driven evaluations, specifically heuristic evaluation and cognitive-visual risks analysis. Specialized AI agents were used to support both heuristic evaluation and cognitive-visual risk analysis, while human experts provided contextual interpretation and validation of the identified usability issues. The results demonstrate how combining AI support with human expertise can enhance the robustness of safety-related software, thus enabling HRA practitioners to perform more reliable risk assessments with reduced interface-induced biases.

1. INTRODUCTION

Traditional probabilistic safety assessment (PSA) frameworks for nuclear power plants have heavily relied on first-generation human reliability analysis (HRA) methodologies, such as the Technique for Human Error Rate Prediction (THERP) [1], the Accident Sequence Evaluation Program (ASEP) [2], and the Cause-Based Decision Tree (CBDT) [3]. However, when applied to modern digital control rooms, these legacy models exhibit structural limitations, as they depend highly on subjective analyst judgments and outdated analog data, leading to high uncertainty in quantifying cognitive errors.

To overcome these limitations, the Korea Atomic Energy Research Institute (KAERI) developed the Empirical data-Based crew Reliability Assessment and Cognitive Error analysis (EMBRACE) methodology [4,5]. EMBRACE provides a modern framework grounded in the Human Reliability data EXtraction (HuREX) database [6], which aggregates empirical operator performance data from full-scope digital simulators. By deconstructing complex emergency procedures into minimum units known as primitive tasks (PTs), EMBRACE ensures realistic and objective human error probability quantification.

Despite its theoretical advances, applying the EMBRACE framework manually requires a sizeable cognitive and computational burden, as analysts must deconstruct several procedure steps and calculate complex dependencies by hand. This may induce secondary human errors during the analysis process itself, threatening consistency and reproducibility. To resolve this, EMBRACE Support Software (ESS) was developed to computerize the complex calculation logic and guarantee the reliability, reproducibility, and traceability of analysis. However, since the ESS itself is a tool used to quantify human error, any usability weakness in its interface could paradoxically introduce the very kind of analyst-induced bias the methodology aims to control. Its interface quality hence is a safety-relevant attribute rather than a merely ergonomic one.

The primary objective of this study is to systematically evaluate the user interface (UI) architecture of the ESS and assess its practical viability through a comprehensive usability evaluation. An innovative methodology based on a hybrid human–machine interface (HMI) diagnostic approach was employed, combining human domain experts in heuristic and cognitive interface evaluation and an AI-driven analytics platform called BANCONE [7] developed by RE:LAB S.r.l. [8]. This combined approach enabled the identification of critical UI bottlenecks. Remedying these vulnerabilities will ensure that the ESS can be seamlessly and reliably deployed within global nuclear regulatory frameworks with minimized interface-induced biases.

2. THE EMBRACE THEORETICAL FRAMEWORK

2.1. Architecture Of Human Error Probability Quantification

Within the rigorous analytical structure of the EMBRACE framework, the total human error probability (HEP) for any defined human failure event (HFE) is not derived from a single monolithic estimate. Instead, it is synthesized from the probabilistic summation of two distinct, mathematically independent failure vectors: the failure probability of timely performance (FPtp) and the failure probability due to cognitive error (FPce) in procedural tasks. Operating under the rare event approximation hypothesis standard in probabilistic safety engineering, the total event probability is treated as the direct sum of these two constituent components. This allows analysts to independently assess whether an operator may fail due to a lack of time or due to a fundamental cognitive misunderstanding of the procedure.

2.2. Mathematical Modeling Of Timely Performance Failure (FPtp)

The FPtp component serves to quantify the explicit probability that the actual temporal duration required by the operating crew to execute a designated sequence of tasks will exceed the maximum physical and thermal-hydraulic time window permitted before a systemic core compromise or release event occurs. In the nomenclature of the framework, this is the relationship between the time required (Tr) and the time available (Ta). The calculation of Tr is not a simple estimate of a single action; rather, it comprehensively aggregates the sequential time consumed by cognitive cue recognition, procedural retrieval, team communication, sequential step execution, and the final physical implementation of the safety action.

The mathematical evaluation of this temporal failure is modeled utilizing a lognormal distribution equation. The lognormal distribution is particularly suited for human response times because the overall task duration results from the multiplicative composition of several sub-processes (cue recognition, procedure retrieval, communication, and execution), which, by the multiplicative form of the central limit theorem, naturally yields a lognormal distribution. This theoretical structure is consistent with the positive skewness empirically observed in human performance data, where a non-negligible fraction of response times extends well beyond the median due to characteristics of persons, environments, or tasks [9]. The equation is expressed as:

$$FPtp = 1 - \Phi\left[\frac{ln(Ta\ /\ Tr)}{\sigma}\right]. \tag{1}$$

In this formula, Φ represents the standard normal cumulative distribution function. Based on rigorous statistical modeling of the extensive HuREX database concerning operator response times under various emergency and transient conditions in APR1400 simulators, the standard deviation or shape parameter (σ) is fixed at a constant 0.3403 for baseline calculations [10]. This specific constant effectively normalizes the expected variance in human behavioral speed during critical incidents, grounding the theoretical formula directly in empirical reality.

2.3. Cognitive Error And Primitive Task Deconstruction (FPce)

Conversely, the FPce metric isolates the probability of operator failure stemming from cognitive misinterpretations, navigational disorientation, or execution errors during the sequential fulfillment of written or digital procedural steps. To quantify this, the EMBRACE methodology mandates the deconstruction of overarching, complex procedures into micro-level actions, defined as PTs. Based on a deep linguistic and ergonomic analysis of the procedure syntax, the framework categorizes all potential operator operations into 22 distinct PT types.

These 22 PTs are distributed across four broad cognitive domains that map to the standard information processing model of human cognition: information gathering, situation interpreting, response planning, and execution. In addition, each individual PT is intrinsically linked to a nominal primitive error probability (PEP), a statistical value drawn directly from the APR1400 HuREX empirical pool. By assigning specific probabilities to granular actions, such as verifying a discrete alarm state versus evaluating a continuous parameter trend, the framework achieves an unparalleled level of precision. Table 1 delineates the specific cognitive types and their baseline failure values.

Table 1: PT Types and Their PEPs

Cognitive Domain	Primitive Task Type	PEP
Information Gathering	Checking discrete state - Verifying alarm occurrence	1.10E-04
	Checking discrete state - Verifying the state of an indicator	1.10E-04
	Checking discrete state - Synthetically verifying information	1.51E-03
	Measuring parameter - Reading a simple value	1.10E-04
	Measuring parameter - Comparing parameters	1.10E-04
	Measuring parameter - Comparing in graph constraint	1.10E-04
	Measuring parameter - Comparing for abnormality	1.10E-04
	Measuring parameter - Evaluating a trend	3.08E-04
Situation Interpreting	Diagnosing	2.16E-03
	Identifying the overall status	2.16E-03
	Predicting	2.16E-03
Response Planning	Entering the succeeding step in the procedure	1.85E-04
	Transferring to another procedure	5.62E-03
	Transferring to another step in the same procedure	5.62E-03
	Directing information acquisition	6.22E-05
	Directing (simple/dynamic) manipulation	1.96E-03
	Directing external notification/request	1.85E-04
Execution	Manipulation - Simple (discrete) control	1.84E-03
	Manipulation - Simple (continuous) control	1.53E-02
	Manipulation - Dynamic manipulation	1.49E-02
	Notifying/requesting outside the main control room	1.84E-03

The baseline PEP values provided in Table 1 represent idealized probabilities extracted under controlled, nominal conditions. To adapt these baseline statistical probabilities to the dynamic, chaotic, or degraded contextual realities of a specific accident sequence, the EMBRACE framework applies a structured matrix of performance shaping factors (PSFs). The base PEP values are mathematically magnified based on plant-specific degradation of operational conditions, using established weight multipliers.

The EMBRACE methodology assigns varying impacts of PSFs for each important procedure step selected for the given HFE. Specifically, it applies different weights depending on whether the step is carried out for a cognitive “procedure transition” or a physical “equipment manipulation”. Table 2 outlines these factors and their corresponding weights.

Table 2: PSFs and Contextual Multipliers

Performance Shaping Factor (PSF)	Weight During Procedure Transition Step	Weight During Equipment Manipulation Step
Complexity of required task	3	3
Subjective stress	[High] 5, [Medium] 2.5	[High] 5, [Medium] 2.5
Complexity of human–machine interface	3	5
Procedure quality	5	5
Support function of computer-based procedure	2	3
Crew dynamics	1	1
Communication level	2	2
Training level	5	3
Inattentiveness	10	10

The FPce calculation integrates the sum of all PSF-weighted PEPs across the entire designated operational path. This raw sum is then mitigated by multiplying by the recovery failure probability (RFP), which explicitly acknowledges the likelihood that multi-person operational crews functioning in a main control room (MCR) will peer-check and self-correct minor cognitive errors before they manifest into systemic failure. The inclusion of RFP thereby provides a more realistic and balanced final error probability.

2.4. Multi-Factor Dependency Analysis

One limitation of earlier generations of HRA was their overly simplistic, static approach to inter-HFE dependency. Legacy systems primarily managed dependency through logic decision trees that forced analysts to assign broad, highly subjective categorical levels of dependence (e.g., zero, low, moderate, high, or complete) between HFEs. This approach often led to artificial probability cliffs, where a slight shift in analyst opinion could cause the calculated risk to fluctuate significantly.

The EMBRACE framework supersedes this conventional structure by deploying a sophisticated, multi-factor analytical equation designed to derive the exact, fractional conditional human error probability (HEP(B|A)) between a preceding failure event (HFE A) and a subsequent failure event (HFE B). The governing mathematical model is articulated as

$$HEP(B|A) = [TRI + \{PTS + CRD\} * RF] * CS + HEP(B) * ACE. \qquad (2)$$

This algebraic formula isolates and quantifies six independent mechanisms of operational dependency, considering the contextual relationship between two events.

First, the temporal resource insufficiency (TRI) variable, modeled on a lognormal distribution, mathematically captures the probability that the prolonged execution of HFE A starves the operator of the necessary time required to successfully address HFE B.

Second, procedure transition similarity (PTS) evaluates the mental model and common cause effect. By deploying a sequence alignment technique—specifically, a normalized adaptation of the Smith–Waterman algorithm—PTS quantifies the structural similarities in the procedural navigation paths of the two events, penalizing scenarios where operators might fail to notice situational divergences.

Third, cue recognition dependency (CRD) evaluates shared instrumentation. If the specific alarm or indicator cue prompting HFE B is physically identical to the cue for HFE A without introducing novel sensory stimuli, CRD defaults to a high-risk penalty value (0.5), acknowledging that the operator may dismiss the cue as part of the initial event; otherwise, it is nullified.

Fourth, the recovery factor (RF) incorporates the mathematical difference in available time frames. If the time margin for HFE B significantly exceeds the time margin for HFE A plus the time required for a complete procedural verification loop, RF mitigates the dependency penalty, reflecting the crew’s opportunity to reset their situational awareness.

Fifth, crew sameness (CS) operates as a binary variable indicating whether the exact same operating crew handles both sequences in the same shift. This fundamentally acknowledges the persistence of flawed shared mental models within a single team.

Finally, the additional contextual effect (ACE) acts as a conditional multiplier invoked exclusively when the failures within HFE A actively escalate the subjective stress or task complexity pertinent to HFE B, such as triggering a cascade of simultaneous alarms that flood the displays.

2.5. Parameter Uncertainty

Given the inherently probabilistic nature of the underlying empirical data, quantifying the parameter uncertainty of the final calculated HEP is meaningful for decision-making, in that it can ensure that risk margins are adequately conservative. The EMBRACE framework models parameter uncertainty through a lognormal distribution, establishing the statistical bounds by assigning an error factor (EF), which is the ratio between the 95th and 50th percentiles, or equivalently, the 50th and 5th percentiles of the distribution. These are as follows.

- HEP < 0.001, EF = 10
- 0.001 ≤ HEP < 0.006, EF = 7
- 0.006 ≤ HEP < 0.01, EF = 5
- 0.01 ≤ HEP < 0.03, EF = 4
- 0.03 ≤ HEP < 0.5, EF = 2
- 0.5 ≤ HEP, EF = 1

3. EMBRACE SUPPORT SOFTWARE (ESS) DEVELOPMENT

3.1. Three-Tier System Architecture and Database Integration

To support the processing of various datasets while maintaining calculation responsiveness, the ESS was constructed upon a standard three-tier system architecture. Such separation of concerns helps ensure that data queries do not overly constrain the user interface or calculation speeds.

The foundational Database Layer integrates four distinct environments: the Procedure Structure DB, which manages the logical flow and branching networks of MCR procedures; the Procedure Sentence DB, which contains the text of the procedures mapped to their corresponding PTs; the HuREX DB, which stores the empirical failure rates and time variance statistics; and the PSF/RFP DB, which handles the contextual multipliers and recovery logic. By combining these databases, the ESS provides a structured environment to support comprehensive analysis.

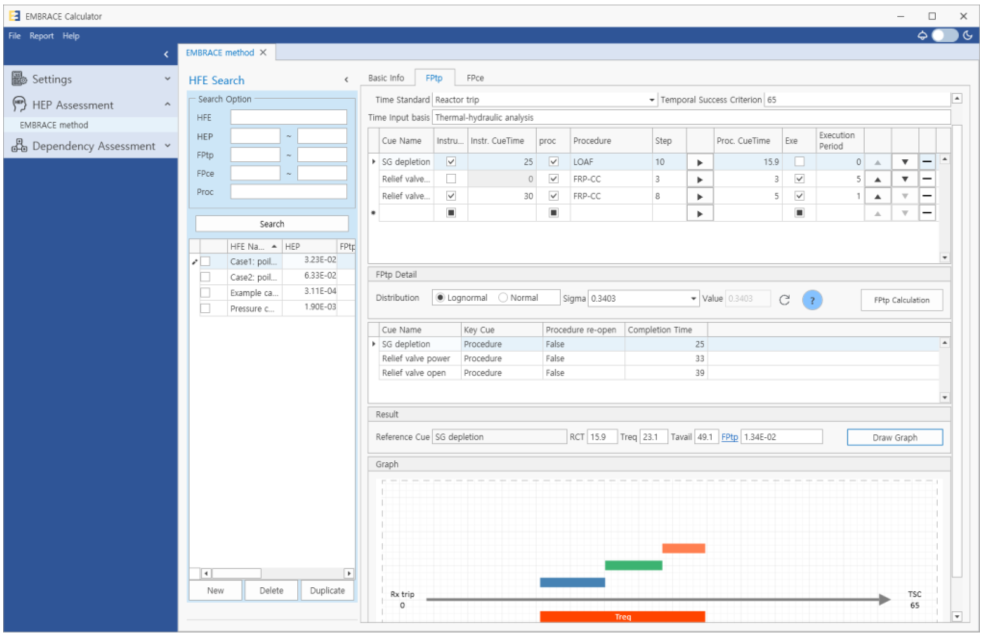
Above this layer sits the Computational Logic Layer, which acts as the primary processing engine of the software. It interprets user-defined procedure inputs, queries the underlying databases, and executes automated mapping algorithms. This layer handles the integrations required for the lognormal distributions of Ta and Tr, as well as the calculation of variables for the multi-factor dependency equations.

As the third tier, the user-facing Graphical User Interface (GUI) Layer provides an interactive dashboard. To assist in reducing the analyst’s cognitive load, the GUI reflects alterations in variables (such as altering a stress PSF from medium to high) in the final probability output. It also generates flowcharts and temporal graphs, offering the user visual oversight of the calculation process.

3.2. FPtp Calculation Module

The ESS includes computational modules corresponding to specific stages of the HRA workflow. Among them, the FPtp Calculation Module facilitates time-based reliability assessments via a four-step logic: scenario definition, timeline modeling, distribution definition, and probability calculation. When the analyst inputs fundamental timestamps, for instance, system cue initiation, expected procedure entry periods, and estimated manipulation durations, the internal algorithm calculates the Reference Cue Time. After the analyst inputs the cue time information and selects the shape parameters, the module computes FPtp. It also renders a timeline graph, plotting the boundaries between the time required and the time available along a chronological axis, providing visual documentation of temporal margins. Figure 1 shows a screenshot of the FPtp Calculation Module interface.

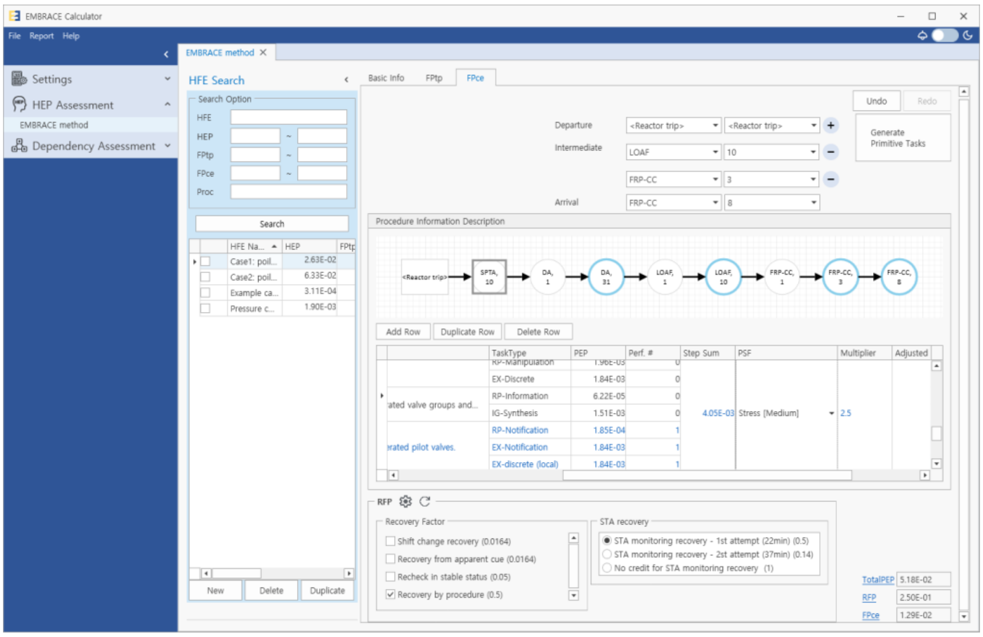
Figure 1: FPtp Calculation Interface



3.3. FPce Calculation Module

The FPce Calculation Module supports three analysis processes: procedure visualization, task analysis, and probability calculation (see Figure 2). When the analyst designates waypoints of the procedure steps to be conducted for the given HFE, the ESS queries the internal Procedure Structure DB. It utilizes a bidirectional breadth-first search (BBFS) algorithm to navigate the interconnected emergency procedures, and the resultant procedure flow is visualized with a directed acyclic graph. Following path generation, the system converts the procedure sentences into sets of PTs by cross-referencing the Procedure Sentence DB. This feature lessens the required manual effort, allowing the analyst to focus on reviewing the system's suggestions and adjusting the importance of the PTs and the number of PT performances. Once all PTs are selected, the system compiles the baseline PEP and applies the PSF weights and the RFP, streamlining the calculation of the final cognitive failure probability.

Figure 2: FPce Calculation Interface



3.4. Dual-Track Dependency Evaluation Modules

To maintain compatibility with previous HRA approaches and allow for comparative benchmarking, the software includes the Electric Power Research Institute (EPRI) dependency model. As shown in Figure 3, the ESS interface guides the analyst through a digital format of the EPRI decision tree, prompting categorical choices regarding intervening success, crew continuity, and cognitive demands, which culminate in 1 of 19 terminal nodes mapped to THERP equations.

Additionally, the EMBRACE Dependency Module operationalizes the six-variable equation detailed in Section 2.4. The interface provides the analyst with evaluation fields for the specific metrics (TRI, PTS, CRD, RF, CS, and ACE; see Figure 4), and once the situational parameters or similarity values are entered, the logic engine resolves the equation. This yields a fractional dependency probability as an alternative to the categorical outputs of the EPRI model.

Figure 3: Dependency Assessment Interface Based on the EPRI Method

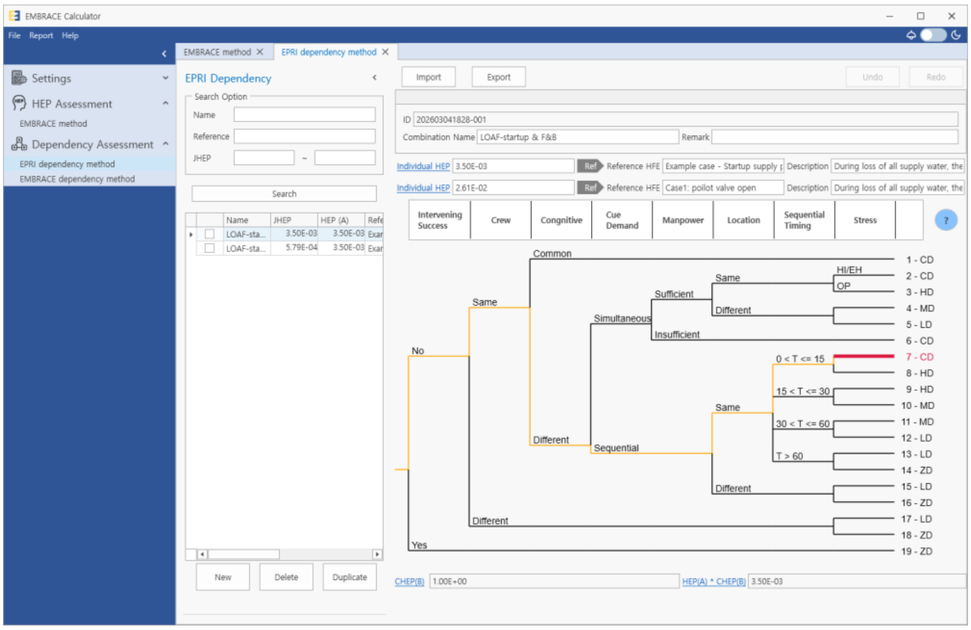
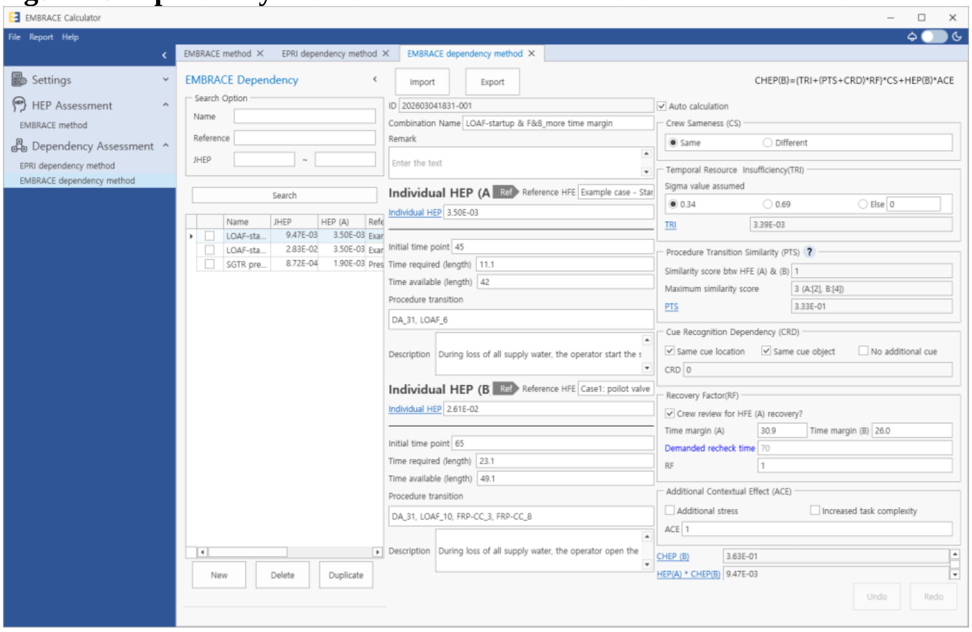


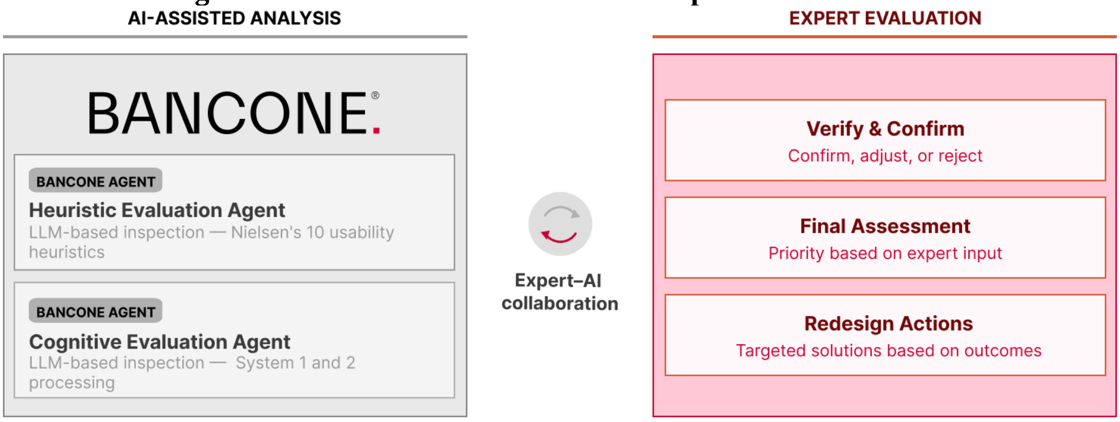
Figure 4: Dependency Assessment Interface Based on the EMBRACE Method



4. USABILITY EVALUATION AND SUGGESTED REDESIGN IMPROVEMENT USING BANCONE

This study examined the usability of the ESS through an AI-assisted expert evaluation approach based on collaboration between human experts and specialized AI agents. The objective was to identify interface weaknesses that could interfere with efficient and reliable use in HRA-related analyses, while also evaluating how AI-supported screening can complement expert judgment. In this study, BANCONE was used as the AI platform to support both heuristic and cognitive-visual risks analysis by assisting experts in identifying, structuring, and interpreting usability issues (see Figure 5).

Figure 5: BANCONE for EMBRACE: Expert-AI collaboration



The evaluation was conducted by a five-member expert team with backgrounds in human factors, HMI design and usability, human–computer interaction (HCI) research, and Human Factors & Ergonomics, working in collaboration with domain-instructed AI agents integrated within BANCONE. The team's role was to validate and interpret the AI-supported findings, ensuring that the analysis remained grounded in domain-specific expert judgment. The AI agents, trained on domain-specific knowledge, cooperated with the experts throughout the analysis process. Their role was to identify and structure the issues, while the experts validated and interpreted the AI-assisted findings. This configuration was designed to combine the broad issue-detection capability of AI with the contextual reasoning of experienced analysts, ensuring that the analysis remained grounded in context-specific judgment.

The results demonstrated that collaboration with specialized AI agents can enhance expert performance by improving analysis efficiency. In particular, a 43% reduction in the time required to complete the analysis was observed, while comparable diagnostic accuracy levels were maintained. No significant differences were found in heuristic-identification accuracy, suggesting that the tool supports analysts by directing attention toward relevant issues rather than replacing human expertise. Further in-depth analyses will be presented in a future publication.

4.1. Heuristic Evaluation

The methodology of the AI heuristic evaluation agent is grounded in Nielsen’s ten usability heuristics [11], a widely used inspection framework for identifying interface problems by comparing a system against established usability principles. The ten heuristics include (1) visibility of system status, (2) match between system and the real world, (3) user control and freedom, (4) consistency and standards, (5) error prevention, (6) recognition rather than recall, (7) flexibility and efficiency of use, (8) aesthetic and minimalist design, (9) help users recognize and recover from errors, and (10) help and documentation. These principles are especially useful for complex analytical software because they reveal whether the system supports efficient task execution, reduces unnecessary cognitive burden, and helps users recover from mistakes.

In this study, the Heuristic Evaluator Agent in BANCONE assisted experts in reviewing the ESS interface against these principles. The analysis focused on how the software supports task flow, feedback, labeling, navigation, and error handling during multi-step HRA calculations. Particular attention was given to interface elements that undermine users’ spatial orientation within the information architecture, thereby hindering their understanding of active states, calculated outputs, or workflow progress. The expert review also examined whether the interface terminology, controls, and layout were consistent with user expectations in a technical analysis environment. The heuristic evaluation highlighted a few recurring themes worth addressing in future iterations. The most consistently recurring observations were related to consistency between interface components and interaction conventions, followed by visibility of system status. Strengthening these aspects would help ensure that users receive clearer feedback regarding ongoing operations and action completion. Given that analytical workflows involve repeated cycles of data entry, condition definition, output generation, and result inspection, refinements in these areas would support user efficiency and confidence.

The analysis also revealed shortcomings in error prevention and recovery. Some actions were not sufficiently protected against accidental execution, and the interface did not always provide strong support for undoing or confirming critical operations. In addition, the system depended heavily on technical abbreviations and compact screens without enough contextual support, making it difficult for users to recognize functions quickly. These findings suggest that the interface would benefit from clearer labels, stronger feedback, and more explicit support for safe operation.

4.2. Cognitive and Visual Risk Evaluation

The Cognitive and Visual Risk Evaluation is a method developed by RE:LAB, designed to assess interface usability through a cognitive psychology lens. The method draws on the theoretical framework proposed by Kahneman [12], specifically the distinction between fast intuitive processing (System 1) and slow analytical reasoning (System 2), and operationalizes it into a structured evaluation protocol applicable to complex human-machine interfaces. A comprehensive methodological description of the approach will be presented in the aforementioned publication.

In this framework, interface problems are examined according to how they affect perception, comprehension, attention, memory load, and decision-making. This approach is particularly relevant for EMBRACE, because analysts must interpret procedural information, manage dependencies, and reason through multiple layers of calculation while avoiding misunderstanding or omission.

The AI agent in BANCONE supported experts by organizing usability findings into categories related to System 1 processing, System 2 processing, and System 1–System 2 conflict. System 1 issues refer to problems that interfere with immediate visual recognition, intuitive understanding, and perceptual clarity. System 2 issues refer to problems that increase the effort needed for conscious reasoning, state tracking, or step-by-step analysis. Conflict issues arise when the interface invites an intuitive action that is inconsistent with the actual system logic, forcing the user to detect and correct the mismatch through deliberate reasoning.

The cognitive review identified several aspects of the EMBRACE interface that contribute to the overall mental workload experienced during analysis. Some screens currently require users to infer meaning from information-dense layouts, specialized terminology, or distributed information elements. In other instances, the cues supporting awareness of the workflow progress could be made more explicit, which at present may lead users to track states and dependencies mentally. These conditions could increase the cognitive effort required for accurate comprehension and may slow analytical work, particularly in tasks involving frequent transitions between procedural structures, task interpretation, and output verification.

A particularly noteworthy observation concerned the presence of System 1–System 2 interaction patterns. In several cases, interface cues suggest one intuitive action or interpretation while the underlying logic calls for another, requiring users to engage in deliberate reasoning to reconcile the two. Such patterns can lead to rapid but incorrect judgments, especially under time pressure or when handling complex technical information. For a safety-oriented analysis tool, addressing these patterns is particularly valuable, as their resolution would directly support both efficiency and the reliability of the final assessment.

4.3. Interface Improvement

The AI–human hybrid methodology also provided guidance on several recommended interface improvements. First, the software should provide persistent feedback for critical actions, such as data loading, saving, analysis execution, and report generation. Second, destructive or irreversible actions should require confirmation and should be paired with recoverability mechanisms whenever possible. Third, the visual structure should be revised so that essential results and workflow states are easier to identify at a glance. In addition, the interface should reduce cognitive burden by clarifying technical terminology and improving the relationship between inputs, intermediate steps, and outputs. Tooltips, contextual help, and clearer labels would help users recognize functions without excessive memory effort. The layout should also better support the analytical flow by grouping related information more coherently and reducing visual clutter. These changes would improve both usability and trust in the software as an HRA support tool.

Overall, the AI-assisted heuristic and cognitive evaluations showed that the ESS has a strong computational foundation. However, its interface would benefit from further refinement to better support and fully leverage the system's analytical potential. The combined use of Nielsen-based heuristic inspection and cognitive psychology-based review, supported by AI agents, enabled a structured, multilayered evaluation of the system. This approach led to the identification of both surface-level usability and aesthetic issues, as well as deeper interaction and cognitive problems. These findings provide a clear basis for subsequent interface redesign aimed at improving clarity, safety, and analytical efficiency.

5. DISCUSSION AND CONCLUSION

5.1. Synthesis of Diagnostic Findings and System Remediation

The development of the ESS represents a practical advancement in the domain of nuclear-industry HRA. By translating the empirically driven EMBRACE methodology into a computational architecture, the ESS addresses several operational limitations found in legacy manual tools. The integration of the

HuREX database into the software's logic layer, combined with pathway-finding algorithms like BBFS, allows analysts to systematically map procedure sentences and cognitive error likelihood. Furthermore, the software's ability to auto-generate dynamic procedural flowcharts and timeline graphs enhances transparency, providing regulatory authorities with clear visual traceability of how specific human error values are derived.

However, the AI–human hybrid usability evaluation revealed that analytical capabilities must be paired with adequate interface design. In other words, results demonstrated that the initial version of the ESS needs stronger user safeguards. The absence of parameter validation logic, complexities in dependency inputs, and certain system bugs indicated that a cumbersome UI could inadvertently induce analyst errors during the assessment process. Resolving these UI bottlenecks is necessary for ensuring the reliability of the software's output as a regulatory tool. Based on these findings, the development team is actively implementing the recommended redesigns to improve the platform's stability and user experience.

5.2. Comparative Characteristics Against the EPRI HRA Calculator

It is desirable to compare the developed software with the EPRI HRA Calculator [13] to provide a clearer understanding of the ESS's characteristics. For HEP quantification, the current version of the ESS supports the EMBRACE method, whereas the EPRI calculator is a general-purpose tool supporting various methods such as ASEP, CBDT, HCR/ORE, THERP, and SPAR-H.

Regarding workflow and interface architecture, the ESS employs a hierarchical lateral navigation structure designed to facilitate a more sequential, workflow-oriented analysis. In contrast, the EPRI HRA Calculator primarily utilizes a top-menu and module-based navigation system. In terms of procedure mapping, the ESS directly integrates procedure logic and suggests PTs based on HuREX empirical data, minimizing the manual cross-referencing often required by the EPRI tool. For timeline analysis, both systems support the comparison of temporal data and provide time-series graph visualizations. Furthermore, the ESS incorporates an algorithm to determine reference cue times by accounting for multi-cue scenarios.

When evaluating parameter uncertainty, the ESS allows analysts to select rules from THERP (lognormal), SPAR-H (beta), or EMBRACE (lognormal), whereas the EPRI software applies methodology-specific rules to determine uncertainty bounds. For dependency analysis, the ESS offers two approaches: (1) EPRI's decision tree-based method, and (2) KAERI's recently developed multi-variable equation-based calculation. In contrast, the EPRI system relies on established decision trees but maintains a practical advantage in the North American market due to its ability to automatically generate HFE combinations through integrations with external PRA software (e.g., CAFTA/RiskSpectrum). While the EPRI HRA Calculator has already secured a large user base, the ESS is still in the software development stage. Therefore, to ensure the acceptance of the ESS, it is necessary to provide a more user-friendly interface and a sufficient user familiarization process.

5.3. Future Extensions and Implications

The work presented in this paper traces a coherent path from methodological foundation to practical deployment. The EMBRACE framework overcomes the limitations of first-generation HRA methods by grounding error quantification in empirical simulator data and decomposing procedures into PTs, but its analytical granularity imposes a substantial cognitive burden on analysts when applied manually. The ESS was developed to resolve this tension: by automating the decomposition of HFEs through objective algorithms and visualizing procedural flows and temporal margins, it reduces inter- and intra-analyst variability and fosters greater consistency in safety evaluations.

Computational rigor alone, however, does not guarantee reliable outcomes. The hybrid AI–human evaluation applied in this study, combining heuristic inspection and cognitive risk analysis through the BANCONE platform, identified interface weaknesses that could otherwise propagate into the final

HRA results, while achieving a 43% reduction in analysis time with precision comparable to expert-only evaluations. Since usability refinement is inherently iterative, AI-supported evaluation of this kind represents a particularly valuable instrument, and the ESS team intends to continue employing it as the interface evolves. Future studies will further investigate the methodological implications of this hybrid approach.

Finally, the modular architecture of the ESS, where the database layer is separated from the computational logic engine, provides the adaptability needed to follow the evolution of the nuclear industry. As advanced reactor designs such as SMRs and microreactors introduce highly automated control ecosystems, the ESS can be extended by updating its underlying databases to evaluate human reliability in these new operational contexts.

Acknowledgements

This work was partly supported by the Nuclear Safety Research Program through the Korea Foundation Of Nuclear Safety (KoFONS) of the Republic of Korea (No. 2106054).

References

[1] A. D. Swain and H. E. Guttman, "*Handbook of Human Reliability Analysis with Emphasis on Nuclear Power Plant Applications*," NUREG/CR-1278, U.S. NRC, (1983).
[2] A. D. Swain, "*Accident sequence evaluation program: Human reliability analysis procedure*," NUREG/CR-4772; SAND-86-1996, Sandia National Labs/U.S. NRC, (1987).
[3] EPRI, "*An approach to the analysis of operator actions in probabilistic risk assessment*," EPRI Report TR-100259, Electric Power Research Institute, (1992).
[4] Y. Kim and J. Kim, "*EMBRACEing HuREX Data: A Fundamental Approach to Develop a Plant-Specific Human Reliability Analysis Method*," Nuclear Engineering and Technology, (2026). (Under revision).
[5] Y. Kim, J. Kim, and S. Y. Choi, "*Handbook of Human Reliability Analysis Using the EMBRACE-APR1400 Method*," KAERI/TR-10997/2025, KAERI, (2025).
[6] Y. Kim, "*Estimate Comparison of Human Reliability Regression Models in Differently Classified Human Errors*," ASRAM2022 (Asian Symposium on Risk Assessment and Management 2022), Daejeon, Korea, (2022).
[7] RE:LAB, "BANCONE Official Website." [Online]. Available: https://bancone.ai/
[8] RE:LAB, "RE:LAB Official Website." [Online]. Available: https://www.re-lab.it/
[9] Y. Kim and W. S. Kim, "*Which Distribution Best Describes Human Performance Time?*," Transactions of the Korean Nuclear Society Autumn Meeting 2023, pp. 26-27, (2023).
[10] Y. Kim, J. Kim, J. Park, S. Y. Choi, S. Kim, W. Jung, H. E. Kim, and S. K. Shin, "*An HRA Method for Digital Main Control Rooms—Part I: Estimating the Failure Probability of Timely Performance*," KAERI/TR-7607/2019, KAERI, Daejeon, Korea, (2019).
[11] J. Nielsen, "*How to conduct a heuristic evaluation*," Nielsen Norman Group, vol. 1, no. 1, p. 8, (1995).
[12] D. Kahneman, Thinking, Fast and Slow. New York, NY, USA: Macmillan, (2011).
[13] EPRI, "*EPRI HRA Calculator 6.0b*," Electric Power Software Center, (2022).