C2ST
C2 Science and Technology

Research Article
https://doi.org/10.26599/C2ST.2026.9700004

An active resilience planning method for C2 kill-webs based on hypergraphs and dynamic resilience potential

Lijian Sun[1,2,†], Yun Zhou[1,†], Chao Chen[1,*]

[1] College of Systems Engineering, National University of Defense Technology, Changsha 410073, China
[2] College of Electronic Engineering, Naval University of Engineering, Wuhan 430030, China

ABSTRACT

The kill-web enhances the survivability and mission completion capability of combat systems through networked redundancy and adaptive organization. However, existing resilience metrics for command and control (C2) structures still rely on passive reaction paradigms, failing to capture the multi-path redundancy and active regulation requirements of kill-webs. This paper proposes an active resilience planning method for C2 kill-webs based on hypergraph modeling and dynamic resilience potential. First, the C2 kill-web is modeled as a directed hypergraph, where hyperedges are defined as combinations of functional units that collaboratively accomplish the same kill task. Hypergraph node centrality (cut-vertex betweenness and uniqueness factor) is proposed to quantify structural redundancy. Second, a dynamic resilience potential (DRP) function is constructed, integrating load margin, alternative path diversity, and historical overload risk, enabling real-time perception of vulnerable regions. Third, an adaptive early warning mechanism using exponentially weighted moving average-cumulative sum (EWMA-CUSUM) statistical process control is implemented, along with a risk-penalized load pre-migration mechanism. Finally, information timeliness is introduced as a complementary resilience dimension, yielding a resilience-AoI joint metric (RAJM). Simulations show that active resilience planning (ARP) improves the RAJM integral by 45%–86% over static and passive methods, and provides approximately 20 s of early warning. Overall, this work offers a systematic framework for resilient C2 organizational structures aligned with the JADC2 vision.

KEYWORDS

command and control; kill-web; hypergraph; resilience planning; information timeliness

1 Introduction

Since the concept of the kill chain was formally introduced by the U.S. Air Force in 1996, it has evolved from the six-stage F2T2EA model to the three-phase "sense–C2–act" model[1]. The core idea is to decompose a complete strike process into sequential links and improve overall effectiveness by optimizing each link. However, with emerging operational concepts such as mosaic warfare, distributed lethality, and multi-domain operations[2], joint operations are rapidly evolving toward multi-domain, high-intensity, and dynamic confrontation. The traditional linear kill chain reveals three fundamental limitations: a) sensitivity to single-point failure—loss of any critical node can break the entire chain; b) slow response—re-planning a new chain in a dynamic battlefield requires substantial computation and coordination time; c) lack of self-healing capability—the system cannot autonomously reconfigure after being attacked. Against this backdrop, the kill-web has emerged as an evolutionary form[3]. It is a networked, adaptive ecosystem that integrates sensation, C2, strike, and support elements through highly connected information networks, providing inherent multi-path redundancy and self-repair capability. Nevertheless, the networked architecture imposes new requirements on the resilience of C2 organizational structures[4], raising three core questions: how to characterize the structural redundancy and multi-path nature of a kill-web; how to shift the resilience paradigm from passive absorption to active avoidance; and how to design effective early warning and migration strategies under complex dynamic environments.

Traditional modeling based on ordinary graphs and linear paths cannot accurately capture the combinational logic that "a single kill task is accomplished by a collaborative combination of multiple functional units." Hypergraphs, where a hyperedge can connect any number of vertices, naturally express such many-to-many relationships. Yet existing centrality measures for C2 structures mainly rely on meta-path or path enumeration[5-6], which face combinatorial explosion when scaled to kill-web scenarios. Simultaneously, most resilience metrics follow the passive paradigm of "failure → performance loss integral → recovery"[7-10], without exploiting the structural redundancy and real-time awareness to pre-emptively migrate loads before an attack. Moreover, in dynamic operations, system performance exhibits non-stationary fluctuations due to random task arrivals and processing saturation—a phenomenon often overlooked, leading to biased assessment because a fixed baseline cannot separate normal load drift from attack-induced damage. In the broader engineering domain,

Received: July 26, 2026; Revised: August 18, 2026; Accepted: September 6, 2026

† These authors contributed equally to this work and share first authorship.
*Chao Chen, chenc1997@nudt.edu.cn

Citation: Sun L J, Zhou Y, Chen C. An active resilience planning method for C2 kill-webs based on hypergraphs and dynamic resilience potential[J]. *C2 Science and Technology*, 2026. https://doi.org/10.26599/C2ST.2026.9700004.

© Chinese Institute of Command and Control. Publishing services by Tsinghua University Press. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/).

"active resilience" has gained attention through predictive maintenance and statistical process control[11-12], but its application to C2 kill-webs remains extremely rare.

Compared with the above works, our approach is uniquely positioned in four aspects: a) modeling—using hypergraphs rather than ordinary graphs to naturally characterize many-to-many collaborative relationships; b) paradigm—shifting from passive absorption to active avoidance, moving the resilience planning window forward; c) method—introducing statistical process control (EWMA-CUSUM) for adaptive early warning with a dynamic baseline, and designing a risk-penalized migration strategy; d) measurement—proposing a RAJM that complements mission capability with information freshness.

Addressing the above challenges, this paper proposes an active resilience planning method for C2 kill-webs based on hypergraph modeling and dynamic resilience potential. The main contributions are as follows:

1) A kill-web hypergraph model is proposed, with cut-vertex betweenness centrality and uniqueness factor defined to quantify structural redundancy, breaking through path-based characterization.

2) A DRP function is constructed, fusing load margin, topological substitutability, and historical overload risk into a real-time warning indicator.

3) An EWMA-CUSUM adaptive early warning mechanism detects significant DRP decline, providing approximately 20 s of advance warning.

4) A risk-penalized load pre-migration strategy avoids secondary overload by penalizing criticality and high load rates of receiving hyperedges.

5) Information timeliness is introduced as a complementary resilience dimension via RAJM, and the mean-reverting property of the load process is mathematically derived, revealing the necessity of a dynamic baseline.

2 Active resilience planning model for kill-webs

2.1 Kill-web hypergraph model

Following the framework established in Ref. [13], a C2 organizational structure is represented as a directed graph $G=(V,E,\Gamma)$, where $V=C\cup S\cup A\cup L$ is the set of organizational elements, C is the set of C2 units, S the set of sensation units, A the set of action units, and L the set of logistics units; E is the set of organizational relationships; and Γ is the set of operational constraints.

The core feature that distinguishes a kill-web from a traditional kill chain is that the same kill task can be collaboratively accomplished by multiple groups of functional units in different combinations. This "many-to-many" combination relationship cannot be accurately captured by an ordinary graph—ordinary edges only connect two nodes and cannot express the semantics of "a group of nodes together constitutes a collaborative scheme." A hypergraph allows an edge to connect any number of vertices, naturally fitting this scenario[14].

Definition 1 (Kill-web hypergraph). A kill-web hypergraph is defined as $H=(V,E,T)$, where V is the node set (i.e., organizational elements); $E_H\subseteq 2^V\setminus\{\emptyset\}$ is the set of hyperedges, where each hyperedge $e\in E_H$ is a non-empty subset of nodes, representing a group of functional units that can collaboratively accomplish a certain kill task; $E_H(v)=\{e\in E_H|\in e\}$ denote the set of hyperedges containing node v; $T=\{t_1,t_2,\cdots,t_{K_T}\}$ is the set of target units.

Each hyperedge e is associated with a target unit $t(e)\in T$. For each target $t_i\in T$, let $E_H^{t_i}=\{e\in E_H: t(e)=t_i\}$ denote the set of hyperedges that can engage target t_i. A hyperedge $e=\{v_1,v_2,\cdots,v_k\}$ is included in E_H if and only if: a) e contains at least one sensation unit, one C2 unit, and one action unit, and the organizational relationships among them can satisfy the functional requirements for closing a kill chain; b) the organizational relationships among the nodes in e satisfy the constraint set Γ.

To characterize dynamic resilience, each node $v\in V$ has a fixed capacity $C(v)>0$; the time-varying load $L(v,t)\geq 0$ is determined by the kinetic equations in Section 2.3.

Definition 2 (Time-varying kill coverage). For a hyperedge e, its kill coverage $\kappa(e,t)$ at time t is defined as the probability that this hyperedge inflicts effective kill on target $t(e)$. Considering the time-varying availability $a_v(t)\in[0,1]$ of each node and assuming independent operation of units, with functional units of the same type within a hyperedge in collaborative redundant parallel and different types in sequential series, we have

$$\kappa(e,t)=\Big(1-\prod_{v\in e\cap S}\big(1-a_v(t)\,q_v\big)\Big)\Big(1-\prod_{v\in e\cap C}\big(1-a_v(t)\,r_v\big)\Big)\Big(1-\prod_{v\in e\cap A}\big(1-a_v(t)\,p_v\big)\Big) \tag{1}$$

where q_v is the detection probability of a sensation unit, r_v is the correct decision probability of a C2 unit, and p_v is the hit probability of an action unit. This product form reflects the functional sequence logic of the kill chain: sensation, decision, and action must all succeed for the entire kill task to be completed. When all $a_v(t)=1$, this equation degenerates to static kill coverage.

Assumption 1 (Non-negativity of load and hard capacity constraint). The load $L_v(t)$ of any node v always satisfies $0\leq L_v(t)\leq \beta C_v$, where $\beta>0$ is a physical upper-bound coefficient. In numerical simulation, hard truncation ensures numerical stability.

Assumption 2 (Spatio-temporal independence of task arrivals). The task arrival process $\lambda_e(t)$ for hyperedge e follows an independent Poisson process with intensity ν_e, and task generation across different hyperedges is mutually independent. This assumption simplifies the expectation calculation of load superposition.

Proposition 1 (Mean-reverting property of the load process). In the absence of attack ($u_A\equiv 0$) and under constant node availability a_v, the expected load $\mathbb{E}[L_v(t)]$ converges exponentially with rate $a_v\mu_v^0$ to the steady-state value $L_v^*=\sum_{e:v\in e}\nu_e/a_v\mu_v^0$.

Proof. The load update equation $\dot{L}_v=\sum_{e:v\in e}\lambda_e(t)-a_v\mu_v^0 L_v$ contains $\lambda_e(t)$ as independent Poisson processes with intensity ν_e. Taking expectations on both sides and using $\mathbb{E}[\lambda_e(t)]=\nu_e$ and the independence between Poisson arrivals and load state, we obtain the deterministic ordinary differential equation $\frac{\mathrm{d}\mathbb{E}[L_v]}{\mathrm{d}t}=-a_v\mu_v^0\mathbb{E}[L_v]+\sum_{e:v\in e}\nu_e$. This is a first-order linear equation whose analytical solution is $\mathbb{E}[L_v(t)]=\mathbb{E}[L_v(0)]\mathrm{e}^{-a_v\mu_v^0 t}+L_v^*\big(1-\mathrm{e}^{-a_v\mu_v^0 t}\big)$.

As $t\to\infty$, $\mathbb{E}[L_v(t)]\to L_v^*$, and the convergence rate is determined by $a_v\mu_v^0$. The proposition is proved. □

This proposition reveals the mathematical root of mission capability degradation before an attack: when the initial load is below the steady-state value, the expected load must monotonically increase, causing the load penalty term $\mathrm{e}^{-\lambda_L L/C}$ to decrease, and consequently the expected mission capability $P(t)$ naturally declines. This inference will be verified in the experimental analysis in Section 4.

2.2 Hypergraph node centrality

Traditional node criticality $K(v,t) = \| \{e : v \in e\} \| / \| E_H \|$ (i.e., the proportion of hyperedges incident to the node) lacks discriminability when hyperedges heavily overlap—many nodes may have similar degrees, failing to distinguish which nodes truly act as "cut-vertices" in hypergraph connectivity. This paper proposes two complementary indicators.

Definition 3 (Hypergraph cut-vertex betweenness centrality). The hypergraph cut-vertex betweenness centrality of node v is defined as:

$$B(v) = \sum_{\substack{e_i,e_j \in E_H,\ e_i \neq e_j,\\ |e_i \cap e_j|>0}} \frac{\sigma_{e_i,e_j}(v)}{\sigma_{e_i,e_j}} \tag{2}$$

where $\sigma_{e_i,e_j} = |e_i \cap e_j|$ is the number of shared nodes between hyperedges e_i and e_j, and $\sigma_{e_i,e_j}(v) = 1$ if $v \in e_i \cap e_j$ and 0 otherwise. The summation traverses only hyperedge pairs that share nodes, avoiding division by zero. This indicator measures the pivotal role of a node in connecting different hyperedges.

Definition 4 (Hyperedge uniqueness factor).

$$U(v) = \ln\left(1 + \frac{1}{|E_H(v)|}\sum_{e \in E_H(v)} \frac{1}{|e|}\right) \tag{3}$$

This factor measures the average "exclusivity" of hyperedges that a node participates in—if a node frequently appears in large-scale hyperedges, its uniqueness is low (its role is easily substitutable); if it appears only in a few small-scale hyperedges, its uniqueness is high (its role is difficult to replace). If a node does not participate in any hyperedge, $U(v) = 0$.

The improved node criticality is synthesized as:

$$K^*(v) = \alpha_B r(B(v)) + \alpha_U r(U(v)) \tag{4}$$

where $\alpha_B + \alpha_U = 1$, and $\mathrm{rank}(\cdot)$ is a rank normalization function that maps the elements of a vector to $[0,1]$ by sorting in ascending order (minimum 0, maximum 1).

2.3 System state evolution equations

The system state is described by node availability $a_v(t) \in [0,1]$ and load $L_v(t) \geq 0$.

Attack model. The attacker inflicts damage on nodes with intensity $u_A(v,t) \in [0,1]$, where $u_A = 1$ means complete node failure and $u_A = 0$ means no attack. This paper considers three attack modes:

Random attack: Randomly select 12 nodes, with attack intensity uniformly random in $[0.5, 0.95]$;

Deliberate attack: Select the top 12 nodes sorted by K^*, attack intensity as above;

Adaptive attack: Select the 12 nodes with the highest current load rate L_v/C_v, with attack intensity $u_A = \min(0.95, L_v/C_v)$ truncated to $[0,1]$, so that the attack effect is positively correlated with the node's current load stress and always within the domain.

Defense repair. The defender adopts a dynamic repair strategy, with repair effort intensity inversely related to the current DRP:

$$u_{\mathrm{rep}}(v,t) = (1 - a_v(t))\big(1 + \lambda_R(1 - D(v,t))\big) \tag{5}$$

where $\lambda_R > 0$ is the DRP sensitivity coefficient, and the result is truncated to $[0,1]$. This design makes vulnerable nodes (low DRP) receive stronger repair effort. Note that $u_{\mathrm{rep}}(v,t)$ represents the repair effort intensity, and its ultimate contribution is uniformly controlled by the baseline repair coefficient η_D in the state equation.

State update equations. Node availability change:

$$\dot{a}_v = -\eta_A u_A(v,t) a_v + \eta_D u_{\mathrm{rep}}(v,t)(1 - a_v) \tag{6}$$

where $\eta_A > 0$ is the attack damage coefficient and $\eta_D > 0$ is the baseline repair coefficient. Node load change

$$\dot{L}_v = \sum_{e \in E_H(v)} \lambda_e(t) - a_v \mu_v^0 L_v \tag{7}$$

where $\mu_v^0 > 0$ is the task processing rate. Load pre-migration is implemented by directly adding/subtracting L_v at each discrete time step.

2.4 Resilience-AoI joint metric

Definition 5 (Kill-web information timeliness). For a hyperedge e, its information timeliness is taken as the maximum age among all sensors within the hyperedge since their latest information generation time, accounting for the impact of sensor availability on information update frequency:

$$A(e,t) = \max_{s \in e \cap S}\left(\frac{t - \tau_{\mathrm{gen}}(s)}{a_s(t) + 0.1}\right) \tag{8}$$

where $\tau_{\mathrm{gen}}(s)$ is the generation time of the latest information from sensor s. Adding 0.1 in the denominator prevents division by zero and reflects the physical fact that effective information age increases sharply when availability drops. The overall average AoI is defined as:

$$A_{\mathrm{avg}}(t) = \frac{1}{|E_H|}\sum_{e \in E_H} A(e,t) \tag{9}$$

Definition 6 (RAJM). The resilience-AoI joint metric (RAJM) of a kill-web at time t is defined as:

$$R(t) = P(t)\mathrm{e}^{-\xi \cdot A_{\mathrm{avg}}(t)/A_{\max}} \tag{10}$$

where $\xi > 0$ is the timeliness sensitivity coefficient and $A_{\max}$ is the maximum allowable information delay for the operation. The mission capability $P(t)$ is defined as:

$$P(t) = \sum_{t_i \in T} w_i \max_{e \in E_H^{t_i}}\left[\kappa(e,t)\,\mathrm{e}^{-\lambda_L \cdot \max_{v \in e}\frac{L_v(t)}{C_v}}\right] \tag{11}$$

where $w_i = 1/|T|$, $\lambda_L > 0$ is the load penalty coefficient, and $\kappa(e,t)$ is calculated by Eq. (1). RAJM measures both the system's mission completion capability (via $P(t)$) and the timeliness of information (via the exponential decay term), compensating for the deficiency of traditional resilience metrics that ignore information freshness[15-16].

3 Active resilience planning method

3.1 Dynamic resilience potential function

Definition 7 (Node dynamic resilience potential). The dynamic resilience potential $D(v,t)$ of node v at time t is defined as the weighted sum of three factors:

$$D(v,t) = \alpha\left(1 - \frac{L(v,t)}{C(v)}\right) + \beta\big(1 - K^*(v)\big) + \gamma\big(1 - H(v,t)\big) \tag{12}$$

The load margin term $(1 - L/C)$ reflects the current overload risk—the lower the value (load closer to capacity), the more vulnerable the node.

The node criticality term $(1 - K^*)$ reflects structural redundancy—the higher the criticality, the greater the impact if the node

is attacked.

The historical overload risk $H(v,t)=\frac{1}{T}\int_{t-T}^{t}\frac{L(v,\tau)}{\sum_{u\in V}L(u,\tau)}\mathrm{d}\tau$, where T is a hyperparameter ($T=10$ s in this paper). This term captures the proportion of total network load that the node has borne over a past period, used to identify "fatigue" nodes that frequently undertake a high proportion of tasks.

The weights satisfy $\alpha+\beta+\gamma=1$.

A higher DRP value indicates a healthier node (high load margin, low criticality, low historical overload); a lower DRP value indicates a more vulnerable node, making it a target for warning and potential migration.

3.2 Adaptive early warning mechanism based on EWMA-CUSUM

Using a fixed threshold θ_{DRP} to determine vulnerability generates many false alarms when load traffic is non-stationary—for example, during peak task periods, many nodes' DRP values drop without being attacked. This paper introduces the cumulative sum (CUSUM) method from statistical process control to adaptively track normal system drift through a dynamic baseline. Since vulnerability is characterized by a significant drop in DRP, a downward CUSUM is employed for detection.

Dynamic baseline estimation. For each node v, define the DRP residual:

$$Z_v(t)=D(v,t)-\hat{\mu}_v(t) \tag{13}$$

where $\hat{\mu}_v(t)$ is the exponentially weighted moving average (EWMA):

$$\hat{\mu}_v(t)=\alpha_{\mathrm{EW}}D(v,t)+(1-\alpha_{\mathrm{EW}})\hat{\mu}_v(t-1) \tag{14}$$

with $\alpha_{\mathrm{EW}}\in(0,1)$ as the smoothing coefficient. The EWMA baseline allows the system to adapt to slow time variations in task arrival rates.

Downward cumulative sum statistic. Define the one-sided downward CUSUM:

$$S_v^-(t)=\max\{0,S_v^-(t-1)-Z_v(t)-\delta\} \tag{15}$$

where $\delta>0$ is the tolerance drift, used to filter out small random fluctuations. When the node's DRP is below baseline (i.e., $Z_v(t)<0$), $-Z_v(t)>0$, and the cumulative sum increases; when DRP recovers above baseline, the cumulative sum decreases.

Warning trigger rule. A warning is triggered when $S_v^-(t)>h$ and the system average load rate exceeds 0.5. The dual condition (cumulative sum exceeds limit + global load is high) effectively avoids meaningless warnings when the system is idle.

Definition 8 (Vulnerable region). If a node v satisfies $S_v^-(t)>h$, it is said to be in a vulnerable state. If in a hyperedge e the proportion of nodes in a vulnerable state exceeds $\eta=0.4$ and the hyperedge average load rate exceeds 0.55, then hyperedge e is called a vulnerable hyperedge. Vulnerable hyperedges and their neighborhoods constitute a vulnerable region.

3.3 Heuristic strategy for load pre-migration

When a warning is triggered and an attack has not yet started, the system attempts to perform load pre-migration. This paper adopts a heuristic strategy based on risk penalty, with the core design principle of "avoiding transferring load from a currently vulnerable node to another potentially vulnerable node."

Migration process. For each vulnerable hyperedge e, execute the following steps:

1) Calculate its total load $L_e=\sum_{v\in e}L_v(t)$;

2) Among all non-vulnerable hyperedges, select the hyperedge e' that maximizes the following scoring function as the receiving target:

$$Q(e')=\frac{C(e')-L(e')}{C(e')}-\lambda_K\bar{K}^*(e')-\lambda_\rho\bar{\rho}(e') \tag{16}$$

where $C(e')=\sum_{v\in e'}C(v)$, $L(e')=\sum_{v\in e'}L(v,t)$, $\bar{K}^*(e')$ is the average criticality of nodes within the hyperedge, $\bar{\rho}(e')$ is the average load rate of nodes within the hyperedge, and $\lambda_K>0$ and $\lambda_\rho>0$ are penalty coefficients;

3) Calculate the migratable amount $\Delta L=\min(0.18L_e, 0.75(C(e')-L(e')))$, where the coefficients were determined through preliminary experiments to balance migration intensity and safety margin;

4) Distribute ΔL equally among the nodes of the vulnerable hyperedge (reduce load) and equally among the nodes of the receiving hyperedge (increase load).

The three terms of the scoring function have clear meanings: the first term is the proportion of remaining capacity of the receiving hyperedge (the larger the better); the second term penalizes high criticality of the receiving hyperedge (avoiding transferring load to structurally critical nodes); the third term penalizes high load rate of the receiving hyperedge (avoiding transferring load to already overloaded regions). λ_K and λ_ρ control the penalty intensity.

3.4 Computational complexity analysis

The per-step computational cost of the algorithm mainly comes from three parts: warning statistic update ($O(|V|)$), vulnerable hyperedge identification ($O(|E_H|\cdot\bar{d})$, where $\bar{d}$ is the average hyperedge size), and migration scoring ($O(|E_{\mathrm{frag}}|\cdot|E_H|\cdot\bar{d})$). The total time complexity is

$$O\big(|V|+|E_{\mathrm{frag}}|\cdot|E_H|\cdot\bar{d}\big) \tag{17}$$

In the scenario of this paper ($|V|=60$, $|E_H|=120$, $\bar{d}=5.4$), a single step is completed in microseconds. When scaling to theater-level scales (where $|E_H|$ is approximately 10^4), approximate nearest neighbor search based on hypergraph embedding would need to be introduced to replace full scoring, which belongs to future work[17-18].

4 Experimental validation

4.1 Experimental setup and parameter configuration

Simulation scenario. A large-scale kill-web with 60 nodes is constructed, including 15 sensation units, 20 C2 units, 15 action units, and 10 logistics units, covering 8 target units. A total of 120 hyperedges are formed, each containing 3–8 nodes, with an average hyperedge size of approximately 5.4.

Parameter configuration. The parameters of the model are set as follows:

System parameters: The probability parameters q_v, r_v, p_v are uniformly randomly generated in $[0.75, 0.95]$; node capacity $C(v)$ is randomly selected from $[10, 20]$; initial load is set to 50% of capacity; task arrival rate $\lambda_e(t)$ is a Poisson process with average arrival rate 0.3; node task completion rate $\mu_v^0=0.3$.

DRP weights: Through offline grid search (α,β,γ with step 0.1), $\alpha=0.6, \beta=0.2, \gamma=0.2$ are chosen to highlight the sensitivity of load margin (the fastest-changing dimension).

EWMA-CUSUM parameters: $\alpha_{\mathrm{EW}}=0.15, \delta=0.02, h=0.12$.

Selection basis: through offline pre-experiments, the average run length (ARL) of the residual cumulative sum under normal conditions is kept above 100 time steps, while ensuring timely warning when attack signs appear.

Repair and migration parameters: $\eta_A = 0.15, \eta_D = 0.12, \lambda_R = 6$; $\lambda_K = 1.0, \lambda_\rho = 0.5$.

Timeliness parameters: $\xi = 1.0, A_{\max} = 30$ s.

Load penalty coefficient: $\lambda_L = 0.1$.

Simulation duration: Total simulation time 100 s, attack start time uniformly random within 16–28 s, lasting 50 s.

Compared methods.

1) Static Robustness method (SR): Reinforce the top 10 nodes (approximately 16.7%) based on K^* centrality, capacity increased by 20%, no active warning or migration.

2) Passive cascading failure model (PCF): Fixed repair rate 0.02, no warning or migration.

3) Proposed method (ARP): Fully implements the DRP warning + EWMA-CUSUM + heuristic pre-migration + dynamic repair defined in Section 3.

Evaluation metrics.

1) Mission capability retention rate P_{retain}: the minimum value of $P(t)$ during simulation.

2) Benchmark-crossing time T_{cross}: the time from attack start until $P(t)$ first exceeds a fixed reference line $P_{\text{ref}} = 0.88 \times \bar{P}_{[0,5]}$, where $\bar{P}_{[0,5]}$ is the average mission capability over the initial 5 s. This reference line serves solely as a uniform comparison anchor and does not imply full dynamic recovery. If $P(t)$ never exceeds P_{ref}, T_{cross} is recorded as $100 - t_{\text{attack}}$.

3) RAJM integral: $\frac{1}{T}\int_0^T \text{RAJM}(t)\text{d}t$.

4) Warning advance time: the difference between the warning trigger time and the attack start time.

5) Benchmark-crossing rate: the fraction of runs in which $P(t)$ successfully exceeds P_{ref} after the attack. Dynamic recovery is separately monitored through the DRP and RAJM trajectories.

Statistical setup. Each experiment is repeated 5 times ($n = 5$), reporting mean ± standard deviation. The random seed is set to $2025 + 100r + \text{hash}(m)\text{mod}100$ (r is the repetition number, m is the attack mode).

4.2 Overall resilience performance comparison

Table 1 summarizes the key resilience metrics of the three methods under three attack modes. The best value for each metric is highlighted in bold.

1) Overall advantage in comprehensive mission effectiveness (RAJM).

The RAJM integrals of ARP under random, deliberate, and adaptive attacks are 0.516, 0.533, and 0.512, respectively, representing improvements of 50.0%, 45.6%, and 85.5% over SR, and 63.8%, 59.1%, and 84.8% over PCF. The RAJM integral reflects the cumulative effectiveness over the entire time window—encompassing both mission completion capability $P(t)$ and information timeliness penalty. ARP's sustained lead indicates that active planning not only improves peak performance but, more importantly, maintains higher mission capability and information freshness over long durations. The RAJM integral of ARP under deliberate attack (0.533) is comparable to that under random attack (0.516), and the benchmark-crossing rate reaches 100%, indicating that ARP's pre-migration strategy has a very strong defensive effect against precision strikes targeting critical nodes.

2) Comprehensive improvement in restoration speed and benchmark-crossing rate.

Under random attack, ARP's average benchmark-crossing time is only 34.379 s, 53.4% shorter than SR (73.779 s) and 56.9% shorter than PCF (79.779 s), with a 100% benchmark-crossing rate. Under deliberate attack, although SR achieves a 100% benchmark-crossing rate, its crossing time is as long as 69.397 s; PCF's crossing rate is only 20%, while ARP also achieves 100% crossing rate with an average time of only 30.197 s, significantly faster than both. Under adaptive attack, ARP's crossing rate is 100% with an average time of only 25.359 s, while SR and PCF have crossing rates of 20% and 40%, and crossing times of 77.959 s and 64.759 s, respectively. Note that these crossing times are measured against the static reference P_{ref} for uniform comparison; dynamic recovery is better reflected in the RAJM and DRP evolutions.

3) Challenge of adaptive attacks and robustness of ARP.

Adaptive attacks select the nodes with the highest current load as targets, posing a fundamental challenge to ARP's migration strategy—migration itself could make the receiving node a new high-load target, thus "drawing fire". Nevertheless, ARP's P_{retain} (0.736) remains much higher than SR (0.672) and PCF (0.692); the RAJM integral leads by approximately 85%; the benchmark-crossing time is far superior to the passive methods (25.359 s vs. 77.959 s and 64.759 s); and the crossing rate reaches 100%—remarkably higher than SR's 20% and PCF's 40%. This fully validates the strong robustness of the proposed method.

4.3 Performance evolution curves

Figures 1–3 respectively show the time evolution curves of mission capability $P(t)$, $R(t)$, and average AoI for each method under

Table 1 Resilience measures of the three methods under different attack modes

Attack mode	Method	P_{retain}	T_{cross}/s	RAJM integral	Bench.-cross. rate/%
Random	SR	0.697±0.027	73.779±7.364	0.344±0.020	80
	PCF	0.690±0.028	79.779±3.910	0.315±0.020	20
	ARP (ours)	0.750±0.020	34.379±14.062	0.516±0.023	100
Deliberate	SR	0.677±0.008	69.397±2.500	0.366±0.031	100
	PCF	0.659±0.007	77.197±1.294	0.335±0.027	20
	ARP (ours)	0.751±0.008	30.197±10.688	0.533±0.037	100
Adaptive	SR	0.672±0.015	77.959±3.942	0.276±0.026	20
	PCF	0.692±0.031	64.759±23.003	0.277±0.024	40
	ARP (ours)	0.736±0.010	25.359±10.140	0.512±0.021	100

Note: T_{cross} measures the time until $P(t)$ first exceeds the reference line $P_{\text{ref}} = 0.88\bar{P}_{[0,5]}$ after attack onset. The benchmark-crossing rate reports the fraction of runs achieving this. Dynamic health is evaluated via DRP and RAJM.

the three attack modes. The gray shaded area indicates the attack duration interval (from attack start to 50 s after attack start).

From Figs. 1–3, three important phenomena can be observed:

1) Natural degradation of performance before attack.

In the initial stage before the attack starts, the mission capability $P(t)$ of all methods shows a certain degree of decline. This phenomenon is not caused by attacks but is determined by the mean-reverting property of the load process revealed by Proposition 1 in Section 2.1: the system evolves from the initial load state ($0.5C_v$) toward the steady-state value (generally above $0.6C_v$),

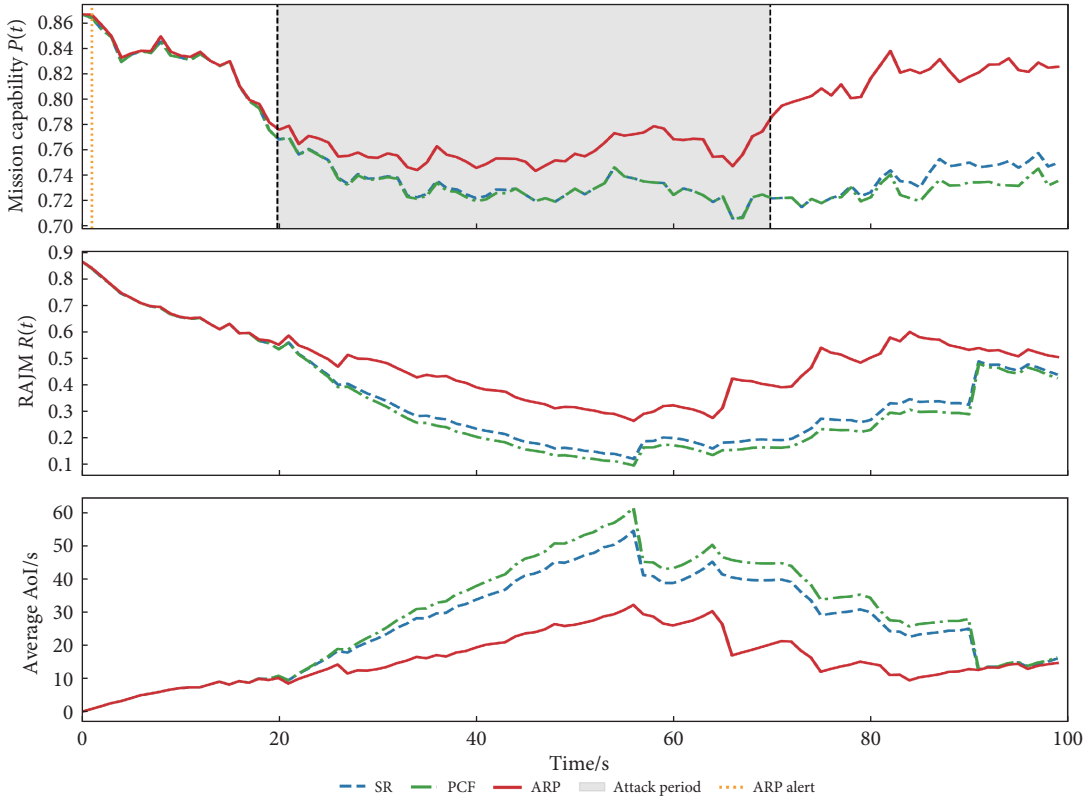

Figure 1 Performance evolution curves under random attack.

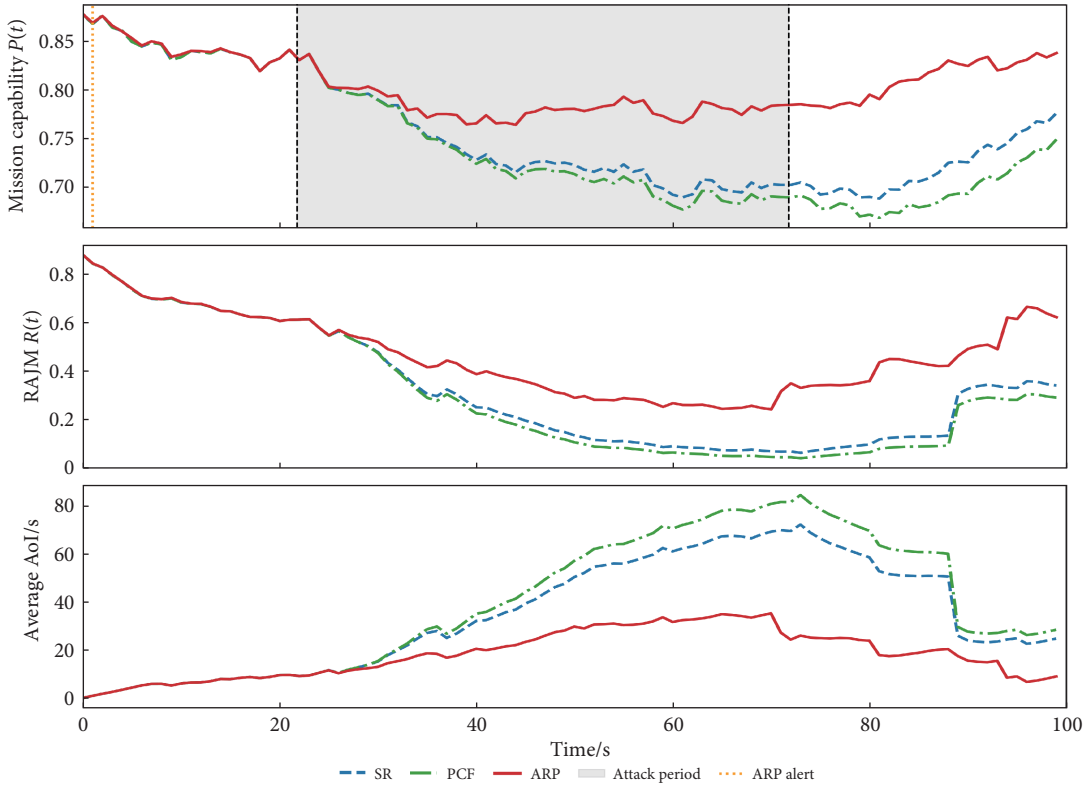

Figure 2 Performance evolution curves under deliberate attack.

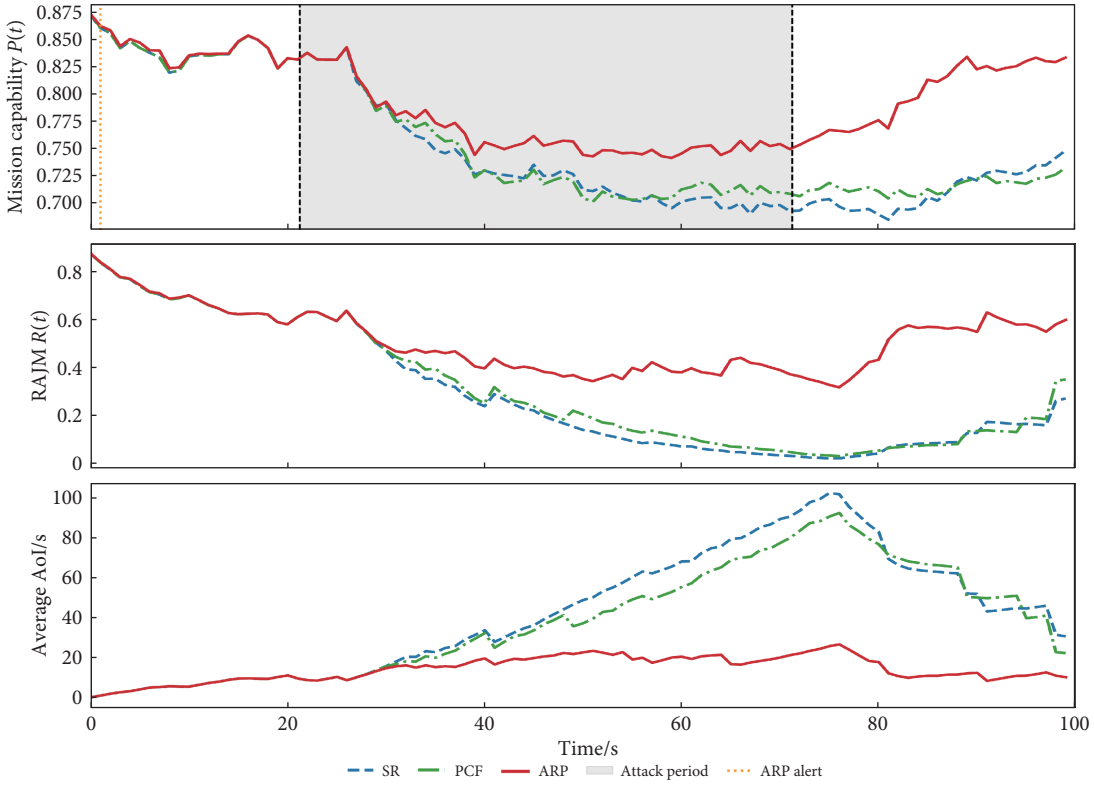

Figure 3 Performance evolution curves under adaptive attack.

causing the load penalty term to increase, and thus the natural expected value of $P(t)$ decreases as the load stabilizes. This phenomenon is consistently observed in simulations of all three attack modes, validating the core assertion of this paper: resilience assessment must establish a dynamic baseline rather than fixing the performance value at the initial moment as the reference. The DRP dynamic baseline proposed in this paper tracks this drift in real time via EWMA, thereby avoiding false alarms from fixed thresholds under non-attack conditions.

2) Post-attack recovery dynamics.

Under deliberate attack (Fig. 2), ARP's $P(t)$ decreases during the attack but remains above 0.75, quickly rebounding after the attack ends and crossing the reference line P_{ref} multiple times after $t \approx 55$ s; in contrast, the $P(t)$ of SR and PCF continues to slide to around 0.65 after the attack, and their restoration speed is significantly slower than ARP. Under adaptive attack (Fig. 3), ARP's $P(t)$ drops but with the smallest amplitude (minimum about 0.74), whereas the $P(t)$ of SR and PCF deteriorates sharply to below 0.7 in the mid-to-late attack phase. ARP's RAJM curve under adaptive attack is clearly higher than the compared methods after $t \approx 40$ s, reflecting the protective effect of the migration strategy on information timeliness.

3) Evolution of information timeliness (AoI).

The average AoI growth rate of ARP is consistently slower than that of the compared methods. Especially during the attack period, the AoI of the passive methods increases sharply, while ARP, through load migration, protects the patency of sensor–C2–action chains, enabling information updates to be maintained. Under deliberate attack, ARP's maximum AoI is only about one-quarter of that of the passive methods.

4.4 Performance of the active early warning mechanism

Table 2 reports the statistical performance of the ARP warning mechanism under the three attack modes.

Table 2 Performance of the ARP active early warning mechanism

Attack mode	Warning advance time/s
Random	18.82±3.28
Deliberate	20.80±1.39
Adaptive	20.24±4.30

The warning advance time ranges from 18.82 to 20.80 s, indicating that the warning moments largely fall within the ideal window (10–25 s)—neither too early, causing resource waste or false alarms, nor too late, losing decision time. In the ARP_deliberate scenario, the warning occurs about 20.80 s before the attack, providing the system with ample migration window—multiple consecutive migration operations have been completed before the attack begins, significantly reducing the load density in vulnerable regions. The warning advance times under random and adaptive attacks are also stable around 18.82 s and 20.24 s, respectively. The warning advance time under adaptive attack remains well within the ideal window, indicating that the EWMA-CUSUM framework can maintain reliable detection performance under highly non-stationary attack modes.

Figure 4 shows the detailed event timeline of ARP under adaptive attack as an example. The alert occurs about 20 s before the attack, followed by multiple load migrations (Migration 1–8), so that the system has significantly reduced the load density of vulnerable regions before the attack arrives. After the attack starts, $P(t)$ decreases but with a controllable magnitude ($P_{\text{retain}} = 0.736$), and quickly crosses P_{ref} after the attack ends (marked as Recover).

4.5 Design principles and parameter sensitivity discussion

Based on the above experimental results, this paper extracts three core design principles for active resilience planning:

Principle 1: Statistical baseline.

The EWMA-CUSUM framework performs outlier detection

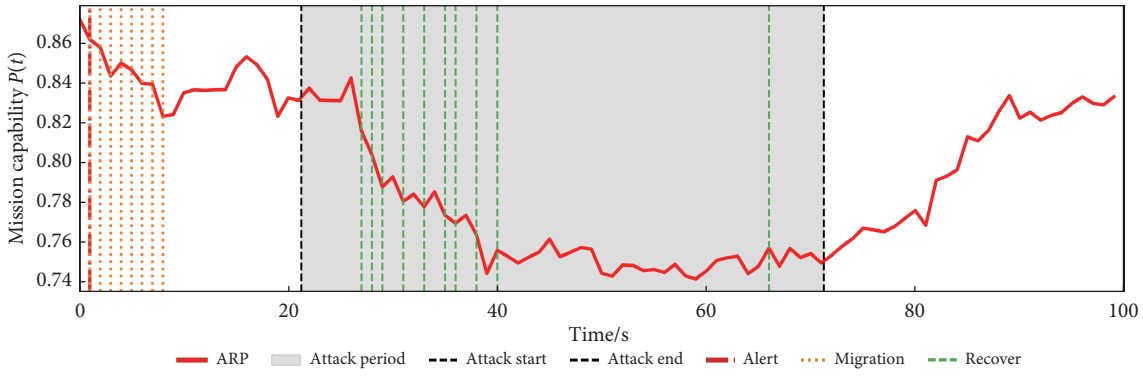

Figure 4 Event timeline of ARP under adaptive attack.

based on a dynamic baseline, avoiding false alarms from fixed thresholds under load fluctuations. The warning advance time is stable at 19–21 s and consistently falls within the ideal window (10–25 s), validating the effectiveness of this principle.

Principle 2: Risk endogenous principle.

The migration scoring function Eq. (16) simultaneously penalizes the criticality (λ_K term) and current load rate (λ_ρ term) of the receiving hyperedge. The effectiveness of this design is most clearly demonstrated under adaptive attacks: ARP's RAJM integral leads SR by 85.5% and PCF by 84.8%. The values $\lambda_K = 1.0$ and $\lambda_\rho = 0.5$ were determined through preliminary experiments: too large a λ_K results in no suitable receiving hyperedge (migration cannot be executed), while too small transfers load to critical nodes; too large a λ_ρ rejects all hyperedges with relatively high load rates, while too small fails to prevent secondary overload.

4.6 Discussion

4.6.1 Computational complexity

As shown in Eq. (17), the per-step time complexity of the algorithm is $O(|E_{\text{frag}}| \cdot |E_H| \cdot \bar{d})$. In this scenario ($|E_H| = 120$), the per-step computation time is about 0.3 ms, and the total simulation time is about 0.4 s, meeting the time constraints for campaign-level decision-making. When scaling to theater-level scales with thousands of hyperedges, approximate nearest neighbor search based on hypergraph embedding should be introduced to replace full scoring.

4.6.2 Parameter sensitivity

The EWMA-CUSUM parameters ($\alpha_{\text{EW}}, \delta, h$) perform robustly in this scenario, but their optimal values depend on the statistical characteristics of the load process (e.g., the coefficient of variation of the task arrival rate). In practical deployment, offline historical data should be used for parameter calibration, or Bayesian optimization could be introduced for adaptive tuning.

4.6.3 Limitations

1) The migration strategy is heuristic and does not guarantee global optimality; future work could introduce convex programming or reinforcement learning for better solutions.

2) The attack model assumes continuous variation of attack intensity, failing to characterize discrete, strategically intended decision attacks; future work could consider a game-theoretic framework.

3) Validation is based on simulation data and needs further testing with wargaming systems or real exercise data.

4) The kill coverage Eq. (1) is based on the assumption of independent unit operation; when common-cause failures or systemic faults exist, correlation models need to be introduced.

5 Conclusions

This paper proposed an active resilience planning method for C2 kill-webs that shifts the resilience paradigm from passive absorption to active avoidance. By modeling the kill-web as a directed hypergraph and introducing cut-vertex betweenness centrality with a uniqueness factor, the method captures the multi-path redundancy inherent in collaborative kill schemes and accurately identifies structurally critical nodes. A Dynamic Resilience Potential function, integrating load margin, topological substitutability, and historical overload risk, provides real-time perception of vulnerable regions. On this basis, an EWMA–CUSUM adaptive early warning mechanism detects significant DRP decline trends and yields approximately 20 s of advance warning, creating a sufficient decision window for pre-attack load migration. The risk-penalized migration strategy further prevents secondary overloads by penalizing criticality and high load rates of receiving hyperedges. Simulations demonstrate that the proposed approach improves the RAJM integral by 45%–86% over static and passive baselines under random, deliberate, and adaptive attacks, with a 100% benchmark-crossing rate and substantially faster restoration. Importantly, the work reveals that kill-web mission capability experiences natural decay even without attacks due to the mean-reverting property of the load process, which underscores the necessity of a dynamic rather than static resilience baseline. Overall, this framework provides a systematic theoretical and methodological foundation for building resilient C2 organizational structures aligned with the JADC2 vision.

Funding

This work was supported by the National Natural Science Foundation of China (62276272), and the National University of Defense Technology Foundation (26-ZZCX-JDZ-41).

Data availability

The datasets generated and/or analysed during the current study are available from the corresponding author on reasonable request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author contribution statement

Lijian Sun: Conceptualization, Methodology, Software, Writing – original draft. Yun Zhou: Conceptualization, Supervision, Validation, Writing - review & editing, Funding acquisition. Chao Chen: Writing – review & editing, Data curation.

Use of AI statement

The authors used deepseek solely for language polishing. The authors reviewed and edited the content and take full responsibility for the final content.

References

[1] Brose C. The kill chain: defending America in the future of high-tech warfare[M]. New York: Hachette Books, 2020.

[2] Kline J E. A tactical doctrine for distributed lethality[EB/OL]. (2016-02-22)[2026-05-22]. https://cimsec.org/tactical)-doctrine-distributed-lethality/.

[3] Jiang J Y, Li J C, Xia B Y, et al. Modeling and resilience analysis of combat systems-of-systems based on kill web[C]//17th Annual System of Systems Engineering Conference. Rochester: IEEE, 2022: 28-35.

[4] Hosseini S, Barker K, Ramirez-Marquez J E. A review of definitions and measures of system resilience[J]. *Reliability Engineering & System Safety*, 2016, 145: 47-61.

[5] Sun L J, Zhou Y, Zhu C, et al. A new important nodes identification method in multi-layer heterogeneous combat network with meta-path centrality[J]. *Journal of Complex Networks*, 2024, 12(3): cnae021.

[6] Sun L J, Zhou Y, Zhang X K, et al. Analysis and evaluation of C2 organizational structure supported by kill chain[J]. *Journal of Command and Control*, 2022, 8(4): 460-470 (in Chinese).

[7] Lei H T, Sun L J, Zhu X Q, et al. A method of analysis, construction and resilience optimization of kill chain[J]. *Journal of Command and Control*, 2024, 10(6): 720-731 (in Chinese).

[8] Wu L, Zhu J H, Zhu C, et al. A modeling approach for kill webs based on functional-information-coordination multidimensional relationships[J]. *Journal of Command and Control*, 2025, 11(4): 405-413 (in Chinese).

[9] Wang K S, Tan J X, Sun P Y, et al. Adversarial resilience enhancement in combat kill-web under sabotage[J]. *Reliability Engineering & System Safety*, 2026, 275(Pt 2): 112805.

[10] Wu X Y, Kan D, Bai Y, et al. Toughness assessment method of combat systems based on time sequence network[J]. *Journal of Air & Space Early Warning Research*, 2024, 38(6): 432-437 (in Chinese).

[11] Vachtsevanos G, Lee B, Oh S, et al. Resilient design and operation of cyber physical systems with emphasis on unmanned autonomous systems[J]. *Journal of Intelligent & Robotic Systems*, 2018, 91(1): 59-83.

[12] Montgomery D C. Introduction to statistical quality control[M]. Hoboken: John Wiley & Sons, Inc. , 2020.

[13] Sun L J, Zhou Y, You H L, et al. An efficient planning method to recommend COA based on new command and control organizational structure model[J]. *IEEE Transactions on Computational Social Systems*, 2024, 11(6): 7359-7372.

[14] Yin M M, Tang K, Xu T, et al. Hypergraph-based high-speed rail hypernetwork analysis and node importance evaluation using operational data: a case study of China[J]. *Journal of Transportation Engineering, Part A: Systems*, 2026, 152(1): 04025113.

[15] Yates R D, Sun Y, Brown D R, et al. Age of information: an introduction and survey[J]. *IEEE Journal on Selected Areas in Communications*, 2021, 39(5): 1183-1210.

[16] Yang H H, Xu C, Wang X J, et al. Understanding age of information in large-scale wireless networks[J]. *IEEE Transactions on Wireless Communications*, 2021, 20(5): 3196-3210.

[17] Smartt C, Sisavath J, Wyatt A J. Graph-based digital engineering ontology for mission design and execution and resulting research opportunities[C]//Proceedings of the 2025 Conference on Systems Engineering Research on Augmented Intelligence in Systems Engineering and Engineered Systems. Cham: Springer, 2025: 77-91.

[18] Stocchero J M, Silva C A, De Souza Silva L, et al. Secure command and control for internet of battle things using novel network paradigms[J]. *IEEE Communications Magazine*, 2023, 61(5): 166-172.