Energy Engineering

Tech Science Press

https://doi.org/10.32604/ee.2026.077070

ARTICLE

Integrated Quality Control and Evaluation Framework for the R&D Process of Waterflooding Sandstone Reservoir Numerical Simulators

Zenghua Zhang[1,2], Yanchun Su[1,2], Zhijie Wei[1,2], Wensheng Zhou[1,2], Chen Liu[1,2], Engao Tang[1,2], Yanhong Wang[3], Shanshan Li[3] and Rui Zhang[3,*]

[1]State Key Lab of Offshore Oil & Gas Exploitation (abbr. SKLOOGE), Beijing, China
[2]CNOOC Research Institute Co. Ltd., Beijing, China
[3]Oilfield Production Research Institute, China Oilfield Services Limited, Tianjin, China

*Corresponding Author: Rui Zhang. Email: zhangruixiaoz@163.com

Received: 01 December 2025; Accepted: 09 March 2026; Published: 27 May 2026

ABSTRACT: Addressing the complexity of quality evaluation during the R&D phase of the waterflooding sandstone reservoir numerical simulator (OSIM), this study establishes a comprehensive assessment framework driven by client-side requirements. The novelty of this work lies in the integration of a client-driven 146-indicator hierarchy specifically tailored for iterative simulator R&D. First, a hierarchical software quality evaluation model is constructed, encompassing four primary dimensions and 146 secondary indicators. To mitigate the subjectivity inherent in traditional weight assignment, the Fuzzy Best-Worst Method (FBWM) is employed to determine indicator weights. Subsequently, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) is integrated to develop a composite ranking algorithm, coupled with specific protocols for acquiring indicator values were established using an 80-case model library. Finally, the proposed framework is validated using nine iterative versions of OSIM and the Commercial Software (CS). Results indicate that the final OSIM version achieved a superior overall ranking with a composite score of 0.962, surpassing the Commercial Software's score of 0.944, verifying the effectiveness of the method. The Root Mean Square Error (RMSE) for field water cut was controlled within 1.8%, demonstrating high simulation precision. This methodology exhibits strong universality, offering a scientific reference for the quality assessment of various reservoir simulators, including thermal and chemical flooding.

KEYWORDS: Reservoir numerical simulation; software quality evaluation; fuzzy best-worst method; TOPSIS method

1 Introduction

In the field of petroleum exploration and development, waterflooding reservoir numerical simulators are central tools for formulating and optimizing field development plans [1–3], where the quality of the software directly dictates the scientific validity of the development decisions. With the continuous iteration and advancement of numerical simulation technologies [4–8], software functional complexity has increased, and user demands for simulation accuracy and computational efficiency have become increasingly rigorous. However, current quality assessments for numerical simulators are often limited to single performance dimensions (such as calculation accuracy or runtime speed), lacking a comprehensive evaluation framework that covers all real-world user scenarios. Furthermore, traditional evaluation mechanisms primarily rely on internal developer testing [9,10], which frequently results in user feedback lagging behind software release, leading to a disconnect between the software's actual performance and user requirements.

There is an urgent need to bridge the gap between developer-centric testing and client-side practical requirements. Recently, Multi-Criteria Decision-Making (MCDM) methods have been widely applied to solve complex evaluation problems under uncertainty. For instance, Zadeh [11] utilized an extended BWM for smog mitigation analysis, while Kolour et al. [12] demonstrated the effectiveness of integrating Fuzzy BWM with DEMATEL for supplier selection in manufacturing. However, most existing MCDM models are applied to supply chain or urban planning. There is a lack of a systematic framework incorporating MCDM specifically for the dynamic, multi-version iteration process of reservoir simulation software.

To address this, this study, taking the development of the waterflooding sandstone reservoir numerical simulator OSIM1.0 as its context, establishes a comprehensive quality evaluation methodology based on the client perspective. Specifically, a quality evaluation model comprising four first-level indicators and 146 second-level indicators was constructed based on practical user needs. Subsequently, the Fuzzy Best-Worst Method [13,14] was employed to determine the weights of these indicators, thereby over-coming the subjectivity inherent in traditional weight assignment techniques. Finally, the Technique for Order Preference by Similarity to Ideal Solution [15–18] was integrated to achieve the quantitative ranking of software quality. This methodology not only provides a scientific basis for version optimization during the development of OSIM soft-ware and offers a practical path for user participation in software development but also possesses the universality to be directly applied to the comparative assessment of different types of simulation software.

2 Software Quality Assessment Framework and Method

To systematically evaluate the quality of reservoir numerical simulators during the iterative R&D process, this study adopts a multi-stage integrated evaluation framework. The overall research process, which encompasses four distinct phases—from indicator construction to final ranking—is illustrated in Fig. 1.

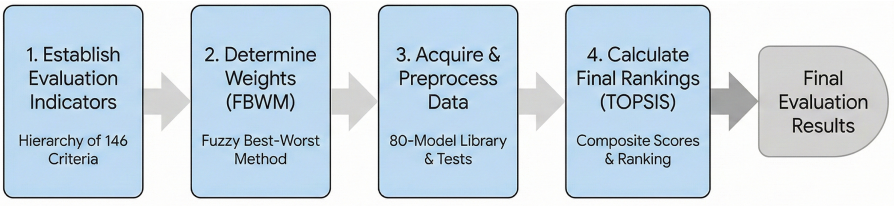


Figure 1: Research framework and evaluation process.

2.1 Construction of Quality Evaluation Indicator System

Targeting the user scenarios of waterflooding sandstone reservoir numerical simulators, this study filters out irrelevant indicators such as security and maintainability—often included in international software quality standards (e.g., ISO/IEC 25010) but not applicable to client-side evaluation—to establish a core quality assessment framework focused on practical application. This system is structured around four core dimensions: Functionality, Performance Efficiency, Compatibility, and Experience (Usability), and comprises 146 second-level indicators (illustrated in Fig. 2). The essence of each dimension is defined as follows:

A. Functionality Evaluation: The primary objective is to assess the software's ability to accurately execute its preset functions. This dimension covers external behavior manifestations, including input data processing, computational kernel accuracy, graphical interface usability, and the correctness and consistency of output results, comprehensively covering all modules such as pre-processing, simulation run, and data visualization.

B. Performance Efficiency Evaluation: As a critical indicator for numerical simulation software quality, this dimension includes two types of sub-indicators: accuracy and computational speed. Accuracy indicators (covering field, pattern, and single-well levels) determine the reliability of the simulation results, while computational speed indicators (including single-core and parallel processing speed) reflect the software's calculation efficiency.
C. Compatibility Evaluation: This aims to ensure stable operation and effective integration across multiple scenarios. The evaluation content includes operating system compatibility (e.g., Windows 7/10/11), third-party software data compatibility (focusing on the convenient import/export of models and calculation results), and stress adaptability (assessing the capacity for handling ultra-large models and processing abnormal data).
D. Experience (Usability) Evaluation: This is a user-centric dimension that assesses the overall user experience across four aspects: design style, interaction logic, operation flow, and business process alignment, ensuring the software interface is friendly, intuitive, and efficient to operate.

2.2 Design of the Comprehensive Evaluation Methodology

Based on the established software quality evaluation indicator system, this paper first employs the Fuzzy Best-Worst Method, incorporating triangular fuzzy numbers, to calculate the corresponding weights for all indicators in the model, thereby ad-dressing the ambiguity and uncertainty inherent in the assessment process. Subsequently, specific numerical acquisition methods for each indicator are proposed for both the test versions developed during the OSIM R&D process and the benchmarking software. Finally, the Technique for Order Preference by Similarity to Ideal Solution is utilized to achieve the quantitative ranking of the overall software quality.

2.2.1 Determination of Indicator Weights

The Fuzzy Best-Worst Method (FBWM) is a highly efficient multi-criteria decision-making (MCDM) approach. Compared to traditional MCDM methods, such as the Analytic Hierarchy Process (which requires $n(n-1)/2$ comparisons), FBWM significantly reduces the number of comparisons needed to only $2(n-1)$, thus improving decision-making efficiency. While newly developed methods like LBWA and FUCOM effectively reduce the number of pairwise comparisons to $n-1$, FBWM (requiring $2n-3$ comparisons) was selected for this study. The slight increase in comparisons provides a necessary redundancy to calculate the Consistency Ratio, ensuring the reliability of expert judgments in this highly subjective domain. Furthermore, by incorporating triangular fuzzy numbers, FBWM can better handle the fuzziness and uncertainty present in the decision-making process, leading to improved weight accuracy. The specific steps of the FBWM are as follows:

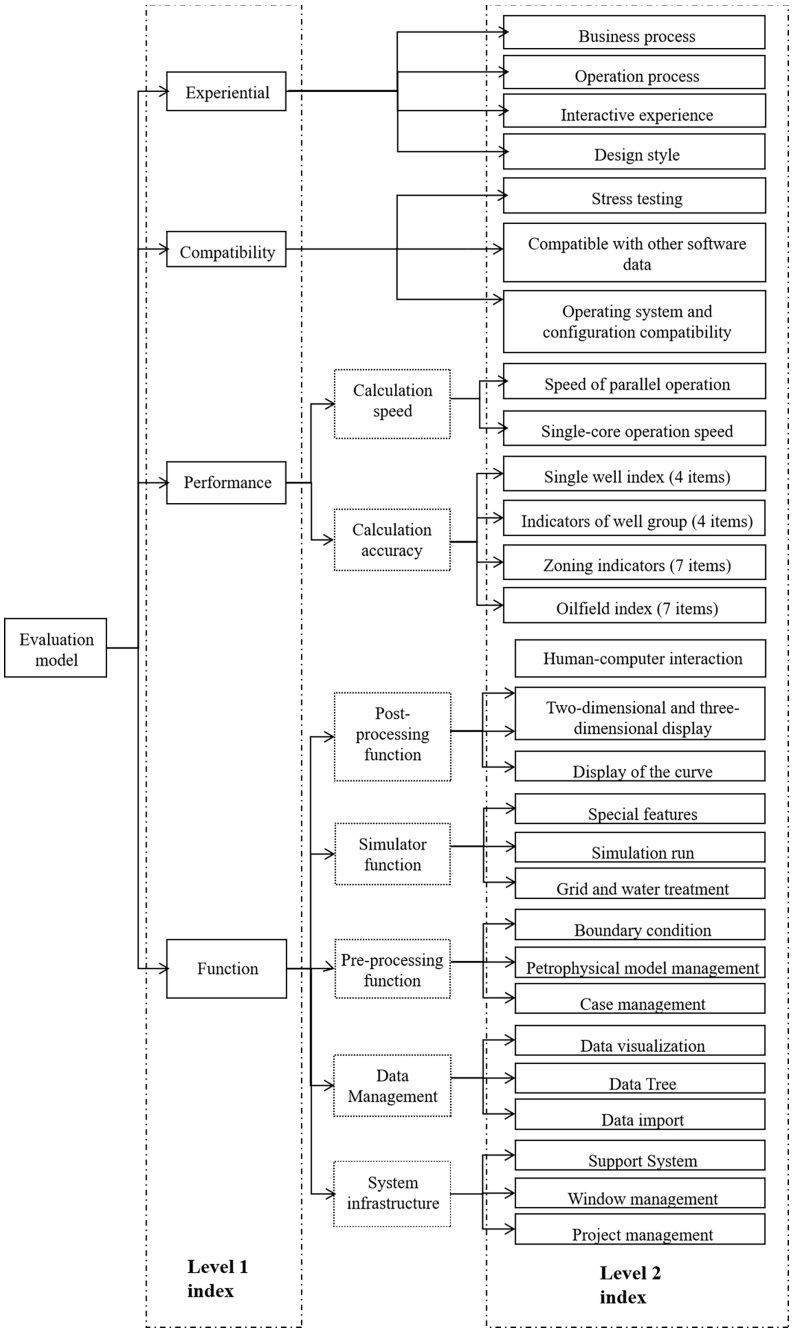


Figure 2: Schematic of the software quality assessment index system.

Step1: Key Criterion Identification: Experts identify the Best Criterion (most influential) and the Worst Criterion (least influential) from the indicator system.

Step2: Fuzzy Quantified Comparison: A quantified analysis is performed on the identified "Best Criterion" and "Worst Criterion". The comparison results are represented using triangular fuzzy numbers $A = (l, m, u)$, where A denotes the fuzzy importance of one criterion over another:

$$\tilde{A}_B = (\tilde{a}_{B1}, \tilde{a}_{B2}, \cdots\cdots, \tilde{a}_{Bn}) \tag{1}$$

$$\tilde{A}_W = (\tilde{a}_{W1}, \tilde{a}_{W2}, \cdots\cdots, \tilde{a}_{Wn}) \tag{2}$$

where $\tilde{a}_{Bi}$ and $\tilde{a}_{Wi}$ are triangular fuzzy numbers for the best and worst comparisons, $\tilde{a}_{Bi} = (l_{Bi}, m_{Bi}, u_{Bi})$ represents the lower, median, and upper limits of the best fuzzy number, and the worst fuzzy number is similar.

Step3: Weight Calculation: The optimal weight of the decision index is represented by nonlinear optimization, and the objective function is:

$$\min \max \left\{ \left| \frac{\tilde{\omega}_B}{\tilde{\omega}_i} - \tilde{a}_{Bi} \right|, \left| \frac{\tilde{\omega}_i}{\tilde{\omega}_W} - \tilde{a}_{Wi} \right| \right\} \tag{3}$$

where $\tilde{\omega}_B$, $\tilde{\omega}_i$, $\tilde{\omega}_W$ are fuzzy weights.

Using the Graded Mean Integration Representation (GMIR) for defuzzification, the optimization is transformed into a linear programming problem:

$$\begin{cases} \min \xi \\ \quad \left| \dfrac{\omega_B}{\omega_i} - a_{Bi} \right| \le \xi, i = 1, 2, \cdots n \\ s.t. \quad \left| \dfrac{\omega_i}{\omega_W} - a_{Wi} \right| \le \xi, i = 1, 2, \cdots n \\ \quad \sum_{i=1}^{n} \omega_i = 1 \\ \quad \omega_i \ge 0, i = 1, 2, \cdots n \end{cases} \tag{4}$$

In the above formula, ξ is the consistency index, ω_B, ω_i, ω_W are the weights after defuzzification, a_{Bi}, a_{Wi} are the comparison results after defuzzification.

2.2.2 Acquisition of Evaluation Indicator Values

Given that the established quality evaluation model encompasses both qualitative and quantitative indicators, along with a large number of criteria (146 secondary indicators), the selection and acquisition of reasonable numerical values for these indicators is crucial for the assessment methodology. For the 146 indicators, a normalization step is applied during preprocessing: quantitative metrics are scaled to a dimensionless range of [0, 1] relative to benchmarking software, while qualitative assessments are converted into numerical scores based on the established rubric to facilitate TOPSIS evaluation. To this end, all test versions developed during the OSIM R&D process were managed and sequentially numbered as OSIM-X_i (i = 1, 2, ……). To facilitate testing, evaluation, and ensure the comparability of results, a dedicated Waterflooding Model Library containing 80 actual field models from various sea areas was established. This library primarily covers different sandstone reservoir types and ensures comprehensive coverage of common practical application scenarios, including large models (effective grid count > 1 million) 17 cases, medium models (200 thousand < effective grid count ≤ 1 million) 25 cases, and small models (effective grid count ≤ 200 thousand) 38 cases. Based on the practical experience of the R&D team and users during software testing, practical methods for numerical acquisition were summarized for the primary and secondary indicators across Functionality, Performance, Compatibility, and User Experience:

a. Functionality Indicators: Evaluation is conducted using the 80-model Waterflooding Model Library. The final score for a functionality point is calculated based on the successful, improvement-required, or failure status (corresponding scores of 1, 0.5, and 0, respectively) during the simulation runs involving that function. If the number of models covering a specific function point is less than 10, corresponding models are randomly selected and supplemented with the required function point configuration to

ensure adequate test cases (≥10). Modified models are stored separately and not included in the main Model Library.

b. Performance Efficiency Indicators: Computation Speed: Under the same hardware configuration, the elapsed time for both single-core and parallel multi-core operations is statistically analyzed, and the ratio of time consumed relative to the benchmarking commercial software (e.g., ECLIPSE) is calculated.

Accuracy: Seven indicators are set for the Field level, four for the Pattern level, and four for the Single Well (Producer) level. The agreement rate (M) for each model is calculated using the Root Mean Square Error (RMSE). The final indicator value is taken as the average of the agreement rates across all participating models, used for benchmarking against commercial software accuracy levels.

c. Compatibility Indicators: Compatibility objectives include operating system compatibility (Windows 7/10/11), third-party software data compatibility (importing models and results), and stress adaptability (handling large models and abnormal data inputs). Since these objectives are not easily quantifiable, this paper adopts an error form recording method for evaluation. The indicator value is defined as the proportion of issue forms related to a specific type (e.g., OS compatibility) relative to the total number of recorded forms.

d. Usability/Experience Indicators: A user rating approach is employed. To ensure objectivity in the evaluation of qualitative metrics, a detailed scoring rubric (Table 1) is established based on user interaction frequency and interface consistency metrics. Testing groups assess the software across four dimensions—design style, interaction logic, operational flow, and business process alignment—with the average score obtained on a hundred-point scale. This assessment aims to identify deficiencies in user experience and guide subsequent optimization to ensure the final delivery of a user-friendly and efficient mature software product.

Table 1: Scoring rubric for experience and usability indicators.

Dimension	Level 2 Index	Scoring Description & Criteria	Score Range
Experience /Usability	Design Style	1. UI Consistency: Uniformity of icons, fonts, and button styles. 2. Color Harmony: Professionalism of the color scheme and paper texture background.	90–100: Excellent 75–89: Good 60–74: Fair <60: Poor
	Interaction Logic	1. Path Efficiency: Clicks required for a standard simulation setup. • <3 clicks: 100 pts. • 3–5 clicks: 80 pts. • >5 clicks: <60 pts. 2. Feedback: Presence of immediate response or progress bars.	100 (High Efficiency) 80 (Acceptable) <60 (Low Efficiency)
	Operation Process	1. Workflow Smoothness: Continuity from pre-processing to simulation run. 2. Fault Tolerance: Error prompts for abnormal data input before execution.	0–100 pts (Quantitative assessment)

(Continued)

Table 1 (continued)

Dimension	Level 2 Index	Scoring Description & Criteria	Score Range
	Business Alignment	1. Professionalism: Alignment with reservoir engineering habits. 2. Integration: Ease of importing/exporting multi-well group and field-level data.	0–100 pts (User consensus rating)

2.2.3 Comprehensive Software Ranking

Following the determination of indicator weights, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) is employed to achieve the quantitative ranking. Although recent studies like Mahmudova [17] have proposed advanced extensions using Linear Diophantine Fuzzy Z-numbers for urban transportation, our study adopts the classical TOPSIS integrated with Fuzzy BWM weights. This combination offers a balanced trade-off between computational complexity and the robustness required for software version benchmarking. TOPSIS is a widely used statistical analysis ranking method based on the core idea of utilizing a normalized raw decision matrix to identify the Positive Ideal Solution (A^+, the optimal solution) and the Negative Ideal Solution (A^-, the worst solution). The first step involves establishing the decision matrix and performing standardized weighting. The weighting coefficients used are the indicator weights calculated by the Fuzzy Best-Worst Method (FBWM), as shown below:

$$S = \begin{pmatrix} & I_1 & I_2 & \cdots & I_n \\ X_1 & x_{11} & x_{12} & \cdots & x_{1n} \\ X_2 & x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ X_m & x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \tag{5}$$

Establish a decision matrix, and setting an evaluation set $X = (X_1, X_2, \cdots, X_m)$; and an index set $I = (I_1, I_2, \cdots, I_n)$. The value of the evaluation object X_i to the index I_i is denoted as x_{ij}.

$$r_{ij} = \omega_j \times \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}} \, i = 1, 2, \cdots, m; j = 1, 2, \cdots, n \tag{6}$$

Then, determine the positive and negative ideal indicators. All indicators in this paper are set as maximum type indicators. If there is a minimum ideal indicator, it needs to be transformed first. The positive and negative ideal expressions are:

$$R^+ = (r_1^+, r_2^+, \cdots, r_m^+) = (\max\{r_{11}, r_{21}, \cdots, r_{n1}\}, \max\{r_{12}, r_{22}, \cdots, r_{n2}\}, \cdots, \max\{r_{1m}, r_{2m}, \cdots, r_{nm}\}) \tag{7}$$

$$R^- = (r_1^-, r_2^-, \cdots, r_m^-) = (\min\{r_{11}, r_{21}, \cdots, r_{n1}\}, \min\{r_{12}, r_{22}, \cdots, r_{n2}\}, \cdots, \min\{r_{1m}, r_{2m}, \cdots, r_{nm}\}) \tag{8}$$

On the basis, calculating the Euclidean distance between the *i*th evaluation object pair and the positive and negative ideals:

$$D_i^+ = \sqrt{\sum_{j=1}^{n} \left(r_j^+ - r_{ij}\right)^2} \tag{9}$$

$$D_i^- = \sqrt{\sum_{j=1}^{n} \left(r_j^- - r_{ij}\right)^2} \tag{10}$$

Finally, the final score of the ith evaluation object can be calculated and sorted.

$$P_i = \frac{D_i^-}{D_i^+ + D_i^-} \tag{11}$$

To address the rank reversal problem common in TOPSIS, a fixed set of alternatives (Versions X1–X9 plus the Commercial Software) was maintained throughout the evaluation, and absolute ideal solutions were defined based on industry standards.

3 Application Example and Analysis

This study selects nine test versions of the OSIM software (designated as OSIM-X1 to OSIM-X9) and the commercial software (abbreviated as CS) as evaluation subjects to validate the effectiveness and reliability of the comprehensive evaluation methodology proposed herein.

3.1 Determination of Evaluation Indicator Weights

Based on the quality evaluation model established in this study, five experts in reservoir numerical simulation were invited to qualitatively assess the weights of the indicators using the FBWM method. Due to the excessive number of secondary indicators (146), a hierarchical evaluation strategy was adopted to enhance efficiency and maintain the focus of expert judgment: weights were first determined for the primary indicators, and then separately for the secondary indicators belonging to each primary indicator. The final comprehensive weights for the secondary indicators were obtained by multiplying the respective primary and secondary weights. Taking the primary indicator evaluation as an example, each expert first identified and selected the Best Criterion and the Worst Criterion, performed a relative importance assessment, and utilized Triangular Fuzzy Numbers for fuzzification. The fuzzy pairwise comparison results for the primary indicators are presented in Table 2, and the resulting primary indicator weights, calculated based on each expert's judgment, are summarized in Table 3.

Table 2: Fuzzy pairwise comparison matrix for primary evaluation indicators.

Expert	Indicators		Function	Performance	Compatibility	Experience
Expert 1	Best indicator	Function	(1, 1, 1)	(1, 3/2, 2)	(9/2, 9/2, 9/2)	(2, 5/2, 3)
	Worst indicator	Compatibility	(9/2, 9/2, 9/2)	(7/2, 4, 9/2)	(1, 1, 1)	(3/2, 2, 5/2)
Expert 2	Best indicator	Function	(1, 1, 1)	(1, 1, 1)	(5/2, 3, 7/2)	(2, 5/2, 3)
	Worst indicator	Compatibility	(5/2, 3, 7/2)	(5/2, 3, 7/2)	(1, 1, 1)	(1, 3/2, 2)
Expert 3	Best indicator	Performance	(1, 3/2, 2)	(1, 1, 1)	(2, 5/2, 3)	(3/2, 2, 5/2)
	Worst indicator	Compatibility	(3/2, 2, 5/2)	(2, 5/2, 3)	(1, 1, 1)	(1, 3/2, 2)
Expert 4	Best indicator	Function	(1, 1, 1)	(1, 1, 1)	(9/2, 9/2, 9/2)	(1, 3/2, 2)
	Worst indicator	Compatibility	(9/2, 9/2, 9/2)	(7/2, 4, 9/2)	(1, 1, 1)	(5/2, 3, 7/2)
Expert 5	Best indicator	Function	(1, 1, 1)	(1, 3/2, 2)	(2, 5/2, 3)	(3/2, 2, 5/2)
	Worst indicator	Compatibility	(2, 5/2, 3)	(3/2, 2, 5/2)	(1, 1, 1)	(1, 1, 1)

Table 3: Final weights of primary evaluation indicators.

Indicators	Expert 1	Expert 2	Expert 3	Expert 4	Expert 5	Comprehensive Fuzzy Weight	Final Weight
Function	(0.399, 0.429, 0.409)	(0.357, 0.375, 0.389)	(0.286, 0.318, 0.341)	(0.399, 0.407, 0.409)	(0.179, 0.385, 0.4)	(0.324, 0.3828, 0.3896)	0.3655
Performance	(0.399, 0.381, 0.409)	(0.357, 0.375, 0.389)	(0.381, 0.397, 0.409)	(0.399, 0.402, 0.409)	(0.183, 0.308, 0.333)	(0.3438, 0.3726, 0.3898)	0.3687
Compatibility	(0.089, 0.095, 0.091)	(0.143, 0.125, 0.111)	(0.143, 0.127, 0.114)	(0.089, 0.09, 0.091)	(0.357, 0.154, 0.133)	(0.1642, 0.1182, 0.108)	0.1301
Experience	(0.114, 0.095, 0.091)	(0.143, 0.125, 0.111)	(0.191, 0.159, 0.136)	(0.114, 0.101, 0.091)	(0.282, 0.154, 0.133)	(0.1688, 0.1268, 0.1124)	0.136

Consistent with the primary indicator weighting procedure, the FBWM method was employed to assess the weights of the secondary indicators, yielding their initial weights. Subsequently, the initial weight of each secondary indicator was multiplied by the weight of its corresponding primary indicator to obtain the final comprehensive weight of the secondary indicator (Table 4).

Table 4: Partial comprehensive weights of secondary indicators.

Level 1 Index	Level 1 Index Weight	Level 2 Index	Level 2 Index Weight	Level 2 Index Comprehensive Weight
Function	0.3655	Simulation run	0.0311	0.01137
		Reservoir fluid	0.0239	0.00874
		Relative Penetration Settings	0.022	0.00804
		Control conditions	0.0216	0.00789
		.....		
Performance	0.3687	Oilfield water cut	0.0981	0.03617
		Oilfield pressure	0.0981	0.03617
		Parallel computing speed	0.0955	0.03521
		Oilfield daily oil production	0.0882	0.03252
		.....		
Compatibility	0.1301	Compatible with operating system and configuration	0.3	0.03903
		Compatible with other software data	0.444	0.05776
		Stress testing	0.222	0.02888
Experience	0.136	Design style	0.167	0.02271
		Interactive experience	0.333	0.04529
		Operation process	0.167	0.02271
		Business process	0.333	0.04529

3.2 Software Testing and Ranking Results

The indicator values for each software version were acquired according to the methodology described in Section 2.2.2. Utilizing the comprehensive weights listed in Table 4, the TOPSIS method is applied to calculate the overall composite score and ranking for each evaluated subject. The results, summarized in Table 5, demonstrate that the quality level of the OSIM software has continuously improved with each iteration. As illustrated in Fig. 3, the composite score index increased rapidly from version X1 (0.020) to X6 (0.683). After version X7, the software quality stabilized with scores consistently exceeding 0.8. This growth reflects the systematic resolution of functionality gaps and performance bottlenecks identified in earlier stages. From version X1 to X6, the composite scores increased rapidly, and after version X7, the software quality stabilized (with composite scores consistently exceeding 0.8). Notably, the OSIM-X9 version achieved a higher comprehensive evaluation score than the benchmarking software under this quality model. The analysis suggests that this superior performance is primarily due to several factors: the score jump from X6 to X7 corresponds to the refactoring of the parallel computing kernel (MPI optimization), which significantly improved the "Performance Efficiency" score. The increase in X8 is attributed to the adoption of the new Ribbon-style visualization module. The continued refinement of the software during development led to progressively complete functionality and significantly improved calculation accuracy, closely approaching that of the benchmarking software; simultaneously, a distinct advantage in parallel computation speed was realized; finally, the personalized design aspects of the software resulted in slightly better user experience scores compared to the benchmarking software.

Table 5: Evaluation results of OSIM test versions.

Object of Evaluation	Distance to a Positive Ideal Solution	Distance to Negative Ideal Solution	Composite Score Index	Ranking
OSIM-X1	0.044	0.001	0.020	10
OSIM-X2	0.033	0.011	0.249	9
OSIM-X3	0.026	0.018	0.407	8
OSIM-X4	0.030	0.021	0.411	7
OSIM-X5	0.019	0.027	0.588	6
OSIM-X6	0.014	0.030	0.683	5
OSIM-X7	0.007	0.038	0.847	4
OSIM-X8	0.003	0.042	0.925	3
OSIM-X9	0.002	0.044	0.962	1
CS	0.003	0.043	0.944	2

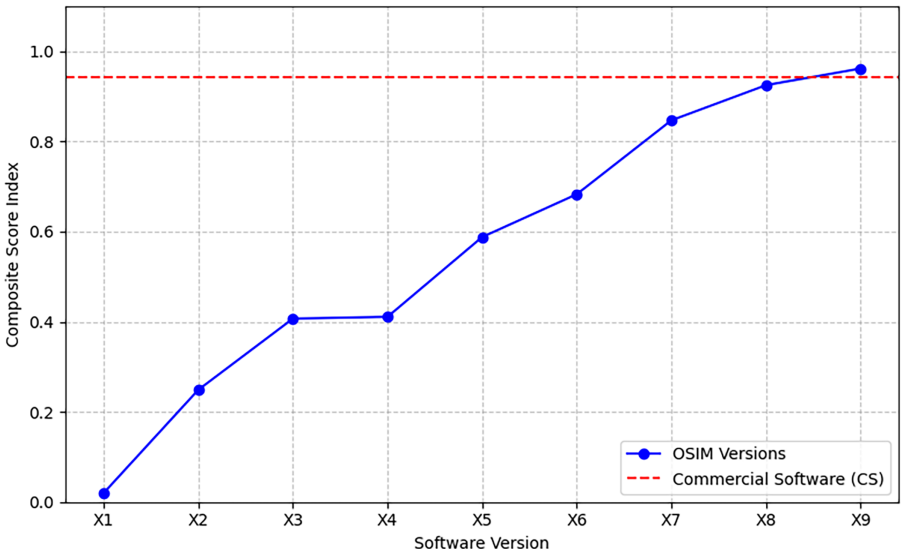


Figure 3: Iterative evolution of the composite score index for OSIM versions.

Table 6 presents a partial performance comparison between the OSIM-X9 version of the waterflooding sandstone reservoir numerical simulator and the commercial benchmarking software. Regarding computational efficiency, the average execution time across 80 models indicates that the OSIM-X9 version (3620 s) is significantly lower than that of CS (5239 s), highlighting its advantage in parallel computing. In terms of accuracy, the average Root Mean Square Error (RMSE) for critical engineering indicators such as field water cut, daily oil production, and average reservoir pressure were 1.8%, 0.82%, and 1.52%, respectively, demonstrating that the overall accuracy meets the requirements of practical reservoir engineering applications.

Table 6: Comparison between numerical simulation software and commercial software for offshore water drive sandstone oilfield.

Model	Time Consumption (s)		Accuracy Difference (Some Indicators) (RMSE%)						
	(OSIM-X9)	(CS)	Oilfield Water Cut	Oilfield Daily Oil Production	Oilfield Gas-Oil Ratio	Average Reservoir Pressure	Average Water Cut of Oil Wells	Daily Average Oil Production	Average Bottom-Hole Pressure
ST-1	7328	17,582	0.2	0.4	0.7	0.3	1.2	1.4	0.8
ST-2	16,607	7576	0.5	0	0.8	0.1	3.4	0	4.4
ST-3	1060	622	0.1	0.3	0.1	0.3	0	0	0.4
ST-4	14,440	9303	0.8	0	0.8	0.1	1.5	0	0.2
ST-5	1698	1481	0.8	0	50	0.1	1	0	0.8
ST-6	12,904	13,748	0	0.2	0	0	0	0.2	0
ST-7	3146	19,956	0.5	0.4	14.1	1.1	1.6	0.7	1.1
					.....				
ST-74	3257	2663	2.1	0	0.6	0	1.2	0	0.2
ST-75	4351	127	0.5	0	0.3	0.1	0.1	0	0
ST-76	4907	3847	2.8	0.1	1	0.1	2.8	0.1	1.8
ST-77	314	1366	1.9	0	0	0	3.5	0	0.2
ST-78	5747	10,751	0.1	0.1	1.5	0	0.2	0.2	0.4

(Continued)

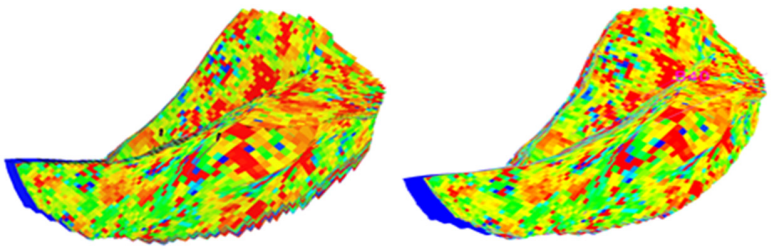
Table 6 (continued)

Model	Time Consumption (s)		Accuracy Difference (Some Indicators) (RMSE%)						
	(OSIM-X9)	(CS)	Oilfield Water Cut	Oilfield Daily Oil Production	Oilfield Gas-Oil Ratio	Average Reservoir Pressure	Average Water Cut of Oil Wells	Daily Average Oil Production	Average Bottom-Hole Pressure
ST-79	10,383	16,650	0.1	0.6	0	0.2	0.2	1.5	0.4
ST-80	3561	17,891	1.2	0.2	1.1	1.7	3.5	0	0.2
Average	3620	5239	1.21	0.66	4.11	0.68	1.8	0.82	1.52

3.3 Results Analysis

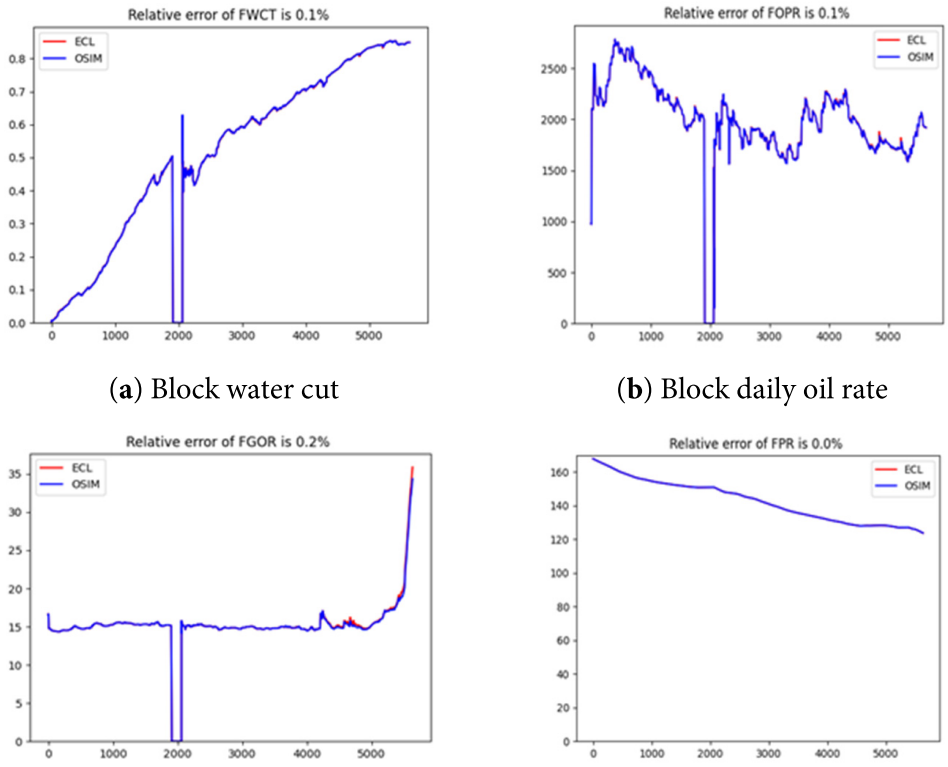
This section presents a detailed analysis of test results using a typical model block as an example. The model represents a fault block reservoir with an average effective thickness of 4.5 m and an *in-situ* oil viscosity of 8.5 mPa·s. Currently, one production well is deployed, with a daily oil production of 28 m^3, a comprehensive water cut of 84%, and an oil recovery factor of 21.6%. The grid dimensions are $67 * 77 * 275$, containing 116,000 active cells.

As shown in Fig. 4, the oil saturation field and pressure distribution field at the final time step simulated by both OSIM and CS are presented. Figs. 5 and 6 plot the prediction error curves for field and single-well indicators, respectively. It can be observed from the figures that the property fields simulated by OSIM-X9 are highly similar to the CS results. Furthermore, the relative prediction errors for key indicators such as water cut and daily oil production are all less than 0.1%, which thoroughly validates the reliability of the computational accuracy of the OSIM-X9 version. Regarding computational efficiency, the simulation time for OSIM was 0.09 h, compared to approximately 0.21 h for CS, indicating OSIM's significant advantage in efficiency.



(**a**) CS (**b**) OSIM

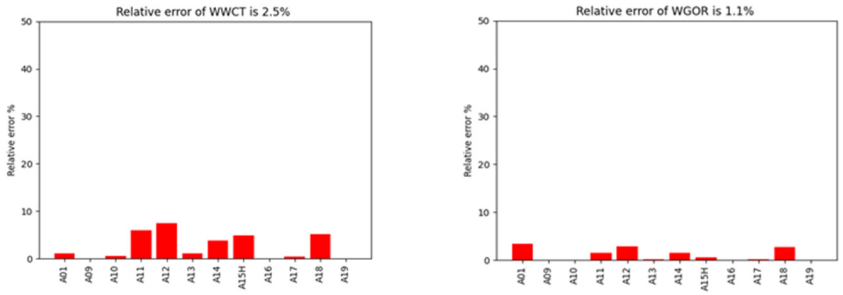
Figure 4: Oil saturation field of a model reservoir at the final time step.



(**a**) Block water cut (**b**) Block daily oil rate

(**c**) Gas-oil ratio (**d**) Formation pressure

Figure 5: Comparison of field performance curves for a representative model.



(**a**) Individual well water cut (**b**) Individual well GOR

Figure 6: Comparison of individual well performance curves for a representative model.

4 Discussion

Unlike traditional black-box testing, our model includes 146 indicators covering subjective dimensions like “Design Style” and “Interaction Logic”. The framework is designed for the process of R&D, allowing for dynamic weight adjustment and continuous feedback, whereas previous models were mostly for post-release evaluation.

4.1 Comparison with Previous Studies

Traditional software evaluation frameworks primarily emphasize stability, security, and technical code metrics. While these are essential, they often result in a "user feedback lag" where the software performs well in tests but fails to align with the complex business logic of petroleum engineers. Our findings confirm that by shifting the evaluation focus to the client side, OSIM-X9 achieved a higher comprehensive score (0.962) than the benchmarking commercial software (0.944). This discrepancy is largely due to the "Experience" and "Functionality Alignment" indicators, where OSIM-X9 provides a more intuitive workflow for offshore waterflooding scenarios compared to generalized commercial tools.

4.2 Implications for Future Research

The success of the proposed framework provides a scientific reference for evaluating a wider range of petroleum engineering software. Specifically, the methodology exhibits strong universality and can be directly extended to chemical flooding simulator by adjusting the secondary indicators to reflect specific physical processes like polymer shearing. Furthermore, future studies could explore the integration of machine learning algorithms to automate the acquisition of "Performance Efficiency" data, reducing the manual testing effort required for large-scale model libraries.

4.3 Limitations and Potential Impact

Despite its advantages, this study has certain limitations that should be noted:

a. Expert Subjectivity: The FBWM relies on the qualitative judgments of the expert panel. While defuzzification methods were used to enhance accuracy, the final indicator weights may still exhibit minor variations if a different panel is chosen.
b. Domain Specificity: The current indicator system is strictly optimized for waterflooding sandstone reservoirs. Direct application to naturally fractured reservoirs or unconventional shale reservoirs may result in biased scores because the physical mechanisms (e.g., dual-porosity effects) are not yet fully weighted in the Functionality dimension.
c. Impact on Results: These limitations imply that while the current ranking of OSIM versions is highly reliable for sandstone reservoirs, the absolute scores should be viewed as benchmarks within this specific geological context.

5 Conclusions

a. This paper establishes a client-side-based software quality evaluation model covering four primary indicators—Functionality, Performance, Compatibility, and Experience—and 146 secondary indicators. By integrating the FBWM method (to reduce weight subjectivity), the TOPSIS method (to achieve quantitative ranking), and differentiated indicator value acquisition protocols, a complete quality evaluation framework for numerical simulation software is formed, which provides a scientific basis for version optimization during the software development phase.
b. Case validation demonstrates that after nine versions of iteration, the final test version of the OSIM software (X9) achieved a superior comprehensive quality evaluation result compared to the benchmarking software. Moreover, the proposed methodology exhibits strong universality, making it directly applicable to the quality comparison of different waterflooding simulators and providing a valuable reference for the quality assessment of other types of reservoir simulators, such as thermal recovery and chemical flooding simulators. Future research will focus on extending this framework to Chemical Flooding (Polymer/Surfactant) simulators and integrating automated testing tools to feed data directly into the evaluation model.

Acknowledgement: Not applicable.

Funding Statement: This research was funded by the CNOOC's Major Science and Technology Projects during the 14th Five-Year Plan Period (Grant No. KJGG2021-0506).

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization: Zenghua Zhang and Rui Zhang; methodology: Zenghua Zhang and Yanchun Su; software: Zhijie Wei and Wensheng Zhou; validation: Chen Liu, Engao Tang and Yanhong Wang; formal analysis: Yanhong Wang; investigation: Zenghua Zhang; resources: Zenghua Zhang; data curation: Chen Liu; writing—original draft preparation: Zenghua Zhang; writing—review and editing: Rui Zhang and Shanshan Li; visualization: Zhijie Wei; supervision: Rui Zhang; project administration: Zenghua Zhang. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Aziz K, Settari A. Petroleum reservoir simulation. London, UK: Applied Science Publishers; 1979.
2. Peaceman DW. Fundamentals of numerical reservoir simulation. Amsterdam, The Netherlands: Elsevier; 1977, doi:10.1016/s0376-7361(08)x7013-x.
3. Han D. Reservoir simulation basics. Beijing, China: Petroleum Industry Press; 2015.
4. Zhao GZ, Lan YB, Kuang T, He X, Wang QZ, Li QN, et al. Research progress and application of high-precision reservoir numerical simulation technology. Daqing Pet Geol Dev. 2024;43(4):152–60. (In Chinese).
5. Zhang R, Wang T, Wang F, Zhu XC, Wang YH, Kou SY, et al. A time-varying flow model of reservoir permeability based on effective cumulative water phase flux. Pet Geol Recovery Effic. 2025;32(1):162–73. (In Chinese).
6. Lian PQ, Ji BY, Duan TZ, Jiang FG. CPU and GPU hybrid parallel numerical simulation for large-scale complex reservoirs. China Sci Online. 2020;15(5):537–41. (In Chinese).
7. Dogru AH, Fung LSK, Middya U, Al-Shaalan TM, Pita JA, HemanthKumar K, et al. A next-generation parallel reservoir simulator for giant reservoirs. In: Proceedings of the SPE/EAGE Reservoir Characterization & Simulation Conference; 2009 Oct 19–21; Abu Dhabi, United Arab Emirates. p. 170. doi:10.3997/2214-4609-pdb.170.spe119272.
8. Wang S, Chen Z. An efficient machine learning-based approach for history matching of reservoir simulation models. Fuel. 2023;332:126107.
9. Fu B. Software quality and testing. 2nd ed. Beijing, China: Tsinghua University Press; 2023. (In Chinese).
10. Pressman RS. Software engineering: a practitioner's approach. 9th ed. New York, NY, USA: McGraw Hill; 2019.
11. Zadeh LA. Fuzzy sets. Inf Control. 1965;8(3):338–53. doi:10.1016/S0019-9958(65)90241-X.
12. Kolour HR, Momayezi V, Momayezi F. Enhancing supplier selection in public manufacturing: a hybrid multi-criteria decision-making approach. Spec Decis Mak Appl. 2026;3(1):1–20. doi:10.31181/sdmap31202629.
13. Rezaei J. Best-worst multi-criteria decision-making method. Omega. 2015;53:49–57. doi:10.1016/j.omega.2014.11.009.
14. Guo S, Zhao H. Fuzzy best-worst multi-criteria decision-making method and its applications. Knowl Based Syst. 2017;121(5):23–31. doi:10.1016/j.knosys.2017.01.010.
15. Hwang CL, Yoon K. Multiple attribute decision making. Berlin/Heidelberg, Germany: Springer; 1981. doi:10.1007/978-3-642-48318-9.
16. Chen L, Wang YZ. Research on evaluation and decision-making method integrating entropy weight coefficient and TOPSIS. Control Decis. 2003;18(4):456–9. (In Chinese).

17. Mahmudova S. Application of the TOPSİS method to improve software efficiency and to optimize its management. Soft Comput. 2020;24(1):697–708. doi:10.1007/s00500-019-04549-4.
18. Anjum R, Mirza MU, Kausar N, Ali R. Decision-making framework for urban transportation using linear Diophantine fuzzy Z-numbers with Dombi aggregation, TOPSIS and VIKOR methods. Spec Oper Res. 2025:1–34. doi:10.31181/sor4155.