

Vitalii Tkachov

METHOD OF RESTORING CRITICAL INFORMATION SYSTEM SERVICES ON A MOBILE PLATFORM UNDER CONDITIONS OF CONTROLLED DEGRADATION

The subject of this work is methods for restoring critical services of an information system on a mobile platform under intermittent connectivity and controlled degradation. *The objective of the study* is to develop a service-oriented recovery method that ensures a measurable reduction in the shortfall in meeting service quality threshold requirements within connectivity windows, taking into account resource constraints and the ability to complete service state reconciliation within the current connectivity window. The study used the following **methods**: system analysis, mathematical modeling of service states and profiles, formalization of the shortfall in meeting service quality threshold requirements, a predicate description of the window feasibility of recovery steps, and a computational experiment to evaluate recovery performance metrics. The article aimed to solve the following **tasks**: formalize a recovery step as a transition between service profiles that reduces the shortfall in meeting service quality threshold requirements; introduce a condition for the window feasibility of a recovery step as a combination of resource feasibility and the possibility of completing service state coordination within the current connectivity window; to construct a rule for forming a set of recovery steps within a window that ensures a reduction in the deficit provided at least one step is executed and is consistent with permissible managed degradation actions as a peripheral resource release mechanism. **Achieved results** consist in the implementation of a window feasibility formalism and a system of metrics for evaluating compliance with feasibility conditions, the proportion of windows with deficit reduction, the cumulative deficit of critical services, and recovery time, which makes it possible to distinguish environmental constraints from the properties of control decisions. **Conclusions**: Ignoring the possibility of completing service state coordination within the current connectivity window leads to the emergence of incomplete recovery transitions and discrepancies between the expected recovery effect and the actual deficit dynamics, whereas the proposed method identifies the steps taken within the current connectivity window and ensures a reduction in the deficit of critical services in windows where recovery is possible. This enhances survivability by reducing the duration of degradation episodes of critical functions and decreasing the cumulative deficit. The practical value lies in the possibility of using the method for onboard information systems of UAV groups and other mobile nodes with connectivity windows of variable duration and bandwidth.

Keywords: information system; survivability; critical service; controlled degradation; connectivity window; service recovery.

1. Introduction

Information systems on mobile platforms (ISMP) operate in environments where available computing, power, and network resources are subject to fluctuations in availability, and connectivity between system components is intermittent [1, 2]. Such MISPs are characterized by short operational control intervals, connectivity windows of variable duration and bandwidth, as well as non-stationary load and disruptive influences [3]. Under these conditions, critical functions of ISMPs are implemented by services, the requirements for which are formulated in terms of service quality metrics, and violations of these requirements manifest as a failure to meet service quality thresholds.

The survivability of an information system on a mobile platform is understood as an integral property consisting of the system's ability to maintain or restore the execution of its primary function in real time despite resource shortages, intermittent connectivity, and incomplete observability of states [4–6]. In the context of mobile platforms, ensuring this resilience is closely

linked to risk-oriented management and robust task distribution under dynamically changing operating conditions [7, 8]. At the operational management level, this means that in each short control interval, the ISMP must ensure the minimum acceptable level of performance for the critical service core and, following a degradation episode, perform recovery transitions of critical services to profiles that ensure the fulfillment of threshold requirements for service quality metrics.

Situations of failure in functional capability that are significant for mobile environments often take the form of controlled degradation, where the ISMP maintains operability by reducing the functionality of some non-critical services or by simplifying data processing, thereby freeing up resources to maintain the minimum acceptable level of operability of the critical service core. However, after the degradation episode ends, a separate task of restoring critical services arises, which cannot be reduced to a general redistribution of resources [9]. Restoration requires the allocation of resources and the completion of service state synchronization within available connectivity windows. If reconfigurations are

initiated without verifying the possibility of completing service state synchronization within the current connectivity window, incomplete recovery transitions occur, the volume of unfulfilled state synchronization accumulates, repeated violations of service quality threshold requirements are recorded after partial recovery, and the time to return to an operational state increases.

An additional complication is that the recovery of critical services often relies on auxiliary actions of controlled degradation of non-critical components and data [10–12]. Such actions must remain permissible under correctness and consistency invariants; otherwise, the short-term release of resources leads to long-term consequences in the form of data degradation, state conflicts, or cascading secondary failures. Therefore, it is relevant to develop a recovery method that operates at the service level due to the failure to meet service quality threshold requirements, generates recovery steps only if it is possible to complete service state coordination within available connectivity windows, and aligns with permissible managed degradation actions as a peripheral resource release mechanism.

2. Analysis of recent research and publications

Modern approaches to ensuring the sustainable operation of edge computing systems are evolving toward the integration of mechanisms for adaptation, redundancy, fault tolerance, and coordination between computing and connectivity. A summary of the state of the art is presented in [13], where risk factors characteristic of edge computing are identified, including high node mobility, degraded or intermittent connectivity, and power constraints. A classification of measures to improve resilience is also provided, including adaptive redundancy, monitoring and anomaly detection, joint optimization of computation and connectivity, and multi-level resource coordination.

A separate area consists of studies devoted to self-adaptive mechanisms in edge computing environments. In particular, in [14], the principles of adaptation design are systematized, a taxonomy of adaptive behavior properties is formed, and the limitations of existing approaches are outlined, including the specifics of reactive schemes, insufficient attention to adaptation at the communication infrastructure level, and the limited suitability of centralized control for decentralized network conditions. This directly reinforces the need for methods capable of generating and executing actions

within short connectivity control intervals, implemented via connectivity windows of variable duration and bandwidth, without assuming full synchronization of states in each interval.

In cloud container environments, the field of controlled functionality reduction is actively developing as a means of preserving the availability of critical components during resource shortages. In [15], a cooperative model of controlled functionality degradation is proposed, where the infrastructure operator performs degradation actions based on the criticality of containers. The goal is to maximize the availability of critical services during large-scale failures.

At the same time, the authors of [13–15] largely assume constant availability of control within the data center and do not separately highlight the requirement to complete service state synchronization within available connectivity windows, which is critical for information systems operating on mobile platforms.

For edge computing infrastructures and fog computing environments, methods for recovering from failures of network chains of functions and service routes in resource-constrained multi-layer topologies are important. In [16], post-failure recovery of service function chains in fog networks under multiple and simultaneous failures is considered. Heuristics for rerouting and remapping requests are proposed, taking into account delays and node load. This increases the proportion of restored chains and reduces recovery time without designating separate backup paths. These results are important as an infrastructure foundation; however, they are focused on restoring network links and do not formalize service-oriented recovery steps due to the lack of fulfillment of threshold requirements for quality of service metrics, with control over the ability to complete state synchronization within the current connectivity window. In other words, the survivability metrics of such systems are not formalized.

The study [17] considers the joint control of handover and service migration as a problem of maximizing long-term utility, taking into account service delay and the resource cost of migration. However, this approach formulates the problem as the optimization of system component placement and migration processes. In the task of restoring critical services in a controlled degradation mode, decisions regarding the relocation or reallocation of resources and the preliminary verification of the possibility of completing service state synchronization within the current connectivity window for each recovery transition are fundamental.

Thus, the works [13–17] under consideration cover the general framework of resilience and adaptation of information systems in edge computing environments, infrastructure mechanisms for controlled functionality reduction, post-failure recovery of network links, and the tasks of service migration under mobility. Beyond their immediate focus lies the service-oriented formulation of critical service recovery within the connectivity windows of a mobile platform as a sequence of discrete recovery steps.

For such steps, the possibility of completing service state synchronization within available connectivity windows must be verified, and effectiveness must be assessed by reducing the shortfall in meeting threshold requirements for service quality indicators, taking into account permissible controlled degradation actions as a peripheral tool for freeing up ISMP resources.

3. Research objective and tasks

The objective of the research is to develop a method for restoring critical services of an information system on a mobile platform under controlled degradation conditions, which, in real time under intermittent connectivity and resource constraints, ensures a measurable reduction in the shortfall in meeting service quality threshold requirements.

To achieve this objective, the following tasks have been set:

- formalize the critical service recovery step as a transition between service profiles, leading to a reduction in the shortfall in meeting service quality threshold requirements;
- introduce a window feasibility condition for the recovery step that simultaneously accounts for resource feasibility on the mobile platform and the possibility of completing service state synchronization within the current connectivity window; – develop a rule for forming recovery steps in each connectivity window that guarantees a reduction in the shortfall in meeting service quality threshold requirements provided there is at least one feasible step and does not result in incomplete recovery transitions.

4. Materials and Methods

We consider the set of services S and the subset of critical services $S_c \subseteq S$.

For each service $s \in S$, a finite ordered set of operational profiles is specified:

$$P_s = \{0, 1, \dots, L_s\}, \quad (1)$$

where the elements of the set $0, 1, \dots, L_s$ define the permissible levels of functionality and service quality: 0 corresponds to the minimum permissible level, and L_s to the maximum level. The value L_s determines the number of levels of functionality and service quality available for the service s .

The current service profile at a given point in time t is denoted by $p_s(t) \in P_s$, which is consistent with the definition of the set P_s (1).

For each service, an aggregated service quality metric $q_s(t)$ and a threshold value θ_s are recorded. Subsequently, the shortfall in meeting the threshold requirement for the service quality metric is defined as:

$$d_s(t) = \max\{0, \theta_s - q_s(t)\}, \quad s \in S. \quad (2)$$

If $d_s(t) = 0$, then the requirement is met; if $d_s(t) > 0$, then it is violated. Formula (2) is used as the baseline coordinate of the service state when constructing recovery steps.

To link the quality metric to profiles, a piecewise-constant model of the dependence of the service quality metric [18] on the profile is introduced:

$$q_s(t) = \bar{q}_s(p_s(t)). \quad (3)$$

In (3), $\bar{q}_s(\cdot)$ is a tabular characteristic of the service profile that records the expected quality level for the profile p_s . Subsequently, formula (3) allows us to interpret the change in the profile $p_s(t)$ as a controlled change in $q_s(t)$ and, accordingly, in the deficit $d_s(t)$ according to formula (2).

Intermittent continuity is defined by a sequence of continuity windows

$$W_k = [\tau_k, \tau_k + \Delta_k], \quad k = 1, 2, \dots, \quad (4)$$

where τ_k is the start time of the k -th window; Δ_k is the duration of the k -th window.

The windows (4) define the time intervals within which exchanges for service state coordination are permitted.

The capacity of the n th window (k) is defined as the maximum allowable amount of data for service

state synchronization that can be transmitted within the window:

$$C_k = \int_{\tau_k}^{\tau_k + \Delta_k} b(t) dt, \quad (5)$$

where $b(t)$ is the instantaneous (ideal) channel capacity at the moment t .

Subsequently, the value C_k (5) is used as a constraint on the possibility of completing service state synchronization during the execution of recovery steps within the window (4).

A recovery step is defined as a discrete transition of the service profile one level upward:

$$a \equiv (s, p \rightarrow p+1), \quad s \in S_c, \quad p \in \{0, \dots, L_s - 1\}. \quad (6)$$

Formula (6) specifies that recovery steps are considered specifically for critical services, and the transition is defined between adjacent profiles.

Each step a from equation (6) is associated with:

- a vector of additional platform resources required $\bar{r}(a) \in \mathbb{R}_+^m$;
- the amount of data required to synchronize the state of the services in $g(a) \in \mathbb{R}_+$, which must be

$$\Delta d(a) = \max \left\{ 0, \max \left\{ 0, \theta_s - \bar{q}_s(p) \right\} - \max \left\{ 0, \theta_s - \bar{q}_s(p+1) \right\} \right\}. \quad (8)$$

In (8), the expected reduction in the $d_s(t)$ deficit resulting from the transition between adjacent profiles is determined.

Within the k -th window, a set of recovery steps A_k is selected. Its feasibility is defined by the following total constraints:

$$\sum_{a \in A_k} \bar{r}(a) \leq \bar{R}_k, \quad \sum_{a \in A_k} g(a) \leq C_k. \quad (9)$$

In (9), the first constraint is consistent with the resource part of predicate (7), and the second – with the possibility of completing the coordination of service states within the current capacity-based connectivity window (5). Subsequently, the conditions in (9) are used as the basic rule for the permissible simultaneous execution of multiple recovery steps.

The operational requirement for window-based deficit reduction is formulated as the presence of at least one critical service for which the deficit in meeting the threshold requirements for service quality metrics is reduced within the window. For a set of recovery steps A_k , this requirement is expressed by the condition:

$$\sum_{a \in A_k} \Delta d(a) > 0. \quad (10)$$

transmitted or received to complete state synchronization during the transition.

The window feasibility of the recovery step in the k window is determined by the predicate

$$\varphi(a, k) = 1 \Leftrightarrow \bar{r}(a) \leq \bar{R}_k \wedge g(a) \leq C_k, \quad (7)$$

where \bar{R}_k is the vector of available platform resources at the beginning of the window W_k .

In (7), the inequality $\bar{r}(a) \leq \bar{R}_k$ is evaluated component-by-component. Mechanism (7) is key to filtering out steps for which, within the current connectivity window, it is impossible to complete service state synchronization or the required platform resources are unavailable.

The expected reduction in the deficit for the recovery step (the recovery step effect) is estimated through the reduction in the deficit of fulfilling the threshold requirement (2). For the step $a = (s, p \rightarrow p+1)$, the expected reduction in the deficit is introduced:

Formula (10) is directly related to the definition of $d(a)$ (8) and the deficit $d_s(t)$ (2) and is used as a criterion that the window W_k has ensured an actual reduction in the deficit of critical services. In the context of survivability, condition (10) specifies an operational requirement for recovery as the process of returning the ISMP to the region of admissible states of critical services. In particular, the fulfillment of (10) within connectivity windows means that the ISMP changes its configuration and ensures a measurable reduction in the shortfall of meeting threshold requirements for service quality metrics, i.e., it shortens the intervals during which survivability is achieved solely through controlled degradation.

In the case where the set of recovery steps that can be performed within the current connectivity window according to (7) is empty due to a resource shortage, it is permissible to employ subordinate controlled degradation actions of non-critical components to free up resources. Let B_k be the set of degradation actions in the k -th window; each action $b \in B_k$ is associated with a resource release vector $\bar{u}(b) \in \mathbb{R}_+^m$.

Then the resource constraint in (9) is refined as follows:

$$\sum_{a \in A_k} \bar{r}(a) \leq \bar{R}_k + \sum_{b \in B_k} \bar{u}(b). \quad (11)$$

Formula (11) allows us to formally describe that degradation actions are used solely to ensure the feasibility of recovery and are not considered an independent goal.

To ensure that degradation does not compromise data processing correctness, a predicate of degradation action admissibility is introduced: $\psi(b) = 1$, which is interpreted as the satisfaction of correctness invariants and error constraints. In general form, this is specified by a system of error budgets:

$$\psi(b) = 1 \Leftrightarrow \forall j \in \{1, \dots, J\} : e_j(b) \leq E_j, \quad (12)$$

where $e_j(b)$ is the estimate of the increase in the j -th component of the error or the risk of violating invariants caused by the execution of the degradation action b ; E_j is the permissible limit value of $e_j(b)$.

Formula (12) is used as a constraint on the selection of the set B_k in each window.

5. Research results and their discussion

5.1. Research scenarios and initial experimental conditions

The method is evaluated in a discrete operational cycle with intermittent connectivity, described by a sequence of windows W_k (4) and their capacities C_k (5). Within each window, steps a to restore critical services defined as profile increases (6) may be performed, provided that window feasibility $\varphi(a, k) = 1$ is satisfied according to (7). Service quality is modeled via a profile characteristic $q_s(t) = \bar{q}_s(p_s(t))$ (3), and a violation of the threshold requirement for the service quality indicator is recorded as a deficit $d_s(t)$ (2). When forming the set of recovery steps A_k , aggregate resource constraints and state coordination (9) are used, and the criterion for the actual reduction of the deficit within the window is specified by condition (10).

For the study, a set of scenarios was developed that replicate combinations of resource shortages and window connectivity typical of mobile platforms:

- scenarios with critical connectivity constraints.

The platform's resource vector \bar{R}_k is relatively stable;

however, the windows W_k are short and of variable duration, and capacity C_k varies from window to window. Under these conditions, the key factor is the limit on the amount of data required for state reconciliation, which is specified by the second part of predicate $g(a) \leq C_k$ (7). The goal of this group of scenarios is to verify that the method generates recovery steps only if it is possible to complete service state synchronization within the current connectivity window and ensures a reduction in the deficit provided that at least one step is performed within the current connectivity window;

- scenarios with critical resource constraints. Windows W_k are sufficiently long and C_k generally do not constitute a bottleneck; however \bar{R}_k decrease due to load or degradation of the mobile platform on which the information system operates. In this group, the resource component of predicate (7) and the satisfaction of the aggregate constraint (9) are decisive. Separately, scenarios of a sharp decrease in available resources and scenarios of gradual resource degradation over time are simulated, allowing us to evaluate the method's behavior as the proportion of windows increases in which the set of steps performed within the current connectivity window narrows;

- combined scenarios. At the same time, C_k and \bar{R}_k change, and a shortfall in the threshold requirements for service quality metrics occurs in a cascading manner as a result of a degradation episode. Such modes are closest to real-world operation: within connectivity windows, only small amounts of state coordination are possible, and within platform windows, the platform has limited resources for performing recovery transitions. For this group of scenarios, the fulfillment of condition (10) is verified without the accumulation of unfinished recovery transitions, as well as the reduction of the integral deficit of critical services, determined based on $d_s(t)$ (2) and $\Delta d(a)$ (8).

Figure 1 shows the implementation of window-based connectivity for a mobile platform: the intervals W_k have variable durations and are spaced apart in time, and the capacity C_k varies significantly between windows, reflecting the non-stationarity of throughput within the connectivity windows. It is precisely this variability that imposes operational constraints on service state synchronization and necessitates verifying the feasibility of completing service state synchronization within each current connectivity window in accordance with (5) and (7).

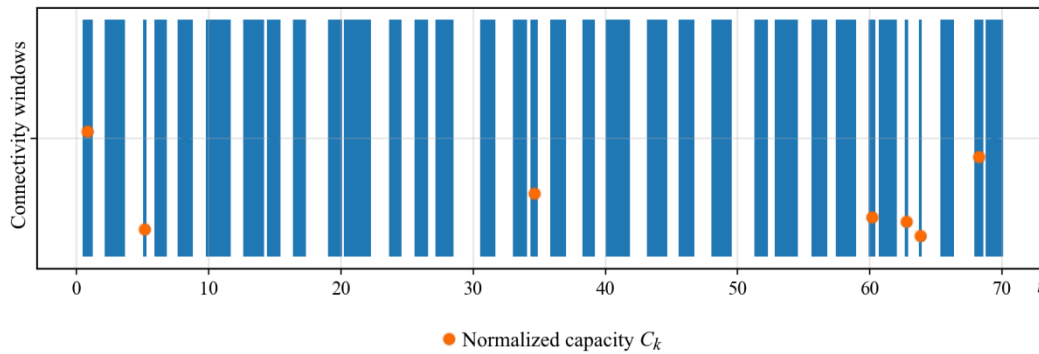


Fig. 1. Connectivity windows W_k and normalized capacity C_k for service state coordination

The initial conditions for all scenarios are defined as the state following a degradation episode. At the start of the simulation, at least one critical service is in a profile $p_s(0)$ for which the deficit $d_s(0)$ (2) is positive, indicating a violation of the service quality threshold requirement. Non-critical services can be transferred to degraded profiles, which is interpreted as a controlled degradation that has already been performed, freeing up part of the resource to maintain the minimum acceptable level of performance of the critical service core. Subsequently, the recovery method operates by forming sets A_k of steps of the form (6), taking into account feasibility (7) and aggregate constraints (9). The initial conditions for all scenarios are defined as the state following a degradation event.

In combined scenarios, the possibility of secondary resource release through controlled degradation is additionally provided for, which is formally accounted for by refining the resource balance (11). Such actions are permitted provided that the feasibility predicate $\psi(b)=1$, defined by the constraints (12), is satisfied. This ensures that degradation is used as an auxiliary mechanism to ensure the feasibility of restoration and excludes the use of degradation actions for the purpose of deliberately altering restoration performance indicators.

5.2. Performance indicators for restoration and methods for calculating them

The method is evaluated based on metrics that directly rely on the shortfall in meeting the threshold requirements for service quality metrics $d_s(t)$ (2), the profile-based quality model $q_s(\cdot)$ (3), the definition of the recovery step (6), the window feasibility

predicate $\varphi(a,k)$ (7), the step effect $\Delta d(a)$ (8), and the condition for window-based deficit reduction within the window (10). A finite simulation period is considered, defined by the sequence K of connectivity windows W_k (4), $k = \overline{1, K}$.

The total deficit of critical services over the simulation period is defined as an integral (in discrete form – cumulative) indicator:

$$D_\Sigma = \sum_{k=1}^K \sum_{s \in S_c} d_s(\tau_k), \quad (13)$$

where $d_s(\tau_k)$ is calculated according to (2) and using $q_s(\tau_k)$ or the tabulated relationship (3). Formula (13) is used to compare methods based on the cumulative deficit of critical services across a sequence of windows.

To account for the varying importance of critical services, a weighted variant is introduced:

$$D_w = \sum_{k=1}^K \sum_{s \in S_c} w_s d_s(\tau_k), \quad w_s > 0, \quad (14)$$

where the weights w_s are set according to the priority scale of critical services. Metric (14) is applied if it is necessary to emphasize the contribution of individual critical services without changing the definition of the deficit (2).

Next, it is necessary to formalize the window feasibility metrics and the satisfaction of feasibility conditions during recovery execution. For each window W_k , a set of potentially feasible recovery steps is determined:

$$F_k = \{a : (s, p \rightarrow p+1) \text{ and } \varphi(a,k)=1\}. \quad (15)$$

Formula (15) establishes that feasibility is defined as a combination of resource feasibility and the ability to complete service state coordination within the current connectivity window.

The proportion of windows in which there is at least one feasible recovery step is defined by the recovery availability metric:

$$\rho_F = \frac{1}{K} \sum_{k=1}^K \mathbf{1}(|F_k| > 0), \quad (16)$$

where $\mathbf{1}(\cdot)$ is an indicator function. Metric (16) characterizes the extent to which environmental conditions (resources and connectivity) allow recovery to be performed, regardless of which actions are selected by the method.

Let A_k be the set of recovery steps actually performed by the method within the window W_k . Compliance with the window feasibility conditions during recovery execution is evaluated by the fraction of steps that are feasible within the current connectivity window as defined in (7):

$$\rho_\varphi = \frac{\sum_{k=1}^K \sum_{a \in A_k} \mathbf{1}(\varphi(a, k) = 1)}{\sum_{k=1}^K |A_k|}. \quad (17)$$

For a method that generates A_k taking into account (7) and (9), $\rho_\varphi = 1$ is expected; whereas for the basic comparison modes, indicator (17) provides a quantitative measure of the occurrence of non-terminating recovery transitions.

Separately, an indicator of state coordination capacity excess is introduced, which is relevant specifically for modes that do not account for state coordination completion conditions within the current connectivity window:

$$G_\varphi = \sum_{k=1}^K \max \left\{ 0, \sum_{a \in A_k} g(a) - C_k \right\}. \quad (18)$$

Indicator (18) is not a substitute for (17), but complements it with a quantitative assessment of the volume of data required for service state coordination that cannot be transmitted within the connectivity windows.

Since the reduction of the deficit within the window is determined by $(\Delta d(a))$ (8) and condition (10), an indicator of the reduction of the deficit within the window is introduced:

$$I_k = \mathbf{1} \left(\sum_{a \in A_k} \Delta d(a) > 0 \right), \quad (19)$$

which is a formalized representation of criterion (10). The proportion of windows in which a deficit reduction is observed is defined as

$$\rho_P = \frac{1}{K} \sum_{k=1}^K I_k. \quad (20)$$

Indicator (20) is applied together with (16): if ρ_F is high but ρ_P is low, this means that steps were taken, but the method did not ensure a reduction in the deficit in the corresponding windows.

For an operational interpretation of the deficit reduction, the reduction in the total deficit of critical services within the window is also used:

$$\Delta D_k = \sum_{s \in S_c} (d_s(\tau_k) - d_s(\tau_k + \Delta_k)). \quad (21)$$

Indicator (21) is used as a consistency check: in a correctly constructed model and when determining $\Delta d(a)$ (8), indicator (19) must correlate with the sign of ΔD_k .

The recovery time is estimated as the moment when the zero-deficit state $d_s(t)$ is reached for each critical service. For the service $s \in S_c$, the recovery time is defined as:

$$T_s = \min \{ \tau_k : d_s(\tau_k) = 0 \}, \quad (22)$$

where τ_k represents the start of the windows (4).

For the ISMP as a whole, an aggregate indicator is used:

$$T_{\max} = \max_{s \in S_c} T_s, \quad (23)$$

which reflects the time required to restore the full critical set of services. Formulas (22)–(23) are used in conjunction with the integral indicators (13)–(14): the former evaluate the restoration rate, while the latter evaluate the cumulative deficit characteristic over the entire simulation period.

The method for calculating the above indicators is as follows. Indicators (13)–(23) are calculated according to the simulation protocol, in which the following are recorded for each window W_k : initial profiles $p_s(\tau_k)$, selected recovery steps A_k , degradation actions if present B_k , available resources \bar{R}_k , capacity C_k , and values $q_s(\tau_k)$ or tabulated values $\bar{q}_s(p_s(\tau_k))$. Next:

Step 1. Deficits $d_s(\tau_k)$ are determined $d_s(t)$ (2) using $q_s(t)$ (3).

Step 2. For each step $a \in A_k$, $\varphi(a, k)$ (7) and $\Delta d(a)$ (8) are calculated.

Step 3. F_k (15) is formed and ρ_F (16) is calculated.

Step 4. Using A_k , ρ_φ (17) and G_φ (18) are calculated.

Step 5. I_k (19), ρ_p (20), and ΔD_k (21) are calculated.

Step 6. Determine T_s (22), T_{\max} (23), and the integrated indicators D_Σ (13) and D_w (14).

The presented system of indicators allows us to distinguish three aspects of recovery quality: the presence of feasible opportunities in the environment ρ_F ; compliance with feasibility conditions and conditions for completing state alignment during the execution of recovery actions ρ_φ , G_φ ; reduction of the deficit and recovery speed ρ_p , ΔD_k , T_{\max} with a direct link to the deficit $q_s(t)$ and the step effect $\Delta d(a)$.

5.3. Basic comparison modes and conditions for correct matching

The comparison was performed using a set of basic modes that differ in the rule for forming sets of recovery steps A_k in the window W_k (4) and in the method of accounting for the deficit $d_s(t)$ (2), the step effect $\Delta d(a)$ (8), and window feasibility $\varphi(a,k)$ (7). In all modes, the recovery steps take the form (6), i.e., they correspond to raising the critical service profile by one level.

Thus, the basic modes are:

1. Mode R0 (passive control). In each window, recovery steps are not performed, i.e., $A_k = \emptyset$. This mode is used as the lower bound for deficit reduction metrics (19)–(21), recovery time (22)–(23), and for estimating cumulative deficit (13)–(14).

2. Mode R1 (resource-constrained recovery without considering the possibility of completing state coordination within the connectivity window). The set A_k is formed from the steps (6) with a focus on maximizing the total effect (8) under the constraint of only the resource part (9). The window capacity C_k and the condition $g(a) \leq C_k$, which is part of predicate (7), are intentionally disregarded. The consequences of this simplification are captured by indicators (17)–(18), where G_φ (18) characterizes the accumulated volume of data for state synchronization that cannot be transmitted within the connectivity windows.

3. Mode R2 (restoration with a constraint on the extent of state coordination and a priority for minimal

coordination). The set A_k consists only of steps for which the condition $g(a) \leq C_k$ holds, in accordance with the second part of (7). Selection is performed with a priority for minimizing $\sum_{a \in A_k} g(a)$, provided that the deficit does not increase; that is, steps with a small $g(a)$ are preferred even if $\Delta d(a)$ is small (8). The resource feasibility is controlled via the first constraint (9). This mode contrasts with R1 and demonstrates the effect of the priority of minimum coordination in the absence of a focus on reducing the deficit on ρ_p (20), ΔD_k (13), T_{\max} (23).

4. Mode R3 (fixed priority of critical services). A fixed order of critical services is specified: s_1, s_2, \dots by importance. In each window, steps (6) are sequentially considered for s_1 , then for s_2 , and so on, with the first step added to A_k for which $\varphi(a,k)=1$ is satisfied and the aggregate constraints (9) are not violated. Mode R3 does not optimize the total effect (8) and provides a reproducible behavioral benchmark where the decision is determined by a fixed order of services.

These modes cover three typical simplifications: ignoring the constraint on the possibility of completing state coordination within the connectivity window (R1), the absence of a focus on reducing the deficit subject to the coordination constraint (R2), and the rejection of adaptive selection in favor of a fixed order of services (R3). Their comparison with the method using deficit (2), window feasibility (7), and condition (10) is performed according to indicators (13)–(23).

5.4. Results for the window feasibility of recovery steps

This subsection analyzes compliance with the window feasibility conditions of recovery steps in terms of the fulfillment of $\varphi(a,k)$ (7) and consistency with the aggregate constraints (9). The main focus is on three aspects: the presence in the mobile platform's computing environment of at least one feasible step F_k according to (15) and the proportion of such windows ρ_F according to (16); the proportion of actually executed steps that are feasible within the current connectivity window, ρ_φ (17); the presence or absence of coordination capacity exceedances, G_φ (18).

Compliance with the feasibility conditions of recovery steps is defined as follows. For the proposed method, the sets of steps A_k are formed such that each selected step belongs to the set of feasible steps F_k (15), and the set of steps satisfies constraint (9). Given the definition of the predicate $\varphi(a, k)$ (7), this means that for all k and all $a \in A_k$, $\varphi(a, k) = 1$ holds. For the indicator (17), this yields $\rho_\varphi = 1$, and for the accumulated capacity excess (18) – the absence of excesses, i.e., $G_\varphi = 0$.

Fig. 2 shows the relationship between the window capacity C_k and the actual amount of state coordination used for two control rules. For the proposed method,

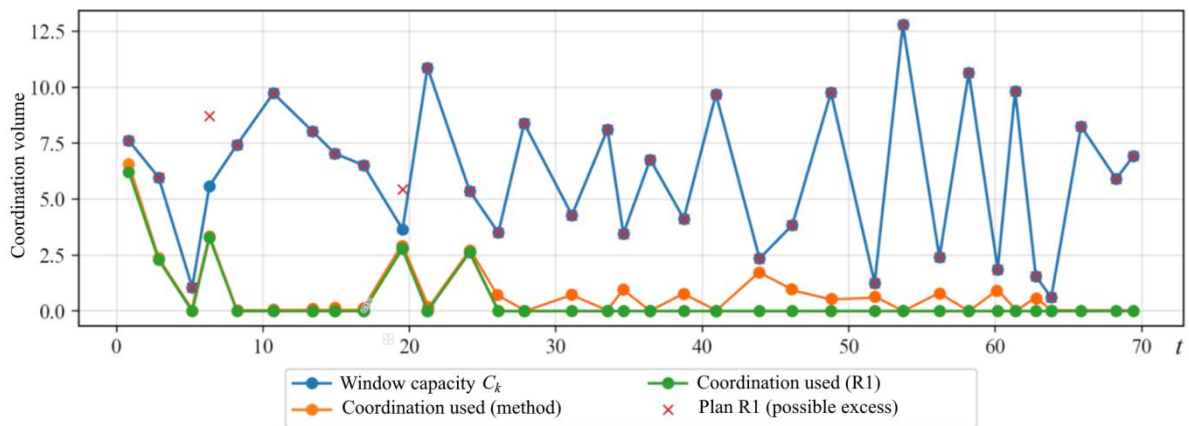


Fig. 2. Comparison of the capacity C_k use for state coordination

This result is a direct consequence of constructing the A_k taking into account (7) and (9) and is confirmed in experiments by the consistent absence of unfinished state-consistency transitions.

For mode R1, in which the window capacity C_k (5) and the synchronization constraint are not taken into account during the formation of A_k , a distinct property is observed: immediately after the execution of the resource part (9), windows arise for which

$$\sum_{a \in A_k} g(a) > C_k,$$

which is directly captured by the indicator $G_\varphi > 0$ in (18) and reduces ρ_φ in (17).

Thus, a comparison of (17)–(18) demonstrates that ignoring the condition $g(a) \leq C_k$, which determines the possibility of completing service state synchronization within the current connectivity window, generates errors that are critical for a mobile platform with intermittent connectivity.

the coordination volume is consistent with the constraint $\sum_{a \in A_k} g(a) \leq C_k$ (9). Therefore, capacity usage does not exceed the window limits. For the base mode R1, the planned coordination volume is additionally shown, which may exceed C_k . This illustrates the consequence of approaches that do not account for the condition $g(a) \leq C_k$, which determines the possibility of completing service state coordination within the current connectivity window during the action formation phase: a portion of the planned recovery transitions remains incomplete within the window, which is related to the second component of the predicate $\varphi(a, k)$ (7).

Modes R2 and R3, unlike R1, explicitly check the condition $g(a) \leq C_k$ in the step selection procedure and adhere to the aggregate constraints (9). Therefore, for them, the metrics ρ_φ (17) and G_φ (18) also indicate correct fulfillment of the feasibility conditions. The differences between the method and these regimes manifest in how the feasibility steps translate into a reduction in the deficit within the window, which is consistent with the indicator I_k (19) and the ratio ρ_p (20).

Next, we consider the availability of recovery within connectivity windows as a property of the environment. The indicator ρ_F (16) characterizes the presence of a recovery possibility in the computational environment: does there exist at least one step a of the form (6) for which $\varphi(a, k) = 1$ (7), i.e., does the ISTM allow recovery to be performed in this window. Analysis of the previously introduced scenarios reveals two consistent patterns.

First, in scenarios with determining connectivity constraints, ρ_F is predominantly determined by the condition $g(a) \leq C_k$ as a component of predicate (7), i.e., by the ratio between the data volume required to reconcile the state $g(a)$ and the capacity C_k (5). Even with sufficient resources \bar{R}_k , the sets F_k (15) are reduced to steps with small $g(a)$, and for small C_k , they may become empty, which reduces ρ_F in (16).

Second, in scenarios with deterministic resource constraints, ρ_F is determined primarily by the condition $\bar{r}(a) \leq \bar{R}_k$ as a component of predicate (7) and by the resource constraints in (9): the set F_k (15) may be non-empty for $g(a) \leq C_k$, but recovery steps become unavailable due to a shortage of \bar{R}_k for any candidate a . In combined scenarios, these two effects manifest simultaneously, underscoring the appropriateness of precisely predicate (7) as a common condition for window feasibility.

To interpret which constraint actually determines the composition of the set A_k , it is advisable to evaluate the residual capacity after selecting steps. For each window, a residual state-consistency capacity is introduced:

$$S_k^g = C_k - \sum_{a \in A_k} g(a), \quad (24)$$

which must be non-negative for modes that satisfy the second constraint (9), and characterizes how fully the capacity of the window C_k (5) is utilized.

Similarly, the vector residual state-consistency capacity is introduced:

$$\bar{S}_k^r = \bar{R}_k - \sum_{a \in A_k} \bar{r}(a), \quad (25)$$

which is non-negative component-wise under the first constraint (9) and indicates whether the solution was constrained by the platform resource.

To reduce capacity utilization to a dimensionless quantity, the window utilization coefficient is applied:

$$u_k = \frac{\sum_{a \in A_k} g(a)}{C_k}, \quad C_k > 0. \quad (26)$$

$u_k \leq 1$ applied to modes that satisfy constraint (9). $u_k > 1$ is possible for modes that do not consider this constraint during formation A_k .

The obtained values (24)–(26) provide an unambiguous interpretation of the modes of the

previously introduced scenarios. In scenarios with determining connectivity constraints, the proposed method is characterized by binding the coordination constraint itself: the residual state coordination capacity S_k^g (24) is small in a significant portion of the windows, while u_k (26) approaches unity. This means that the method utilizes the capacity of the state coordination window and generates recovery transitions within the bounds of constraint (9). In contrast, in scenarios with resource-constrained conditions, the resource component is more frequently constrained: the residual resource vector \bar{S}_k^r (25) has small components, while S_k^g (24) may remain relatively large, i.e., the window capacity is not fully utilized due to a lack of resources for steps (6).

For mode R1, the coefficient u_k (26) exceeds 1 in some windows, and the magnitude of this excess is accumulated by the indicator G_φ (18). This is direct quantitative confirmation that the formation of the A_k without accounting for constraint (9) leads to incomplete coordination exchanges within the connectivity windows and repeated violations of the threshold requirements for service quality indicators.

The results obtained show that the proposed method ensures compliance with the window feasibility conditions for recovery steps: the executed steps are feasible within the current connectivity window according to (7) and satisfy the aggregate constraints (9), as reflected in ρ_φ (17) and the absence of violations of G_φ (18). The indicator ρ_F in (16) distinguishes situations where recovery is possible within the window from situations where the environment has no feasible steps according to (15). The residual state coordination capacity (24), the residual resource vector (25), and the coefficient (26) specify which constraint is decisive in each scenario class.

5.5. Results for window-based reduction of critical service shortages

This subsection analyzes the method's ability to convert the recovery capabilities available on the mobile platform into an actual reduction in the deficit of critical services. To this end, the deficit reduction indicator within the window I_k (19), the proportion of windows in which a reduction in the deficit is observed ρ_p (20), and the magnitude of the reduction in the total deficit of

critical services within the window ΔD_k (21) are used. Additionally, the recovery availability metric from ρ_F (16) is used, which allows distinguishing environmental constraints from the quality of management decisions.

Since ρ_F (16) represents the proportion of windows in which there is at least one recovery step that can be performed within the current connectivity window, and ρ_p (20) reflects the proportion of windows in which the method resulted in an actual reduction of the deficit (19), it is appropriate to estimate the conversion rate of feasibility into deficit reduction:

$$\eta = \frac{\rho_p}{\rho_F + \varepsilon}, \quad (27)$$

where $\varepsilon > 0$ is a small number to avoid division by zero in the case of $\rho_F = 0$.

Formula (27) is used as a generalized indicator of how effectively the method utilizes windows in which recovery is possible.

In scenarios with determining connectivity constraints, the set F_k (15) is often restricted to steps with a small amount of coordination $g(a)$ due to the constraint $g(a) \leq C_k$, which is part of predicate (7). Under such conditions, the indicator ρ_F (16) decreases, reflecting the environmental constraints. The key point is that windows with $|F_k| > 0$ create an opportunity to reduce the deficit, which is realized through the selection A_k satisfying condition (10), and thus $I_k = 1$ (19). This is why η (27) is interpreted as a characteristic of the formation rule A_k .

For the proposed method, the reduction of the deficit in the window is ensured by the selection construct

A_k , which combines the feasibility requirement (7), the total constraints (9), and condition (10). This means that if there is at least one step a from $\varphi(a, k) = 1$ and $\Delta d(a) > 0$ (8), the method forms A_k such that $\sum_{a \in A_k} \Delta d(a) > 0$ (10), i.e., ensures $I_k = 1$ (19). In this sense, progress is a controlled property of the solution, related to deficits (2) and step effects (8).

Next, we consider the magnitude of the deficit reduction within the window and its stability. The indicator I_k (19) captures the fact of the deficit reduction but does not reflect its magnitude. Therefore, the analysis is supplemented by the average magnitude of the positive deficit reduction:

$$\overline{\Delta D^+} = \frac{\sum_{k=1}^K \max\{0, \Delta D_k\}}{\sum_{k=1}^K I_k + \varepsilon}. \quad (28)$$

Formula (28) characterizes the average magnitude of the reduction in the total deficit of critical services in those windows where a deficit reduction is recorded.

Fig. 3 shows the dynamics of the total deficit of critical services and its accumulation over time, reflecting the integral characteristic of the system's operation in degraded modes. It can be seen that after a series of windows in which completion steps of recovery are possible, the total deficit decreases to zero and remains there, while the accumulated deficit reaches a plateau, corresponding to the completion of recovery. In this scenario implementation, the curves for the method and R1 are visually similar, which is interpreted as the decisive influence of environmental constraints over differences in control rules. That is why the indicators ρ_F and ρ_p in Fig. 4 are additionally analyzed.

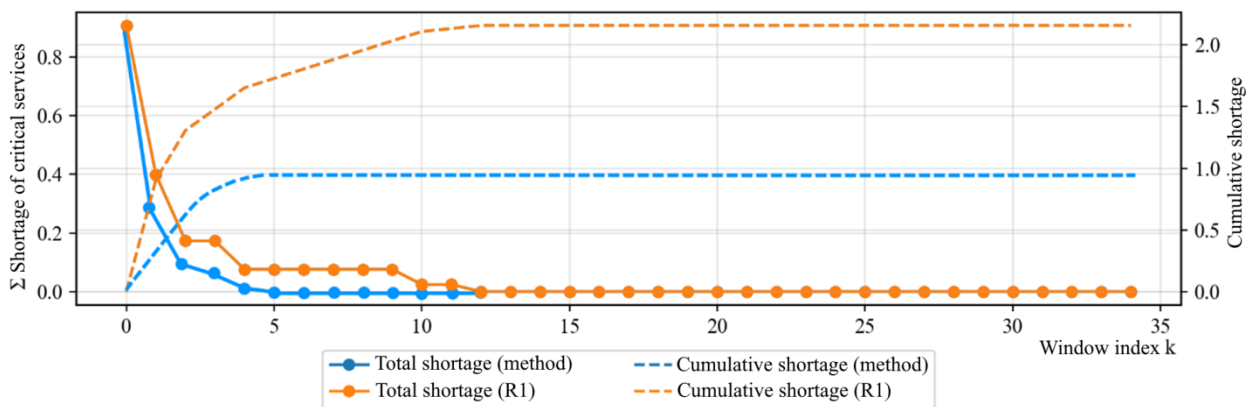


Fig. 3. Total and cumulative deficit of critical services (comparison)

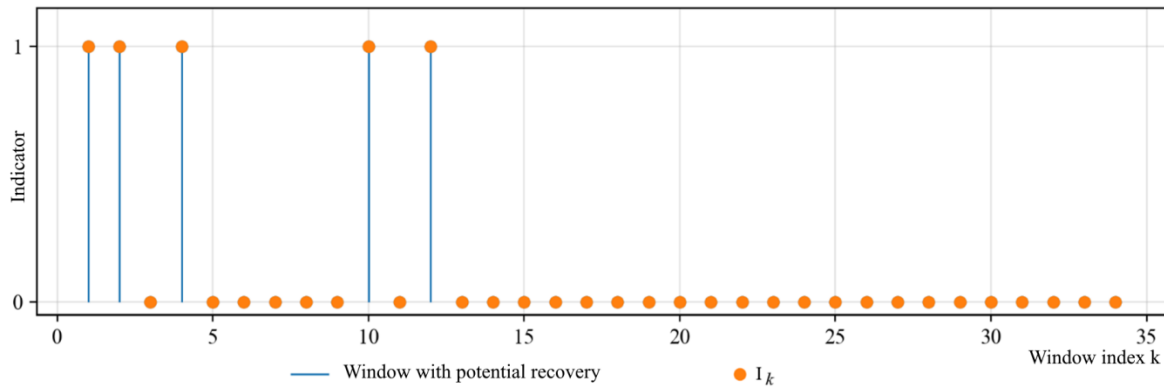


Fig. 4. Indicators of the separation between the possibility of recovery and the actual reduction of the deficit in connectivity windows

Fig. 4 demonstrates the separation of the possibility of recovery and the actual reduction of the deficit in connectivity windows. The vertical markers correspond to windows in which there is at least one recovery step ($\exists a: \varphi(a, k) = 1$), i.e., the environment allows recovery to be performed. The points I_k reflect the actual reduction of the deficit in the window, i.e., the realization of the reduction of the critical service deficit. In this experiment, ρ_F is small (of the order of 0.15), which implies rare windows with recovery potential; accordingly, ρ_p is bounded from above and turns out to be close to ρ_F , which is consistent with the structural constraint $\rho_p < \rho_F$ and explains the shape of the shortage curves in Fig. 3.

In modes that do not account for the constraint on the ability to complete service state coordination within the current connectivity window, a discrepancy arises between the estimated reduction in the deficit based on the aggregate effect (8) and the actual change in the deficit (21). If the synchronization constraint $\sum_{a \in A_k} g(a) \leq C_k$ (9) is violated within the window, as reflected by the excess $G_\varphi > 0$ (18), part of the planned synchronization exchanges cannot be completed. In such a situation, $\sum_{a \in A_k} \Delta d(a)$ may remain positive, but the actual reduction in the deficit ΔD_k (21) becomes unstable and may approach zero in individual windows. This is why it is advisable to analyze the reduction in the deficit using the pair of indicators I_k (19) and ΔD_k (21), rather than solely through step effects (8).

For the proposed method, this discrepancy is eliminated by the feasibility constraint (7) and the total constraints (9), which is consistent with the previous results via ρ_φ (17) and G_φ (18). Therefore, the progress

value in the window according to (21) becomes consistent with the cumulative effect of the selected steps $\sum_{a \in A_k} \Delta d(a)$ in (10), and the average progress value $\overline{\Delta D^+}$ in (28) reflects precisely the effectiveness of the step selection.

The reduction in the deficit is distributed among critical services as follows. Since ΔD_k (21) aggregates the contributions of all critical services, it is important to ensure that recovery does not focus on a single service at the expense of keeping others in a deficit state for an extended period. To achieve this, individual recovery times T_s (22) and the aggregated time T_{\max} (23) are used. If the method reduces the deficit at I_k (19) and ΔD_k (21), but T_{\max} (23) remains high, this means that the reduction in the deficit across the windows is achieved unevenly across services, and certain critical services may be restored significantly later.

For the R3 fixed-priority mode, this situation is typical by design: deficit reduction is prioritized for services with the highest priority, which may increase ρ_p (20) in early windows, but increases the T_{\max} (23) and the cumulative deficit D_Σ (13) due to the prolonged persistence of deficits in lower-priority critical services. In the proposed method, this is addressed by basing the step selection criterion on the deficit (2) and the step effect (8), while priority is determined by the step's contribution to reducing the deficit and its window feasibility (7). This is reflected in an increase in ρ_p (20) and a decrease in T_{\max} (23) in the absence of coordination violations G_φ (18).

Next, the relationship between the reduction of the deficit within the connectivity windows and the integral deficit indicators is formalized. The reduction of the deficit according to (19), (21) is local in

nature, whereas the integral estimate is given by the accumulated deficit D_{Σ} (13) and the weighted deficit D_w (14). The reduction of D_{Σ} (13) is determined by ρ_p (20) and $\overline{\Delta D^+}$ (28). When comparing modes, it is important to consider modes in which the deficit reduction is achieved through frequent small steps, and modes in which steps with a large expected effect are applied without controlling the possibility of completing state convergence within the window, leading to the occurrence of overshoots G_{ρ} (18) and unstable dynamics ΔD_k (21).

As a result, the results are consistent with the logic of constructing the indicators. Compliance with the window feasibility conditions (17)–(18) ensures that the planned effect of the steps (8) corresponds to the actual change in the deficit (21). The presence of opportunities in the environment is captured via ρ_f (16), and the effectiveness of their use is assessed via ρ_p (20) and the coefficient η (27). The total impact on the quality of recovery is reflected through the integral indicators (13)–(14) and the recovery time (22)–(23), which are direct functions of the deficit (2) and are therefore correctly linked to condition (10).

5.6. Sensitivity analysis of connectivity parameters and state coordination volume

The sensitivity of critical service recovery under intermittent connectivity is determined by how the sets of recovery steps performed within the current connectivity window F_k (15) and, accordingly, the indicators ρ_f (16), ρ_p (20), η (27), D_{Σ} (13) and T_{\max} (23) while varying the window parameters W_k (4), their capacity C_k (5), and the matching characteristics $g(a)$ included in (7) and (9). The following analysis is conducted at the level of parameterized series, in which individual factors are varied while other scenario parameters remain constant.

To isolate the effects, scaling factors are introduced that are applied to the environment parameters and recovery steps:

1. Scaling of the connectivity window sizes:

$$C_k(\alpha) = \alpha C_k, \quad \alpha > 0. \quad (29)$$

Equation (29) models the change in the capacity of connectivity windows caused by a change in the

window's throughput or duration, assuming constant recovery profiles.

2. Scaling of the volume of recovery step state coordination:

$$g_{\beta}(a) = \beta g(a), \quad \beta > 0. \quad (30)$$

Formula (30) is interpreted as a change in the volume of data required for state synchronization during the execution of recovery steps, for example, due to varying state compactness, different incremental synchronization policies, or varying levels of detail during recovery.

3. Scaling of the available mobile platform resource:

$$\bar{R}_k(\gamma) = \gamma \bar{R}_k, \quad \gamma > 0. \quad (31)$$

Formula (31) reflects different mobile platform load modes or different hardware configurations.

In the subsequent sensitivity analysis, all dependencies are derived as a consequence of defining window feasibility (7) and constructing the sets F_k (15).

Next, the monotonicity of recovery availability within connectivity windows is formalized. The set of feasible steps F_k (15) is considered with respect to the scaling parameters (29)–(31). The following monotonicities directly follow from the definition of the predicate (7):

1. For $\alpha_2 \geq \alpha_1$, the following is enabled:

$$F_k(C(\alpha_1)) \subseteq F_k(C(\alpha_2)), \quad (32)$$

since the condition $g(a) \leq C_k(\alpha)$ in (7) becomes weaker as α increases according to (29), whereas the resource part of (7) remains unchanged. Equation (32) implies that increasing the window size cannot reduce the set of available recovery steps.

2. For $\beta_2 \geq \beta_1$, the opposite inclusion holds:

$$F_k(g\beta_2) \subseteq F_k(g\beta_1), \quad (33)$$

because the condition $g_{\beta}(a) \leq C_k$ in (7) becomes stricter as β increases according to (30). Equation (33) establishes that increasing the scope of recovery step coordination narrows the set of feasible steps.

3. For $\gamma_2 \geq \gamma_1$, the following holds

$$F_k(\bar{R}(\gamma_1)) \subseteq F_k(\bar{R}(\gamma_2)), \quad (34)$$

since the component-wise condition $\bar{r}(a) \leq \bar{R}_k(\gamma)$ in (7) weakens as γ increases according to (31).

Since the recovery feasibility indicator ρ_f (16) is defined by the fact that the set F_k (15) is non-empty, it

follows from (32) and (34) that ρ_F is non-decreasing with respect to α and γ , and from (33) the ρ_F does not increase with respect to β . This is important for interpretation: the variation of ρ_F is, first and foremost, a characteristic of the presence of conditions for recovery in the environment.

The actual reduction in the deficit within the window is estimated by the indicator I_k (19) and the ratio ρ_p (20), while the conversion is estimated by the coefficient η (27). The definitions imply the constraint

$$\rho_p \leq \rho_F, \quad (35)$$

since a reduction in the deficit within the window is possible only if there is at least one feasible recovery step (15) with $\varphi(a,k)=1$ (7), i.e., when $|F_k| > 0$ (16). Inequality (35) is structural and does not depend on the specific selection method A_k .

At the same time, ρ_p and η are determined by the value of ρ_F and the rule for using the set of candidates F_k in the window. The key parameter is the ratio between $g(a)$ and C_k , which defines the recovery resolution. If for most candidates a satisfies $g(a) \approx C_k$, then the set F_k in (15) often contains only 1–2 steps, and the choice of A_k is close to the deterministic bound (7). In this mode, η (27) approaches 1 provided that these steps satisfy $\Delta d(a) > 0$ (8), since then the satisfaction of condition (10) becomes inevitable.

Another limiting regime arises when $g(a) \ll C_k$ for many a . Then F_k (15) increases, and the total constraint (9) becomes dominant. In this case, the sensitivity of ρ_p and η is determined by the ability of the formation rule A_k to ensure $\sum_{a \in A_k} \Delta d(a) > 0$ (10) while satisfying (9). In this mode, the influence of the parameter β (30) is most noticeable: an increase in $g_\beta(a)$ narrows F_k (33) and increases competition between steps for window capacity, which may reduce ρ_p (20) for a constant ρ_F (16) if steps with a small $\Delta d(a)$ (8) remain predominantly feasible.

The cumulative deficit D_Σ (13) and the recovery time T_{\max} (23) are integral consequences of the reduction in the deficit within connectivity windows and its magnitude. From the definition ΔD_k (21) and summation

over windows, it follows that the decrease in D_Σ is ensured simultaneously by an increase in the proportion of windows with a decrease in the deficit ρ_p (20) and an increase in the average positive decrease in the deficit $\overline{\Delta D^+}$ (28). Therefore, the parameters α , β , γ (29)–(31) affect D_Σ and T_{\max} primarily through two channels:

- by changing the proportion of windows in which recovery is possible (via ρ_F (16) based on the monotonicities (32)–(34));

- by changing the magnitude of the deficit reduction within the window (via which specific steps remain feasible in F_k (15) and how they are formed in A_k under constraint (9)).

This implies the following. A decrease in α (29) or an increase in β (30) leads to the constraint $g(a) \leq C_k$ (7) being satisfied less frequently, the sets F_k becoming narrower, ρ_F (16) decreasing, and the recovery skipping some connectivity windows, which increases D_Σ (13) and T_{\max} (23). Conversely, increasing α or decreasing β expands the sets F_k (32)–(33), which potentially reduces D_Σ and T_{\max} , provided that the formation rule A_k ensures the fulfillment of (10) without violating (9).

Analysis of (29)–(35) yields the following operational interpretation. If the environment is characterized by small C_k (5) or high variability Δ_k (4), then the assurance of recovery is determined by the ratio between the available resource \bar{R}_k and the scope of state coordination $g(a)$ (7). In this context, it is advisable to construct recovery profiles such that there are steps with small $g(a)$ and a non-zero effect $\Delta d(a)$ (8). Such steps form non-empty sets F_k (15) under strict window conditions, increase ρ_F (16), and ensure the possibility of satisfying condition (10) while maintaining the feasibility conditions (7), (9).

Thus, the sensitivity of the method to connectivity parameters and coordination volumes is structural in nature and directly follows from the definition of window feasibility (7), the sets of feasible steps (15), and the progress indicators (19)–(21). This allows the obtained results to be interpreted as a consequence of the relationship between window capacity (5), step

coordination volumes ($g(a)$), and available resources (\bar{R}_k), which together determine the domain of admissible recovery actions in each connectivity window.

5.7. Discussion of the Research Results

It is appropriate to interpret the results obtained by distinguishing between two reasons for the lack of a reduction in the shortage of critical services: environmental constraints and the characteristics of management decisions. Environmental constraints are reflected by the non-emptiness of the set of feasible steps F_k (15) and the indicator ρ_F (16), which is directly determined by the window feasibility predicate (7). The properties of a control decision are manifested in how often and by how much the method ensures a reduction in the deficit within the connectivity windows, which is recorded by I_k (19), ρ_p (20), the conversion coefficient η (27), and the values ΔD_k (21) and $\overline{\Delta D}^+$ (28). The decrease in ρ_p is interpreted in conjunction with the change in ρ_F , since recovery within the window is possible provided that $|F_k| > 0$.

From a survivability perspective, the results obtained should be interpreted as confirmation that the recovery of critical services in the ISMP is a controllable component of maintaining survivability. The metrics ρ_F (16) and ρ_p (20) reflect, respectively, the potential and realized ability of the ISMP to restore critical functions within window connectivity. The metrics D_Σ (13) and T_{\max} (23) characterize the duration and intensity of the degradation mode during which the failure to meet the threshold requirements for service quality metrics persists. In this context, compliance with the window feasibility conditions specified by $\varphi(a, k)$ (7) and the constraints (9) is a necessary condition for ensuring real-time recovery, as it eliminates incomplete recovery transitions that prolong degradation intervals and increase the risk of secondary violations of critical requirements.

It is important that the ability to complete service state synchronization within the current connectivity window, defined by the condition $g(a) \leq C_k$ as a component of $\varphi(a, k)$ (7) and the second constraint (9), is a determining constraint for the mobile platform. Ignoring this constraint manifests as the occurrence

of G_φ (18) and a decrease in ρ_φ (17) in modes that form A_k without accounting for the constraint $\sum_{a \in A_k} g(a) \leq C_k$ (9).

In a service-oriented interpretation, this implies an accumulation of uncompleted balancing exchanges and a discrepancy between the expected effect $\sum_{a \in A_k} \Delta d(a)$ (8) and the actual reduction in the deficit ΔD_k (21). In contrast, satisfying conditions (7) and (9) ensures consistency between the local estimate of the effect $\Delta d(a)$ (8) and the actual reduction in the deficit ΔD_k , which is essential for mobile environments with short connectivity windows.

The issue of recovery separability is also important. Sensitivity analysis shows that the relationship between capacity C_k (5) and state synchronization volumes $g(a)$ (7) determines the range of possible recovery steps within the window. The monotonicities (32)–(34) and the structural constraint (35) reflect that an increase in environment availability does not guarantee a reduction in the cumulative deficit D_Σ in (13): if the set of feasible steps F_k (15) grows, but most steps have a small effect $\Delta d(a)$ (8), then the reduction in the deficit may occur in a significant proportion of windows ρ_p (20), but be small in magnitude $\overline{\Delta D}^+$ (28), which increases T_{\max} in (23). If the steps are characterized by $g(a) \approx C_k$, then the choice within the window is determined by the constraint $g(a) \leq C_k$ (7), and the effectiveness of the recovery is determined by the presence of steps with $\Delta d(a) > 0$ (8). It is advisable to construct recovery profiles such that the set of steps of the form (6) contains steps with small $g(a)$ (7) and a non-zero effect $\Delta d(a)$ (8), since they ensure the non-emptiness of F_k (15) and increase ρ_F (16) in strict windowing modes.

Separately, it is worth noting the limits of applicability arising from the previously adopted abstractions. First, service quality is represented by the aggregate metric $q_s(t)$ and the threshold θ_s , while the deficit is defined by formula (2). Such a model is suitable for formalizing deficit reduction; however, in real systems, service may have a multidimensional SLA profile, and the deficit may be determined by a more complex composition of metrics. This does not alter the

feasibility logic (7) or condition (10), but requires refinement of the definition $d_s(t)$ (2) and the step effect $\Delta d(a)$ (8) for the multidimensional case.

Second, discrete profiles P_s (1) and a tabular correspondence of profile quality (3) are used. In ISMPs with continuous control or a large number of modes, a procedure for quantizing profiles and evaluating $\bar{q}_s(\cdot)$ is required. The accuracy of such a procedure will directly affect the correctness of the evaluation $\Delta d(a)$ (8).

Third, the method relies on an estimate of the window capacity C_k (5) and the state alignment volume $g(a)$ (7). In practical implementation, these values are determined via telemetry and may be subject to error. Overestimation C_k or underestimation $g(a)$ leads to steps for which state coordination completion within the current connectivity window is impossible being incorrectly identified as feasible, which manifests as an increase in G_ρ (18) and a discrepancy between $\sum_{a \in A_k} \Delta d(a)$ (10) and the actual ΔD_k (21). Underestimating C_k or overestimating $g(a)$, conversely, narrows F_k (15) and reduces ρ_F (16), which may artificially increase D_Σ (13) and T_{\max} (23) under constant environmental conditions. In applied scenarios, it is advisable to monitor the quality of estimation C_k and $g(a)$, as well as to use the residual state alignment capacity (24) and the residual resource vector (25) as indicators of approaching the feasibility boundaries.

Fourth, this study does not introduce specific constraints on the intensity of reconfigurations as a separate control mechanism. The key elements in this work are the feasibility conditions for steps within the window, defined by $(\varphi(a, k))$ (7) and the constraints (9), and condition (10). If the applied control loop requires additional stabilization regarding switching frequency, such constraints can be introduced as external regulations based on known solutions [19], [20].

The scope of application of the proposed method corresponds to architectures in which a mobile platform combines the functions of an edge computing node and a coordinator for a group of subordinate unmanned aerial vehicles (UAVs). A typical ISMP is the onboard information system of the carrier vehicle, which provides mission planning and task distribution among vehicles,

situational awareness based on the aggregation of sensor streams, navigation correction and reference map alignment, the generation of control commands and execution monitoring, and the secure exchange of telemetry for critical flight safety parameters. Intermittent connectivity in such a system is determined by flight geometry, shielding, radio channel conditions, and electronic countermeasures. This defines connectivity windows with variable duration and bandwidth for communication between the carrier aircraft, subordinate aircraft, and the ground control station, and limits the window capacity for state synchronization. During a degradation episode, the ISMP performs a controlled reduction of non-critical service profiles, specifically by reducing the update frequency of secondary telemetry streams, switching to coarser video processing models, deferring the synchronization of non-critical logs, and limiting map detail. These actions free up computational resources and bandwidth to maintain the minimum acceptable level of performance for the critical service core. Once conditions have stabilized, the task of restoring critical services to their normal profiles arises, with the feasibility of restoration determined by the availability of onboard resources and the ability to complete service state synchronization within the current connectivity window. This applies to the synchronization of map increments, updated task parameters, integrity check data, sensor aggregation parameters, and other state components that must be transmitted or received before the end of the window. Under such conditions, the application of the recovery step window feasibility predicate (7) and the condition of deficit reduction within the window (10) ensures the formation of only those transitions that elevate the critical service profile, for which state synchronization can be completed within the available window. This reduces the likelihood of accumulating incomplete reconfigurations, repeated violations of service quality thresholds, and erroneous recovery, which in a UAV group translates into a loss of coordination and reduced mission stability.

Thus, service-oriented recovery, based on the deficit (2) and the step effect (8) and subject to window feasibility (7) and aggregate constraints (9), ensures a measurable reduction of the deficit within connectivity windows, as reflected by the metrics (19)–(23) and integral measures (13)–(14). The limits of applicability are determined by the accuracy of estimating the parameters of windows (4) and (5), the characteristics of state coordination within (7), as well as the level

of detail of service profiles (1)–(3). This provides a basis for refining profile models and procedures for evaluating feasibility parameters without altering the logic of window-based recovery, as defined by equations (7), (9), and (10).

6. Conclusions and prospects for further research

This paper solves the problem of restoring critical services of an information system on a mobile platform under conditions of controlled degradation with intermittent connectivity and resource constraints. The proposed method presents recovery as a sequence of discrete steps to improve service profiles, with performance monitoring through the reduction of the shortfall in meeting threshold requirements for service quality indicators. This formulation distinguishes recovery transitions that ensure an actual reduction in the deficit from reconfigurations that do not lead to a reduction in the deficit within connectivity windows.

It is shown that the correctness of the recovery of critical ISMP services is determined by the resource feasibility of actions and the possibility of completing service state coordination within the current connectivity window. It has been established that ignoring the constraint on state coordination within the window leads to the accumulation of unfinished coordination exchanges, discrepancies between the expected effect of recovery actions and the actual dynamics of the deficit, as well as to repeated violations of threshold requirements after partial recovery.

In the context of improving the survivability of the ISMP, the method provides an operational mechanism for maintaining and restoring critical functions during a connectivity window with variable duration and bandwidth. Reducing the cumulative deficit of critical services and shortening the recovery time of the critical set correspond to a reduction in the duration of degradation episodes of critical services and an increase in the proportion of control intervals in which service quality threshold requirements are met. The method generates recovery steps based on window feasibility and on aggregate resource and state coordination constraints, ensuring recovery without accumulating unfinished transitions.

A distinctive feature of the proposed method is the introduction of a window feasibility condition for a step, which accounts for resource feasibility on the mobile platform and the possibility of completing service state coordination within the current connectivity window. This ensures the pruning of steps for

which completion of coordination within the window is impossible, and consistency between the expected reduction in the deficit and the actual change in the deficit. The evaluation metric system ensures the separation of computational environment constraints and control decision properties, as well as the comparison of recovery modes based on cumulative deficit, the magnitude of deficit reduction within the window, and recovery time.

The value of the results is determined by the method's suitability for implementation as an operational recovery control module in a window-based ISMP, particularly in onboard information systems of UAV groups. The method ensures the return of critical services to their normal profiles without accumulating unfinished coordination exchanges and while maintaining the acceptability of controlled degradation as a peripheral tool for freeing up resources. This reduces the likelihood of repeated violations of service quality thresholds following partial recovery and shortens the system stabilization time after a degradation episode.

Prospects for further research cover three areas. First, it is advisable to generalize the quality deficit from a scalar metric to a multidimensional profile of requirements for critical services with a formalized rule for aggregating deficits, while maintaining the criterion of deficit reduction within the connectivity window. Second, it is necessary to extend the window feasibility model to account for errors in estimating window capacity and the scope of state coordination [21], including robust feasibility estimates and adaptive adjustment of coordination parameters [22]. Third, it is promising to account for dependencies between services and partial state coordination modes of the ISMP [23], which will allow for the formation of recovery steps with varying granularity and improve the efficiency of using short connectivity windows while preserving the correctness of ISMP recovery.

Conflict of interest

The author declare that they have no conflict of interest with respect to this research, including financial, personal, authorship, or other conflicts that could influence the research and its results presented in this article.

Funding

The study was conducted without financial support.

Data availability

The manuscript has no associated data.

Use of artificial intelligence

The authors confirm that they did not use artificial intelligence technology in the creation of this work.

References

1. Dodonov, O., Gorbachyk, O., Kuznietsova, M. (2023), "Critical Infrastructure Resilience and Cybersecurity of Information Management Systems", *XXIII International Scientific and Practical Conference Information Technologies and Security (ITS 2023)*, pp. 1–10, available at: <https://ceur-ws.org/Vol-3887/paper1.pdf>
2. Fesenko, H., Iliashenko, O., Kharchenko, V., Leichenko, K., Sachenko, A., Scislo, L. (2024), "Methods and Software Tools for Reliable Operation of Flying LiFi Networks in Destruction Conditions", *Sensors*, Vol. 24, No. 17, 5707. DOI: <https://doi.org/10.3390/s24175707>
3. Рубан, І.В., Ткачов, В.М. (2025), "Багаторівнева модель інформаційної системи на мобільній платформі та формалізація критеріїв її живучості", *Вісник Херсонського національного технічного університету*, Т. 2, № 3(94), С. 399–409. DOI: <https://doi.org/10.35546/kntu2078-4481.2025.3.2.51>
4. Dodonov, O., Jiang, B., Dodonov, V. (2019), "Survivability Mechanisms of Complex Computer Systems Based on Common Information Space", *Computer Science*, Vol. 45, No. 7, pp. 41–45. DOI: <https://doi.org/10.19678/j.issn.1000-3428.0053440>
5. Churyumov, G., Tokarev, V., Tkachov, V., Partyka, S. (2018), "Scenario of Interaction of the Mobile Technical Objects in the Process of Transmission of Data Streams in Conditions of Impacting the Powerful Electromagnetic Field", *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, pp. 183–186. DOI: <https://doi.org/10.1109/DSMP.2018.8478539>
6. Рубан, І.В., Ткачов, В.М. (2025), "Модель міжрівневих порушень та політик відновлення інформаційної системи на мобільній платформі", *Телекомунікаційні та інформаційні технології*, №. 3 (88), С. 204–221. DOI: <https://doi.org/10.31673/2412-4338.2025.038721>
7. Yu, C., Rosendo, A. (2021), "Risk-Aware Model-Based Control", *Frontiers in Robotics and AI*, Vol. 8, 617839. DOI: <https://doi.org/10.3389/frobt.2021.617839>
8. Zeng, Y., Wu, L., Li, J., Zhuang, X., Wu, C. (2025), "Resilient Task Allocation for UAV Swarms: A Bilevel PSO-ILP Optimization Approach", *Drones*, Vol. 9, No. 9, 623. DOI: <https://doi.org/10.3390/drones9090623>
9. Та, P.B., Mafeni, V., Kim, Y. (2025), "A Design of Workflow-Based Automated Failure Recovery Framework in IoT Edge Environment", *IEEE Access*, Vol. 13, pp. 101885–101899. DOI: <https://doi.org/10.1109/ACCESS.2025.3578739>
10. Trung, K.N., Tran, M.-N., Kim, Y. (2026), "Enabling Service Continuity for Stateful Service Segmentation in Mobile Edge Computing Toward 6G", *IEEE Access*, Vol. 14, pp. 20589–20604. DOI: <https://doi.org/10.1109/ACCESS.2026.3661972>
11. Casini, D., Pazzaglia, P., Becker, M. (2025), "Managing real-time constraints through monitoring and analysis-driven edge orchestration", *Journal of Systems Architecture*, Vol. 164, 103403. DOI: <https://doi.org/10.1016/j.sysarc.2025.103403>
12. Agrawal, K., Jyothi, S.A. (2025), "Cooperative Graceful Degradation in Containerized Clouds", *Proceedings of ASPLOS 2025*, pp. 214–232. DOI: <https://doi.org/10.1145/3669940.3707244>
13. Ergenç, D., Memedi, A., Fischer, M., Dressler, F. (2025), "Resilience in Edge Computing: Challenges and Concepts", *Foundations and Trends in Networking*, Vol. 14, No. 4, pp. 254–340. DOI: <https://doi.org/10.1561/13000000074>
14. Golpayegani, F., Chen, N., Afraz, N., Gyamfi, E., Malekjafarian, A., Schäfer, D., Krupitzer, C. (2024), "Adaptation in Edge Computing: A Review on Design Principles and Research Challenges", *ACM Transactions on Autonomous and Adaptive Systems*, Vol. 19, No. 3, pp. 1–43. DOI: <https://doi.org/10.1145/3664200>
15. Chu, S., Koe, J., Garlan, D., Kang, E. (2024), "Integrating Graceful Degradation and Recovery through Requirement-driven Adaptation", *Proceedings of the 19th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS '24)*, pp. 122–132. DOI: <https://doi.org/10.1145/3643915.3644090>
16. Siasi, N., Jasim, M.A., Ghani, N. (2024), "Post-fault restoration of service function chains in fog networks", *Computer Networks*, Vol. 251, 110580. DOI: <https://doi.org/10.1016/j.comnet.2024.110580>
17. Guo, F., Peng, M. (2023), "Efficient Mobility Management in Mobile Edge Computing Networks: Joint Handover and Service Migration", *IEEE Internet of Things Journal*, Vol. 10, No. 20, pp. 18237–18247. DOI: <https://doi.org/10.1109/JIOT.2023.3279842>
18. Li, B., Nahrstedt, K. (1998), "An Open Task Control Model for Quality of Service Adaptation", *Proceedings of the 14th International Conference of Advanced Science and Technology (ICAST 98)*, pp. 29–41.

19. Zhang, X., Debroy, S. (2023), "Resource Management in Mobile Edge Computing: A Comprehensive Survey", *ACM Computing Surveys*, Vol. 55, No. 13s, Article 291, 37 pp. DOI: <https://doi.org/10.1145/3589639>
20. Zhang, J., Ning, Z., Waqas, M., Alasmary, H., Tu, S., Chen, S. (2024), "Hybrid edge-cloud collaborator resource scheduling approach based on deep reinforcement learning and multiobjective optimization", *IEEE Transactions on Computers*, Vol. 73, No. 1, pp. 192–205. DOI: <https://doi.org/10.1109/TC.2023.3326977>
21. Colarusso, C., Falco, I., Zimeo, E. (2025), "Business continuity of Cloud-based IoT applications through a seamless continuum", *Internet of Things*, Vol. 33, 101723. DOI: <https://doi.org/10.1016/j.iot.2025.101723>
22. Фролов, Д.Є. (2025), "Модель забезпечення відмовостійкості інформаційних систем на основі багаторівневих структур", *Вісник Херсонського національного технічного університету*, Т. 2, №. 3(94). С. 474–483. DOI: <https://doi.org/10.35546/kntu2078-4481.2025.3.2.61>
23. Zolotukhin O., Filatov, V., Kudryavtseva, M., Shaptala, S. (2025), "Using Artificial Intelligence Methods In Tasks Of Decentralized Control Of A Group Of Unmanned Aerial Vehicles", *Military Strategy and Technology*, №. 2(2), pp. 59–73. DOI: <https://doi.org/10.63978/3083-6476.2025.2.2.06>

Received (Надійшла) 08.02.2026

Accepted for publication (Прийнята до друку) 27.02.2026

Publication date (Дата публікації) 12.03.2026

Відомості про авторів / About the Authors

Ткачов Віталій Миколайович – кандидат технічних наук, доцент, Харківський національний університет радіоелектроніки, докторант кафедри електронних обчислювальних машин; Харків, Україна;

Vitalii Tkachov – Phd (Technical Sciences), Associate Professor, Kharkiv National University of Radio Electronics, Doctoral candidate at the Department of Electronic Computers; Kharkiv, Ukraine;

e-mail: vitalii.tkachov@nure.ua

ORCID ID: <https://orcid.org/0000-0002-6524-9937>

МЕТОД ВІДНОВЛЕННЯ КРИТИЧНИХ СЕРВІСІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ НА МОБІЛЬНІЙ ПЛАТФОРМІ В УМОВАХ КЕРОВАНОЇ ДЕГРАДАЦІЇ

Предметом роботи є методи відновлення критичних сервісів інформаційної системи на мобільній платформі за переривчастої зв'язності та керованої деградації. **Мета дослідження** – розробити сервісно-орієнтований метод відновлення, який забезпечує вимірюване зменшення дефіциту виконання порогових вимог до показників якості обслуговування в межах вікон зв'язності з огляду на ресурсні обмеження й можливості завершення узгодження стану сервісів у поточному вікні зв'язності. У дослідженні застосовано такі **методи**: системний аналіз, математичне моделювання станів і профілів сервісів, формалізація дефіциту виконання порогових вимог до показників якості обслуговування, предикатний опис віконної здійсненості відновлювальних кроків і обчислювальний експеримент для оцінювання показників результативності відновлення. У статті передбачалося розв'язати такі **завдання**: формалізувати крок відновлення як перехід між сервісними профілями, що зменшує дефіцит виконання порогових вимог до показників якості обслуговування; впровадити умову віконної здійсненості кроку відновлення як поєднання ресурсної здійсненості та можливості завершення узгодження стану сервісів у межах поточного вікна зв'язності; побудувати правило формування множини кроків відновлення у вікні, яке забезпечує зменшення дефіциту за наявності хоча б одного здійсненого кроку й узгоджується з допустимими діями керованої деградації як периферійного інструмента вивільнення ресурсів. **Досягнуті результати** полягають у впровадженні формалізму віконної здійсненості та системи показників для оцінювання дотримання умов здійсненості, частки вікон зі зменшенням дефіциту, накопичувального дефіциту критичних сервісів і часу відновлення, що дає змогу відокремлювати обмеження середовища від властивостей керувальних рішень. **Висновки**: ігнорування можливості завершення узгодження стану сервісів у межах поточного вікна зв'язності

призводить до появи незавершених відновлювальних переходів і розбіжності між очікуваним ефектом відновлення й фактичною динамікою дефіциту, тоді як запропонований метод формує кроки, здійснені в поточному вікні зв'язності, та забезпечує зменшення дефіциту критичних сервісів у вікнах, де відновлення є можливим. Це підвищує живучість унаслідок скорочення тривалості деградаційних епізодів критичних функцій та зменшення накопичувального дефіциту. Практична цінність полягає в можливості використання методу для бортових інформаційних систем груп БпЛА та інших мобільних вузлів із вікнами зв'язності зі змінною тривалістю та пропускну здатністю.

Ключові слова: інформаційна система; живучість; критичний сервіс; керована деградація; вікно зв'язності; відновлення сервісу.

Бібліографічні описи / Bibliographic descriptions

Ткачов В. М. Метод відновлення критичних сервісів інформаційної системи на мобільній платформі в умовах керованої деградації. *Автоматизовані системи управління та прилади автоматики*. 2026. № 1 (188). С. 78–97. DOI: <https://doi.org/10.30837/0135-1710.2026.188.078>

Tkachov, V. (2026), "Method of restoring critical information system services on a mobile platform under conditions of controlled degradation", *Management Information System and Devices*, No. 1 (188), P. 78–97. DOI: <https://doi.org/10.30837/0135-1710.2026.188.078>
