Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/200620

Evolution of Mathematical Models in Information Systems Project Management: A Systematic Review and Testable Framework

Umar Ammar Altahtooh

MIS Department, School of Business Administration, Taibah University, Madinah, Saudi Arabia

Received: 27 Sep. 2026, Revised: 12 Oct. 2026, Accepted: 29 Oct. 2026
Published online: 1 Nov. 2026

Abstract: This study examines how the mathematical models used to plan, estimate, monitor and optimize information systems projects have evolved, how they are structurally related, and under which project conditions each family of models adds measurable value. A systematic literature review was conducted in accordance with the updated Preferred Reporting Items for Systematic Reviews and Meta-Analyses statement, combining a primary Google Scholar search with Scopus, Web of Science and IEEE Xplore searches, dual independent screening, quality appraisal and citation snowballing; 142 studies were synthesized. The extracted models were grouped into six clusters and integrated, through decision theory, systems theory, contingency theory and operations research, into a formal analytical framework from which three testable propositions were derived. The propositions were tested with reproducible Monte Carlo experiments on 400 randomly generated project networks and 9,000 simulated cost trajectories. Classical network estimates were systematically optimistic: the bias was nil for single-path schedules but reached 8.6% with eight parallel work-streams, and it was driven by the interaction between parallelism and duration uncertainty rather than by uncertainty alone. Committing delivery dates from simulated completion distributions instead of classical network estimates reduced the expected cost of lateness and idle buffers by 19% on average, and by 86% relative to deterministic schedules, but the net benefit became positive only above identifiable thresholds of parallelism, volatility and modelling cost. Earned value cost forecasts lost accuracy in proportion to the volatility of cost efficiency, with mid-project mean absolute errors rising from 0.8% to 7.7%. The results indicate that model choice in information systems projects should be treated as a contingent cost–benefit decision: deterministic models suffice for serial and stable projects, whereas parallel and volatile projects justify simulation-based and adaptive hybrid models. The study converts a descriptive historical classification into an explicit, testable decision framework that researchers can validate empirically and practitioners can use to select quantitative tools.

Keywords: Decision theory, Earned value management, Information systems projects, Mathematical models

1 Introduction

Information systems (IS) projects combine intangible deliverables, evolving requirements and rapidly changing technology, and they remain among the most failure-prone investments organizations undertake; a substantial tail of IT projects incurs cost and schedule overruns far beyond those anticipated at approval [1]. Quantitative planning and control models are the principal means by which project managers attempt to make such uncertainty explicit and manageable.

The mathematics of modern project management originated in the late 1950s with the Critical Path Method (CPM) [2] and the Program Evaluation and Review Technique (PERT) [3], which represented a project as a network of dependent activities and introduced probabilistic activity-time estimates [4]. In the 1960s, earned value management (EVM) integrated scope, schedule and cost into a single performance-measurement system [5]. From the 1980s, software-specific parametric models such as the Constructive Cost Model (COCOMO) [6] and function point analysis [7] related effort to product size, while later decades added Monte Carlo simulation [8], resource-constrained scheduling [9], system dynamics [10], agile adaptations of earned value [11] and, most recently, machine-learning estimators [12].

Several reviews have examined individual model families: software cost estimation [13], scheduling under uncertainty [14], extensions of the resource-constrained

* Corresponding author e-mail: utahtooh@taibahu.edu.sa

© 2026 NSP
Natural Sciences Publishing Cor.

project scheduling problem [15], system dynamics in project management [16], earned value extensions [5] and data-driven software engineering [17]. Three limitations nevertheless persist. First, the literature is fragmented by model family, so that the formal relationships among families are rarely specified. Second, existing classifications are largely descriptive: they state what each model does but not when one model should be preferred over another. Third, the evolution of these models has seldom been explained by established theory, which leaves claims about their usefulness difficult to test.

This study addresses these limitations through the following problem formulation: for an IS project with given structural and contextual characteristics, which combination of mathematical models minimizes the expected loss of planning and control decisions, net of the cost of modelling? Three research questions follow. RQ1: how have the mathematical models used in IS project management evolved between 1957 and 2025, and which formal capabilities did each era add? RQ2: how are the resulting model clusters structurally and formally related? RQ3: under which project conditions does a more advanced model cluster yield a measurable decision benefit over its predecessors?

The study makes four contributions. It reports a reproducible systematic review conducted under the updated Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA 2020) statement; it grounds the review in decision theory, systems theory, contingency theory and operations research; it converts the descriptive clustering of models into a formal analytical framework with three testable propositions; and it tests these propositions with reproducible computational experiments that include an explicit cost–benefit analysis of model choice. The remainder of the paper is organized as follows. Section 2 describes the review protocol, the theoretical lens and the design of the computational experiments. Section 3 presents and discusses the historical evolution, the structural clusters, the integrative framework, the test results and their implications. Section 4 concludes.

2 Materials and Methods

2.1 Review design and protocol

A systematic literature review (SLR) was conducted following the PRISMA 2020 statement [18], which updates the original PRISMA guideline [19], together with the guidelines for systematic reviews in software engineering [20] and established recommendations for concept-centric reviews [21,22]. The protocol specified the research questions, information sources, search strings, eligibility criteria, quality-appraisal checklist, data-extraction form and synthesis procedures before screening began. The protocol was not registered in a public registry, because the principal registries accept only health-related reviews; the complete protocol and the PRISMA 2020 checklist are provided as supplementary material.

2.2 Information sources and search strategy

Google Scholar was the primary search system, and Scopus, Web of Science Core Collection and IEEE Xplore were searched to validate and extend coverage. The search string combined a model dimension and an application-context dimension with Boolean operators (Table 1) and was translated into the field syntax of each database (title, abstract and keyword fields in Scopus and Web of Science; metadata fields in IEEE Xplore). Because Google Scholar limits query length and displays a maximum of 1,000 results per query, the string was executed as a set of shorter sub-queries, each combining one model term with the context group, and results were screened in relevance order until two consecutive pages yielded no eligible record. The final searches were completed in 2025. Retrieved records were pooled across the four sources before de-duplication, and only the combined total (more than 2,800 records) was recorded. Backward and forward citation snowballing was then applied to all included studies until no new eligible study emerged [23].

2.3 Justification of Google Scholar and treatment of methodological bias

Google Scholar was chosen as the primary system because the topic spans seven decades and several disciplines. Much of the foundational operations research and software engineering literature from the 1950s to the 1980s consists of conference papers, technical reports and books that are poorly indexed in Scopus and Web of Science, and comparative studies show that Google Scholar retrieves substantially more of this material in engineering and computer science [24]. Google Scholar is, however, not suitable as the sole search system of a systematic review: its ranking algorithm is opaque, its Boolean functionality is limited, results are capped, and identical queries are not exactly reproducible over time [25,26]. Table 2 lists the resulting sources of bias and the mitigation applied to each. Two further safeguards are structural. The conclusions about model performance in Section 3.4 rest on controlled computational experiments rather than on outcomes reported in the reviewed studies, so they are not exposed to publication bias. In addition, a benchmark set of 20 seminal works identified independently of the search was used as a recall check, and all 20 were retrieved by the combined strategy.

© 2026 NSP
Natural Sciences Publishing Cor.

Table 1: Systematic search string matrix and Boolean logic

Query dimension	Target keywords and Boolean operators
Group 1: methodologies and models	"mathematical model*" OR "mathematical equation*" OR "mathematical formula*" OR "quantitative model*" OR "stochastic model*" OR "optimization model*" OR CPM OR PERT OR "earned value" OR COCOMO OR "critical path" OR "three-point estimation" OR "Monte Carlo"
Operator	AND
Group 2: application context	"information systems project*" OR "IS project*" OR "software project*" OR "IT project management"

Table 2: Sources of methodological bias and mitigation measures

Source of bias	Risk for this review	Mitigation applied
Indexing and coverage bias	Older or non-journal works missed by a single database	Four information sources; backward and forward snowballing
Ranking and reproducibility bias	Opaque Google Scholar relevance ranking; results not exactly repeatable	Exact strings, search year and pooled record count reported; stopping rule documented
Citation (popularity) bias	Highly cited studies ranked first	Forward snowballing from recent studies; Scopus, Web of Science and IEEE Xplore searches are not citation-ranked
Publication bias	Successful model applications over-reported	Performance claims tested by controlled simulation (Section 3.4), not by reported outcomes
Language bias	English-language dominance	No language filter at search stage; residual bias acknowledged (Section 3.7)
Selection bias	Subjective eligibility decisions	Dual independent screening; agreement measured with Cohen's kappa

2.4 Eligibility criteria

The inclusion and exclusion criteria in Table 3 were applied at both screening stages to reduce ambiguity and selection bias [22].

2.5 Selection process

The searches yielded more than 2,800 records. After approximately 1,150 duplicates were removed, the titles and abstracts of 1,650 records were screened and 1,330 were excluded as irrelevant. Of the 320 full texts assessed for eligibility, 178 were excluded because they did not meet the eligibility criteria in Table 3, leaving 142 studies for synthesis (Figure 1). Two reviewers screened all records independently; disagreements were resolved by discussion and, where necessary, by a third reviewer. Inter-rater agreement was high (Cohen's kappa = 0.87 [27]), which corresponds to almost perfect agreement [28].

2.6 Quality appraisal

Each included study was appraised with a checklist adapted from software engineering review protocols [20]. Four criteria were each scored 0 (not met), 1 (partly met) or 2 (fully met): clarity of the mathematical formulation; rigor of validation (empirical, simulation-based or analytical); relevance to IS projects; and scholarly impact. Studies scoring 6–8 were rated high quality, 4–5 medium and 0–3 low. Only high- and medium-quality studies were retained for detailed synthesis; low-quality studies were used solely to corroborate the historical chronology.

2.7 Data extraction and synthesis

A standardized form captured, for each study, the publication year and authors; the type of model or equation; the historical period and context of development; the key equations and assumptions (deterministic or stochastic, single or multi-project); the treatment of IS characteristics such as uncertainty, scope volatility and agile delivery; reported strengths, limitations and performance; and structural features such as variables, parameters and links to other models. Synthesis proceeded in three steps. First, models were grouped chronologically by era to identify emergence, adaptation and paradigm shifts. Second, thematic synthesis grouped models into six clusters by core function and mathematical foundation. Third, the representative models of each era were coded on eight binary formal capabilities (Section 3.1) to transform the qualitative chronology into a quantitative capability index. Because the models, outcome measures and contexts were heterogeneous, no meta-analysis or meta-regression was performed [29]. Ethical approval was not required for this secondary research.

2.8 Theoretical lens

Four theories were used to explain the evolution of the models and to derive testable relationships between

© 2026 NSP
Natural Sciences Publishing Cor.

Table 3: Inclusion and exclusion criteria

Inclusion criteria	Exclusion criteria
Peer-reviewed journal articles, conference proceedings, and highly cited books or book chapters	Purely qualitative studies without mathematical components
Studies explicitly addressing mathematical models, equations or quantitative techniques for project planning, estimation, scheduling, monitoring, control, risk analysis or optimization	Studies confined to non-IS domains (e.g., construction or manufacturing) unless they demonstrated transfer to IS contexts
Studies focused on, or with substantial discussion of, IS, software or IT projects	Duplicates, non-academic sources (e.g., blogs, white papers) and non-peer-reviewed works
Works providing historical context or comparative analysis of models	Studies published before 1950 or after the search cut-off date in 2025

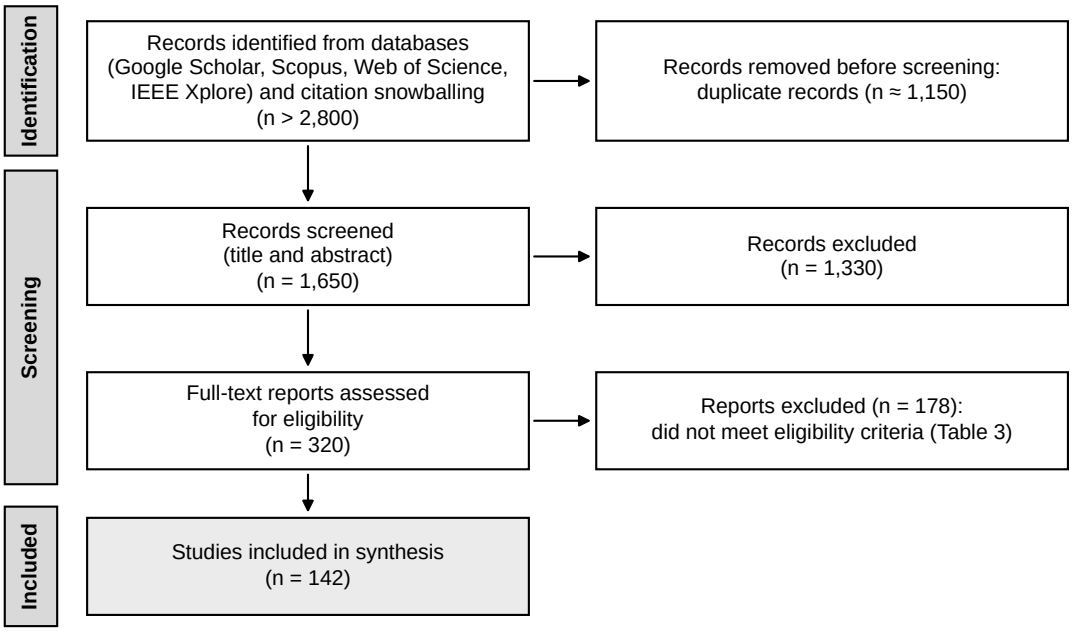


Fig. 1: PRISMA flow diagram of identification, screening, eligibility and inclusion

clusters (Table 4). Decision theory treats a model as an instrument for choosing among actions under uncertainty by comparing their expected utility or expected loss [30]; its Bayesian formulation provides the value of information as the yardstick for deciding whether additional analysis is worth its cost [31], while bounded rationality explains why simpler heuristics persist in practice [32]. Systems theory regards a project as a set of interacting elements whose aggregate behavior emerges from structure and feedback rather than from the properties of individual elements [33,34]; in software projects, rework cycles and Brooks's law are characteristic feedback effects [10,35]. Contingency theory holds that there is no single best way to organize and that effectiveness depends on the fit between an approach and its context [36,37]; project-management research has used it to explain why practices that succeed in one setting fail in another [38,39]. Operations research supplies the formal apparatus of network analysis, mathematical programming and stochastic modelling on which most of the reviewed models are built [40].

2.9 Design of the computational experiments

The propositions derived in Section 3.3 were tested with reproducible Monte Carlo experiments. Simulation was preferred to a meta-analysis of reported results because it allows the contingency variables to be varied under control, provides a known ground truth, and is immune to publication bias. All data are synthetic.

Experiments A and B used 400 randomly generated IS project networks. Each network consisted of a requirements activity, k parallel work-streams (k drawn

© 2026 NSP
Natural Sciences Publishing Cor.

Table 4: Theoretical lenses and their role in the integrative framework

Theory	Core premise	Role in this study	Clusters most informed	Framework construct
Decision theory	Choose the action minimizing expected loss; information has value only if it changes decisions	Defines model value and the cost–benefit adoption rule	C4, C5, C6	Eqs. (12)–(15); P2
Systems theory	System behavior emerges from structure and feedback	Explains merge bias and feedback-based recalibration	C1, C3, C4	Eqs. (9)–(11), (16)–(17); P1
Contingency theory	Effectiveness depends on fit with context	Specifies the moderators z that govern model choice	All clusters	Eq. (18); P1–P3 moderators
Operations research	Formal optimization and stochastic analysis	Supplies the equations linking the clusters	C1, C2, C4, C5	Eqs. (8)–(13)

uniformly from 1 to 8) of two to four activities each, and serial integration-testing and deployment activities. Most-likely durations were drawn uniformly between 5 and 20 days and the work-streams were balanced to similar totals, as planners normally attempt. Activity durations followed the beta-PERT distribution with optimistic time 0.8M and pessimistic time $(1 + \lambda)$M, where M is the most-likely time and $\lambda \in 0.5, 1.0, 1.5, 2.0$ represents requirements volatility. Because durations were sampled from the very distribution that PERT assumes, any difference between PERT and simulation can be attributed to network structure rather than to distributional misspecification. For each network, 10,000 replications were used for planning and an independent set of 10,000 replications for evaluation, which prevents in-sample optimism. In Experiment B a delivery date D was committed under a holding cost of $c_h = 1$ per day of unused buffer and a penalty of $c_p = 4$ per day of lateness (critical fractile 0.8), using three rules: the deterministic CPM duration, the PERT normal approximation and the Monte Carlo quantile.

Experiment C simulated 600 projects at each of five volatility levels $\sigma \in 0, 0.02, 0.04, 0.06, 0.08$, i.e., 3,000 projects evaluated at 25%, 50% and 75% completion (9,000 forecasts). Each project had a budget at completion of 100 spread over 20 periods. Cost efficiency followed a log-normal random walk around a project-specific initial level, with independent period-level reporting noise (standard deviation 0.05). At each checkpoint the CPI-based estimate at completion was compared with the realized final cost. A hybrid forecast was also computed by separating random-walk and noise variance from the observed period cost-performance indices and projecting the remaining cost by Monte Carlo simulation to form an 80% prediction interval.

Hypotheses were evaluated with the Wilcoxon signed-rank test [41], the Friedman test [42], the Kruskal–Wallis test [43] and ordinary least-squares regression with heteroscedasticity-consistent (HC3) standard errors [44], at a significance level of 0.05. The experiments were implemented in Python with NumPy and SciPy using a fixed random seed (20261018); the code and the generated data are provided as supplementary material so that every reported number can be reproduced.

3 Results and Discussion

3.1 Historical evolution and formal capabilities (RQ1)

Foundations (1950s–1960s). CPM was developed by DuPont and Remington Rand in 1957–1958 for plant maintenance projects [2], and PERT was developed in 1958 for the U.S. Navy Polaris program [3]. Both represent a project as a directed acyclic graph; forward and backward passes yield the earliest and latest start (ES, LS) and finish (EF, LF) times and the total float of each activity, and the longest path determines the minimum project duration. PERT added uncertainty through three time estimates, optimistic (O), most likely (M) and pessimistic (P), combined under a beta-distribution assumption:

$$TF = LS - ES = LF - EF \tag{1}$$

$$T_e = \frac{O + 4M + P}{6}, \qquad \sigma_e^2 = \left(\frac{P - O}{6}\right)^2 \tag{2}$$

These models gave early computer-system projects their first quantitative scheduling basis, but manual computation and the absence of mechanisms for requirements change limited their use in IS settings. Analytical work soon showed that PERT's project-level estimate is biased whenever paths merge [45,46,47], a point formalized in Section 3.3.

Integrated performance measurement (1960s–1970s). In 1967 the U.S. Department of Defense issued the Cost/Schedule Control Systems Criteria, the origin of EVM, which integrates scope, schedule and cost [5]. With planned value PV, earned value EV, actual cost AC and budget at completion BAC, the core relations are:

$$CV = EV - AC, \qquad SV = EV - PV \tag{3}$$

$$CPI = \frac{EV}{AC}, \qquad SPI = \frac{EV}{PV} \tag{4}$$

$$EAC = AC + \frac{BAC - EV}{CPI} = \frac{BAC}{CPI} \tag{5}$$

© 2026 NSP
Natural Sciences Publishing Cor.

Adoption in software development was slower than in defense and engineering because earned value is hard to define for intangible deliverables such as code or requirements documents; early IS applications were concentrated in large government contracts.

Software-specific parametric estimation (1970s–1980s). Boehm's COCOMO, calibrated on 63 projects [6], estimated effort in person-months from size in thousands of lines of code (KLOC):

$$PM = a \cdot (KLOC)^b \cdot \prod_j EM_j \tag{6}$$

where a and b depend on the development mode and the effort multipliers EM_j capture 15 cost drivers in Intermediate COCOMO and 17 in COCOMO II [48,49]. Function point analysis offered a technology-independent size measure based on inputs, outputs, inquiries, files and interfaces [7]. This era recognized that software projects differ from construction or manufacturing, although the models assumed waterfall processes and were sensitive to size-counting accuracy [50].

Stochastic, resource-constrained and multi-project models (1990s). The resource-constrained project scheduling problem was formulated with integer programming and solved heuristically because it is NP-hard [51,9]; Monte Carlo simulation replaced single-point PERT estimates with completion-time distributions [8]; system dynamics captured rework and staffing feedback in software projects [10]; and portfolio-level models recognized that IS organizations share resources across many initiatives.

Agile era and hybridization (2000s–2010s). The Agile Manifesto [52] challenged fixed baselines. Earned schedule corrected the tendency of SPI to converge to 1 at project end [53]; with C the number of complete periods for which EV $\geq PV_C$ and AT the actual time:

$$ES = C + \frac{EV - PV_C}{PV_{C+1} - PV_C}, \qquad SPI(t) = \frac{ES}{AT} \tag{7}$$

Story-point-based agile EVM [11], comparative studies of duration-forecasting methods [54] and hybrid simulation models of iterative delivery followed.

Data-driven and AI-enhanced models (2020s). Neural-network and ensemble estimators learn effort and risk relationships from historical project data [12]; agent-based and system dynamics simulations represent earned value and multi-project behavior [55,56]; fuzzy and uncertainty-aware network models address imprecise estimates [57]; and multi-criteria methods support group decisions [58].

To transform this chronology into a quantitative form, the representative models of each era were coded on eight formal capabilities, and a cumulative formal capability index (FCI) was computed as the share of capabilities available (Table 5). The index rises monotonically from 0.13 to 1.00, with the largest single increase in the 1990s, when resource constraints, multi-project scope and dynamic feedback entered the toolkit simultaneously. The index measures the capabilities available to practitioners in each era; it does not measure the frequency of use.

S = stochastic inputs; C = integrated cost–schedule measurement; D = statistical calibration on historical data; R = resource constraints; M = multi-project or portfolio scope; F = dynamic feedback; I = support for iterative delivery; L = automated learning and re-calibration; FCI = number of capabilities / 8.

3.2 Structural analysis of model clusters (RQ2)

Thematic synthesis of the 142 studies identified six clusters, summarized in Table 6. Cluster 1 (network scheduling) models the project as a directed acyclic graph and computes the critical path deterministically (CPM) or probabilistically (PERT) [57]; it captures dependencies well but ignores rework loops and scope change. Cluster 2 (parametric estimation) comprises the COCOMO family and function-point-based models, which are empirical regressions of effort on size and cost drivers and therefore depend on calibration data [50]. Cluster 3 (performance measurement) comprises EVM and its derivatives; it provides objective indices against a performance measurement baseline but presupposes a stable baseline, which is problematic for intangible and evolving IS deliverables [55]. Cluster 4 (probabilistic and simulation models) includes Monte Carlo simulation, system dynamics, Petri nets and queuing models, which yield distributions of outcomes and represent feedback such as Brooks's law [8,56,35]. Cluster 5 (optimization and resource allocation) applies linear, integer, goal and multi-objective programming, often complemented by multi-criteria methods such as the analytic hierarchy process [58,15]. Cluster 6 (hybrid and data-driven models) combines elements of the other clusters, for example agile earned value metrics with machine-learning estimators and simulation of delivery pipelines [17].

3.3 Integrative analytical framework and propositions

The clusters are not competing alternatives but successive layers of one decision system (Figure 2). Let the project be a directed acyclic graph with activity durations d = (d_1, …, d_n) and let Π denote the set of paths from start to finish. The parametric layer (C2) supplies the parameters θ of the duration distributions, the network layer (C1) maps durations to project completion, and the stochastic layer (C4) propagates uncertainty:

$$\theta_i = g(S_i, \beta), \qquad d_i \sim F(\cdot \mid \theta_i, \lambda) \tag{8}$$

© 2026 NSP
Natural Sciences Publishing Cor.

Table 5: Evolution of formal capabilities by era (✓ = capability available; cumulative)

Era	Paradigm	Representative models	S	C	D	R	M	F	I	L	FCI
1950s–60s	Deterministic and probabilistic scheduling	CPM, PERT	✓								0.13
1960s–70s	Integrated performance	Earned value criteria	✓	✓							0.25
1980s	Parametric estimation	COCOMO, function points	✓	✓	✓						0.38
1990s	Stochastic optimization	RCPSP, Monte Carlo, system dynamics, portfolio models	✓	✓	✓	✓	✓	✓			0.75
2000s–10s	Adaptive hybrids	Earned schedule, agile EVM, COCOMO II	✓	✓	✓	✓	✓	✓	✓		0.88
2020s	Intelligent systems	Machine learning, real-time analytics	✓	✓	✓	✓	✓	✓	✓	✓	1.00

Table 6: Structural comparison of the six model clusters

Cluster	Mathematical foundation	Key assumption	Principal output	Main IS limitation
C1 Network scheduling	Graph theory; longest path	Known dependencies; independent durations	Critical path, float	Ignores rework and scope change
C2 Parametric estimation	Non-linear regression	Size drives effort; stable calibration	Effort, cost, duration	Sensitive to size counting; dated drivers
C3 Performance measurement	Ratio indices against baseline	Stable baseline; measurable earned value	CPI, SPI, EAC	Volatile baselines; intangible output
C4 Stochastic and simulation	Probability; differential/difference equations	Specified input distributions or feedback structure	Outcome distributions, dynamics	Data and expertise intensive
C5 Optimization	Mathematical programming	Known objective and constraints	Optimal allocation or schedule	Scalability; input sensitivity
C6 Hybrid and data-driven	Combination; machine learning	Representative training data	Adaptive forecasts and decisions	Opacity; validation scarce

$$T(\mathbf{d}) = \max_{\pi \in \Pi} \sum_{i \in \pi} d_i \tag{9}$$

$$F_T(t) = P\left(T(\mathbf{d}) \leq t\right) \tag{10}$$

Because T(d) is the maximum of linear functions it is convex, and Jensen's inequality gives the merge bias of the classical network estimate:

$$\Delta = E\left[T(\mathbf{d})\right] - T\left(E[\mathbf{d}]\right) \geq 0 \tag{11}$$

Equality holds when a single path is critical with probability one, which occurs either when there are no parallel paths or when durations are deterministic [45,46]. In systems-theoretic terms, the bias is an emergent property: it arises only from the interaction between parallel structure and uncertainty, and neither factor alone produces it. The decision layer (C5) commits a delivery date D that minimizes expected loss, which is the newsvendor problem of operations research [59]:

$$L(D,T) = c_h\left(D-T\right)^+ + c_p\left(T-D\right)^+ \tag{12}$$

$$D^* = F_T^{-1}\left(\frac{c_p}{c_p + c_h}\right) \tag{13}$$

A model m produces a commitment D_m from its own estimate of F_T. Following the value-of-information logic of decision theory [31], the value of model m relative to a simpler model m′ and its net benefit are:

$$V(m \mid m') = E\left[L(D_{m'}, T)\right] - E\left[L(D_m, T)\right] \tag{14}$$

$$NB(m \mid z) = V(m \mid m', z) - \kappa_m \tag{15}$$

where κ_m is the cost of building and running the model and z is the vector of contingency variables. During execution the control layer (C3) tracks cost efficiency η_t, estimated by CPI_t. If efficiency drifts as a random walk with volatility σ, the CPI-based forecast of Equation (5) assumes that the drift stops at time t, so its error grows with σ and with the remaining horizon h:

$$\ln \eta_t = \ln \eta_{t-1} + \varepsilon_t, \quad \varepsilon_t \sim N(0, \sigma^2) \tag{16}$$

$$\mathrm{Var}\left[\ln \eta_{t+h} \mid \eta_t\right] = h\,\sigma^2 \tag{17}$$

Observed performance is fed back to re-estimate θ and F_T, which closes the loop and constitutes the hybrid layer (C6). Finally, contingency theory implies that the preferred model depends on context:

$$m^*(z) = \arg\max_m \; NB(m \mid z), \qquad z = (k, \lambda, \sigma, \ldots) \tag{18}$$

Three testable propositions follow. P1 (systems theory): the optimism of network-only PERT estimates increases with the number of parallel work-streams k, and duration uncertainty λ raises the bias only through its interaction with k. P2 (decision theory and economics): the expected commitment loss is ordered CPM > PERT > Monte Carlo, the value of stochastic modelling increases with k and λ, and its net benefit is positive only above a threshold set by the modelling cost κ. P3 (contingency theory): the error of the CPI-based estimate

© 2026 NSP
Natural Sciences Publishing Cor.

at completion increases with cost-efficiency volatility σ and decreases as the project approaches completion.

3.4 Computational test of the propositions (RQ3)

Table 7 reports the regression tests and Figure 3 shows the corresponding mean responses.

P1. Across the 400 networks the PERT estimate understated the simulated mean completion time by 5.5% on average (SD 3.6%; Wilcoxon signed-rank $p < 0.001$). The bias was nil for single-stream projects (−0.02%) and rose monotonically to 8.6% with eight parallel streams (Spearman $\rho = 0.71$, $p < 0.001$). Consistent with Equation (11), the main effect of λ was not significant ($p = 0.249$), whereas the interaction $\ln k \times \lambda$ was strongly significant ($t = 25.84$), and the model explained 93% of the variance (Table 7; Figure 3a). The practical consequence is substantial: a date that PERT promises with 80% confidence was met in only 60.5% of the evaluation replications on average, and in only 46.9% for eight-stream projects. P1 is supported.

P2. Mean expected commitment loss, in units of one day of buffer cost, was 66.9 for the deterministic CPM rule, 11.7 for the PERT rule and 9.4 for the Monte Carlo rule (Friedman $\chi^2 = 778.6$, $p < 0.001$; both pairwise Wilcoxon tests $p < 0.001$). Simulation therefore reduced expected loss by 19.4% relative to PERT and by 86.0% relative to CPM, and it was superior to PERT in 97.3% of networks. The value of stochastic modelling rose from approximately zero at $k = 1$ to 4.8 day-cost units at $k = 8$ and from 1.1 at $\lambda = 0.5$ to 3.4 at $\lambda = 2.0$; the $\ln k \times \lambda$ interaction was again the dominant term (Table 7; Figure 3b). The small negative main effect of λ refers to single-stream projects, for which the value is effectively zero. The net benefit depended on modelling cost: simulation was worthwhile in 97.3% of networks when $\kappa = 0$, in 75.3% when $\kappa = 0.5$, in 62.5% when $\kappa = 1$ and in 41.8% when $\kappa = 2$ day-cost units. Model choice is therefore a contingent economic decision rather than a matter of methodological sophistication. P2 is supported.

P3. At mid-project the mean absolute error of the CPI-based estimate at completion rose from 0.8% with stable cost efficiency to 2.2%, 4.1%, 5.9% and 7.7% as σ increased to 0.08 (Kruskal–Wallis $H = 1167.6$, $p < 0.001$). Errors were largest early in the project (12.0% at 25% completion for $\sigma = 0.08$) and smallest late (3.6% at 75%), and the negative $\sigma \times$ completion interaction confirms that volatility matters most when much work remains (Table 7; Figure 3c). The hybrid forecast, which adds a stochastic layer to EVM, produced 80% intervals that covered the realized cost in 83–95% of cases when $\sigma \leq 0.02$, but coverage fell to 61–76% at $\sigma = 0.08$ because volatility estimated from a few periods is itself uncertain. Hybrid models therefore improve the communication of forecast risk, but their calibration is also contingent on volatility and on the length of the performance history. P3 is supported.

3.5 Theoretical implications

The review moves the field from a descriptive taxonomy to a formally linked, theory-grounded system of models. First, the framework shows that the six clusters are connected by explicit equations: parametric models generate network parameters, network structure determines how uncertainty propagates, stochastic distributions determine optimal decisions, and earned value monitoring feeds back into estimation. Second, decision theory supplies a common metric, the reduction of expected decision loss, against which models from different eras can be compared. This reframes model evolution as an increase in decision value that becomes worthwhile as IS projects become more parallel and more volatile. Third, systems theory explains why classical network models fail in exactly those projects: merge bias is an emergent property of structure combined with uncertainty, and the experiments confirmed that uncertainty without parallelism produces none. Fourth, the results lend formal support to contingency views of project management [38,39]: no cluster dominates in all contexts, and the contingency variables k, λ and σ predict where each cluster adds value. The persistence of simple models in practice is consistent with bounded rationality [32] and, as Experiment B shows, can be economically rational when modelling costs are high relative to decision value. Finally, the evidence supports the shift from predictive to predictive–prescriptive analytics, in which models not only forecast outcomes but also prescribe commitments and interventions.

3.6 Practical and economic implications

The framework yields a selection rule that project management offices can apply. For serial, stable projects (one or two work-streams, low volatility), CPM or PERT schedules combined with standard EVM are sufficient, because the value of simulation is close to zero. For projects with several parallel work-streams, such as enterprise system implementations with concurrent configuration, data-migration and integration streams, delivery dates should be committed from simulated completion distributions, because PERT-based 80% commitments may in practice be met only about half of the time. Where cost efficiency is volatile, CPI-based forecasts should be accompanied by stochastic intervals and earlier re-baselining. The decision to adopt a more advanced model should be made explicitly with Equation (15). For example, if one day of reserved buffer costs USD 5,000, the mean value of simulation in Experiment B corresponds to about USD 11,300 per commitment

© 2026 NSP
Natural Sciences Publishing Cor.

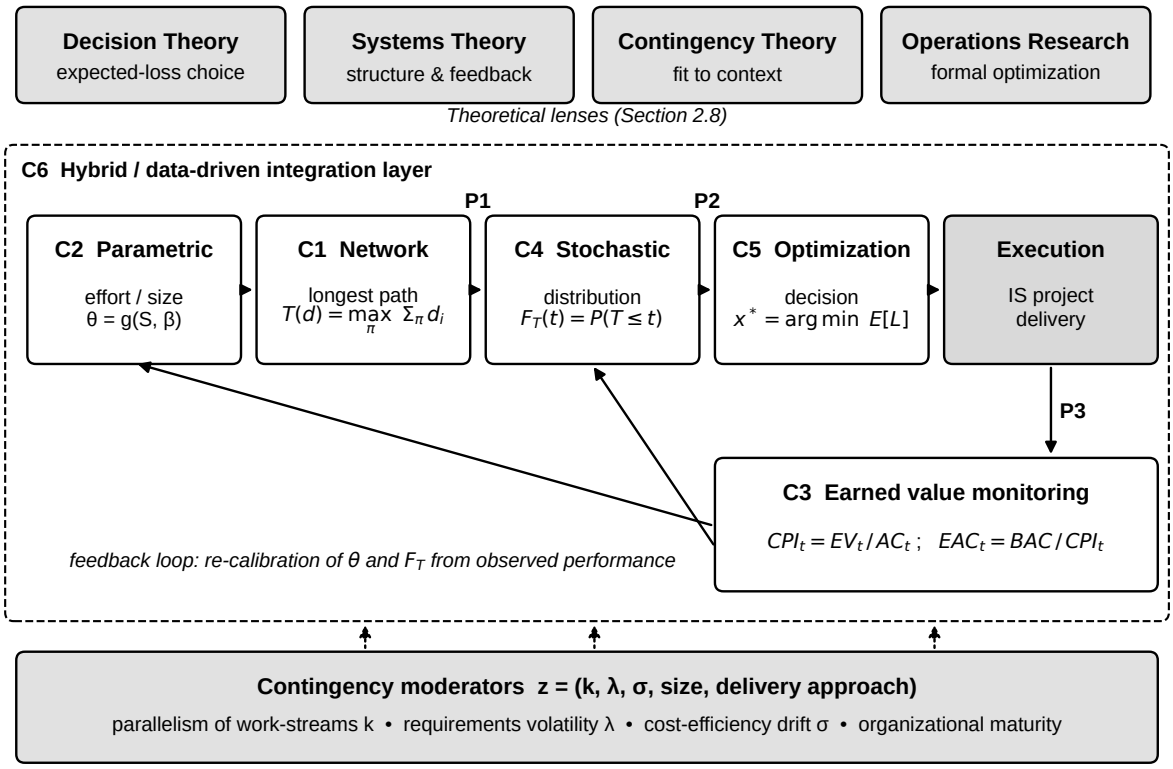


Fig. 2: Integrative analytical framework linking the six model clusters, the four theoretical lenses, the contingency moderators and the propositions P1–P3

Table 7: Regression tests of the propositions (OLS with HC3 robust standard errors)

Proposition (dependent variable)	Predictor	Coefficient	Robust SE	t	p
P1: PERT optimism bias, % (n = 400; R^2 = 0.93)	ln k	1.060	0.122	8.66	< 0.001
	λ	0.119	0.103	1.15	0.249
	ln k × λ	2.396	0.093	25.84	< 0.001
P2: value of stochastic model per 100 days (n = 400; R^2 = 0.70)	ln k	0.813	0.192	4.24	< 0.001
	λ	−0.449	0.205	−2.19	0.029
	ln k × λ	1.370	0.163	8.39	< 0.001
P3: absolute EAC error, % (n = 9,000; R^2 = 0.40)	σ	179.55	5.45	32.97	< 0.001
	Completion share	−1.471	0.202	−7.28	< 0.001
	σ × completion share	−185.31	8.33	−22.24	< 0.001

decision, which justifies simulation whenever building and running the model costs less. Beyond model selection, training for IS project managers should cover the assumptions and failure conditions of the equations, not only their computation, and sponsors should require probabilistic rather than single-point reporting of completion dates and costs.

3.7 Limitations and directions for future research

Several limitations should be noted. Despite the mitigation measures in Table 2, the reliance on Google Scholar as the primary search system limits exact reproducibility, and grey literature, proprietary industry reports and non-English studies remain under-represented. The clustering and the capability coding in Table 5 involve judgement, and alternative taxonomies could yield different insights. The computational experiments use stylized networks and cost processes; they establish internal validity and the logical consistency of the framework, but not the magnitude of effects in real projects. The parameters of the loss function, the volatility ranges and the network structures were chosen to be plausible rather than estimated, and literature published after the 2025 cut-off is not included.

© 2026 NSP
Natural Sciences Publishing Cor.

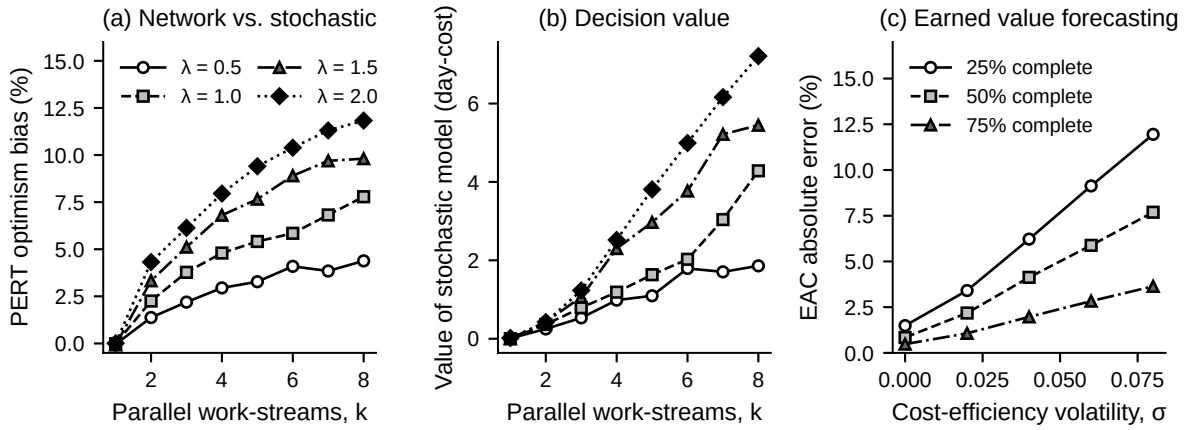


Fig. 3: Mean responses in the computational experiments: (a) PERT optimism bias by number of parallel work-streams and volatility; (b) value of stochastic modelling for deadline commitment; (c) absolute error of the CPI-based estimate at completion by volatility and completion stage

In addition, record counts were not retained separately for each information source, and reasons for full-text exclusion were not recorded individually, which limits exact replication of the identification and eligibility stages.

Future research should, first, test P1–P3 empirically with real project data, for example from industry benchmarking repositories, using multilevel models that relate forecast errors to project structure and volatility. Second, it should develop hybrid models that recalibrate classical equations with real-time project telemetry and explainable machine learning, and should measure their net benefit with Equation (15). Third, mathematical structures are needed for scaled agile frameworks and DevOps pipelines that link value delivery and technical debt to earned value. Fourth, multi-agent and fuzzy-stochastic models could integrate human factors such as cognitive bias and team dynamics. Fifth, optimization frameworks must incorporate environmental, social and governance goals along with time, cost and quality considerations. In the end, there are no lightweight models available for small and medium-sized companies and digital start-ups.

4 Conclusions

This study traced the evolution of mathematical models in IS project management from the deterministic networks of the 1950s to today's adaptive and data-driven hybrids, and organized them into six formally linked clusters. By grounding the review in decision theory, systems theory, contingency theory and operations research, it converted a descriptive classification into an analytical framework with three testable propositions. Reproducible experiments supported all three: classical network estimates become increasingly optimistic as parallelism and uncertainty interact; simulation-based commitments reduce expected decision loss, but only above identifiable thresholds of complexity and modelling cost; and earned value forecasts degrade in proportion to cost-efficiency volatility. No single model is universally superior. The appropriate combination of models depends on project structure, volatility and the economics of modelling, and the framework offers both a basis for empirical validation by researchers and a transparent selection rule for practitioners.

Acknowledgments

The author thanks the anonymous reviewer for constructive comments that substantially improved the manuscript.

References

[1] B. Flyvbjerg and A. Budzier, Why your IT project may be riskier than you think, Harv. Bus. Rev. **89**, 23-25 (2011).
[2] J. E. Kelley and M. R. Walker, Critical-path planning and scheduling, in Proc. Eastern Joint Computer Conference, 160-173, (1959).
[3] D. G. Malcolm, J. H. Roseboom, C. E. Clark and W. Fazar, Application of a technique for research and development program evaluation, Oper. Res. **7**, 646-669 (1959).
[4] M. Pratyusha and K. Ranjan, A comparative study on critical path method and project evaluation and review technique in construction under uncertain environment, in Data-Driven Modelling with Fuzzy Sets, CRC Press, Boca Raton: 150-164, (2024).

© 2026 NSP
Natural Sciences Publishing Cor.

[5] A. Nizam and A. Elshannaway, Review of earned value management (EVM) methodology, its limitations, and applicable extensions, J. Manag. Eng. Integr. **12**, 59-70 (2019).
[6] B. W. Boehm, Software Engineering Economics, Prentice Hall, Englewood Cliffs, (1981).
[7] A. J. Albrecht, Measuring application development productivity, in Proc. Joint SHARE/GUIDE/IBM Application Development Symposium, 83-92, (1979).
[8] K. A. Kirytopoulos, V. P. Leopoulos and V. K. Diamantas, PERT vs. Monte Carlo simulation along with the suitable distribution effect, Int. J. Proj. Organ. Manag. **1**, 24-46 (2008).
[9] C. Artigues, S. Demassey and E. Néron (Eds.), Resource-Constrained Project Scheduling: Models, Algorithms, Extensions and Applications, John Wiley & Sons, London, (2013).
[10] T. Abdel-Hamid and S. E. Madnick, Software Project Dynamics: An Integrated Approach, Prentice Hall, Englewood Cliffs, (1991).
[11] T. Sulaiman, B. Barton and T. Blackburn, AgileEVM – earned value management in Scrum projects, in Proc. AGILE 2006 Conference, IEEE: 7-16, (2006).
[12] B. Şengüneş and N. Öztürk, An artificial neural network model for project effort estimation, Systems **11**, 91 (2023).
[13] M. Jørgensen and M. Shepperd, A systematic review of software development cost estimation studies, IEEE Trans. Softw. Eng. **33**, 33-53 (2007).
[14] W. Herroelen and R. Leus, Project scheduling under uncertainty: survey and research potentials, Eur. J. Oper. Res. **165**, 289-306 (2005).
[15] S. Hartmann and D. Briskorn, A survey of variants and extensions of the resource-constrained project scheduling problem, Eur. J. Oper. Res. **207**, 1-14 (2010).
[16] J. M. Lyneis and D. N. Ford, System dynamics applied to project management: a survey, assessment, and directions for future research, Syst. Dyn. Rev. **23**, 157-189 (2007).
[17] A. Yalçıner, A. Dikici and E. Gökalp, Data-driven software engineering: a systematic literature review, in Proc. European Conference on Software Process Improvement, Springer, Cham: 19-32, (2024).
[18] M. J. Page, J. E. McKenzie, P. M. Bossuyt, I. Boutron, T. C. Hoffmann, C. D. Mulrow et al., The PRISMA 2020 statement: an updated guideline for reporting systematic reviews, BMJ **372**, n71 (2021).
[19] D. Moher, A. Liberati, J. Tetzlaff and D. G. Altman, Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement, BMJ **339**, b2535 (2009).
[20] B. Kitchenham and S. Charters, Guidelines for performing systematic literature reviews in software engineering, Technical Report EBSE-2007-01, Keele University and Durham University, UK, (2007).
[21] J. Webster and R. T. Watson, Analyzing the past to prepare for the future: writing a literature review, MIS Q. **26**, xiii-xxiii (2002).
[22] O. Zawacki-Richter, M. Kerres, S. Bedenlier, M. Bond and K. Buntins, Systematic Reviews in Educational Research: Methodology, Perspectives and Application, Springer, Wiesbaden, (2020).
[23] C. Wohlin, Guidelines for snowballing in systematic literature studies and a replication in software engineering, in Proc. 18th International Conference on Evaluation and Assessment in Software Engineering, ACM, New York: 1-10, (2014).
[24] A. Martín-Martín, E. Orduna-Malea, M. Thelwall and E. Delgado López-Cózar, Google Scholar, Web of Science, and Scopus: a systematic comparison of citations in 252 subject categories, J. Informetr. **12**, 1160-1177 (2018).
[25] M. Gusenbauer and N. R. Haddaway, Which academic search systems are suitable for systematic reviews or meta-analyses? Evaluating retrieval qualities of Google Scholar, PubMed, and 26 other resources, Res. Synth. Methods **11**, 181-217 (2020).
[26] N. R. Haddaway, A. M. Collins, D. Coughlin and S. Kirk, The role of Google Scholar in evidence reviews and its applicability to grey literature searching, PLoS ONE **10**, e0138237 (2015).
[27] J. Cohen, A coefficient of agreement for nominal scales, Educ. Psychol. Meas. **20**, 37-46 (1960).
[28] J. R. Landis and G. G. Koch, The measurement of observer agreement for categorical data, Biometrics **33**, 159-174 (1977).
[29] J. P. T. Higgins, J. A. López-López and A. M. Aloe, Meta-regression, in Handbook of Meta-Analysis, Chapman and Hall/CRC, Boca Raton: 129-150, (2020).
[30] J. von Neumann and O. Morgenstern, Theory of Games and Economic Behavior, Princeton University Press, Princeton, (1944).
[31] H. Raiffa and R. Schlaifer, Applied Statistical Decision Theory, Harvard University Press, Boston, (1961).
[32] H. A. Simon, A behavioral model of rational choice, Q. J. Econ. **69**, 99-118 (1955).
[33] L. von Bertalanffy, General System Theory: Foundations, Development, Applications, George Braziller, New York, (1968).
[34] J. D. Sterman, Business Dynamics: Systems Thinking and Modeling for a Complex World, Irwin/McGraw-Hill, Boston, (2000).
[35] F. P. Brooks, The Mythical Man-Month: Essays on Software Engineering, Addison-Wesley, Reading, (1975).
[36] P. R. Lawrence and J. W. Lorsch, Organization and Environment: Managing Differentiation and Integration, Harvard University Press, Boston, (1967).
[37] L. Donaldson, The Contingency Theory of Organizations, Sage, Thousand Oaks, (2001).
[38] D. Howell, C. Windahl and R. Seidel, A project contingency framework based on uncertainty and its consequences, Int. J. Proj. Manag. **28**, 256-264 (2010).
[39] B. J. Sauser, R. R. Reilly and A. J. Shenhar, Why projects fail? How contingency theory can provide new insights – a comparative analysis of NASA's Mars Climate Orbiter loss, Int. J. Proj. Manag. **27**, 665-679 (2009).
[40] F. S. Hillier and G. J. Lieberman, Introduction to Operations Research, 10th ed., McGraw-Hill, New York, (2015).
[41] F. Wilcoxon, Individual comparisons by ranking methods, Biom. Bull. **1**, 80-83 (1945).
[42] M. Friedman, The use of ranks to avoid the assumption of normality implicit in the analysis of variance, J. Am. Stat. Assoc. **32**, 675-701 (1937).
[43] W. H. Kruskal and W. A. Wallis, Use of ranks in one-criterion variance analysis, J. Am. Stat. Assoc. **47**, 583-621 (1952).

© 2026 NSP
Natural Sciences Publishing Cor.

[44] J. G. MacKinnon and H. White, Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties, J. Econom. **29**, 305-325 (1985).
[45] C. E. Clark, The greatest of a finite set of random variables, Oper. Res. **9**, 145-162 (1961).
[46] D. R. Fulkerson, Expected critical path lengths in PERT networks, Oper. Res. **10**, 808-817 (1962).
[47] K. R. MacCrimmon and C. A. Ryavec, An analytical study of the PERT assumptions, Oper. Res. **12**, 16-37 (1964).
[48] B. W. Boehm, C. Abts, A. W. Brown, S. Chulani, B. K. Clark, E. Horowitz et al., Software Cost Estimation with COCOMO II, Prentice Hall, Upper Saddle River, (2000).
[49] B. Boehm, R. Valerdi, J. Lane and A. W. Brown, COCOMO suite methodology and evolution, CrossTalk **18**, 20-25 (2005).
[50] S. Sholiq, R. Sarno, E. S. Astuti and M. A. Yaqin, Implementation of COSMIC function points (CFP) as primary input to COCOMO II: study of conversion to line of code using regression and support vector regression models, Int. J. Intell. Eng. Syst. **16**, 92-103 (2023).
[51] J. Blazewicz, J. K. Lenstra and A. H. G. Rinnooy Kan, Scheduling subject to resource constraints: classification and complexity, Discrete Appl. Math. **5**, 11-24 (1983).
[52] M. Fowler and J. Highsmith, The agile manifesto, Softw. Dev. **9**, 28-35 (2001).
[53] W. Lipke, Schedule is different, The Measurable News, 31-34 (2003).
[54] S. Vandevoorde and M. Vanhoucke, A comparison of different project duration forecasting methods using earned value metrics, Int. J. Proj. Manag. **24**, 289-302 (2006).
[55] M. Castañón-Puga, R. F. Rosales-Cisneros, J. C. Acosta-Prado, A. Tirado-Ramos, C. Khatchikian and E. Aburto-Camacllanqui, Earned value management agent-based simulation model, Systems **11**, 86 (2023).
[56] C. Barua, System dynamics modeling of critical chain project management in multi-project engineering environments, Am. J. Data Sci. Anal. **6**, 43-85 (2025).
[57] F. Abbasi and T. Allahviranloo, Realistic solution of fuzzy critical path problems, case study: the airport's cargo ground operation systems, Granul. Comput. **8**, 617-632 (2023).
[58] R. V. Françozo, L. S. V. U. Junior, E. S. Carrapateira, B. C. S. Pacheco, M. T. Oliveira, G. B. Torsoni and J. Yari, A web-based software for group decision with analytic hierarchy process, MethodsX **11**, 102277 (2023).
[59] K. J. Arrow, T. Harris and J. Marschak, Optimal inventory policy, Econometrica **19**, 250-272 (1951).

Umar Ammar Altahtooh is an Associate Professor at Taibah University in Madinah, Saudi Arabia (formerly an MIS Engineer at the Madinah Municipality). He has over 24 years of practical and academic experience in MIS, business, and project management. He holds a PhD in Project Management from the University of Manchester, Faculty of Engineering, UK (2014), an MBA from Missouri State University, College of Business, USA (2009), and a BSc in Management Information Systems (MIS) from King Fahd University of Petroleum and Minerals, College of Industrial Management, Dhahran (2002). He has published research papers, including one that developed Project End Theory by introducing the concept of Time Error into project management principles. Dr. Altahtooh is a PMP (Project Management Professional) and holds several other professional certifications and academic awards. He is also the author of the book "Introduction to Project Management".

© 2026 NSP
Natural Sciences Publishing Cor.