Quantitative Models for Evaluating the Efficiency and Resilience of a Cyber Range for Educational and Research Purposes*

Roman Yaroviy [1], Olena Skliarenko[1,*], Anatolii Kozynets[1], Valerii Lakhno[2]

[1] Private Higher Educational Establishment "European University", Vernadskoho Akademika Blvd 16 , 03115 Kyiv, Ukraine
[2] National University of Life and Environmental Sciences of Ukraine, Heroiv Oborony 15, 03041 Kyiv, Ukraine

Abstract

This paper presents a comprehensive approach to evaluating the efficiency and resilience of a cyber range designed for educational, training, and research purposes in cybersecurity. A generalized methodology for constructing integral objective functions is proposed, enabling the combination of key performance, reliability, and fault-tolerance indicators into a single quantitative index. Two mathematical models are developed: the first is based on a weighted sum of metrics (throughput, resource utilization efficiency, probability of successful session completion, and system resilience coefficient), while the second serves as an optimization criterion for resource scheduling and load management tasks. Particular attention is paid to a peak-load scenario that involves simultaneous execution of laboratory classes, Capture The Flag (CTF) competitions, and research tasks. For this scenario, resource utilization coefficients are calculated for CPU, RAM, storage, and network subsystems, as well as the probability of successful completion for all active sessions, overall system throughput, and average response time. The results show that even with 140 concurrent virtual machines, the system operates at only about 60% of CPU capacity, with a significant memory and network headroom, ensuring the absence of service queues and failures. Recommendations are formulated for scaling and load distribution to maintain stable operation under stress conditions. The proposed models can be applied not only for monitoring and performance evaluation but also for automated scheduling of educational events, resource allocation optimization, and enhancement of training quality. These results can serve as the foundation for the development of adaptive load management systems capable of real-time control, thereby improving the reliability, efficiency, and cost-effectiveness of cyber range operation.

Keywords

Cyber range, performance indicators, system resilience, mathematical modeling, throughput, objective function, performance evaluation, simulation modeling, optimization

1. Introduction

Cyber ranges are increasingly used as a critical component in cybersecurity education and training, providing realistic, isolated environments for practical exercises, Capture The Flag competitions, and red/blue team scenarios. As the demand for such platforms grows, so does the need for systematic evaluation of their performance and reliability. A cyber range must not only provide sufficient computational and network resources but also guarantee a high probability of successful session completion under peak load [1,3-5].

During the training of students majoring in F5 «Cybersecurity and Information Protection», an essential stage is the acquisition of practical skills in monitoring information security events and implementing protective measures under conditions simulating real-world attacks on organizational

* Applied Information Systems and Technologies in the Digital Society (AISTDS-2025), October 01, 2025, Kyiv, Ukraine
1* Corresponding author.
roman.yaroviy@e-u.edu.ua (R. Yaroviy), olena.skliarenko@e-u.edu.ua (O.Skliarenko), anatoliy.kozinets@e-u.edu.ua (A. Kozynets), lva964@nubip.edu.ua (V. Lakhno)
0000-0001-8978-8137 (R. Yaroviy), 0000-0001-6555-1223 (O. Skliarenko), 0000-0002-3926-4981 (A. Kozynets), 0000-0001-9695-4543 (V. Lakhno)

information systems. One of the most effective approaches to ensuring high-quality practical training is the integration of a cyber range into the educational process [11,13,16].

A cyber range is a multifunctional hardware-software complex designed to conduct cyber exercises by simulating computer attacks and enabling participants to practice effective responses. Such a platform provides students with an opportunity to analyze complex attacks, develop collaboration skills, and gain experience in localizing threats and eliminating the consequences of cybersecurity incidents. Currently, several advanced solutions are available, such as the Federal Cyber Defense Skilling Academy (CISA), the Georgia Cyber Innovation & Training Center, the SANS Institute platform, and CyberUkraine, offering comprehensive attack simulations including phishing, distributed denial-of-service attacks, malware deployment, network infrastructure exploits, and data confidentiality breaches [2,9,10].

The deployment of such platforms, however, requires significant financial resources and may be difficult for institutions with limited budgets, as the return on investment can be lengthy. This highlights the relevance of research aimed at creating small-scale educational cyber ranges that use open-source software and virtualization to simulate attacks [6-8]. The development of such systems involves the joint efforts of technical, methodological, and organizational teams responsible for architecture design, training material development, exercise scripting, planning, and testing [15,17]. Ultimately, a cyber range reproduces the infrastructure of a typical enterprise network, including servers, workstations, network devices, corporate software, and security tools, within a virtual environment [12]. This makes it possible to model realistic attack-defense scenarios and assess the efficiency of technical and organizational response measures. The need to ensure reliable, measurable, and scalable operation of such systems justifies the development of quantitative models for evaluating their efficiency and resilience, which is the focus of this paper [14].

2. Research Methodology

The study is based on a combination of system analysis and mathematical modeling. Quantitative indicators of cyber range operation were identified, including throughput, resource utilization efficiency, probability of successful session completion, resilience coefficient, waiting time, and load function. The first model represents an integral objective function constructed as a weighted sum of these indicators, allowing the overall performance score to be determined under different operating conditions. The second model serves as an optimization function for resource allocation tasks, making it possible to balance performance, delay, and risk of overload.

A peak-load scenario was modeled, corresponding to 140 concurrent virtual machines running laboratory exercises, CTF competitions, and research tasks. Resource utilization coefficients for CPU, RAM, storage, and network were calculated, as well as the probability of successful completion of all sessions. The sensitivity of the models to variations in weighting coefficients and load scenarios was studied using simulation methods, including Monte Carlo techniques. The modeling process employed orchestration systems, statistical analysis tools, and software environments such as Python and MS Excel to validate the robustness of the proposed approach.

3. Evaluation of the Efficiency and Resilience of a Cyber Range under Combined Loads

3.1. Technical specifications of the cyber range

- Computational Power (R_{cpu}): Each server has two Intel Xeon processors with 12 cores each (24 physical cores per server) running at a frequency of ~2.5–3.0 GHz. In total, 20 servers provide approximately 480 cores (≈960 threads with Hyper-Threading enabled). This is equivalent to about 1.2–1.4 thousand GHz of total frequency available for virtualization.

- RAM (R_{ram}): Each server is equipped with 128 GB of RAM. Therefore, the total memory volume is 20 × 128 GB = 2560 GB (≈2.5 TB) available for deploying virtual machines (VMs).

- Storage Systems ($R_{storage}$): Each server is equipped with local SSD drives of about ~2–4 TB. A shared network storage system may also be available for VM images. We estimate the total available storage capacity at ~50 TB (across all servers). Disk access speeds reach hundreds of thousands of IOPS, minimizing input-output delays.

- Network Bandwidth ($R_{network}$): The network infrastructure provides high throughput. Each server is connected via 10 Gbit/s to the network core. Workstations are connected via 1 Gbit/s to the training network. This configuration provides sufficient data transfer between Red/Blue Team segments and to the Internet simulator, even under peak loads (attacks, DDoS simulations, etc.).

- Software Resources ($R_{software}$): The system has a sufficient set of licenses and software. Licenses for the hypervisor covering 20 hosts are available, allowing all servers to be used. There are enough VM images and templates for training environments. Blue Team tools (SIEM, IDS/IPS) are installed with 50 client licenses for each of the key products, which is sufficient for all active participants. Overall, software resources are adequate, and no license limitations restrict VM deployment.

3.2. Load Scenario (Calculation of U(t))

Let us consider the peak usage case of the cyber range when multiple activities occur simultaneously: a CTF competition is underway while a training course with laboratory work is running in parallel. We also account for minor background loads from research tasks. The event stream intensities at time t_0 (the peak situation) are:

- $U_{courses}(t_0)$: A laboratory session is simultaneously conducted for a group of 20 students (all workstations are occupied). Each student deploys one training VM. Hence, the course stream = 20 events (20 VM deployments).

- $U_{exercises}(t_0)$: In parallel, instructors (White Team) launch several autonomous training exercises for additional groups or testing. Let this add 10 events (10 VMs for demonstration attacks/scenarios).

- $U_{ctf}(t_0)$: At the same time, a Capture The Flag competition is held with remote participants. Suppose 50 participants are simultaneously solving tasks. On average, each participant requires 2 active VMs (one for attack and one for defense/target). Thus, the total number of CTF events is about 100. This stream is pulsating in nature, but at the peak, we take the maximum load.

- $U_{research}(t_0)$: Background research activity – for instance, administrators launch several test VMs to verify new software or simulate an APT attack. Let this be 4 VMs at this moment.

Thus, the total intensity at time t_0 is:

$U(t_0) = 26 + 10 + 100 + 4 = 140$ events simultaneously

This corresponds to 140 active virtual machines running in one time interval (peak concurrent sessions/environments).

3.3. Calculation of key performance indicators for this scenario.

Overall load $L(t)$. To estimate the overall load $L(t)$ we compare the requirements $U(t)$ with the resources $R(t)$. Because the resources are heterogeneous, we analyze the most critical component. In our case the computing resources (CPU) are decisive, since memory and network are used less intensively relative to the capabilities of the servers.

CPU utilization. Each event (VM) on average consumes, say, 2 vCPUs (a dual-core VM). For 140 simultaneous VMs about 280 vCPUs are needed. The available resource is 480 cores (about 480 vCPUs equivalent). Thus the relative CPU load

$\tau_{cpu} = 280480 = 0.58$, {≈58% of cluster capacity}

That is, more than half of the CPU resources are occupied, but there is still a substantial reserve until the critical level (42% of the CPU capacity remains). This corresponds to a coefficient $L(t_0) \approx 0.58$ (when normalized by CPU). The system operates in normal mode, L(t) < 1, there is no overload.

Memory utilization. If each VM consumes an average of 4 GB of RAM, then 140 VMs require ≈ 560 GB of memory. This is only ≈ 22% of the available 2,560 GB. Thus $\eta_{ram} \approx 0.22$. Memory is not a bottleneck in this scenario (reserve ~78%).

Network utilization. Assume that an average virtual machine generates approximately 2 Mbit/s of traffic (including user activity, updates, attack traffic, etc.). For 140 VMs, this results in a total of about 280 Mbit/s. Even if the traffic is unevenly distributed and reaches several Gbit/s during peaks (for example, during a DDoS attack on one of the segments), the network infrastructure (10 Gbit/s per server, 200+ Gbit/s core) is capable of handling the load. The approximate average utilization of the main link remains below 3% (280 Mbit/s out of 10 Gbit/s per server). Therefore, the network resource is not overloaded, and $\eta_{network}$ remains very low on average (only a few percent).

Conclusion for L(t). In the peak scenario, the total relative load is approximately $L(t_0) \approx 0.58$ (58% of CPU resources, which represent the most heavily utilized component). This value is below the threshold level of 1, meaning the system operates with a performance margin. Under such conditions, the cyber range remains both stable and efficient, with no queues or service failures caused by resource shortages.

$$\mathrm{T} = \frac{1}{t_s} N_{ks}, \tag{1}$$

The average duration of a single session is denoted as t_s, and the number of sessions that can run simultaneously is denoted as N_{ks}.

At the same time, there are 140 sessions. If the average session lasts, say, 4 hours (laboratory work or a CTF task), then over an 8-hour training day one "station" can host two sessions (morning and afternoon). With 140 parallel sessions this yields up to 280 sessions per day. Theoretically, over a full 24-hour day with continuous operation one could support about 840 sessions. In reality the schedule is not continuous, but the cyber range can conduct several hundred training events per day at full load, which is a very high figure. This confirms that the system throughput T is high due to the ability to run different segments (training courses, CTF, exercises) in parallel without interference.

- Resource utilization efficiency η. As noted, the CPU is loaded at about 58% and RAM at about 22% at the peak. One can calculate these coefficients for our scenario:

$\eta_{cpu}(t_0)$= (number of events × average CPU per event) / R_{cpu} ≈ 0.58. That is, a little more than half of the processor cores are utilized. The rest remain free, providing a reserve in case of spikes or additional tasks.

$$R_{cpu} = \frac{140 * 2}{480} \approx 0.583\ (58.3\%). \tag{2}$$

Thus a little over half of the processor cores are engaged. The remaining cores stay free, providing a reserve for load spikes or additional tasks.

$\eta_{ram}(t_0)$ = (number of events × average RAM per event) / R_{ram} ≈ 0.219 (21.9%). Most of the memory remains unused; this is normal because the system is equipped with memory with a reserve for different scenarios (some specific tasks may require much more RAM per VM).

$\eta_{storage}(t_0)$ – the percentage of storage utilization depends on how many images and data are loaded. For example, if each of the 140 VMs uses an image of ~20 GB, then ~2.8 TB out of ~50 TB (≈5.6%) are occupied. During operation (logs, dumps) a few more percent may be added. Thus storage is not fully utilized either.

$\eta_{network}(t_0)$ – as mentioned, an average network load of a few hundred Mbit/s versus the available tens of Gbit/s gives only a few percent. Even peaks (say 5 Gbit/s when simulating a large DDoS attack on one of the segments) will be < 3% of the core network and 50% on a single server (if all the traffic goes through that one server).

Conclusion about η. In this scenario resources are used in a balanced way with sufficient reserve. The CPU is the most loaded (~60%), which is close to optimal for efficiency (not idle but not overloaded). Other resources are used less intensively, which indicates the possibility of scaling.

Probability of successful session completion P_{succ}. Based on the obtained $L(t_0) \approx 0.58$ and the known high resilience of the system C(t), one can expect a very high probability of successful servicing of all sessions. Formally: $P_{succ} = P\{C(t_0) > C_{min} \cap L(t_0) \leq L_{max}\}$. Under our conditions $L(t_0)$ is less than the threshold Lmax (which can be taken as ≈1 or slightly lower), meaning the load does not exceed the critical level.

Additionally, $C(t)$ due to isolation, redundancy, and protection, remains high. The absence of failures ($C(t) > C_{min}$) depends on several stochastic factors $x_{iattack}, x_{ihuman}, x_{itech}$:

- The probability of an external attack on the cyber range during an event is low (the system is isolated, and Cprotection is high).
- Human factors (administrator errors) at that time are minimized through procedures and supervision (the White Team monitors the events).
- Hardware failures are rare: the failure of 1 out of 20 servers is an unlikely event over a short period (e.g., probability ~1–5% per day if servers are reliable). Even if a single server fails, the system has redundancy: at 58% load, shutting down one server (~5% capacity) slightly increases the overall L(t) to ~61% — the remaining servers can take over its VMs, or they are automatically restarted on backup resources (thanks to the orchestrator and $C_{redundancy}$). Thus, PsuccP_{succ}Psucc decreases only slightly.

A qualitative estimate of P_{succ}: under normal conditions (no failures), all 140 sessions complete successfully (probability ~100%). Considering the small probability of random issues, P_{succ} can be estimated at ~0.95–0.99 (95–99%). In other words, nearly all sessions complete successfully. The system was designed for reliable operation, so even at peak load, infrastructure-related failures are rare.

For comparison, if the load were critical L(t) ⟶ 1 simultaneously with low resilience (e.g., a hardware failure occurs, $C(t) < C_{min}$)), P_{succ} would drop sharply. Our cyber range, however, due to resource reserves and system resilience, maintains a high probability of successfully conducting all planned activities.

Average waiting/response time τ. Under moderate load (L < 1) the automated orchestration system provides quick resource allocation for new events. An M/M/1 queuing model for estimating waiting time gives an approximate formula.

$$\tau \approx \frac{U(t)}{R(t)\,(R(t) - U(t))} \tag{3}$$

If one interprets R(t) as the total throughput (e.g., the number of VMs that can be initialized per unit time) and U(t) as the intensity of requests to start VMs, τ can be estimated. At low load τ will be close to zero (requests are serviced almost immediately). When L(t) = 0.58 the waiting time is still small; only at very high loads does it grow significantly.

A simplistic formula may yield a negative denominator (which would mean that 140 simultaneous requests exceed a linear throughput of 20 VMs/minute). However, in practice the orchestrator launches VMs in parallel on many servers, so the effective throughput is much higher. If we distribute 140 launches across 20 servers, each server needs to start only 7 VMs, which is quite feasible.

Thus the above calculation shows that the described cyber range (20 HP servers, 26 workstations, a powerful network) has significant resource reserves R(t) and a high resilience coefficient C(t). At typical and even peak loads U(t) the system operates efficiently: the overall load is maintained below the threshold (L(t) < 1), resources are not saturated, failures and queues are minimized.

In this scenario we see that even with a combined load (simultaneous CTF + training courses) the system does not reach its limits. This means that the cyber range can scale to even more complex events. For example, the residual resources allow the number of simultaneous participants to be increased.

4. Objective Functions for Evaluating Cyber Range Performance

Let us explore the objective functions for evaluating the operation of the cyber range.

For a formal evaluation of the cyber range's performance, quantitative indicators reflecting various aspects of its operation are considered. The key metrics of cyber range efficiency include the following.

- Throughput T: the number of events or sessions serviced per unit time. Higher throughput means greater efficiency and productivity of the system (more users or events processed in a given time).

- Probability of successful session completion P_{succ}: the proportion of sessions that finish successfully (without failures). This metric characterizes the reliability and success of the system – a high value of P_{succ} indicates a low frequency of errors or failures (for example, P_{succ} =0.98 means that 98% of sessions proceed without disruptions).

- Resource utilization efficiency η: reflects how efficiently the resources of the cyber range (CPU, memory, etc.) are used. It can be defined as the ratio of utilized resources to available resources (e.g., average CPU load) or as the ratio of performance to resource consumption. High η means that resources are not idle but are also not overused.

- Resilience coefficient C(t): a measure of fault tolerance or resilience of the system as a function of time. This coefficient may be defined in different ways – for example, as the level of maintained functionality during failures or attacks, the speed.

5. Model 1. Generalized integral evaluation of cyber range efficiency

Model description. The first model combines the main performance indicators of the cyber range (efficiency, success and resilience) into a single integral evaluation. This is achieved by calculating a weighted sum or other combination of metrics using weighting coefficients. The idea is similar to the scalarization technique in multi-criteria optimization: each criterion is assigned a weight and they are summed into a single index. Such an approach is often used to obtain a performance score that accounts for multiple KPIs (Key Performance Indicators).

Formalization. Suppose we need to take into account three groups of indicators corresponding to:

efficiency (E) – reflects productivity and the rational use of resources;

success (S) – reflects reliability and quality of work (absence of failures);

resilience (R) – reflects resistance to overloads and failures.

Then the integral objective function can be written as a weighted sum:

$$F_1 = \alpha \cdot E + \beta \cdot S + \gamma \cdot R, \tag{4}$$

where α, β and γ are weight coefficients corresponding to the importance of each group of indicators. The sum of the weights is usually normalized (for example α + β + γ = 1), but this is not mandatory – the weights can be selected according to priorities without normalization.

Component details.

Let us define which specific metrics are included in E, S and R.

The efficiency component (E) may combine throughput capacity and resource efficiency, as well as take delays into account.

$$E = f_E(T, \eta, \tau) = \omega_T \frac{T}{T_{ref}} + \omega_\eta \frac{\eta}{\eta_{ref}} + \omega_\tau \frac{\tau}{\tau_{ref}}, \tag{5}$$

where $T_{ref}, \eta_{ref}, \tau_{ref}$ denotes certain reference (normalizing) values for scaling, and τ, T, η are local weights within the efficiency component (these can be selected or assumed equal for simplicity).Thus E increases with increasing throughput and resource efficiency, but decreases as waiting time increases. Including τ with a minus sign means that large delays reduce the overall score.

The success component (S) can be directly represented by the probability of successful session completion, i.e. $S = P_{succ}$.

If necessary, this component can be extended with other reliability indicators, such as the frequency of errors or error rate. In this case P_{succ} is a convenient metric between 0 and 1 that directly reflects the fraction of successful operations (the closer to 1, the better).

The resilience component (R) may take into account the resilience coefficient C(t) as well as the system's behavior under load L(t). One way to define R is to assess the ability to maintain performance at high load. For example:

$$R = f_R(C, L) = C_{avg} \frac{T_{high\ load}}{T_{normal}}, \tag{6}$$

where C_{avg} is the average value of the resilience coefficient over a certain interval or scenario, and $T_{high\ load}/T_{normal}$ is the ratio of throughput at peak load L(t) to throughput under normal conditions. Such a formula gives an intuitive understanding of resilience: if at peak load the system maintains, say, 80% of its nominal throughput, then it is sufficiently resilient. In the general case one can simplify by assuming R = C_{avg} or another aggregated resilience metric if data are available.

Substituting these components into the formula for F_1 yields the full integral evaluation. For example, one possible expression (with normalization of metrics) may look like this:

$$F_1 = \alpha \left(\frac{T}{T_{ref}} + \frac{\eta}{\eta_{ref}} - \frac{\tau}{\tau_{ref}} \right) + \beta P_{succ} + \gamma C_{avg}, \tag{7}$$

where α, β and γ are the weights for efficiency, success and resilience, respectively. In practice the choice of specific forms f_E, f_S, f_R may vary. The main thing is that these three groups of criteria are combined linearly with given weights. Such a linear performance index converts a multidimensional assessment into a single number that can be tracked over time or used to compare different configurations of the range. Similar approaches are used in network management, for example to compute a hotspot score – an integral "hotness" index of a network segment, which is a weighted combination of many KPIs (failures, delays, overloads, etc.) with corresponding weights.

Stochastic weight coefficients. The weights α, β and γ can be regarded as random (stochastic) parameters if the significance of the criteria changes by scenario or is not clearly defined. This means that instead of fixed values one can specify probability distributions for the weights and perform a Monte Carlo analysis, calculating the expected value, variance and distribution of F1 over many random weight realizations. Stochastic weights make the integral function flexible and better suited for practical situations where priorities may change; for example, under high load resilience should be a priority, while under normal load efficiency becomes more important.

Interpretation. The value of the integral function F1 provides a generalized assessment of the cyber range's efficiency. Its economic and practical meaning is that it combines different aspects of performance into a single score that can be used by decision-makers. An increase in F1 typically indicates improvements in throughput, reliability or resilience and a reduction in delay. Conversely, a decrease points to problems that require attention.

- The efficiency component (with weight α) reflects the productivity per unit of resource and has the economic meaning of return on invested resources. The more tasks processed (T) with full use of memory, storage and network ($\eta \approx 1$) and minimal delay ($\tau \to 0$), the higher the return.
- The success component (with weight β) shows the quality and reliability of the service. A high P_{succ} means fewer failed sessions, i.e., less wasted effort and time. In practical terms this may mean fewer reworks, higher user satisfaction and saving human and machine resources.
- The resilience component (with weight γ) characterizes the stability of the system under stress. If the system is resilient it will avoid expensive downtime and recover quickly after problems. Economically this reduces losses from interruptions and ensures continuity of the educational process.

Thus the objective function F1 allows management or automated monitoring systems to quickly assess how balanced and well the cyber range is performing overall. Its value can be used to compare different periods, configurations or event types: an increase signals improvement, while a decrease suggests the need for optimization.

6. Model 2. Alternative objective function for optimization tasks

Model description. The second objective function is intended for direct use in optimization problems such as scheduling tasks on the cyber range or modelling usage scenarios of its resources. It can serve as the fitness function for algorithms (genetic or heuristic) that search for the best schedule.

Main idea. Build F2 so that it explicitly reflects the trade-offs between different goals important in planning. In the context of scheduling or resource management of the cyber range one usually has to balance:

- maximizing throughput (so as to serve as many sessions as possible in the available time);
- maintaining a high probability of success (not overloading the system until success decreases);
- minimizing waiting time for new sessions (users should not wait long for their turn);
- efficient use of resources (avoiding both idle time and over- load ing) ;
- ensuring resilience (avoiding operating modes close to failure).

Based on this, one can propose an objective function of the "benefit minus cost" type. For example, maximize the number of successfully completed sessions per unit time (this is the benefit) while minimizing delays and the risk of failure (the costs).

Possible formalization. Consider a certain plan or allocation scenario X (for example, a schedule for launching N sessions on the available servers, or a distribution of resources among competing requests). Define the function as a sum of weighted terms:

$$F_2(X) = \alpha'(T_{succ}(X)) + \beta'(\eta_{avg}(X)) - \gamma'(\underline{\tau}(X)) - \delta'(L_{peak}(X)) \quad (8)$$

Explanation of this expression:

- $T_{succ}(X)$ – the effective throughput for plan X, i.e., the number of sessions successfully completed per unit of time, can be expressed as P_{succ} under the conditions of implementation X (since not all initiated sessions will successfully complete under high load). This term, with the coefficient α', reflects the objective of maximizing the number of successful services per unit time – in effect, a pursuit of high productive throughput.
- $\eta_{avg}(X)$– the average resource utilization efficiency under scenario X is included with a positive weight β' in order to encourage full loading of the cyber range. For instance, if the servers operate at only 50% capacity on average, the plan can be improved by consolidating tasks (within reasonable limits so as not to compromise the session success rate). This component ensures that resources are not wasted, which is particularly important in scheduling problems where resources are costly or limited.
- $\underline{\tau}(X)$ – the average waiting time (or total waiting of all tasks) in plan X. It is subtracted with coefficient γ′ (we penalize waiting). Thus optimizing F2 stimulates solutions where each session starts quickly and resources are used productively.
- $L_{peak}(X)$– the peak load or overload index in scenario X may represent, for example, the maximum number of concurrent sessions or the percentage of resource utilization at the most loaded moment. Including the term $\delta'(L_{peak}(X))$ (or alternatively $\delta'(L_{peak}(X))$- $\delta'(L_{safe}(X)$ to penalize exceedance of a safe threshold) ensures that any plan pushing the

system into a risk zone of overload receives a lower score. This penalty is directly related to system resilience: it indirectly promotes plans that maintain the load at a stable, sub-critical level, thus preserving system reliability and avoiding scenarios where C(t) experiences a sharp decline.

The coefficients α', β', γ' and δ' are tuned according to the priorities of the problem. For instance, if the most critical objective is to serve the maximum possible number of users, α' can be assigned a very large value while the penalty terms are kept smaller. Conversely, if avoiding waiting queues is a higher priority, β' can be increased, and so forth.

The presented form of F2 is just one possible variant. Depending on the type of optimization problem, the alternative objective function can take other forms. We give a few examples.

– Multiplicative form (geometric):

$$F_2^{/}(X) = TP_{succ}C, \tag{9}$$

which essentially maximizes throughput, success and resilience simultaneously. This criterion is strict: if any of the factors is small, the product decreases sharply. It can be useful if you need to ensure high performance on all criteria and avoid trade-offs.

– Benefit-to-cost ratio

$$F_2^{//}(X) = \frac{TP_{succ}}{k_1 T + \frac{k_2}{C}}, \tag{10}$$

where the numerator is an indicator of successful performance and the denominator is a composite cost metric (average waiting time and the risk of failure, expressed as C or probability of failure). The goal would be to minimize this ratio of cost relative to benefit.

Regardless of the specific form, F2 serves as the objective function for planning, i.e., the criterion that the optimization algorithm attempts to improve by exploring possible scenarios for using the cyber range.

Interpretation of F2. Practically, this function can be viewed as the “utility” of a given way of operating the cyber range. Maximizing F2 means finding a schedule or resource distribution that ensures:

– as many sessions as possible are successfully completed in the given period (users get results and the system delivers maximum value);
– resources work productively without idle time (which justifies investment in the infrastructure);
– users/tasks hardly wait (time is money; reducing waiting increases the efficiency of the personnel and equipment);
– at the same time there are no extreme loads such that the system begins to fail or accumulate queues (that is, a balance is maintained where the system operates in a stable regime without risk of accidents, which could lead to longer recovery and additional expenses).

In other words, F2 can be associated with the economic benefit from operating the cyber range in mode X minus the operating costs/risks of that mode. This makes it very convenient for planning: the plan with the highest F2 will be the most profitable.

7. Practical application of the objective functions

7.1. Simulation modeling

Both functions F1 and F2 can be used as criteria within simulations. For example, by modeling different load scenarios L(t) (gradual traffic growth, sudden peaks, node failures, etc.), F1 can be calculated for each scenario as an aggregated score. This allows for comparison of scenarios and identification of "bottlenecks." F1 drops sharply under a certain type of load, this signals that the cyber range requires improvement under these conditions (e.g., scaling resources, optimizing code, etc.).

It is also possible to introduce weight distributions α, β, γ for different scenarios to model varying customer requirements or situations (for instance, for educational purposes, session success might be more important, whereas for stress-testing, fault tolerance is prioritized). Simulation with stochastic weights provides a distribution of possible integral scores and demonstrates how consistently the system meets different requirements.

Below in Figure 1 we provide graphs that illustrate the impact of individual key variables on the values of the objective functions F1 and F2. For each graph all other parameters are fixed at typical levels and one variable is changed.

The parameters are set as T=0.8 (moderate throughput), η=0.6 (60% resource utilization), P_{succ}=0.95 (95% success rate), and R=C=0.9 (high resilience). It can be observed that as latency increases, the value of F1 decreases linearly. The minimum latency (left side of the graph) corresponds to the maximum F1≈3.3, while approaching the threshold value (1.0 on the graph) reduces F1 to approximately 2.2. This demonstrates that response speed critically affects the overall efficiency: even with high throughput and success rates, excessive delays significantly reduce system effectiveness. This behavior aligns with practical metrics — for example, the mean time to respond (MTTR) should be minimized to reduce the impact of incidents.

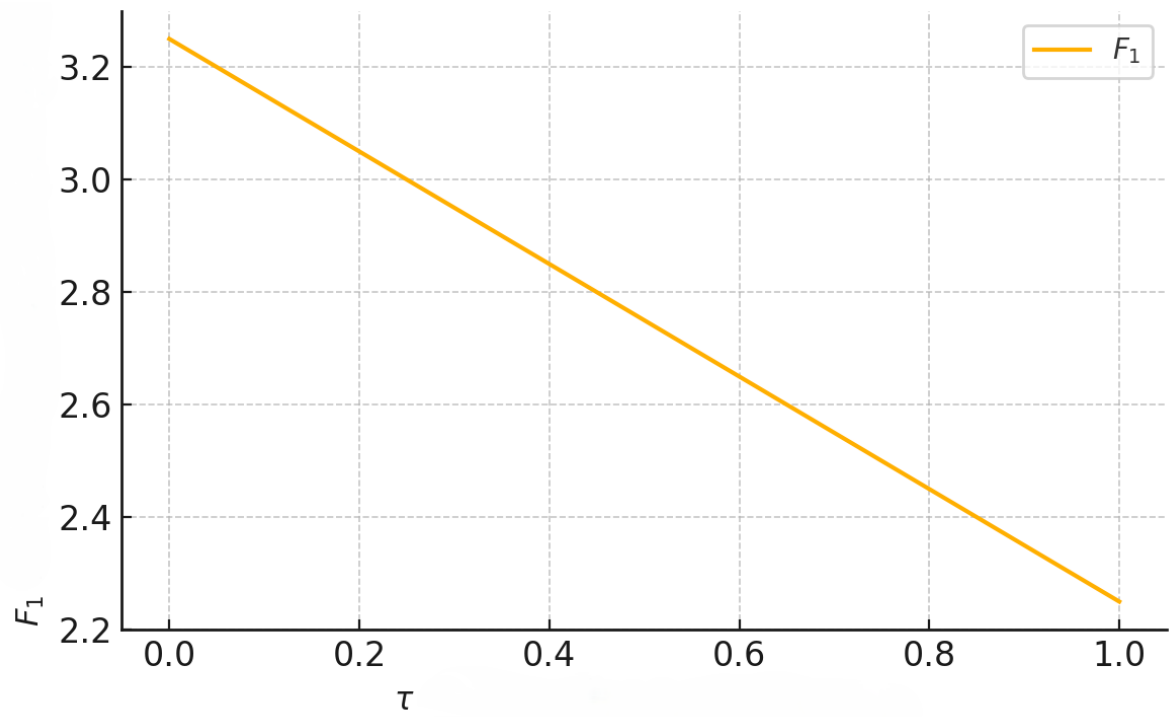

Figure 1. Dependence of the integral evaluation F1 on the reaction time τ (normalized).

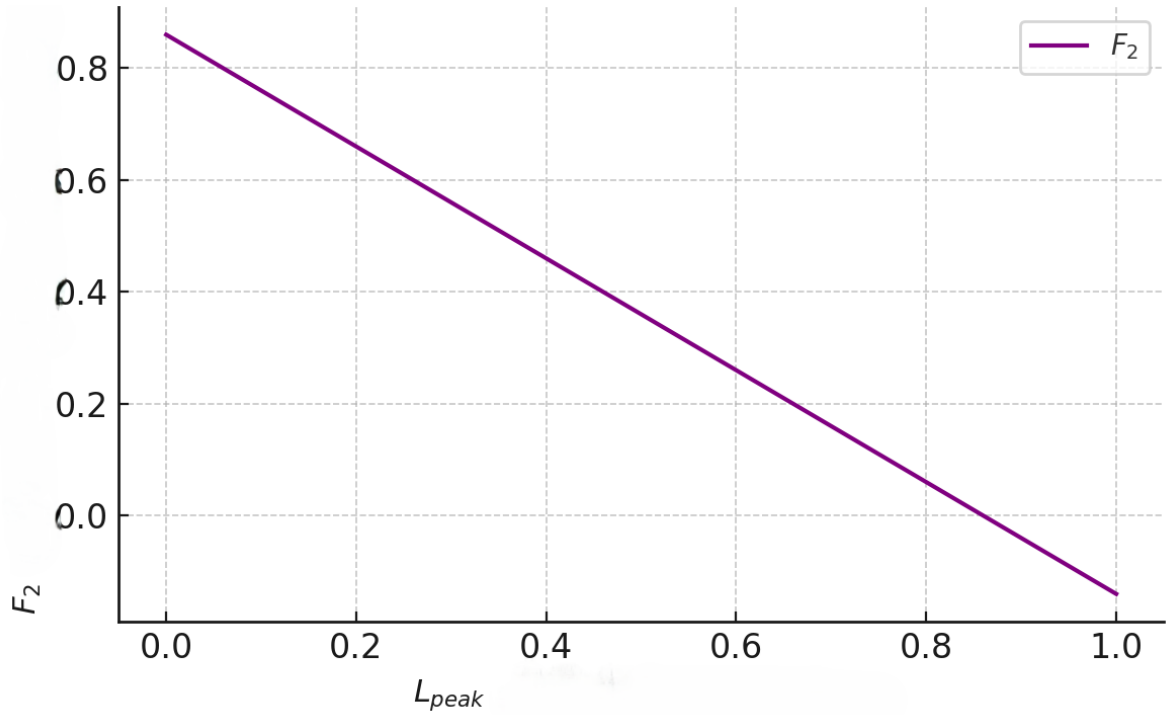

Figure 2. Dependence of the optimization function F2 on the magnitude of the peak load L_{peak}.

Under zero peak load (an idealized case with no peaks, $L_{peac} = 0$, the target function F2=0. (arbitrary units). As L_{peac} increases to 1 (i.e., 100% resource utilization), F2 decreases almost linearly to nearly 0. This reflects the negative effect of peak loads: the closer the system approaches full resource exhaustion during peak moments, the lower the overall efficiency according to the F2 criterion. In practice, this corresponds to the need to avoid operating at the system's maximum capacity — it is recommended to maintain a resource margin (e.g., keeping CPU and memory utilization below ~70–90%) to ensure stability.

In Figure 2 it can be seen that at $L_{peac} = 0.5$ (50% load) the function F2 is still quite high (~0.4–0.5), whereas at L_{peac} =1.0 it is almost zero, indicating an undesirable regime without resource reserves.

7.2. Three-dimensional dependencies and interaction of parameters

Next we will examine the dependencies of the objective functions on two variables simultaneously. The 3D surfaces illustrate the joint influence of parameters on the resulting indicator.

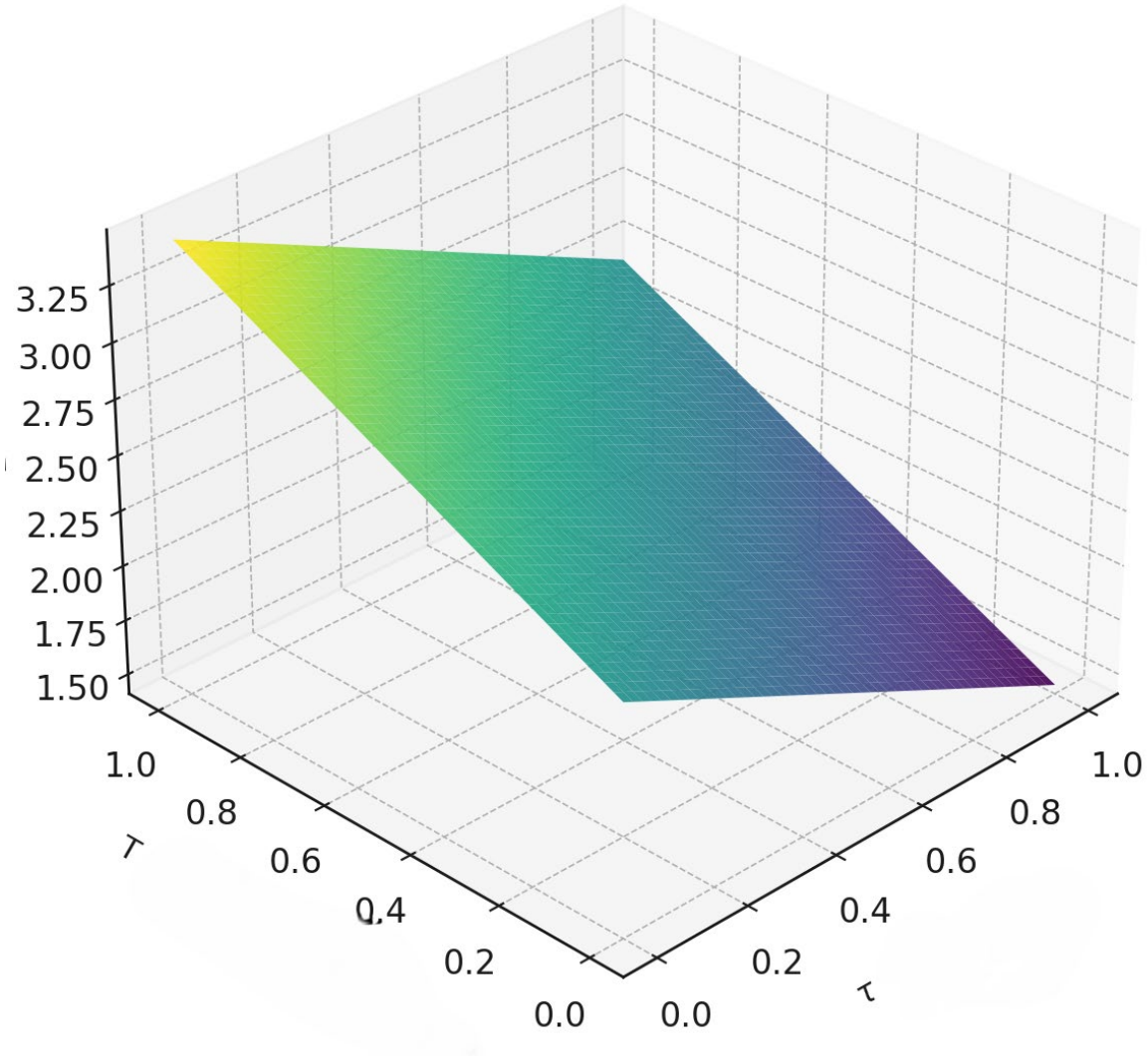

Figure 3. Surface of F1 in the coordinates τ (reaction time) and T (throughput capacity).

Here η = 0.6, P_{succ}=0.95, R=0.9; the weights α = β = γ = 1. The bright yellow area (in the far left corner) corresponds to the minimum reaction time (τ ≈ 0) and maximum throughput (T = 1); in this case F1 reaches its highest values (≈ 3.3 on the graph). The purple area (on the right closer to the viewer) corresponds to a large delay (τ ≈ 1) and low load (T = 0), giving the smallest F1 ≈ 1.5. The surface shows an approximately linear dependence: F1 increases proportionally to T and inversely to τ. As system throughput increases and delays decrease simultaneously, the integral assessment improves almost in a straight line. This picture confirms that to maximize F1 it is necessary to ensure a balance of high performance and low delay. In practice this means that scheduling on the cyber range should strive to achieve the best task execution (maximum events/transactions) while meeting response time requirements (for example, not exceeding the time limit for processing a request).

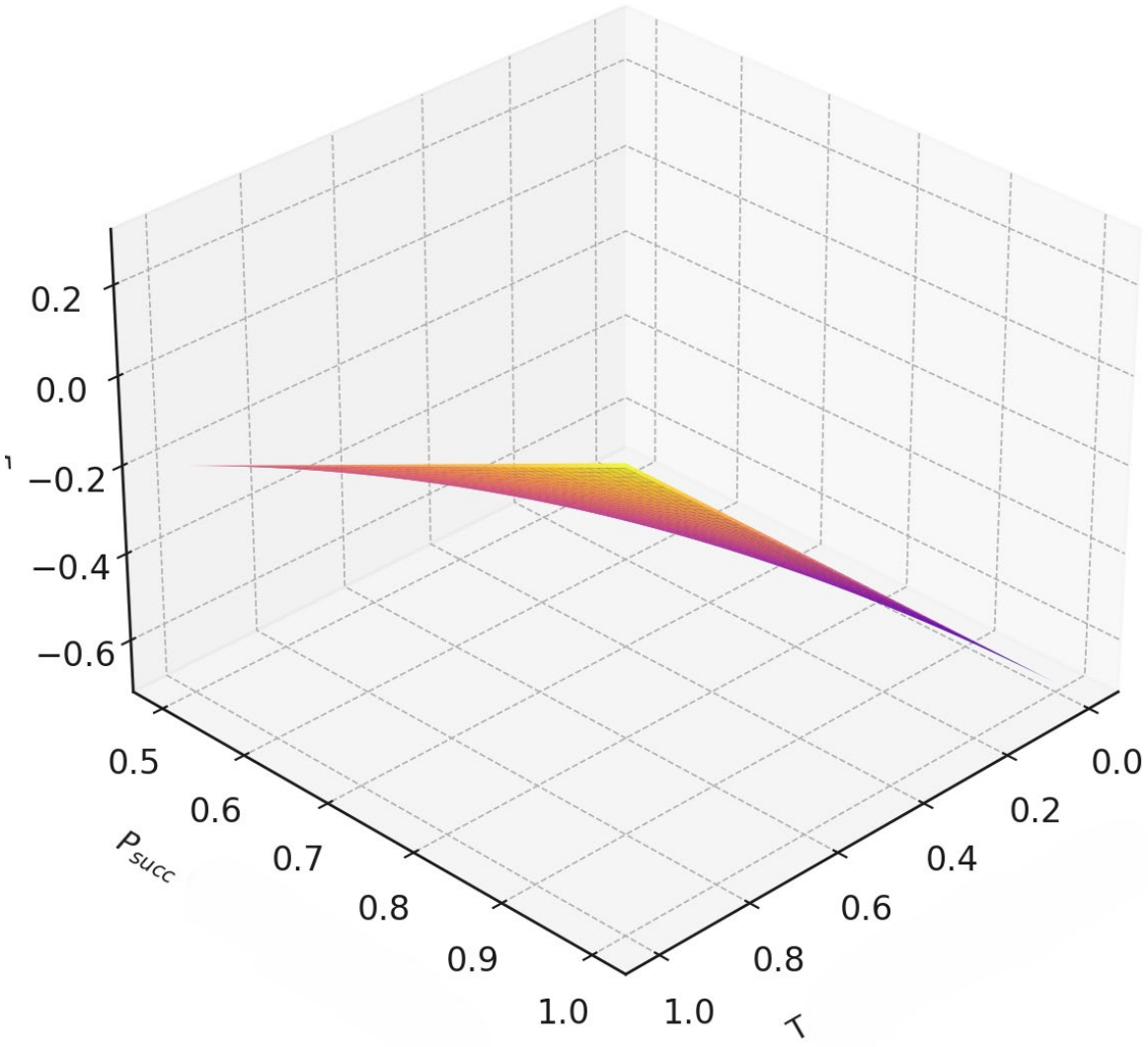

Figure 4. Surface of F2 in the coordinates T (throughput capacity) and P_{succ} (probability of success).

Parameters for building Figure 4: η = 0.6, R = 0.5, L_{peak}=0.8 and the weight coefficients α′ = β′ = γ′ = 1. It can be seen that the maximum value of F2 is achieved at the point where T is close to 1 (high load) and $P_{succ}$is close to 1 (high reliability). On the graph, this region corresponds to a slight peak (yellow–orange "bump") at T≈1, P_{succx}≈1, where F2≈0.3. Conversely, if either the load is very low or the success probability is far from 1, F2 takes on negative values (purple plane at the bottom, ~0.6). This indicates that for a high F2, the system must operate near optimal conditions: executing many tasks successfully. Low P_{succ} significantly reduces the target function even at high TTT, which aligns with the requirement to maintain a success rate of at least 95%. For example, at P_{succ}=0.6, even maximum throughput (T=1) does not yield a high F2 due to substantial losses from failures.

Thus, the surface demonstrates a nonlinear interaction: efficiency according to F2 drops sharply if Psucc decreases, and this cannot be compensated by increasing T. The optimum is reached near the far-right corner (maximum success probability and high load). In practical terms, this confirms that cyber range planning must ensure high reliability when scaling load — for example, by providing additional resources or fault-tolerant mechanisms — so that even at peak load the probability of successful execution remains ≥95%.

7.3. Automated scenario analysis

Consider several potential scenarios or configurations for using the cyber range (e.g., different schedules or different resource allocations among components). By calculating F1 for each scenario, it is possible to rank them according to overall efficiency. However, it is more appropriate to use F2, especially when optimization is the goal.

For example, search algorithms (genetic algorithms, greedy methods, or dynamic programming) can be employed to adjust scheduling parameters with the aim of maximizing F2. In this way, the system automatically identifies the scenario that provides the best trade-off between the number of tasks served, speed, and reliability. During such analysis, F2 serves as the objective function that the algorithm attempts to optimize.

Constraints can also be incorporated into the function — for instance, ensuring that Psucc does not fall below a certain threshold or that τ does not exceed a specified maximum. These constraints reflect critical requirements that cannot be violated. Automated analysis thus explores different alternatives (possibly using queue or load simulations) and, using F2 as the evaluation criterion, selects the optimal scenario.

7.4. Adaptive planning and real-time management

If the cyber range operates in real-time with variable load (e.g., student groups connect for training at different times), the target functions can be integrated into an automatic load management system. For instance, a monitoring system continuously calculates the current F1, and if it detects a downward trend (due to increasing τ or decreasing P_{succ}), it can alert a dispatcher or an automatic orchestrator to take action (e.g., scale infrastructure, postpone the launch of new sessions, etc.).

A more advanced approach involves an adaptive controller that dynamically adjusts the schedule to maximize F2. This can be implemented via a prioritization system: under high load, the penalty weights for τ and overload are increased, making the scheduling algorithm more conservative and preventing queues from growing excessively. When the load decreases, the efficiency weight can be raised again to utilize all available resources. In other words, the coefficients $\alpha, \beta, \delta, \tau$ in F2 can adaptively change depending on the system state or external priorities, while the management system continuously makes decisions aimed at maximizing the current value of the target function.

7.5. Use in schedule planning

For long-term planning (e.g., scheduling training sessions for a week in advance when the number of participants and types of tasks are known), F2 can be used to identify the optimal allocation of sessions over time and resources. This is a typical scheduling problem with an optimality criterion. The F2 model allows formulating it as an integer programming or queue optimization problem, where the variables represent how many sessions to run in parallel in each time slot, how many resources to allocate, and so on. The objective is to maximize F2 (or minimize F2 if it is more convenient to treat it as a cost function). Solving such a problem provides an optimal (or near-optimal) schedule. For example, it may show that it is better to run no more than N sessions simultaneously to keep P_{succ} above 95%, while other sessions are queued, slightly increasing τ, because the overall trade-off is more favorable. This approach ensures that scheduling decisions are based on a quantitative criterion rather than intuition, selecting the timetable that best balances the given priorities (efficiency vs. service quality).

7.6. Limitations

Despite the development of two formalized models (F1 and F2) and their simulation-based analysis, the present study has several limitations. In particular, the obtained results are still of a model-based nature. Therefore, future research will focus on verifying the proposed models using empirical indicators of the cyber range system's performance and stability.

Several simplifications were also applied during the modeling process. Specifically, an average workload on the virtual machines of the cyber range was assumed. Moreover, the weighting coefficients α, β, and γ were determined experientially based on priority. To enhance objectivity, the next stage of the research should involve formalizing their determination through multi-criteria optimization methods, such as NSGA-II or NSGA-III [18, 19].

At this stage, it will also be appropriate to compare the proposed models with existing approaches to assessing cyberinfrastructure performance, including the methodologies developed by MITRE and NIST. The implementation of these refinements in future work is expected to improve the reproducibility of the results and enable the integration of the models into cyber range monitoring systems.

7.7. Comparing strategies and configurations

If different architectural configurations of the cyber range exist (e.g., varying numbers of servers, different load balancing algorithms, or different security protocols that can affect performance), the target function F1 can be used as a single metric for comparison. For example, configuration A yields F1=0.8 under typical load, while configuration B yields F1=0.9. This may indicate that configuration B is generally better in terms of overall efficiency, success rate, and resilience. If the configurations have different strengths (one offering higher throughput, another better resilience), comparisons can be performed using different sets of weights α, β, γ, reflecting various usage scenarios. This enables selecting the configuration that is optimal according to the specified priorities. This highlights an important point: involving experts to determine the weights is a way to formalize requirements for the cyber range. Depending on their input, the weights are adjusted, and the system is optimized according to these criteria.

Finally, let us emphasize that the proposed objective functions are tools for decision-making. They help to quantitatively measure what has been qualitatively formulated as requirements for the cyber range. Combining metrics into one objective makes it possible to focus on maximizing useful work while minimizing resource consumption and losses. Their application helps to ensure an integrated approach to management and optimization.

In conclusion, it should be emphasized that the proposed target functions serve as decision-making tools. They help quantitatively measure what has been qualitatively formulated as requirements for the cyber range. By combining key metrics into a single indicator, we obtain a clear and manageable parameter. Its practical significance lies in balancing performance and quality: maximizing useful work with minimal resource consumption and losses (due to delays or failures). Applying these functions in practice—whether for monitoring or automated planning—enables organizations to operate their cyber ranges more efficiently, ensuring a high return on investment in infrastructure and training, while simultaneously maintaining high quality and reliability in educational or testing processes.

8. Conclusions

This study introduced a systematic approach to the quantitative evaluation of cyber range performance and resilience under combined load conditions. Two mathematical models were proposed: the first (F1) is formulated as an integral objective function that aggregates key metrics such as throughput, resource utilization efficiency, probability of successful session completion, and system stability into a single performance score. The second model (F_2) was developed as an optimization function for resource scheduling, enabling decision-makers to balance performance, latency, and risk of overload in a formalized manner.

Simulation experiments, including Monte Carlo analysis, were performed for a peak-load scenario involving 140 concurrent virtual machines executing laboratory exercises, CTF competitions, and research tasks. The results demonstrated that CPU utilization remains below 60 percent, memory and network subsystems operate with sufficient reserve capacity, and the probability of successful

session completion exceeds 95 percent, ensuring that the system maintains a high level of availability and service quality even under stress conditions. Sensitivity analysis confirmed the robustness of the proposed models to variations in weighting coefficients and workload distribution, making them applicable for a wide range of educational and research scenarios.

The proposed objective functions can be incorporated into monitoring dashboards as formal key performance indicators (KPIs) and used for continuous performance tracking, benchmarking of different configurations, and proactive detection of potential bottlenecks. In addition, the optimization-oriented model F2 provides a decision-support tool for administrators, allowing adaptive resource allocation and dynamic load balancing based on current operating conditions.

Future work should focus on expanding the models to handle heterogeneous workloads with prioritization schemes, integrating predictive analytics to forecast performance degradation, and implementing automated control loops for real-time orchestration. The use of machine learning techniques for anomaly detection, workload classification, and predictive scheduling represents a promising avenue for further improving the efficiency, resilience, and cost-effectiveness of cyber range infrastructures.

Overall, the study provides both a methodological and practical contribution to the design and operation of educational and research cyber ranges, offering tools that can help institutions enhance the quality of training, increase resource utilization efficiency, and ensure long-term scalability of their platforms.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] E. Ukwandu, M.A. Ben Farah, H. Hindy, D. Brosset, D. Kavallieros, R. Atkinson, C. Tachtatzis, M. Bures, X. Bellekens. *A review of cyber-ranges and test-beds: Current and future trends.* Sensors, 20(24) (2020), 7148. URL: https://doi.org/10.3390/s20247148.

[2] State Center for Cyber Protection. *Cyber training center.* (2024). URL: https://scpc.gov.ua/uk/cyber-trainer (accessed: 17.07.2025).

[3] D.R. Hokanson, B.J. Borghetti, J.M. Casey. *Evaluating authentic cybersecurity training in cyber ranges.* In: Proc. 54th ACM Technical Symposium on Computer Science Education, Vol. 1 (2023), pp. 846–852. URL: https://dl.acm.org/doi/10.1145/3544548.3581046.

[4] Tecnalia Research & Innovation. *Cyber Range Laboratory.* URL: https://www.tecnalia.com/en/infrastructure/cyber-range-laboratory (accessed: 17.07.2025).

[5] Group-IB. *Blue team cybersecurity for your business needs.* (2023). URL: https://www.group-ib.com/resources/knowledge-hub/blue-team/ (accessed: 17.07.2025).

[6] Leonardo Cyber & Security Solutions. *Cyber Range Brochure.* (2023). URL: https://cybersecurity.leonardo.com/documents/16277703/18499467/Cyber+Range+LQ+%28mm08974%29.pdf (accessed: 17.07.2025).

[7] J.D. Christopher. *The 2024 state of ICS/OT cybersecurity: Our past and our future.* SANS Institute Blog (2024). URL: https://www.sans.org/blog/the-2024-state-of-ics-ot-cybersecurity-our-past-and-our-future/ (accessed: 17.07.2025).

[8] S. Yevseiev, V. Ponomarenko, O. Laptiev, O. Milov et al. *Synergy of building cybersecurity systems.* Monograph. Kharkiv: PC TECHNOLOGY CENTER (2021), 188 p.

[9] NATO CCDCOE. *Locked Shields 2024 sets the stage for advancing global cyber defence.* (2024). URL: https://ccdcoe.org/news/2024/locked-shields-2024-sets-the-stage-for-advancing-global-cyber-defence/ (accessed: 17.07.2025).

[10] P. Nespoli, M. Albaladejo-González, J.A. Ruipérez-Valiente, J. Garcia-Alfaro. *SCORPION Cyber Range: Fully customizable cyberexercises, gamification, and learning analytics to train cybersecurity competencies.* arXiv preprint (2024). URL: https://arxiv.org/abs/2401.12594.

[11] T. Nadeem, Q. Zhang, S.C. Sundaramurthy. *A systematic method for measuring the performance of a cyber analyst in a SOC.* Computers & Security, 124 (2023), 102986. URL: https://www.sciencedirect.com/science/article/pii/S0167404822003510.

[12] MITRE. *Cyber resiliency metrics, measures of effectiveness, and scoring.* (2021). URL: https://www.mitre.org/sites/default/files/2021-11/prs-18-2579-cyber-resiliency-metrics-measures-of-effectiveness-and-scoring.pdf.

[13] N. Chouliaras, G. Kittes, I. Kantzavelou, L. Maglaras, G. Pantziou, M.A. Ferrag. *Cyber Ranges and TestBeds for Education, Training, and Research.* Applied Sciences, 11(4) (2021), 1809. doi:10.3390/app11041809.

[14] V.A. Lakhno, D.Y. Kasatkin, O.V. Skliarenko, Y.O. Kolodinska. *Modeling and Optimization of Discrete Evolutionary Systems of Information Security Management in a Random Environment.* In: Machine Learning and Autonomous Systems. Smart Innovation, Systems and Technologies, vol. 269. Springer, Singapore (2022), pp. 9–22. URL: https://doi.org/10.1007/978-981-16-7996-4_2.

[15] Cloud Range Cyber. *Measuring SOC performance in the cyber range.* (2023). URL: https://www.cloudrangecyber.com/news/measuring-soc-performance-in-the-cyber-range.

[16] L. Arsenovych, O. Nikolaievsky, O. Skliarenko, L. Lytvynenko, I. Kydriavskyi. *Organization of Training with the Use of Digital Technologies for Ensuring Cybersecurity in the Educational Space.* WSEAS Transactions on Computer Research, 12 (2024), pp. 524–536. doi:10.37394/232018.2024.12.51.

[17] A. Fesenko, O. Toroshanko, Y. Shcheblanin, I. Mykhalchuk, M. Pyroh. *Ensuring the Availability of Information Resources Through the Use of an Intelligent Communication Network with the Drift of Parameters of Autonomous Segments.* In: Proc. IT&I 2023 – Information Technology and Implementation, CEUR Workshop Proceedings, Vol. 3624 (2023), pp. 456–461.

[18] Ch. Khammassi, S. Krichen. *A NSGA2-LR wrapper approach for feature selection in network intrusion detection.* Computer Networks, 172 (2020), 107183. doi:10.1016/j.comnet.2020.107183.

[19] D. Hua et al. *A Multi-Stage NSGA-III Optimization Model for False Data Injection Attacks in Integrated Power-Hydrogen Cyber-Physical Systems.* IET Journal (peer-reviewed entry) 2025). doi:10.1049/rpg2.70022.