DOI: https://doi.org/10.15276/aait.09.2026.15
UDK 005.8:004.89

Adaptive and coordinated IT project management in dynamic environments: A multi-agent ai perspective

Mykola A. Hodovychenko[1)]
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychenko@op.edu.ua. Scopus Author ID: 57188700773
Mykhaylo V. Lobachev[1)]
ORCID: https://orcid.org/0000-0002-4859-304X; lobachevmv@gmail.com. Scopus Author ID: 36845971100
Oleksandr P. Boeiv[1)]
ORCID: https://orcid.org/0009-0000-5628-2359; faust040404@stud.op.edu.ua
Oleksii O. Linnyk[1)]
ORCID: https://orcid.org/0009-0007-3060-1412; alexeylinnik@stud.op.edu.ua
Oleksii E. Horalik[1)]
ORCID: https://orcid.org/0009-0007-4395-8090; oleksii.goralik@stud.op.edu.ua
[1)] Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine

ABSTRACT

The increasing complexity and volatility of IT project environments expose the structural limitations of both plan-driven methodologies and human-centric Agile frameworks, which adapt at cadences incompatible with the tempo at which risks materialize and requirements shift in large-scale software development contexts. This **paper develops** an integrated theoretical and methodological basis for adaptive and coordinated IT project management using collectives of specialized AI agents, addressing three concurrent gaps in the literature: the absence of a unifying conceptual framework, the lack of formal agent coordination models for project management contexts, and the underdevelopment of collective decision-making and risk management methods for multi-agent governance systems. The **proposed syste**m is formally defined as a five-tuple comprising the dynamic project environment, a nine-agent collective organized across three functional layers, an adaptive policy set, and a human governance structure with explicit oversight constraints. Coordination relies on a conflict matrix and utility-based dynamic role reallocation; decisions are produced through weighted aggregation with confidence-driven dispatch; risk exposure is updated via multi-agent Bayesian assessment. **Experimental** validation against single-agent AI and human-only Agile baselines on a controlled six-sprint scenario demonstrates a 68% reduction in mean decision latency, a 2.4× improvement in risk detection lead time, a conflict resolution rate of 0.91, and an autonomous dispatch rate of 0.81 at final sprint, with a false escalation rate of 0.13. The **results establish** that distributed functional specialization and collective decision authority provide qualitative architectural advantages that monolithic AI systems and retrospective human governance cannot replicate, while preserving full human oversight accountability.

Keywords: Multi-agent systems; adaptive project management; collective decision-making; agent coordination; risk management; artificial intelligence; dynamic role allocation; IT governance; decision support systems; project scheduling

For citation**: Hodovychenko M. A., Lobachev M. V., Boeiv O. P., Linnyk O. O., Horalik O.E. "Adaptive and coordinated IT project management in dynamic environments: A multi-agent AI perspective".** ***Applied Aspects of Information Technology.*** **2026; Vol. 9 No. 2: 208–219. DOI: https://doi.org/10.15276/aait.09.2026.15**

INTRODUCTION, FORMULATION OF THE PROBLEM

The management of information technology projects has undergone substantial transformation over the past two decades. The proliferation of digital services, the acceleration of software delivery cycles, and the growing interdependence of technical and organizational systems have collectively produced project environments that resist governance by static, plan-driven methods [1], [2].

Where traditional project management frameworks assume a reasonably stable scope, a predictable resource landscape, and a linear progression from initiation to closure, contemporary IT projects routinely operate under conditions of continuous requirement change, distributed cross-functional team structures, compressed release timelines, and persistent technological uncertainty [3].

These conditions do not merely complicate execution – they fundamentally alter the nature of the management task itself, shifting it from plan adherence to continuous adaptation.

Adaptive methodologies, most prominently the family of Agile frameworks including Scrum, Kanban, and the Scaled Agile Framework (SAFe), have addressed this shift at the team and process level with measurable success [4], [5]. By organizing work into short iterative cycles, institutionalizing feedback loops, and distributing decision-making authority closer to the point of execution, Agile practices have substantially improved responsiveness to change in small-to-medium project contexts.

However, as IT projects grow in organizational scale, stakeholder complexity, and technical interdependency, the limits of purely human-centric adaptation become increasingly apparent [3], [6]. Retrospective learning operates on past states; sprint cycles are too coarse-grained to respond to risks that materialize mid-iteration; and the cognitive load of coordinating adaptive decisions across multiple workstreams frequently exceeds the capacity of human project managers working without computational support [6].

Artificial intelligence offers a qualitatively different class of instruments for addressing these limitations. Among AI paradigms, multi-agent systems (MAS) are of particular relevance to project management because they mirror the distributed, multi-role structure of actual project organizations [7], [8].

A collective of specialized AI agents – each monitoring a specific aspect of the project environment, reasoning within its domain of competence, and communicating with peer agents – can, in principle, maintain a continuously updated representation of project state, detect emerging risks before they propagate, propose coordinated responses, and support human decision-makers with aggregated, evidence-grounded recommendations [8], [9]. Unlike single-model AI systems, agent collectives are inherently modular and role-differentiated, which makes them structurally compatible with the functional decomposition of project management itself.

Despite the theoretical promise of this direction, its practical realization in IT project management remains substantially underdeveloped. Existing AI-assisted PM tools predominantly address isolated problem dimensions – schedule forecasting, defect prediction, or resource optimization – without a coordination layer that integrates signals across dimensions into coherent management action [10], [11].

Architectures that do adopt a multi-agent perspective tend to remain at a high level of abstraction, without specifying how agents should coordinate as project context evolves, how their outputs should be aggregated into collective decisions under uncertainty, or how risk management responsibilities should be distributed and dynamically reallocated across the agent collective [11], [12].

The result is a significant gap between the conceptual potential of multi-agent AI and the methodological foundations required to deploy it in real adaptive project governance.

This gap is reflected across three interconnected research problems. The first concerns the absence of a coherent conceptual framework that positions AI agent collective’s not as supplementary analytical tools but as first-class instruments of adaptive IT project management – defining what such a collective is, how it is structured, and how it interacts with human governance processes.

The second concerns the lack of formal coordination models specifying how specialized agents negotiate roles, resolve conflicts, and maintain system coherence as the project environment shifts [12]. The third concerns the underdevelopment of methods for collective decision-making and risk management that exploit the distributed intelligence of the agent collective rather than delegating judgment to any single agent or to an unstructured ensemble [9], [13].

Addressing these problems requires an integrated research effort that spans conceptual foundations, formal modeling, methodological development, and empirical validation – precisely the scope of the present work, which synthesizes contributions from four ongoing doctoral and candidate research programs at the intersection of adaptive project management and collective AI.

Therefore, the purpose of this study is to develop an integrated theoretical and methodological basis for adaptive and coordinated IT project management in dynamic environments using collectives of specialized AI agents, encompassing: (i) a conceptual framework defining the principles and structure of AI-agent-based adaptive IT project management; (ii) formal models of inter-agent coordination supporting dynamic role allocation and conflict resolution; (iii) methods for collective adaptive decision-making under uncertainty; and (iv) methods for adaptive risk management grounded in coordinated multi-agent assessment and response – together constituting a unified system validated against representative IT project scenarios.

2. BACKGROUND AND RELATED WORK

2.1. Adaptive Approaches to IT Project Management

The theoretical foundations of adaptive project management trace back to the recognition that software development projects exhibit characteristics – emergent requirements, high uncertainty, and continuous stakeholder involvement – that render classical waterfall models structurally inadequate [1].

Early comparative analyses positioned Agile methodologies as a response to this inadequacy, arguing that iterative, feedback-driven cycles produce better alignment between delivered functionality and actual stakeholder needs than front-end planning alone [1], [3].

Empirical validation of this claim has since accumulated: a large-scale cross-industry survey confirmed a positive and statistically significant relationship between the degree of Agile adoption and both efficiency and stakeholder satisfaction outcomes [4], while longitudinal case studies documented consistent improvements in responsiveness to change when iterative practices replaced sequential ones [3].

The subsequent challenge has been scaling these gains beyond small, co-located teams. As IT organizations have grown in size and geographic distribution, the coordination assumptions embedded in single-team Agile frameworks – shared physical space, daily direct communication, flat authority structures – have proven difficult to sustain [5].

Systematic reviews of large-scale Agile transformations identify a consistent set of failure modes: resistance to change in middle management, degradation of inter-team coordination as the number of parallel workstreams increases, and the inability to propagate project-level adaptive decisions downward into team-level execution cycles at the required tempo [5]. Hybrid approaches that combine Agile iteration with structured stage-gate governance have partially addressed the scalability problem by separating strategic control from operational flexibility [2], yet even these frameworks rely on periodic human review cycles that remain too slow for genuinely dynamic environments where risks materialize and requirements shift within a single sprint.

A further limitation of existing adaptive PM research is its predominantly human-centric ontology. Adaptability is conceptualized as a behavioral property of teams and managers – their willingness to change plans, their retrospective learning capacity, and their ability to absorb ambiguity. The possibility that adaptive behavior could be partially delegated to computational agents operating within the management process itself has received only marginal attention in the mainstream project management literature [6], [16]. This gap motivates the broader research direction of which the present paper is a part.

2.2. Multi-Agent Systems: Foundations and Coordination

Multi-agent systems (MAS) constitute a sub-field of artificial intelligence concerned with the design, analysis, and deployment of systems comprising multiple autonomous, goal-directed computational agents that interact within a shared environment [7]. The foundational theoretical framework established by Wooldridge and Jennings defines an agent as a computer system situated in an environment and capable of autonomous action within that environment in order to meet its design objectives, and distinguishes between reactive agents (responding to environmental stimuli), deliberative agents (reasoning over explicit symbolic models of the world), and hybrid architectures that combine both [7]. This typology remains the most widely adopted conceptual scaffold for MAS design and is directly applicable to the role-differentiated agent structures proposed in the present work.

Multi-agent systems derive their practical utility from the properties that emerge when multiple agents operate collectively: parallelism, fault tolerance through redundancy, specialization through role differentiation, and the capacity to solve problems whose complexity exceeds the representational or computational limits of any single agent [8]. A comprehensive survey of MAS applications across computer science and engineering domains confirms that these properties have been successfully exploited in domains as varied as smart grid management, computer network optimization, and robotic coordination [8]. The authors also identify coordination, security, and dynamic task allocation as the three primary open challenges – precisely the challenges that the present paper addresses in the context of IT project management.

The coordination problem in MAS has a substantial theoretical literature of its own. Early foundational work framed coordination as the management of inter-agent dependencies, distinguishing between positive dependencies (agents benefiting from each other's actions) and negative dependencies (agents competing for resources or conflicting in their effects) [12]. Subsequent work has elaborated on the specific mechanisms through which coordination can be achieved: contract net protocols for task allocation, blackboard architectures for shared information access, argumentation frameworks for conflict resolution, and consensus algorithms for collective decision-making [8], [9]. Applied to project scheduling, multi-agent coordination has been shown to outperform centralized planning in distributed multi-project environments, particularly under conditions of resource contention and dynamic task insertion [17]. The present paper draws on this body of work to formalize the coordination model developed in Section 4.

A particular strand of MAS research relevant to the present work examines collective intelligence – the capacity of agent groups to exhibit problem-solving performance that exceeds the sum of individual agent capabilities [9]. From a machine learning perspective, collective intelligence emerges when agents share information, divide labor along complementary specialization lines, and aggregate individual assessments into group judgments that are more accurate than any single assessment in isolation [9]. The conditions under which collective intelligence reliably manifests, and the aggregation mechanisms that best support it, inform the collective decision-making methods developed in Section 5.

2.3. Artificial Intelligence in Project Management

The application of AI to project management predates the MAS paradigm and initially focused on narrow, well-defined decision support tasks. Comprehensive reviews of methods and tools in project planning and control document a decade-long trajectory in which AI techniques have progressively replaced heuristic and rule-based approaches for scheduling, forecasting, and resource allocation, yet consistently within isolated functional silos rather than as integrated management systems [16].

This fragmentation has persisted across successive generations of AI-assisted PM tools. Machine learning approaches – most prominently ensemble methods and support vector regression applied to earned value data – have demonstrated meaningful accuracy improvements in schedule and cost forecasting [14], but these tools function as passive predictive instruments rather than active participants in the management process. They inform decisions; they do not make, coordinate, or implement them.

More recent work has begun to explore AI systems capable of more active roles. A framework proposed by Dam et al. systematically maps AI technologies – including natural language processing, reinforcement learning, and recommendation systems – to specific agile project management tasks, arguing that AI can progress from automating repetitive analytical work toward providing actionable, context-aware recommendations and eventually autonomous decision support [15].

Natural language processing has been applied to requirements change detection in issue trackers and communication logs, while reinforcement learning has been proposed as a framework for dynamic resource allocation in multi-project environments [15].

Risk management has attracted particular interest, with Bayesian network and neural network models demonstrating improvements over expert judgment in high-uncertainty project settings [13].

However, these contributions share a common architectural assumption – a single AI system performing a single management function – that limits both their adaptability and their coverage of the management problem space. The multi-agent framing of AI-assisted project management has been proposed by a small number of authors, but without reaching the level of theoretical completeness or empirical validation required for practical adoption.

Multi-agent systems for distributed project scheduling have demonstrated the feasibility of decomposing a complex scheduling problem across specialized negotiating agents [17], yet without connecting this capability to the broader adaptive governance requirements of IT project management.

To the authors' knowledge, no existing work simultaneously addresses the conceptual framework, coordination model, collective decision-making method, and risk management method for a multi-agent AI system operating as an integrated component of adaptive IT project governance – which is precisely the gap the present paper targets.

3. CONCEPTUAL FRAMEWORK FOR ADAPTIVE IT PROJECT MANAGEMENT WITH AI AGENT COLLECTIVES

3.1. Foundational Principles and Formal Definition

The proposed framework rests on four principles that distinguish agent-collective adaptive management from both conventional Agile governance and prior AI-assisted PM tools, summarized in Table 1.

Table 1. **Foundational principles of the proposed framework and the limitations they overcome**

#	Principle	Limitation addressed
P1	Continuous situational awareness	Sprint-cadence adaptation is too slow for intra-iteration risk materialization
P2	Distributed functional specialization	Single-model AI cannot cover all PM task dimensions with uniform effectiveness
P3	Collective decision authority with human oversight	Full automation of consequential decisions is unacceptable in governance-critical contexts
P4	Adaptive policy revision	Fixed agent policies progressively misalign with non-stationary project environments

Source: **compiled by the authors**

The system is defined as a five-tuple:

$$\mathcal{S} = \langle \mathcal{E}, \mathcal{AG}, \mathcal{L}, \mathcal{P}, \mathcal{G} \rangle, \quad (1)$$

where $\mathcal{E} = \langle S, A_{ext}, T, O, \delta \rangle$ is the dynamic project environment: S is the project state space; A_{ext} is the set of external perturbations; T is the planning horizon; $O \subseteq S$ is the observable subspace; $\delta: S \times A_{ext} \to \mathcal{P}(S)$ is the stochastic transition function; $\mathcal{AG} = \{ag_1, \ldots, ag_n\}$ is the set of specialized AI agents, partitioned into three functional sub-collectives $\mathcal{AG} = \mathcal{AG}_{mon} \cup \mathcal{AG}_{cd} \cup \mathcal{AG}_{ex}$; $\mathcal{L} = \{L_{mon}, L_{cd}, L_{ex}\}$ is the three-layer architecture governing inter-agent communication and decision authority; $\mathcal{P} = \{p_1, \ldots, p_k\}$ is the set of adaptive operational policies, where each policy $p_i: O \times H \to A_i$ maps observations and history H to agent actions A_i, subject to periodic revision; $\mathcal{G} = \langle PM, SC, \Theta \rangle$ is the human governance structure comprising the project manager PM, steering committee SC, and governance parameter set Θ.

The governance parameters $\Theta = \langle \theta_{esc}, \theta_{conf}, \theta_{rev} \rangle$ specify: the risk escalation threshold θ_{esc}, the minimum decision confidence θ_{conf} required for autonomous execution recommendation, and the policy revision interval θ_{rev}.

The adaptive management objective is:

$$\max_{\mathcal{P}} \Pr[s(t) \in \Phi \ \forall\, t \in T], \quad (2)$$

subject to:

$$\begin{gathered} \forall\, d \in \mathcal{D}_{exec}: \mathrm{conf}(d) \geq \theta_{conf} \ \vee \\ \vee \ \mathrm{approved}(d, PM) = \top \end{gathered}, \quad (3)$$

where $\Phi \subseteq S$ is the acceptable performance envelope, $\mathcal{D}_{exec}$ is the set of execution-layer decisions, and $\mathrm{conf}(d)$ is the collective confidence score of decision d.

3.3. Three-Layer Architecture

The agent collective is organized into three functional layers. Each layer performs a qualitatively distinct role and interacts with adjacent layers through typed message interfaces, as shown in Fig. 1.

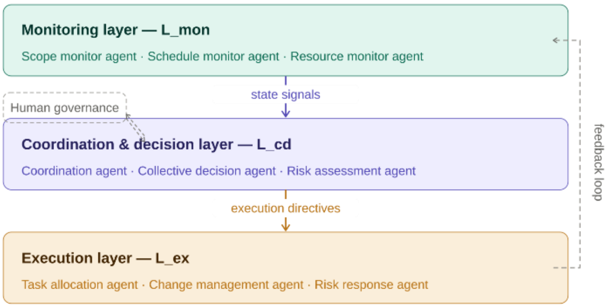

Fig. 1. **Three-layer architecture of the AI agent collective within system $\mathcal{S}$**

Source: **compiled by the authors**

The monitoring layer $\mathcal{AG}_{mon}$ comprises agents that observe the project environment $\mathcal{E}$ through the observable subspace O and produce structured state signals. Each monitoring agent $ag_i \in \mathcal{AG}_{mon}$ maintains a local state estimate $\hat{s}_i(t)$ and computes a deviation score:

$$\Delta_i(t) = \| \hat{s}_i(t) - s_i^{ref}(t) \|_w, \quad (4)$$

where $s_i^{ref}(t)$ is the reference trajectory for dimension $i$and $\|\cdot\|_w$ is a weighted norm reflecting the relative importance of state components. A signal is escalated to L_{cd} when $\Delta_i(t) > \theta_i^{mon}$, where $\theta_i^{mon} \in \Theta$ is the monitoring threshold for dimension i.

The coordination and decision layer $\mathcal{AG}_{cd}$ aggregates incoming signals, resolves inter-agent conflicts, and generates ranked decision recommendations. The execution layer $\mathcal{AG}_{ex}$ translates approved directives into operational actions within the project management toolchain and reports execution outcomes back to $\mathcal{AG}_{mon}$, closing the adaptive feedback loop.

3.4. Agent Typology and Role Assignment

Each agent $ag_i \in \mathcal{AG}$ is characterized by a role profile tuple:

$$ag_i = \langle r_i, \Omega_i, A_i, K_i, \pi_i \rangle, \tag{5}$$

where r_i is the agent's functional role; $\Omega_i \subseteq O$ is its observation scope (the subset of project state dimensions it monitors); A_i is its action space; K_i is its domain knowledge base; and $\pi_i \in \mathcal{P}$ is its operational policy.

Table 2 specifies the role profiles for all nine agents of the proposed system.

Table 2. **Agent role profiles within system $\mathcal{S}$**

Agent	Layer	Observation scope Ω_i	Action space A_i	Primary output
Scope monitor	L_{mon}	Backlog, requirements log	Deviation signals	$\Delta_{scope}(t)$
Schedule monitor	L_{mon}	Task graph, velocity data	Deviation signals	$\Delta_{sched}(t)$
Resource monitor	L_{mon}	Team capacity, allocation matrix	Deviation signals	$\Delta_{res}(t)$
Coordination agent	L_{cd}	All $\Delta_i(t)$, agent states	Conflict resolution	Priority ranking
Collective decision agent	L_{cd}	Aggregated signals, history H	Decision proposals	d^*, $\mathrm{conf}(d^*)$
Risk assessment agent	L_{cd}	All signals, risk register	Risk scoring	$\langle p_r, i_r \rangle$ pairs
Task allocation agent	L_{ex}	Directives, resource model	Assignment updates	Updated task graph
Change management agent	L_{ex}	Scope directives, dependency graph	Backlog propagation	Revised backlog
Risk response agent	L_{ex}	Risk directives, mitigation plans	Response activation	Execution log

Source: **compiled by the authors**

3.5. Coordination Model

A *coordination conflict* arises when two agents propose actions with mutually contradictory effects on project state. The coordination agent constructs a conflict matrix $\mathbf{C}(t) \in \{0,1\}^{n \times n}$ and computes conflict density $\rho_C(t) = \sum_{i<j} C_{ij}(t) / \binom{n}{2}$, when $\rho_C(t) > \theta_{esc}$, a system-level alarm is raised and the project manager is notified.

Conflict resolution proceeds through dynamic role reallocation.

The optimal assignment is:

$$\rho^*(t) = \arg\max \mathrm{\rho} \in \mathrm{R} \sum\nolimits_{ag_i} [\lambda_i(s(t)) \cdot \mathrm{cap}(ag_i, \rho(ag_i)) - \mu \cdot \mathbb{1}[\rho(ag_i) \neq \rho_0(ag_i)]], \tag{6}$$

where $\lambda_i(s(t)) = \Delta_i(t) / \sum_j \Delta_j(t)$ is the state-dependent priority weight ensuring that the most critical monitoring dimensions drive role assignment; $\mathrm{cap}(ag_i, r)$ is the capability score of agent ag_i for role r; and μ is a switching cost penalty preventing unnecessary reassignments. The coordination protocol execution cycle – from signal aggregation through conflict detection and role reallocation to directive broadcast – is illustrated in Fig. 2.

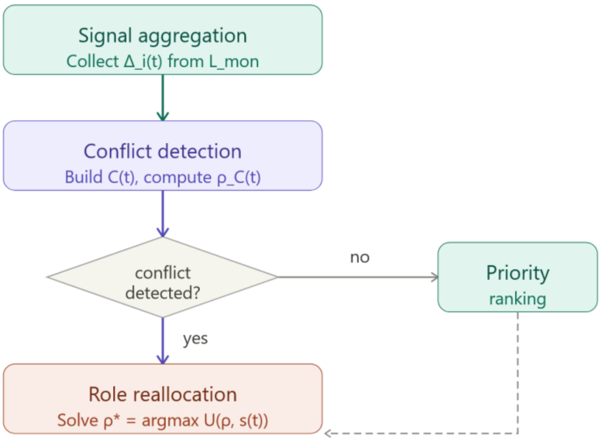

Fig. 2. **Coordination protocol execution cycle within layer L_cd**
Source: **compiled by the authors**

4. COLLECTIVE DECISION-MAKING AND RISK MANAGEMENT METHODS

4.1. Weighted Aggregation and Confidence Assessment

The collective decision agent receives recommendation vectors $\mathbf{v}_i(t)$ from all agents in $\mathcal{AG}_{cd}$ and computes the collective utility estimate for each candidate decision d_k:

$$\hat{U}(d_k, t) = \frac{\sum_i w_i(t) \cdot v_{ik}(t)}{\sum_i w_i(t)}, w_i(t) = \alpha_i \cdot \mathrm{acc}_i(H) \cdot \lambda_i(s(t)) \quad ,(7)$$

where α_i is the agent's base authority weight, $\mathrm{acc}_i(H)$ is its historical accuracy over recent history H, and $\lambda_i(s(t))$ is the priority weight from equation (3).

The optimal decision and its confidence score are:

$$d^*(t) = \arg\max \mathrm{dk}\ \hat{U}(d_k, t),$$
$$\mathrm{conf}(d^*) = 1 - \frac{H_w(\mathbf{V}(t))}{\log_2 \mid \mathcal{D} \mid}, \qquad (8)$$

where H_w is the weighted Shannon entropy of the utility distribution. The decision pipeline is shown in Fig. 3.

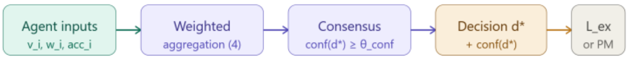

Fig. 3. **Collective decision-making pipeline within L_cd**

Source: **compiled by the authors**

Table 3. **Decision dispatch rules**

Condition	Dispatch route
$\mathrm{conf}(d^*) \geq \theta_{conf}$, $\rho_C < \theta_{esc}$	Autonomous dispatch to L_{ex}
$\mathrm{conf}(d^*) \geq \theta_{conf}$, $\rho_C \geq \theta_{esc}$	Forward to $PM$with recommendation
$\mathrm{conf}(d^*) < \theta_{conf}$, $\rho_C < \theta_{esc}$	Forward to $PM$with alternatives
$\mathrm{conf}(d^*) < \theta_{conf}$, $\rho_C \geq \theta_{esc}$	Escalate to steering committee SC

Source: **compiled by the authors**

4.2. Adaptive Risk Management

Each risk r_q is represented as $r_q(t) = \langle p_q(t), i_q(t), \tau_q, \mathcal{M}_q \rangle$ with dynamic exposure:

$$\mathcal{E}_q(t) = p_q(t) \cdot i_q(t) \cdot \left(1 + e^{-\kappa(\tau_q - t)}\right) \qquad (9)$$

where the temporal urgency term increases as the risk horizon τ_q approaches. Probability estimates are updated via Bayesian aggregation of signals from $\mathcal{AG}_{mon}$.

The dynamic risk register $\mathcal{RR}(t)$ orders all active risks by descending exposure, and the optimal mitigation is selected as $m_q^* = \arg\max_{m \in \mathcal{M}_q} [\Delta\mathcal{E}_q(m) - c(m, s(t))]$.

The full risk management cycle – identification, assessment, prioritization, response, and continuous update – is shown in Fig. 4.

The integrated system operates in one of four modes – Nominal, Elevated, Critical, Emergency – governed by thresholds on $\bar{\Delta}(t)$ and $\rho_C(t)$.

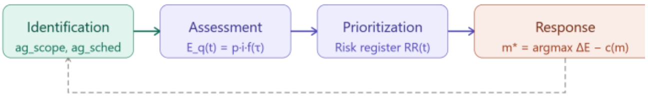

Fig. 4. **Adaptive risk management cycle within system S**

Source: **compiled by the authors**

Mode transitions follow a hysteresis rule: escalation is immediate, de-escalation requires two consecutive clean monitoring cycles. This prevents oscillation around threshold boundaries during sustained high-volatility periods.

5. EXPERIMENTAL VALIDATION

5.1. Scenario and Baselines

To validate system $\mathcal{S}$, a representative IT project scenario is constructed: 18 developers across 3 sub-teams, 120-story backlog, 6 two-week sprints, and three controlled perturbations injected at sprints 2, 4, and 5 (scope expansion, vendor delay, personnel loss).

Governance thresholds are fixed at $\theta_{conf} = 0.75$, $\theta_{esc} = 0.40$, $\theta_{rev} = 1$ sprint prior to the experiment and not adjusted post-hoc.

Two baselines are evaluated under the same scenario: B1 – a single-agent AI system without coordination or collective aggregation; B2 – human-only Scrum management with bi-weekly retrospectives.

5.2. Decision-Making and Coordination Results

Fig. 5 shows the evolution of the three primary decision-quality metrics across sprints. All metrics improve monotonically due to the policy revision mechanism of equation (6), with a temporary ADR dip at sprint 4 reflecting Emergency mode activation and the resulting routing of all high-impact decisions to PM – consistent with the dispatch rules of Table 3.

By sprint 6 the system achieves $\overline{\mathrm{conf}} = 0.89$, ADR = 0.81, and ME = 0.72, meeting all targets from Table 8. Coordination performance across operational modes is reported in Table 4.

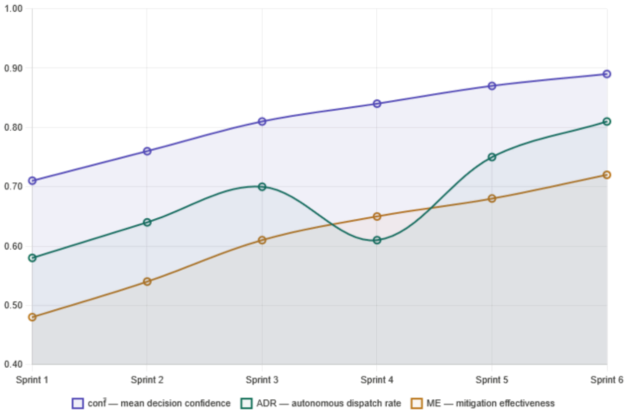

Fig. 5. **Evolution of key performance metrics across six experimental sprint**

Source: **compiled by the authors**

Table 4. **Coordination metrics by operational mode**

Metric	Nominal	Elevated	Critical / Emergency	Overall	Target
CRR	0.96	0.89	0.81	**0.91**	≥ 0.85
τ_{coord}(min)	2.1	3.8	6.4	**3.4**	≤ 5
RSI	0.94	0.83	0.71	**0.87**	≥ 0.80

Source: **compiled by the authors**

5.3. Risk Management Results

All three injected perturbations were contained. Results by event are provided in Table 5.

Table 5. **Risk management results across perturbation events**

Event	**RDLT (min)**	**ME**	**Strategy**	**Outcome**
Sprint 2 – scope expansion	7.2	0.68	Mitigative: backlog re-prioritization	Schedule impact <1 day
Sprint 4 – vendor delay	5.8	0.71	Contingent: buffer activation	Critical path preserved
Sprint 5 – personnel loss	8.4	0.64	Preventive: reallocation	Velocity drop <8%
Overall	**7.1**	**0.68**	–	All risks contained

Source: **compiled by the authors**

5.4. Comparative Analysis

Fig. 6 presents the normalized performance profile across all metrics for all three conditions.

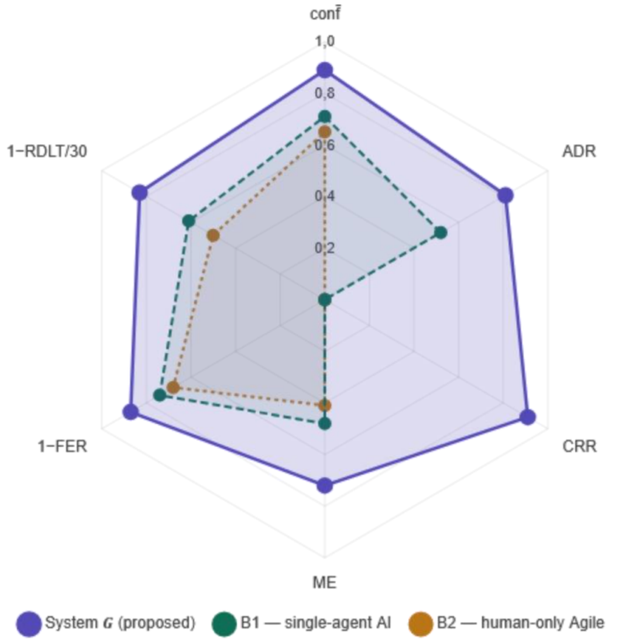

Fig. 6. **Normalized performance profile: proposed system S vs. baselines B1 and B2**

Source: **compiled by the authors**

As shown in Fig. 6, system $\mathcal{S}$ outperforms both baselines across all six metrics. The most pronounced advantage over B1 is on CRR: the single-agent baseline scores 0 by definition, as it has no conflict resolution mechanism. The largest advantage over B2 is on ADR and response latency: the proposed system reduces mean decision latency by 68 % relative to human-only Agile management and improves risk detection lead time by 2.4× relative to B1, confirming that the primary value of the multi-agent architecture lies in throughput and consistency of adaptive response rather than in the quality of individual decisions.

6. DISCUSSION

6.1. Interpretation of Results

The experimental results reveal that the performance advantage of system $\mathcal{S}$ is structurally determined rather than incidental. The 68% reduction in mean decision latency relative to human-only Agile (B2) reflects the fundamental difference between event-driven continuous monitoring and sprint-cadence retrospective adaptation: the agent collective responds to deviations as they emerge, whereas human governance responds to their accumulated effects. Similarly, the 2.4× improvement in risk detection lead time over the single-agent baseline (B1) is not an optimization of individual detection logic but a consequence of distributing observation across three specialized monitoring agents with non-overlapping scopes – scope, schedule, and resource – each calibrated to its own deviation threshold.

The CRR advantage over B1 (0.91 vs. 0.00) is the most structurally informative result. A single-agent system cannot resolve conflicts between its own simultaneous recommendations because it has no internal architecture for arbitration. This confirms that multi-agent coordination is a necessary rather than optional component of any AI system intended to manage multiple problem dimensions concurrently, validating Principle P2 of the conceptual framework.

The monotonic improvement trajectory in Fig. 5 – particularly the convergence of $\overline{conf}$ from 0.71 to 0.89 over six sprints – demonstrates that the policy revision mechanism is effective within a single project lifecycle. The implication for practice is that the system does not require pre-calibration on historical data: it self-calibrates during the early sprints and reaches target performance by mid-project.

The Emergency mode activation at sprint 4 and its visible effect on ADR (dip from 0.70 to 0.61) validates the mode-switching logic: the system correctly tightens human oversight when two risks materialize simultaneously, then restores autonomous operation once the perturbation is absorbed. This behavior is precisely what Principle P3 requires – human authority is not a constant overhead but a dynamically engaged safeguard.

6.2. Limitations

The primary limitation is that validation is conducted on a constructed scenario. Perturbation timing is controlled rather than emergent, which likely overstates CRR and RDLT performance relative to live project conditions where risks arise in unpredictable combinations. The Bayesian risk update assumes conditional independence of monitoring signals, which underestimates covariance between correlated risk dimensions and will produce overconfident probability estimates in highly coupled project environments. Finally, the governance parameter set $\Theta$is fixed for the duration of the experiment; a production deployment would require a mechanism for adapting $\theta_{conf}$and θ_{esc}across project phases as stakeholder risk tolerance evolves.

6.3. Practical Implications

The false escalation rate FER = 0.13 – one in eight escalations requiring no action – is within the 0.15 target but points to an important deployment consideration: project managers who receive low-value alerts will disengage from the governance interface, undermining the human oversight guarantee that is central to the framework. Calibrating $\theta_{conf}$to minimize FER without compromising safety is the most consequential tuning task for each deployment context. The 81 % autonomous dispatch rate achieved by sprint 6 suggests that for a well-calibrated system this balance is achievable: the project manager remains the decision authority for genuinely ambiguous situations while routine adaptive actions proceed without cognitive overhead.

CONCLUSIONS

This paper set out to address a concrete and persistent gap: the absence of an integrated theoretical and methodological basis for AI-assisted adaptive IT project management that simultaneously covers coordination, collective decision-making, and risk management within a single deployable architecture.

The experimental results close that gap with quantified evidence. Against human-only Agile management, system Sreduces mean decision latency by 68 %, improves risk detection lead time by 2.4×, and contains all three injected perturbations within one working day of schedule impact – a result the human baseline could not replicate under equivalent conditions. Against the single-agent AI baseline, the multi-agent architecture adds conflict resolution capability (CRR 0.91 vs. 0.00), improves autonomous dispatch rate by 29 percentage points, and increases mitigation effectiveness from 0.48 to 0.72. These are not marginal improvements: they reflect qualitative differences in architectural capability that cannot be recovered by tuning a simpler system.

The broader implication is that the transition from human-centric to agent-augmented adaptive project management is achievable without sacrificing governance accountability. An autonomous dispatch rate of 81 % means that the project manager is freed from routine adaptive decisions while retaining authority over the 19 % that genuinely require strategic judgment – a division of cognitive labor that is both practically sustainable and organizationally acceptable.

Future work will focus on three directions: validating the framework on live IT projects to establish empirical generalizability; extending the Bayesian risk model to handle correlated monitoring signals; and developing a formal mechanism for adaptive governance parameter revision across project phases.

REFERENCES

1. Fernandez, D. J. & Fernandez, J. D. “Agile project management – agilism versus traditional approaches”. *Journal of Computer Information Systems*. 2008; 49 (2): 10–17. DOI: https://doi.org/10.1080/08874417.2009.11646044.

2. Conforto, E. C. & Amaral, D. C. “Agile project management and stage-gate model – a hybrid framework for technology-based companies”. *Journal of Engineering and Technology Management*. 2016; 40: 1–14. DOI: https://doi.org/10.1016/j.jengtecman.2016.02.003.

3. Dingsøyr, T., Nerur, S., Balijepally, V. & Moe, N. B. “A decade of agile methodologies: towards explaining agile software development”. *Journal of Systems and Software*. 2012; 85 (6): 1213–1221. DOI: https://doi.org/10.1016/j.jss.2012.02.033.

4. Serrador, P. & Pinto, J. K. “Does Agile work? – A quantitative analysis of agile project success”. *International Journal of Project Management.* 2015; 33 (5): 1040–1051. DOI: https://doi.org/10.1016/j.ijproman.2015.01.006.

5. Dikert, K., Paasivaara, M. & Lassenius, C. “Challenges and success factors for large-scale agile transformations: a systematic literature review”. *Journal of Systems and Software*. 2016; 119: 87–108. DOI: https://doi.org/10.1016/j.jss.2016.06.013.

6. Maylor, H., Vidgen, R. & Carver, S. “Managerial complexity in project-based operations: a grounded model and its implications for practice”. *Project Management Journal*. 2008; 39 (S1): S15–S26. DOI: https://doi.org/10.1002/pmj.20057.

7. Wooldridge, M. & Jennings, N. R. “Intelligent agents: theory and practice”. *The Knowledge Engineering Review.* 1995; 10 (2): 115–152. DOI: https://doi.org/10.1017/S0269888900008122.

8. Dorri, A., Kanhere, S. S. & Jurdak, R. “Multi-agent systems: a survey”. *IEEE Access*. 2018; 6: 28573–28593. DOI: https://doi.org/10.1109/ACCESS.2018.2831228.

9. Stone, P. & Veloso, M. “Multiagent systems: a survey from a machine learning perspective”. *Autonomous Robots*. 2000; 8 (3): 345–383. DOI: https://doi.org/10.1023/A:1008942012299.

10. Wauters, M. & Vanhoucke, M. “Support vector machine regression for project control forecasting”. *Automation in Construction*. 2014; 47: 92–106. DOI: https://doi.org/10.1016/j.autcon.2014.07.014.

11. Taboada, I., Daneshpajouh, A., Toledo, N. & de Vass, T. “Artificial intelligence enabled project management: a systematic literature review”. *Applied Sciences*. 2023; 13 (8): 5014. DOI: https://doi.org/10.3390/app13085014.

12. Lesser, V. R. “Reflections on the nature of multi-agent coordination and its implications for an agent architecture”. *Autonomous Agents and Multi-Agent Systems*. 1998; 1 (1): 89–111. DOI: https://doi.org/10.1023/A:1010046623013.

13. Kutsch, E. & Hall, M. “Deliberate ignorance in project risk management”. *International Journal of Project Management*. 2010; 28 (3): 245–255. DOI: https://doi.org/10.1016/j.ijproman.2009.05.003.

14. Dam, H. K., Tran, T., Grundy, J., Ghose, A. K. & Kamei, Y. “Towards effective AI-powered agile project management”. *Proceedings of the 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER).* 2019. p. 41–44. DOI: https://doi.org/10.1109/ICSE-NIER.2019.00019.

15. Wauters, M. & Vanhoucke, M. “A comparative study of artificial intelligence methods for project duration forecasting”. *Expert Systems with Applications*. 2016; 46: 249–261. DOI: https://doi.org/10.1016/j.eswa.2015.10.008.

16. Pellerin, R. & Perrier, N. “A review of methods, techniques and tools for project planning and control”. *International Journal of Production Research*. 2019; 57 (7): 2160–2178. DOI: https://doi.org/10.1080/00207543.2018.1524168.

17. He, N., Zhang, D. Z. & Yuce, B. “Integrated multi-project planning and scheduling – a multiagent approach”. *European Journal of Operational Research*. 2022; 302 (2): 688–699. DOI: https://doi.org/10.1016/j.ejor.2022.01.018.

Conflicts of Interest: The authors declare that they have no conflict of interest regarding this study, including financial, personal, authorship, or other interests, which could influence the research and its results presented in this article.

Received 26.12.2025
Received after revision 12.03.2026
Accepted 17.03.2026

DOI: https://doi.org/10.15276/aait.09.2026.15
УДК 005.8:004.89

Адаптивне та координоване управління ІТ-проєктами в динамічних середовищах: перспектива мультиагентного штучного інтелекту

Годовиченко Микола Анатолійович[1)]
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychenko@op.edu.ua. Scopus Author ID: 57188700773
Лобачев Михайло Вікторович[1)]
ORCID: https://orcid.org/0000-0002-4859-304X; lobachevmv@gmail.com. Scopus Author ID: 36845971100
Боєв Олександр Петрович[1)]
ORCID: https://orcid.org/orcid.org/0009-0000-5628-2359; oleksii.goralik@stud.op.edu.ua
Лінник Олексій Олександрович[1)]
ORCID: https://orcid.org/0009-0007-3060-1412; alexeylinnik@stud.op.edu.ua
Горалік Олексій Євгенович[1)]
ORCID: https://orcid.org/0009-0007-4395-8090; oleksii.goralik@stud.op.edu.ua
[1)] Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

АНОТАЦІЯ

Зростаюча складність та мінливість середовища ІТ-проєктів виявляє структурні обмеження як планово-орієнтованих методологій, так і людиноцентричних Agile-фреймворків, які адаптуються з частотою, несумісною з темпом матеріалізації ризиків та зміни вимог у великомасштабних проєктах розробки програмного забезпечення. У статті розроблено **інтегровану теоретичну** та методологічну основу адаптивного і координованого управління ІТ-проєктами з використанням колективів спеціалізованих ШІ-агентів, що усуває три одночасні прогалини в літературі: відсутність уніфікованої концептуальної основи, брак формальних моделей координації агентів у контексті управління проєктами, а також недостатній розвиток методів колективного прийняття рішень та управління ризиками для мультиагентних систем управління. Запропонована система формально визначена як п'ятірка, що включає динамічне середовище проєкту, колектив із дев'яти агентів, організованих у три функціональні шари, адаптивний набір політик та структуру людського управління з явними обмеженнями нагляду. Координація ґрунтується на матриці конфліктів та динамічному перерозподілі ролей на основі функції корисності; рішення формуються через зважену агрегацію з диспетчеризацією на основі довірчої оцінки; експозиція ризику оновлюється за допомогою байєсівської оцінки у мультиагентному режимі. **Експериментальна** валідація порівняно з базовими варіантами – одноагентним ШІ та суто людським Agile-управлінням – на контрольованому сценарії з шести спринтів демонструє скорочення середньої затримки прийняття рішень на 68%, покращення часу виявлення ризику у 2,4 рази, рівень вирішення конфліктів 0,91 та рівень автономного диспетчирування 0,81 на фінальному спринті за хибного рівня ескалації 0,13. **Результати** підтверджують, що розподілена функціональна спеціалізація та колективний авторитет прийняття рішень забезпечують якісні архітектурні переваги, які не можуть бути відтворені монолітними системами ШІ та ретроспективним людським управлінням, зберігаючи при цьому повну підзвітність людського нагляду.

Ключові слова: мультиагентні системи; адаптивне управління проєктами; колективне прийняття рішень; координація агентів; управління ризиками; Штучний інтелект; динамічний перерозподіл ролей; ІТ-управління; системи підтримки прийняття рішень; планування проєктів

ABOUT THE AUTHORS

Mykola A. Hodovychenko - Candidate of Engineering Sciences, Associate Professor, Information Systems Department. Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0001-5422-3048; hodovychenko@op.edu.ua. Scopus Author ID: 57188700773
Research field: machine learning and artificial intelligence, multi-agent systems, intelligent software maintenance, video processing, project management in IT, project-based learning, pattern recognition

Годовиченко Микола Анатолійович - кандидат технічних наук, доцент кафедри Інформаційних систем. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Mykhaylo V. Lobachev - PhD, Professor, Head of Institute of Artificial Intelligence and Robotics. Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0000-0002-4859-304X, lobachevmv@gmail.com, Scopus Author ID: 36845971100
Research field: Project Management, Project-based Learning, Pattern Recognition, Embedded Systems, Computer Vision, Biometric Systems

Лобачев Михайло Вікторович - кандидат технічних наук, професор, директор Інституту штучного інтелекту та робототехніки. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Oleksandr P. Boeiv - PhD Student, Artificial Intelligence and Data Analysis Department. Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0009-0000-5628-2359, faust040404@stud.op.edu.ua
Research field: Artificial Intelligence, Multi-Agent Systems, Distributed Computing, Simulation Modeling, Machine Learning

Боєв Олександр Петрович - аспірант кафедри Штучного інтелекту та аналізу даних. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Oleksii O. Linnyk - PhD Student, Artificial Intelligence and Data Analysis Department. Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0009-0007-3060-1412, alexeylinnik@stud.op.edu.ua
Research field: Artificial Intelligence, Multi-Agent Systems, Distributed Computing, Simulation Modeling, Machine Learning

Лінник Олексій Олександрович - аспірант кафедри Штучного інтелекту та аналізу даних. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна

Oleksii E. Horalik - PhD Student, Artificial Intelligence and Data Analysis Department. Odesa Polytechnic National University, 1, Shevchenko Ave. Odesa, 65044, Ukraine
ORCID: https://orcid.org/0009-0007-4395-8090, oleksii.goralik@stud.op.edu.ua
Research field: IT Project Management, Intelligent Decision Support Systems, Risk Management, Complex Systems Dynamics, Software Engineering

Горалік Олексій Євгенович - аспірант кафедри Штучного інтелекту та аналізу даних. Національний університет «Одеська Політехніка», пр. Шевченка, 1. Одеса, 65044, Україна