IMMC

February 2026

Summary Sheet

Team 2026005

Problem. Allocate limited resources to reduce anthropogenic harm in large national parks (poaching, snares, roadkill, disease introduction, wildfire), while ensuring that detections can be converted into timely interventions.

Core idea. Convert spatial inputs into a priority map on a 1 km grid, then use a staged budget portfolio and daily patrol routing to maximize expected protection under uncertainty.

Model outputs. (i) priority map of cells and aggregated big blocks, (ii) budgeted allocation (K, G, U) for rangers/GPS/acoustic sensors, (iii) daily patrol routes anchored to bases, (iv) budget–protection curve.

Key recommendations.

- Fund **minimum viable ranger response** first ($K_{\min}$), otherwise sensing does not reduce damage.
- Deploy GPS only if full coverage is affordable: improves accountability, blocks blind trips, supports speed enforcement to reduce roadkill.
- Allocate remaining budget between more teams vs. boundary acoustic sensors via Monte Carlo scenario evaluation.
- Wildfire: monitor small events; trigger early professional response once size/severity threshold is exceeded.

Data layers used. Wildlife distributions, water sources, important plants, roads/access, POIs/bases, existing monitoring, fire/no-fire mask.

Transferability. Same workflow applies to Altai and Wood Buffalo by replacing mobility constraints, seasonality, and local threat profiles.

Letter to the IMMC

Team 2026005
February 2026

Dear IMMC Committee,

We are pleased to share our proposal for a reasonable and sustainable approach to the protection of national parks. Our Graph-Based Decision Model provides a practical decision framework for Etosha National Park and tests how the same logic transfers to two very different parks (Altai Reserve and Wood Buffalo).

We started by converting the available spatial information (wildlife distribution, water availability, important plant areas, roads/access, and existing monitoring) into a daily priority picture, and then used it to (i) select a staged budget portfolio and (ii) generate daily patrol routes that are feasible within a shift and not fully predictable.

Our strategy.

1. **Minimum viable response first:** fund enough ranger teams so alerts can be acted upon within a daily shift. If staffing is below this threshold, better sensing mostly produces information without outcomes.

2. **GPS if and only if full coverage is affordable:** deploy GPS on *all* operational vehicles or not at all. Partial GPS creates predictable blind trips, which can be exploited; full coverage improves accountability, reduces corruption opportunities, and enables enforceable speed compliance to reduce roadkill.

3. **Then tune the remaining split:** after staffing minimum and the GPS decision, allocate the remaining budget between more teams and boundary acoustic monitoring using scenario-based evaluation, because the best mix depends on uncertainty like where threats occur, how quickly teams can respond, and how detection changes behavior.

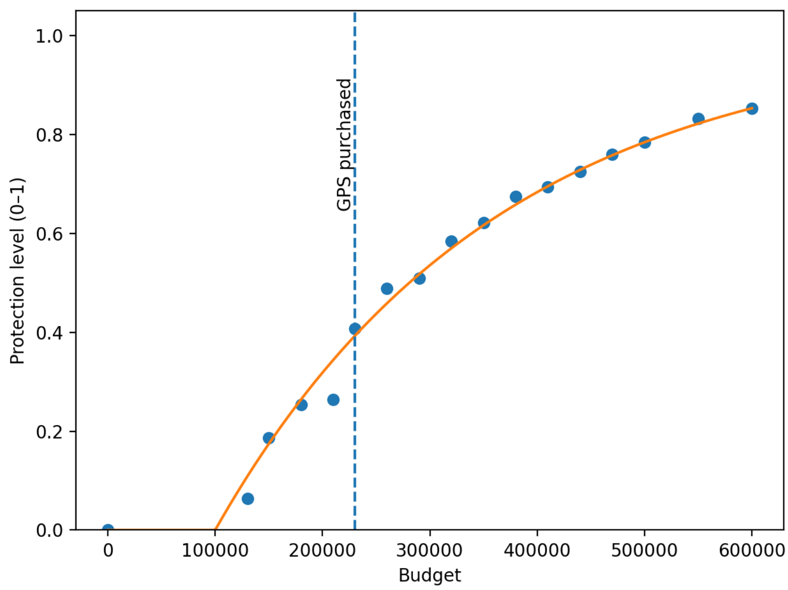


Figure 1: Budget versus protection improvement (illustrating diminishing marginal returns).

Figure 1 highlights a key planning insight: protection gains show **diminishing marginal returns**. This is why the first spending must remove the bottleneck (response capacity), then fund the hardest-to-bypass leverage point, and only then optimize the remaining trade-off (teams vs. boundary sensors).

How this changes field operations. Each day, patrol effort is concentrated in the highest-value areas identified by the priority picture, while routes include controlled variation to reduce predictability. When a detection arrives (vehicle anomalies from GPS, boundary acoustic alerts, or direct observation), the nearest feasible team is dispatched. Protection is tracked through operational outcomes such as time-to-intercept and expected prevented loss under uncertainty.

Threat-by-threat intervention logic (what we recommend in practice).

- **Poaching and illegal harvesting:** treat as deliberate, adaptive threats. Boundary acoustic monitoring helps detect incursions, but its coverage is local and can be avoided if sensor locations become known. Full vehicle GPS coverage is harder to bypass, covers the entire road network, reduces gate/road misuse, and supports accountability (including corruption risk reduction).
- **Snares and passive traps:** response time is decisive; routes must revisit high-risk corridors often enough to find and save trapped animals rather than only completing a shortest loop.
- **Roadkill:** reduce via enforceable speed limits backed by GPS-based compliance monitoring and meaningful penalties.
- **Fence gaps, wildlife dispersal, and disease introduction:** repair fence gaps where fencing is intended, reducing predictable crossing points used by both animals and humans; this also increases the effectiveness of boundary monitoring by shrinking feasible corridors. For disease introduction, use entrance screening and operational reporting/removal of severely diseased animals found during patrols.
- **Wildfire:** monitor via frequent satellite fire products; allow small low-severity fires to remain under observation, but once a fire exceeds a size/severity threshold or threatens priority habitats, trigger *early* professional response to avoid catastrophic spread.

What decision-makers get.

- A clear spending rule: **(1) response capacity, (2) full GPS if feasible, (3) tune the rest by scenarios.**
- Feasible daily patrol plans focused on high-value areas, with reduced predictability.
- Decision outputs that are directly operational: expected response times, expected prevented loss, and the budget–benefit curve.
- A transferable workflow: adapt by updating local layers (mobility constraints, boundary structure, assets, seasonality).

Limitations. Results depend on the quality of spatial inputs and simplified assumptions about detection and response. Some seasonal likelihoods are best interpreted as *relative* within a season rather than exact probabilities. Adversaries can adapt, so route variation and periodic re-planning are necessary. Finally, technology effectiveness depends on real operational pipelines (communication, dispatch discipline, vehicle availability).

Sincerely,
Team 2026005

Contents

1 Introduction

1.1 Background

National parks are crucial for preserving biodiversity and maintaining unique ecosystems. Despite ongoing efforts, the challenge of ensuring their protection remains unresolved. The hardest part is the proper allocation of limited human and technological resources to combat harmful risks, such as illegal entry, poaching (including snaring), disease spread, fires, and other mostly anthropogenic risks. Etosha National Park is an example of such a struggling reserve. Founded in 1907, it is one of Africa's largest national parks, one of the oldest, and Namibia's most popular tourist attraction. The reserve has repeatedly been targeted by poachers due to the presence of rare species, specifically black rhinos. Existing protection measures have not fully eliminated the threat of poaching.

The advancement of technology both improves safety and introduces new vulnerabilities alongside protective capabilities. There are many ways to reduce risk that were introduced throughout the years like drones, cameras, patrols, and even trained dogs. However, the developed system still fails to prevent some of the main dangers to species. The mere implementation of technology is not sufficient if the coordination is limited. As Joe Noci, Senior Advisor for the Drone Center of Excellence Innovation, explained in an interview about drone use in Namibia, "We have used UAVs extensively here in Namibia, in many applications and conditions, and there are many problems in application, as well as many successes." Despite these efforts already in the 2010s, poaching remains an ongoing challenge.

Therefore, a coordinated allocation strategy is required. In this study, Team 2026005 proposes a mathematically grounded optimization model for the effective allocation of personnel and technological resources to protect vulnerable species and their habitats, taking into account their features, landscape, and other critical details.

1.2 Problem Restatement

In this study, we define protection in measurable and operational terms, accounting for limitations in available personnel and technological resources. First, we identify key anthropogenic threats, prioritize endangered species, and analyze risk mitigation strategies. Second, we develop an algorithm that accounts for animal movement, spatially distributed risks, and their temporal dynamics. Third, based on the algorithm output, we construct optimized patrol strategies under multiple scenarios. Finally, we evaluate the adaptability of the model by applying it to two additional parks under differing environmental and threat conditions.

We define *protection* as minimizing the expected anthropogenic damage to populations of high-priority species through timely detection and strategic intervention.

Assumptions

Uniform within-area presence probability. Using spatial distribution maps to identify potential habitats, we assume that within each identified area the probability of species presence is uniform over time, with an increase in probability around waterholes during the dry season.

Justification. This assumption simplifies the spatial modeling process and is sufficient for patrol allocation, since any territory with a non-zero probability of species presence requires protection. In real-world applications, the assumption can be refined using ranger observations and updated field data, which are typically more detailed than information available in scientific articles.

2 Key Risks and Factors Identification

The factors were identified based on our assumptions and further research into security-related issues. Below we list the risks that have been confirmed as important to consider.

Road-related wildlife mortality. Collisions between wild animals and vehicles are one of the main causes of anthropogenic mortality in Etosha National Park and its surroundings. Documented cases include frequent collisions with nocturnal birds and mammals, a seasonal increase in collisions with birds during the summer migration period, and isolated incidents such as the death of 108 warthogs on the B1 highway (July 2012), a collision with a springbok (2012), and a rhino being hit by a Ministry of Environment and Tourism vehicle (2015). Although roadkill is usually local in nature, it contributes to the overall decline in wildlife populations and represents a constant anthropogenic pressure on animal populations.

Illegal harvesting of resources. Illegal harvesting of biological resources creates environmental and management problems in Etosha. Overharvesting can lead to ecosystem degradation, which in turn can affect the quality of the habitat of dependent species. Confiscations and arrests reported in 2024-2025 indicate continued incursions into the park for the purpose of illegal harvesting. Effective monitoring is crucial to prevent resource depletion that can undermine biodiversity conservation.

Wildlife dispersal beyond park boundaries. The movement of large predators and other wild animals outside the park poses risks to the animals themselves and to residents of nearby settlements. Among recent incidents, a pride of six lions left Etosha and killed livestock between Casablanca and Oshivel. In the past (1970s–1980s), several lions were shot on surrounding farms after dispersing due to water shortages. The construction of artificial watering holes has prevented such dispersals from recurring, but renewed dispersal due to alternative environmental factors, such as lack of food, may again cause animals to leave the park and risk being killed.

Wildfire. Wildfires represent a large-scale disturbance factor in Etosha, with both natural (lightning-induced) and anthropogenic ignition sources. Historical records (1970–1979) indicate 30 confirmed lightning-caused fires, 11 probable lightning events, and 15 fires originating from artificial or external sources. Fire risk is seasonally concentrated, with peak occurrence in September during the late dry season. Major recent events include a 2,545 km^2 fire in October 2020 and large-scale fires in 2021 and 2025 affecting approximately 30–48% of the park area in some years. Documented wildlife mortality includes rhinoceroses, giraffes, elephants, and lions. Given their spatial scale and potential impact on priority species and habitats, wildfires constitute a high-impact ecological risk requiring structured monitoring and response.

Transboundary disease transmission. The transmission of pathogens from surrounding livestock or human settlements to wild animal populations poses an epidemiological risk at park boundaries. Documented cases include outbreaks of rabies in kudu (1977) originating in neighboring agricultural areas, subsequent spread to lions (1983–1984), and confirmed cases near the eastern border areas. Of the 6,190 wildlife deaths recorded between 1975 and 1990, 115 (approximately 2%) were associated with rabies, most of which occurred near human infrastructure. Additional cases include rabies mortality among reintroduced wild dogs (1995) and Rift Valley fever in springbok (2011) during a regional outbreak among domestic livestock.

In this paper, **TD** is modeled specifically as *border-driven spillover risk* (external introduction near park boundaries), not as internal endemic disease dynamics.

Organized poaching and illegal wildlife trade. Poaching in Etosha involves both random and organized criminal activities, facilitated by breaches in fencing and insider information. Documented methods include driving cattle through gaps in the fence to cover tracks, night hunting near watering holes, using stolen ammunition, and hiding rhino horns for later extraction. Reports from western Esotica indicate leaks of information about rhino locations, resulting in the discovery of several carcasses. Additional tactics include diversionary wire traps and smuggling networks using concealed transport and forged documents. These patterns demonstrate adaptive strategies and justify modeling poaching as a structured, deliberate threat.

Wire snaring and passive trapping. Illegal wire snares are a widespread and inexpensive method of hunting wild animals in Etosha. In 2023, 62 active snares were seized in three days in the vicinity of Okaukueho, and there were also cases of serious injuries requiring euthanasia (e.g., a springbok that got its neck caught in a snare). Snares are mainly used to catch medium-sized ungulates, but they can also injure other species.

Rejected (Low-Priority) Factors

We also considered several factors that were ultimately rejected due to their insignificant observed impact on wildlife and the state of the ecosystem in Etosha. These risks were excluded because we found no incidents that would confirm a negative impact, indicating that the park administration is successfully managing them. However, when scaling the model to other reserves, the set of factors should be reassessed, as threat profiles may vary.

Internal endemic disease outbreaks. **Anthrax.** Anthrax outbreaks are endemic in Etosha and are seasonal in nature (for example, the peak incidence among zebras occurs in March-April, and among elephants in November). Control measures are limited to targeted preventive vaccination of rare species (e.g., black rhinos, roan antelopes), because large-scale carcass burning or water disinfection are logistically impossible across the park's vast territory. Anthrax is considered a conditionally manageable ecological process rather than a constantly modeled risk for intervention.

Rabies. Outbreaks of rabies within the park account for a relatively small proportion ($\sim$2%) of documented wildlife mortality (1975–1990). Once symptoms appear, rabies is incurable. Control measures include culling of suspect animals, laboratory confirmation, and carcass disposal. Given its relatively limited proportional impact, rabies is not considered a major factor in the baseline distribution model.

We consider the measures taken to be sufficient.

Solid waste and plastic contamination. Improper waste disposal can pose a risk of ingestion or entanglement for wildlife; however, no significant wildlife mortality has been identified in Etosha. National regulations prohibiting the use of plastic bags in protected areas (since 2018) and enforcement mechanisms (fines and penalties) indicate the effectiveness of risk reduction measures, resulting in this factor being given low priority in the current model.

Unsafe tourist proximity to wildlife (without harmful intentions). Isolated incidents involving tourists leaving vehicles or approaching wild animals (e.g., rhinos) have been documented and publicly condemned by park management. Although such events pose a safety risk, no confirmed environmental impact on the population has been recorded, and enforcement measures appear adequate.

Tourist-induced ignition events. During the period under review, there were no documented cases of forest fires in Eto'o directly linked to tourist behavior. Consequently, this factor is not considered a distinct modeled risk.

2.1 Risk coefficient and threat prioritization.

To integrate identified threats into the model, we prioritize them using a composite spatiotemporal coefficient

$$\mathrm{RiskCoeff}_i(g) = W_i^{\mathrm{impact}} \cdot w_i(g), \tag{1}$$

where W_i^{impact} is the intrinsic (structural) importance of threat i independent of seasonality, and $w_i(g)$ is the regime-specific (e.g., monthly) likelihood or intensity of threat i under regime g. This separation preserves stable conservation priorities (impact) while allowing operational priorities to shift with seasonality (likelihood).

Criteria for intrinsic impact. To quantify the importance of threats, we evaluate each threat using five criteria:
(i) **Spatial extent** — the fraction of the park's usable habitat that can be affected;
(ii) **Severity** — the magnitude of harm to priority species populations and the land of the park we aim to protect;
(iii) **Reversibility** — the expected recovery time and the degree to which damage is irreversible;
(iv) **Recurrence** — the persistence of impact through repeated or continuous exposure;
(v) **Response window** — the time available for intervention before most damage becomes unavoidable.
For all criteria, higher scores correspond to a higher impact (worse outcomes).

Deriving a qualitative hierarchy. To avoid bias toward any particular threat criteria, we construct the hierarchy by reasoning about the consequences of each criterion at its maximum level, comparing them independently. To prevent inadvertently favoring any criterion, we consider a low-impact baseline scenario (e.g., human-generated pollution). If the **spatial extent** becomes large enough to affect the entire habitat, animals may be forced to relocate rapidly, leaving the protected area and facing substantially higher mortality risk due to human activity outside its boundaries. Thus, rapid habitat loss outweighs other consequences, establishing **spatial extent** as the highest-priority criterion. Conditional on similar extent, **Severity** puts lives of animals on risk, forcing them to leave the land, however, the process may be long-term, which settles the criterion on the second place.places animals' lives at risk and may force them to leave the area. However, this process can be long-term, placing the criterion second. At its maximum, **reversibility** has the strongest but long-term implications, making it third. **Recurrence** captures cumulative damage from repeated exposure over time. Finally, the **response window** influences the long-term consequences of delayed or absent intervention. This yields the qualitative hierarchy:

$$\text{Spatial extent} > \text{Severity} > \text{Reversibility} > \text{Recurrence} > \text{Response window}. \tag{2}$$

AHP conversion to quantitative weights. The Analytic Hierarchy Process (AHP) will be used to convert qualitative ordering into quantitative. We construct a reciprocal pairwise-comparison matrix $A^{(C)} = [a_{pq}]$, where a_{pq} represents how much more important criterion p is than criterion q based on previously determined importance. To map rank differences to the standard Saaty scale, we apply a deterministic rule: if two criteria are adjacent in the ordering, we set the comparison to 3; a gap of two positions maps to 5; a gap of three maps to 7; and a gap of four maps to 9. If $r(p)$ is the position of criterion p in the ordering (1 = most important), define $\Delta = r(q) - r(p)$ and set

$$a_{pq} = \begin{cases} 1, & \Delta = 0, \\ 3, & \Delta = 1, \\ 5, & \Delta = 2, \\ 7, & \Delta = 3, \\ 9, & \Delta \geq 4, \end{cases} \qquad a_{qp} = \frac{1}{a_{pq}}, \quad a_{pp} = 1. \tag{3}$$

For the five criteria, this produces the reciprocal matrix:

$$A^{(C)} = \begin{pmatrix} 1 & 3 & 5 & 7 & 9 \\ \frac{1}{3} & 1 & 3 & 5 & 7 \\ \frac{1}{5} & \frac{1}{3} & 1 & 3 & 5 \\ \frac{1}{7} & \frac{1}{5} & \frac{1}{3} & 1 & 3 \\ \frac{1}{9} & \frac{1}{7} & \frac{1}{5} & \frac{1}{3} & 1 \end{pmatrix}. \tag{4}$$

Criterion weights are computed using the normalized-column method, row-averaging, and renormalization. The resulting weights are:

$$\begin{aligned} w_{\text{Spatial extent}} &= 0.5028, \quad & w_{\text{Severity}} &= 0.2602, \\ w_{\text{Reversibility}} &= 0.1344, \quad & w_{\text{Recurrence}} &= 0.0678, \\ w_{\text{Response window}} &= 0.0348. \end{aligned} \tag{5}$$

We verified internal consistency of the criteria matrix using the AHP Consistency Ratio (CR). For the 5×5 criteria matrix we obtained $CR \approx 0.053$, below the standard acceptability threshold of 0.10.

Threat scoring rubric. To obtain threat-specific impact weights, each threat i is assigned ordinal ratings $s_{i,c} \in \{1, \dots, 5\}$ under each criterion c using an anchored rubric to ensure consistent scoring across threats.
Spatial extent (fraction of usable habitat potentially affected): 1 ($< 1\%$), 2 (1–5%), 3 (5–15%), 4 (15–30%), 5 ($> 30\%$).
Reversibility (recovery time): 1 (days–weeks), 2 (up to one season), 3 (1–3 years), 4 (3–10 years), 5 ($>$ 10 years or effectively irreversible).
Recurrence (persistence): 1 (rare/one-off), 2 (episodic), 3 (regular seasonal/annual), 4 (multiple times per year or prolonged periods), 5 (near-continuous).
Response window (time-to-act): 1 (days–weeks), 2 (several days), 3 (1–2 days), 4 (hours), 5 (minutes or effectively no actionable window).
Severity (potential population-level harm): 1 (minimal, no detectable population effect), 2 (low, localized harm), 3 (moderate, affects local groups), 4 (high, can cause local declines), 5 (critical, can cause major declines or mass mortality). Ratings are normalized to $\hat{s}_{i,c} = s_{i,c}/5 \in [0,1]$ and

Table 1: Threat-by-criterion ratings (ordinal scale 1–5). Criteria: spatial extent E, severity S, reversibility (recovery burden) R, sustained pressure L, and response window urgency T. Threat codes: RM (road mortality), WF (wildfire), OP (organized poaching), SN (snaring), IH (illegal harvesting), DB (dispersal beyond boundaries), TD (transboundary disease).

Threat	**E**	**S**	**R**	**L**	**T**
RM (Road-related wildlife mortality)	2	3	2	3	3
WF (Wildfire)	5	4	5	3	5
OP (Organized poaching & illegal wildlife trade)	3	5	5	3	5
SN (Wire snaring & passive trapping)	2	4	4	4	4
IH (Illegal harvesting of resources)	3	2	3	4	2
DB (Wildlife dispersal beyond park boundaries)	4	3	3	3	2
TD (Transboundary disease transmission)	4	4	4	3	3

aggregated as

$$W_i^{\text{impact}} = \sum_c w_c \cdot \hat{s}_{i,c}, \tag{6}$$

followed by normalization across threats so that $\sum_i W_i^{\text{impact}} = 1$. Using the threat-by-criterion ratings (RM road mortality, WF wildfire, OP organized poaching, SN snaring, IH illegal harvesting, DB dispersal beyond boundaries, TD transboundary disease), the final normalized impact weights are:

$$\begin{aligned} W_{\mathrm{WF}}^{\mathrm{impact}} &= 0.192176, \\ W_{\mathrm{TD}}^{\mathrm{impact}} &= 0.162674, \\ W_{\mathrm{OP}}^{\mathrm{impact}} &= 0.161063, \\ W_{\mathrm{DB}}^{\mathrm{impact}} &= 0.144751, \\ W_{\mathrm{SN}}^{\mathrm{impact}} &= 0.124982, \\ W_{\mathrm{IH}}^{\mathrm{impact}} &= 0.115731, \\ W_{\mathrm{RM}}^{\mathrm{impact}} &= 0.098622. \end{aligned} \tag{7}$$

Sensitivity checks. In order to assess robustness, we conduct two sensitivity checks and report the stability of both the threat ranking and the induced operational priorities. For each perturbation experiment, and under representative regimes g, we recompute the full vector W^{impact}, the ordering of threats, and the resulting composite coefficients $\mathrm{RiskCoeff}_i(g)$. We then quantitatively assess the stability of the ranking by calculating Spearman's ρ and Kendall's τ rank correlations between the baseline and modified rankings. Prioritization is considered reliable when the correlations remain high, for example, $\rho \geq 0.9$, and when the top k threats, with $k = 3$, remain unchanged or differ only by neighboring permutations.

For the test with weight perturbation ($\pm 20\%$ at a time), each criterion weight w_c is taken at $\pm 20\%$, the weight vector is renormalized so that its sum equals one, and W^{impact} and the resulting rating are recalculated. The maximum absolute change in the weight of any threat is also tracked,

$$\Delta_{\max} = \max_i \left| W_{i,\mathrm{pert}}^{\mathrm{impact}} - W_{i,\mathrm{base}}^{\mathrm{impact}} \right|, \tag{8}$$

and the minimum baseline deviation between neighboring threats is reported, since significant changes in the rating are only expected if two threats have nearly equal baseline scores.

For the ordinal-score perturbation test (± 1 on $s_{i,c}$), we perturb each ordinal rating by ± 1 (clipped to $[1, 5]$) and recompute W^{impact}. Because the aggregation is linear, changing a single score affects W_i^{impact} by at most $w_c/5$ per criterion (before normalization), providing a conservative bound on sensitivity to scoring noise. In addition, we perform a stress-test in which all scores for a single threat are simultaneously shifted by ± 1 (clipped to $[1, 5]$) to emulate coherent expert bias; the ranking is considered robust if the top-k set remains invariant under this stronger perturbation. Since $\mathrm{RiskCoeff}_i(g) = W_i^{\mathrm{impact}} w_i(g)$ couples intrinsic impact to regime-specific likelihoods, we repeat the sensitivity checks under at least two contrasting regimes (e.g., late dry season vs. wet season) to ensure that robustness conclusions are not artifacts of a single likelihood profile.

Regime-based monthly likelihood model. We model monthly variation in threat likelihood using a regime-based approach driven by two dominant seasonal drivers: (i) climatic seasonality (dry vs. wet months) and (ii) visitor pressure (popular vs. non-popular months). Each month is assigned to exactly one regime:

- G_1 (Dry + popular): July–September,
- G_2 (Wet + popular): October–November,
- G_3 (Wet): December–February,
- G_4 (Dry): March–June.

To keep the likelihood module tractable and focused on threats that directly affect patrol and monitoring decisions, we restrict the analysis to seven threats: road-related wildlife mortality (RM), wildfire (WF), organized poaching and illegal wildlife trade (OP), wire snaring and passive trapping (SN), illegal harvesting of resources (IH), wildlife dispersal beyond park boundaries (DB), and transboundary disease transmission (TD). For each regime g, we estimate a regime-specific vector of relative likelihood weights $w^{(g)} = (w_i^{(g)})_{i=1}^n$ over these threats, where $n = 7$ and

$$\sum_{i=1}^{n} w_i^{(g)} = 1. \tag{9}$$

Month-level weights are mapped via the regime assignment function $g(m)$:

$$w_i(m) = w_i^{(g(m))}. \tag{10}$$

From expert rankings to quantitative likelihood weights (AHP mapping). Because incident-level data are incomplete and not consistently reported across threat types, we convert structured expert rankings into quantitative likelihood coefficients using a deterministic AHP-style pairwise comparison procedure. For each regime g, we first define an ordered list of threats from most likely to least likely. Let $r_i^{(g)} \in \{1, \dots, n\}$ denote the rank position of threat i in regime g (rank 1 is most likely). We then construct a reciprocal pairwise comparison matrix $A^{(g)} = [a_{ij}^{(g)}] \in \mathbb{R}^{n\times n}$, where each entry represents how much more likely threat i is than threat j in regime g.

To deterministically translate rank gaps into Saaty-scale comparisons, define

$$\Delta = r_j^{(g)} - r_i^{(g)}, \tag{11}$$

and set

$$a_{ij}^{(g)} = \begin{cases} 1, & \Delta = 0, \\ 3, & \Delta = 1, \\ 5, & \Delta = 2, \\ 7, & \Delta = 3, \\ 9, & \Delta \geq 4, \end{cases} \qquad a_{ji}^{(g)} = \frac{1}{a_{ij}^{(g)}}, \quad a_{ii}^{(g)} = 1. \tag{12}$$

This yields a consistent, auditable mapping from an expert ordering to a standard AHP comparison structure. From each matrix $A^{(g)}$, we compute the likelihood weight vector $w^{(g)}$ using the normalized-column method: each column is normalized by its sum, and the weight for each threat is the average of its normalized row; the resulting vector is normalized to sum to one. These weights represent *relative* likelihood within a regime rather than absolute event probabilities. Uncertainty is handled via sensitivity analysis by perturbing (i) rank positions $r_i^{(g)}$ or (ii) the rank-to-scale mapping above, and re-evaluating downstream patrol recommendations.

Regime-specific likelihood rankings (inputs). The regime-specific likelihood rankings used as inputs are:

- G_1 (Dry + popular; July–September):
  RM > WF > OP > SN > IH > DB > TD.
  Peak visitor pressure increases road exposure and road–wildlife encounters, while dry conditions increase wildfire likelihood and spread potential. Dry-season constraints can also increase feasibility of illegal wildlife targeting (OP, SN), whereas cross-boundary disease transmission is relatively less prominent than in wet months.

- G_2 (Wet + popular; October–November):
  RM > SN > OP > IH > WF > DB > TD.
  Traffic remains high, keeping RM dominant; SN and OP remain persistent under changing ecological conditions. Wildfire likelihood decreases relative to late dry season, and wet-season movement/disease dynamics are not yet dominant compared to tourism-driven exposure.

- G_3 (Wet; December–February):
  TD > DB > IH > SN > OP > WF > RM.
  Wet conditions support transmission pathways at the wildlife–livestock interface and increase cross-boundary movement as water and forage become more widely available. Tourism intensity is lower, reducing road exposure, and fuel flammability is reduced, lowering wildfire likelihood.

- G_4 (Dry; March–June):
  WF > OP > SN > DB > IH > RM > TD.
  This regime is dry but with lower visitor pressure than peak months, shifting dominance to climate- and ecology-driven threats: increasing dryness elevates wildfire likelihood, while wildlife becomes progressively more constrained by water availability, increasing predictability and the feasibility of poaching and passive trapping. Road mortality remains lower than in popular months, and transboundary disease transmission is less facilitated than in wet months.

Numerical regime-specific likelihood weights. Applying the deterministic AHP-style mapping from rank gaps to the Saaty scale and computing weights via the normalized-column + row-average method yields the following position weights (for ranks 1 to 7):

$$\begin{aligned} &w_{\text{rank 1}} = 0.4092, \quad w_{\text{rank 2}} = 0.2460, \quad w_{\text{rank 3}} = 0.1512, \\ &w_{\text{rank 4}} = 0.0903, \quad w_{\text{rank 5}} = 0.0525, \quad w_{\text{rank 6}} = 0.0309, \\ &w_{\text{rank 7}} = 0.0198. \end{aligned} \tag{13}$$

For each regime, these values are assigned to threats according to the expert likelihood ranking, producing $w^{(g)}$ such that $\sum_i w_i^{(g)} = 1$.

Regime g	RM	WF	OP	SN	IH	DB	TD
G_1 (Dry+popular)	0.4092	0.2460	0.1512	0.0903	0.0525	0.0309	0.0198
G_2 (Wet+popular)	0.4092	0.0525	0.1512	0.2460	0.0903	0.0309	0.0198
G_3 (Wet)	0.0198	0.0309	0.0525	0.0903	0.1512	0.2460	0.4092
G_4 (Dry)	0.0309	0.4092	0.2460	0.1512	0.0525	0.0903	0.0198

Table 2: Regime-specific relative likelihood weights $w^{(g)}$ over the seven modeled threats (sum to 1 in each regime).

3 Graph-Based Decision Model for Resource Allocation and Patrol Routing

3.1 Spatial Discretization and Mobility Graph

We discretize Etosha National Park into a regular grid of $1\,\text{km} \times 1\,\text{km}$ cells. Each cell is a node in a weighted mobility graph $G = (V, E)$, and edges represent feasible moves between neighboring cells.

All layers are processed in a metric CRS (UTM) so that distances and areas are computed in meters. For each cell we store node features used downstream, including: median elevation (DEM), road coverage (total road length), POI counts by type, coverage indicators for animal-distribution polygons and plant polygons, and a fire/no-fire mask. Edges are constructed using an 8-neighborhood (orthogonal + diagonal adjacency).

Each edge (i, j) is weighted by travel time using a road/off-road speed model. We label an edge as *on-road* if the segment between cell centers intersects the road layer; otherwise it is *off-road*. The edge speed is

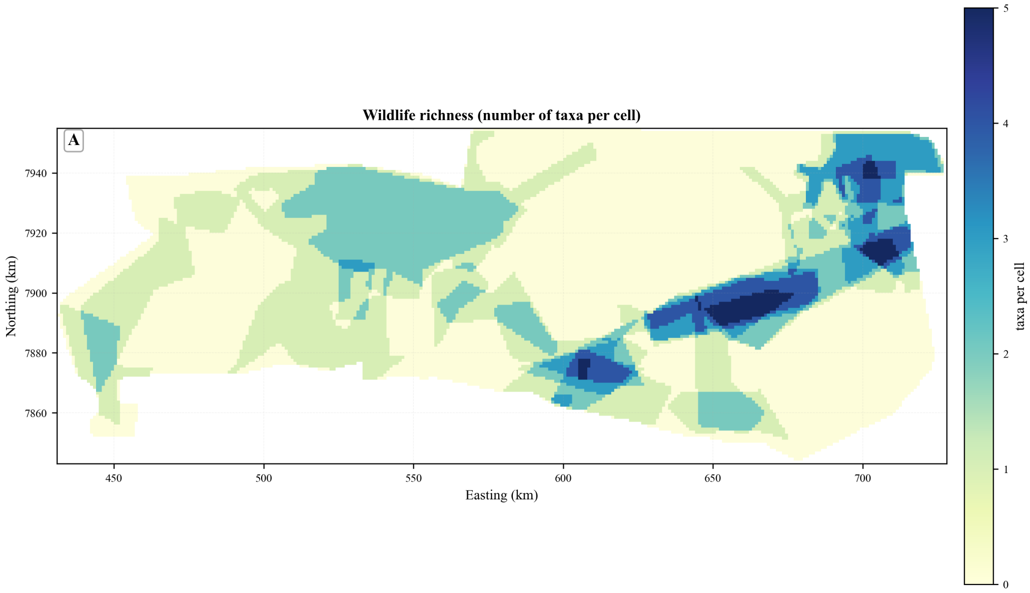
$$v(i,j) = \begin{cases} 60 \text{ km/h}, & \text{on-road}, \\ 30 \text{ km/h}, & \text{off-road}. \end{cases}$$

With

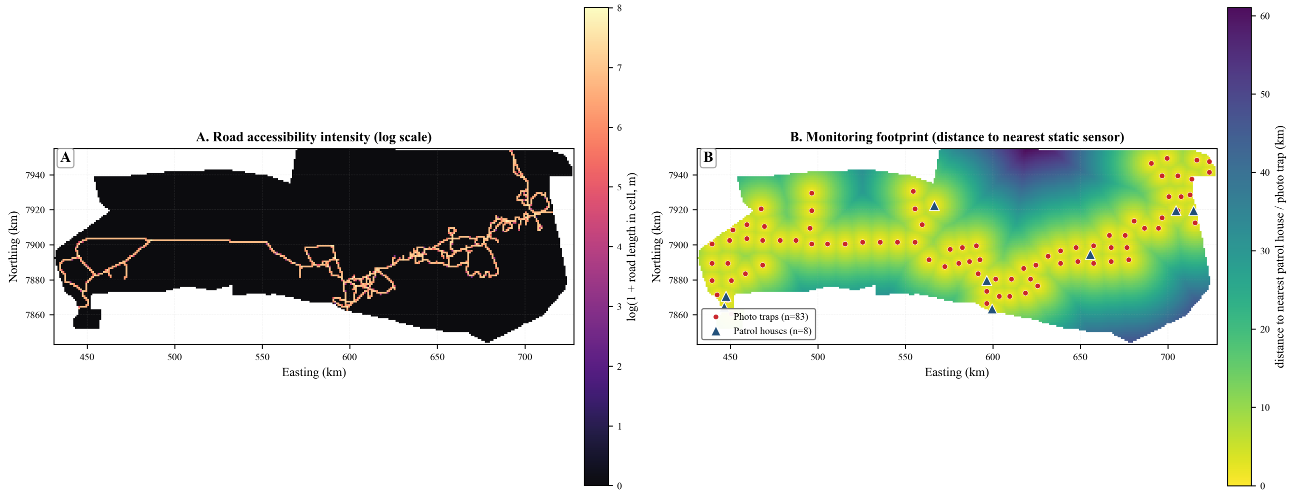
$$\text{distance}_m(i,j) = \begin{cases} 1000, & \text{orthogonal neighbors}, \\ 1000\sqrt{2}, & \text{diagonal neighbors}, \end{cases}$$

the travel time is

$$T(i,j) = \frac{\text{distance}_m(i,j)}{v(i,j) \cdot 1000}.$$



(a) Spatial Distribution of Wildlife Richness (Taxa per Cell).



(b) Infrastructure and Monitoring Footprint.

Figure 2: Core spatial layers used by the priority map and patrol planning.

The resulting graph is stored as `etosha_grid_graph.json` and is used consistently for (i) priority-map computation, (ii) shortest-path travel times (Dijkstra), and (iii) daily patrol route construction.

Key spatial inputs. Figures 2a–2b provide a compact quality check for preprocessing and summarize the two most important spatial inputs for our patrol model: where ecological value is concentrated (wildlife richness) and how accessible/monitored the park is (infrastructure footprint). Additional maps (fire status and important plant coverage) are moved to Appendix.

3.2 Priority Map Construction

For each cell (node) $i \in V$ we compute a patrol priority P_i by combining an intrinsic score S_i with two graph-context layers defined on the mobility graph (Section 3.1).

Intrinsic score. Let $\mathcal{T}_i$ be the set of wildlife taxa whose distribution polygons intersect cell i. Let $a_t \geq 0$ denote the taxon-specific importance weight (unlisted taxa have $a_t = 0$). Define:

$$\begin{aligned} A_i &= \sum_{t \in \mathcal{T}_i} a_t && \text{(wildlife presence score),} \\ H_i &= \mathbb{1}\{\text{a water source is present in } i\} && \text{(water indicator),} \\ B_i &= \mathbb{1}\{\text{an important plant area overlaps } i\} && \text{(plant indicator),} \\ M_i &= \mathbb{1}\{\text{a fixed monitoring asset is present in } i\} && \text{(coverage indicator).} \end{aligned} \tag{14}$$

The intrinsic score is

$$S_i = \big(A_i + 0.5H_i + 0.5B_i\big)\big(1 - 0.5M_i\big), \tag{15}$$

where the multiplicative term penalizes already covered cells to favor marginal coverage expansion.

Two graph-context layers. Let $\mathcal{N}(i)$ be the set of direct neighbors of i (8-neighborhood). Let $\tau(i, j)$ be the shortest-path travel time from i to j on the same edge-time weights (Dijkstra). Define the forward neighborhood $\mathcal{F}(i)$ as the set of the $K = 1000$ nodes with smallest $\tau(i, j)$, excluding i and $\mathcal{N}(i)$. The layer means are

$$\overline{S}_{\mathcal{N}}(i) = \operatorname{avg}\{S_j : j \in \mathcal{N}(i)\}, \qquad \overline{S}_{\mathcal{F}}(i) = \operatorname{avg}\{S_j : j \in \mathcal{F}(i)\}. \tag{16}$$

Exponential mixing. We use raw weights $\tilde{w}_h = e^{-\alpha h}$ for depths $h \in \{0, 1, 2\}$ with $\alpha = 1$, normalize over non-empty layers for node i to obtain (w_0, w_1, w_2), and set

$$P_i = w_0 S_i + w_1 \overline{S}_{\mathcal{N}}(i) + w_2 \overline{S}_{\mathcal{F}}(i). \tag{17}$$

Priority map. Figure 3 visualizes the resulting spatial distribution of $\{P_i\}$ over the $1\,\text{km} \times 1\,\text{km}$ grid.

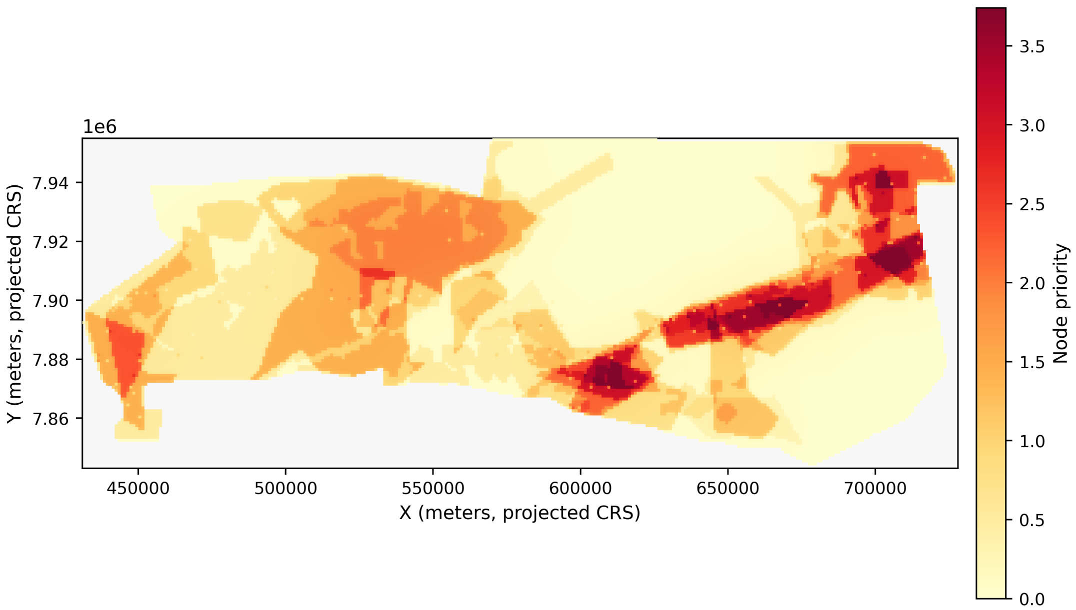


Figure 3: Spatial Distribution of Node Priorities.

3.3 Budgeted Resource Allocation

We allocate a fixed protection budget B across three intervention channels: ranger teams, vehicle GPS tracking, and boundary acoustic monitoring. Allocation follows a staged policy: (i) minimum viable response capacity, (ii) GPS if and only if full coverage is affordable, and (iii) simulation-based tuning of the remaining budget.

Intervention priorities. The staged order reflects two operational principles. First, *detection only matters if it can be converted into timely intervention*: without enough ranger capacity, additional sensing increases alarms but does not reduce realized damage. Second, we prioritize controls that are *harder to bypass* and cover multiple risks simultaneously. Full vehicle GPS coverage provides park-wide accountability for road movements (deterring vehicle-enabled poaching and insider-assisted gate abuse) and also enables enforceable speed control to mitigate roadkill. In contrast, boundary acoustic monitoring improves early detection only on instrumented boundary segments and can be partially bypassed by choosing alternative entry points. Therefore we (a) Secure minimum response capacity first. (b) Deploy GPS only if full coverage is affordable (all-or-nothing): install OBD-II GPS trackers at park entry gates for every operational vehicle; use traces for enforceable speed limits, geofence-based off-road detection, and tamper/unplug alerts. To reduce corruption risk at gates, add a fixed AI-assisted camera check that logs vehicle and tracker ID and does not allow entry until installation is visually verified. (c) Use acoustic sensors as a complementary layer whose scale is tuned jointly with additional rangers via simulation.

Resource 1: ranger teams (minimum viable response). Let c_K be the cost of one ranger team and let K be the number of teams. We first enforce a minimum operational capacity:

$$K \leftarrow K_{\min}, \qquad B_{\text{rem}} \leftarrow B - c_K K_{\min}.$$

The threshold $K_{\min}$ is chosen so that detections can be converted into interventions: if $K < K_{\min}$, even perfect sensing (GPS or acoustic) does not reduce damage because teams cannot react within the daily time budget.

Resource 2: GPS tracking (all-or-nothing). Let G_{full} be the number of vehicles requiring GPS and let c_G be the unit cost. We enforce full coverage or none:

$$G = \begin{cases} G_{\text{full}}, & \text{if } B_{\text{rem}} \geq c_G\, G_{\text{full}}, \\ 0, & \text{otherwise}, \end{cases} \qquad B_{\text{rem}} \leftarrow B_{\text{rem}} - c_G\, G.$$

Partial coverage yields low deterrence because violators can always choose an untracked vehicle/trip: if only a fraction $p = G/G_{\text{full}}$ of trips are tracked, then the probability of avoiding GPS audit is $1-p$, so the expected deterrence benefit scales as p and leaves a large predictable loophole for small p. Therefore we deploy GPS only when $p = 1$ (full coverage), otherwise we treat GPS as ineffective and set $G = 0$.

Resource 3: boundary acoustic monitoring (sound detectors). Let U be the number of acoustic sensors and c_U be the unit cost. Sensors are installed along boundary segments to maximize early detection of illegal entry and incursions. Candidate placements are generated on the boundary/fence line and evaluated by expected detection gain. For any chosen U, placements are obtained by a greedy boundary-coverage procedure.

Simulation-based tuning of the remaining budget. After enforcing $K_{\min}$ and deciding G, we allocate the remaining budget between additional ranger teams and acoustic sensors via Monte Carlo evaluation. For each candidate pair (K, U) satisfying

$$c_K K + c_U U \leq B - c_G G, \qquad K \geq K_{\min},$$

we (i) construct daily patrol routes for K teams (Section 3.5); (ii) simulate stochastic threat and detection scenarios (Monte Carlo); and (iii) compute an outcome metric (expected protected value / expected damage reduction). We then choose

$$(K^\star, U^\star) = \arg\max_{K,U} \ \mathbb{E}\big[\text{ProtectionValue}(K, G, U)\big] \quad \text{s.t. } c_K K + c_U U \leq B - c_G G, \quad K \geq K_{\min}.$$

This yields a budget-feasible portfolio where response exists by construction, GPS is deployed only if fully feasible, and the ranger–sensor trade-off is optimized under uncertainty.

3.4 Big-Cell Aggregation Graph

To make allocation and routing scalable, we aggregate the 1 km × 1 km grid into square *big cells*. Let $q \in \mathbb{Z}_{\geq 1}$ denote the aggregation factor (number of small cells per side), i.e., one big cell corresponds to a $q \times q$ window of small cells.

Big-cell indexing. For a small cell with indices (r, c), its big-cell index is

$$(R, C) = \big(\lfloor r/q \rfloor,\ \lfloor c/q \rfloor\big),$$

and we denote the corresponding big cell by $b_{R,C}$. The big-grid dimensions are

$$n_{\text{big\_rows}} = \left\lceil \frac{n_{\text{rows}}}{q} \right\rceil, \qquad n_{\text{big\_cols}} = \left\lceil \frac{n_{\text{cols}}}{q} \right\rceil,$$

where $n_{\text{rows}} \times n_{\text{cols}}$ is the small-grid size.

Big-node features and edges. For each big cell b, we store its member set of small cells and the aggregated priority

$$S_b = \sum_{i \in b} P_i,$$

as well as its metric geometry (bounding box and centroid). Big cells are connected by 8-neighborhood adjacency (orthogonal and diagonal neighbors), with edge lengths derived from centroid distances.

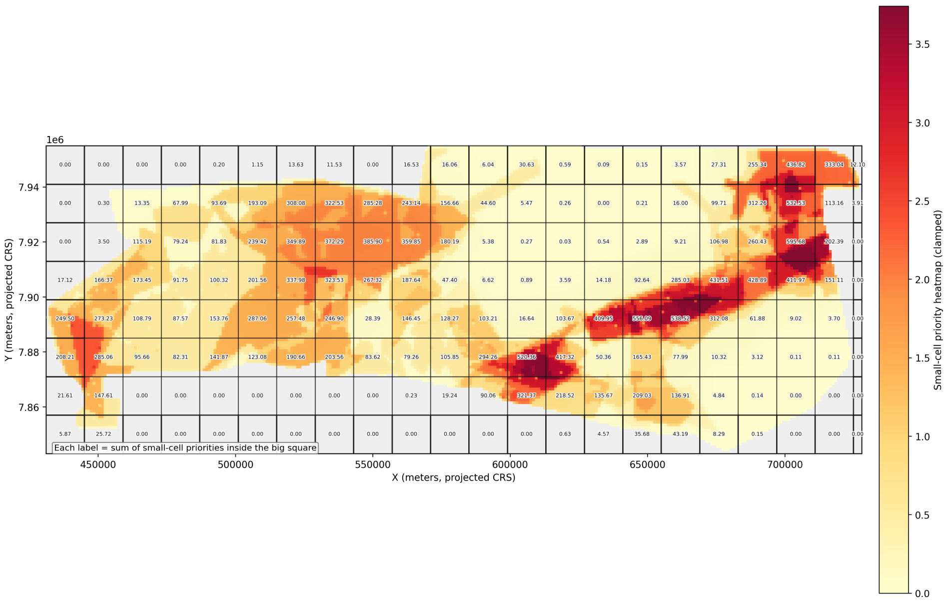


Figure 4: Aggregated Priority Map Across Large Spatial Blocks.

Remark. This step is a geometric aggregation of the grid. More accurate inter-big travel costs via boundary portals are computed in a separate preprocessing step and used by the routing algorithm.

3.5 Daily Patrol Routing Heuristic

Given a fixed allocation decision (Section 3.3)—number of teams K, chosen bases (patrol posts), and the precomputed big-cell travel-time model—we construct one daily route for each ranger team. Each route is a *round trip* that starts and ends at the same base (no overnight relocation).

Spatial level used for routing. Routing is performed on the *big-cell* graph to keep the combinatorics tractable. Each big cell b aggregates the small-grid priorities:

$$S_b = \sum_{i \in b} P_i,$$

and big-cell moves include 8-neighborhood transitions (diagonals allowed). Travel times between big cells are taken from the portal-based shortest-path model (Section 3.3); they already incorporate movement on the $1\,\text{km} \times 1\,\text{km}$ small grid with the constant speed assumption (40 km/h).

Inputs for the routing step. Let $\mathcal{B}$ be the set of available bases (patrol houses). Each base $p \in \mathcal{B}$ is mapped to its nearest big cell $b(p)$ by minimal travel time. For a chosen number of teams K, we distribute teams across bases by an integer vector $m = (m_p)_{p \in \mathcal{B}}$ such that

$$m_p \in \mathbb{Z}_{\geq 0}, \qquad \sum_{p \in \mathcal{B}} m_p = K.$$

This distribution specifies how many independent patrol routes are anchored at each base. The set of candidate big cells to cover is

$$\mathcal{C} = \{b : \; S_b > 0\} \setminus \{b(p) : p \in \mathcal{B}\},$$

i.e., positive-priority cells excluding base cells.

Route representation and time budget. Each patrol route is represented as an ordered sequence of big cells

$$\pi = (b_0, b_1, \ldots, b_L, b_0),$$

where b_0 is the base cell, and the route must satisfy a daily time budget H (hours). Let $T(u, v)$ be the precomputed travel time between big cells u and v. Optionally, we include a within-cell visit overhead $T_{\text{in}}(x)$ (time to traverse/inspect inside a big cell, derived consistently from the portal model). The total route time is

$$\text{Time}(\pi) = \sum_{\ell=0}^{L} T(b_\ell, b_{\ell+1}) \; + \; \sum_{\ell=1}^{L} T_{\text{in}}(b_\ell) \;\; \leq \;\; H.$$

The collected priority is computed with *unique assignment*: each candidate big cell contributes its priority at most once across all patrols (no double counting).

Greedy unique-insertion heuristic. For a fixed base-allocation vector m, we build routes by iterative insertions.

- **Initialization.** For every team, initialize an empty round trip: $\pi \leftarrow (b_0, b_0)$.
- **Marginal insertion metrics.** Consider inserting a candidate cell $x \in \mathcal{C}$ into a current route between consecutive cells $(u \to v)$. The marginal time increase is

  $$\Delta t(u, x, v) = T(u, x) + T(x, v) - T(u, v) + T_{\text{in}}(x). \tag{18}$$

  The marginal priority gain is

  $$\Delta p(x) = S_x, \tag{19}$$

  provided x has not been assigned to any other patrol yet (otherwise $\Delta p(x) = 0$).
- **Feasibility (return-to-base).** An insertion is feasible if the route remains within the budget:

  $$\text{Time}(\pi) + \Delta t(u, x, v) \leq H.$$

- **Selection score.** Among feasible insertions, we select by a marginal efficiency score

$$\mathrm{score}(x; u, v) = \frac{\big(\Delta p(x)\big)^a}{\big(\Delta t(u, x, v)\big)^b} \cdot \frac{\xi}{\Omega}, \tag{20}$$

  where $a, b > 0$ control the gain–time trade-off, ξ is a small random tie-break factor (to diversify near-equal choices), and Ω is an overlap penalty (defined below).

- **Overlap control (hard cap).** To avoid repeatedly using the same transition elements (e.g., identical portal/side transitions) across patrols, we maintain usage counts for transition elements and impose a hard limit (one use per element). If inserting x would exceed this limit, the insertion is forbidden. For soft control, we also include $\Omega \geq 1$ in (20) to downweight insertions that increase overlap even when still feasible.

- **Greedy loop with unique assignment.** At each iteration we evaluate the best feasible insertion for each patrol and choose the globally best (20). After inserting x into the selected patrol route, we mark x as assigned and remove it from the candidate set for all other patrols. We update route caches (time, collected priority, and overlap counts) incrementally and repeat until no feasible insertion exists.

Choosing the best distribution across bases. For a fixed K we evaluate multiple base distributions m (integer compositions of K across bases). For each m we run the greedy unique-insertion procedure above and obtain a complete set of patrol routes. The best solution is selected by lexicographic preference:

maximize total collected priority, and among ties minimize total patrol time.

From big-cell routes to small-grid paths (execution and visualization). The optimization produces routes as sequences of big cells. For execution and mapping, each big-cell transition $(a \to b)$ is reconstructed on the 1 km grid using the portal model. Optionally, within each block we replace the shortest internal path by a *priority-biased* small-grid path that maximizes accumulated small-cell priority under a bounded detour budget:

$$\mathrm{len}(\mathrm{path}) \;\leq\; (1+\delta)\cdot \mathrm{len}(\mathrm{shortest}),$$

where δ is a tunable detour factor. This preserves travel-time feasibility while allowing limited local "sweeps" through high-priority small cells.

4 Simulation and evaluation

Modeled vs. policy-handled threats (scope of the simulation)

While we identify a broader set of anthropogenic risks in Etosha (RM, WF, OP, SN, IH, DB, TD), the quantitative optimization and Monte Carlo engine in this paper simulate only the four threats that are *directly* affected by patrol routing and the deployed detection technologies: **OP** (organized poaching), **SN** (snaring), **IH** (illegal harvesting), and **TD** (transboundary disease). These threats share a common operational structure: an incident emerges at a location, may or may not be detected by our assets, and the final damage depends strongly on how quickly the nearest ranger team can respond.

The remaining three threats are handled through simpler operational protocols that do not require route optimization, and are therefore excluded from the stochastic event generator:

- **WF (wildfire).** We assume continuous satellite-based active-fire monitoring with high temporal frequency (e.g., Meteosat/SEVIRI operational active-fire products are disseminated on a 15-minute cycle; see [48, 49]). Small natural fires are monitored without immediate ecological interference under a minimal-intervention policy; however, once a fire exceeds a predefined severity threshold, we trigger an early call-out to fire services to reduce the probability of large-area burn.

- **RM (road-related mortality).** When the full GPS package is available, we treat roadkill mitigation as an enforcement problem: reduce the speed limit below the current baseline and apply strict penalties based on GPS traces. This effect is *not* explicitly included in the simulated loss function, and is reported as an operational co-benefit of GPS deployment.

- **DB (dispersal beyond park boundaries).** We treat this primarily as an infrastructure maintenance task: seal unintended fence breaches and reinforce weak segments, while keeping designated corridors/gates as controlled exit/entry points. This reduces both unintended wildlife exits and opportunities for illegal human entry; illegal entries are addressed within the modeled OP/IH processes via boundary detection (sound sensors).

Consequently, all reported Monte Carlo results and the objective metric in Section 4.1 are computed only over OP/SN/IH/TD, while WF/RM/DB are addressed through the policy mechanisms above.

4.1 Monte Carlo scenario generation

We evaluate each budget/patrol configuration using repeated stochastic simulations rather than a single fixed scenario. Time is discretized into hours over a horizon of $D \times 24$ steps (for D simulated days). To make configurations comparable, we use a **normalized event pressure** setup: **exactly one risk event is generated per hour**. This does *not* claim a real incident rate for Etosha; instead, it fixes the external event load so that differences in `TotalLoss` reflect only the relative effectiveness of the protection portfolio (rangers/GPS/sound) and routing decisions under the same pressure. The event type is sampled from a **season-dependent categorical distribution** over four modeled risks: **OP** (poaching), **SN** (snaring), **IH** (illegal harvesting), and **TD** (transboundary disease). The season is determined by the calendar date (starting from 2025-01-01) and mapped into one of four season groups G_1–G_4. We derive Table 3 by conditioning the seven-threat regime weights $w_i(g)$ (Table 2) on the simulated subset $\{OP, SN, IH, TD\}$ and renormalizing them to sum to 1.

Table 3: Season groups and event-type sampling probabilities (%).

Group	Season label	Months	OP (%)	SN (%)	IH (%)	TD (%)
G_1	Dry + popular	Jul–Sep	48.18	28.78	16.73	6.31
G_2	Wet + popular	Oct–Nov	29.80	48.49	17.80	3.90
G_3	Wet	Dec–Feb	7.47	12.84	21.50	58.19
G_4	Dry	Mar–Jun	52.40	32.20	11.18	4.22

After sampling the event type, we randomize the event location and (when applicable) its target using simple, transparent rules: (i) **OP/IH**: spawn is chosen **50/50** between *border entry* and *road entry*. For border entry, the specific border cell is sampled with a strong bias toward high-priority cells (sampling weight $\propto$ `priority`^{2}); for road entry, the spawn is sampled from road nodes. Then a target type is sampled with fixed weights (animals 20%, plants 10%, water 70%), and the intruder moves toward the selected target at a fixed intruder speed. (ii) **SN**: the event spawns on an animal node (habitat area), representing trap placement where animals are present. (iii) **TD**: the event spawns near the border (preferably on border animal nodes, with fallbacks) to represent cross-boundary introduction.

4.2 Protection metric over time

We define "protection" as **minimizing expected ecological damage** under budget constraints. Each event accumulates a nonnegative **loss** value that increases when detection and response are delayed, and is capped by a risk-specific maximum. For any simulated period, we compute the overall objective:

$$\texttt{TotalLoss} = \sum_{t=1}^{D\cdot 24} \texttt{loss}_t,$$

as well as a risk-level breakdown:

$$\texttt{LossByRisk}[r], \quad \texttt{EventsByRisk}[r], \quad \texttt{DetectedByRisk}[r], \quad \texttt{UndetectedByRisk}[r],$$

Normalized protection level. We convert total simulated loss into a unitless score in $[0, 1]$ using a fixed linear normalization:

$$\text{ProtectionLevel}(B) = \left[\frac{L_0 - L(B)}{L_0 - L_{\min}}\right]_0^1, \qquad [x]_0^1 = \min\{1, \max\{0, x\}\}, \tag{21}$$

where $L(B)$ is `TotalLoss` at budget B, $L_0 = L(0)$ is the baseline (no budget), and $L_{\min}$ is a chosen constant, which is a best-case reference floor used to anchor the scale. Thus $\text{ProtectionLevel}(0) = 0$ and reaching $L_{\min}$ corresponds to $\text{ProtectionLevel} = 1$.

for $r \in \{\texttt{OP}, \texttt{SN}, \texttt{IH}, \texttt{TD}\}$.

An event can be detected via four channels:

- **Sound trackers** (for border-entry events),
- **GPS tracking** (for road-entry events, only if the full GPS package is purchased),
- **Ranger proximity** (a ranger within 1 km),
- **Fixed photo traps / camera points** (within 300 m; assumed pre-existing infrastructure and therefore not budgeted).

Once detected, we compute the nearest ranger and their **response time** T_{resp} to the event, and apply an SLA-style threshold: the response is considered timely if $T_{\text{resp}} \leq T_{\text{SLA}}$. Importantly, even timely response may still produce nonzero loss if the intruder has already spent time in a high-value habitat cell.

4.3 Results

To understand how fast protection improves under limited resources, we ran a *budget sweep* and plotted the resulting **protection level** $\text{ProtectionLevel}(B)$ (Eq. 21) against the total available budget. Figure 5 shows a clear pattern of **diminishing marginal returns**: initial investments yield the largest improvement in protection, while further spending produces progressively smaller gains (a saturation effect). Operationally, this means that beyond some point, adding more resources is better justified only if the objective requires very high protection levels, because the *cost per additional unit of protection* increases.

In our simulations, a budget around **300,000** already achieves **above 50% protection**, after which the curve noticeably flattens. Therefore, for a cost-effective baseline plan (when funds are limited), we recommend treating $\sim$ 300,000 as a practical stopping point for this scenario; higher budgets still help, but the additional benefit grows slowly.

Concrete staffing interpretation at $B \approx 300{,}000$. Under the baseline unit-cost assumptions (Table 4), the near-optimal configuration around $B \approx 300{,}000$ corresponds to $K = 3$ **ranger patrol teams**. Assuming **5 rangers per team**, this is **15 rangers** in active field capacity. In other words, the recommended "$\sim$ 300,000 stopping point" translates into a small but operationally meaningful staffing level: three independent teams that can cover separate high-priority regions within the same shift-time constraint. All results in Figure 5 are **conditional on the unit-cost assumptions** summarized in Table 4. These values were chosen to be *plausible low-end* estimates based on publicly available ranger compensation ranges and ultra-low-cost hardware listings; however, real procurement and deployment costs can vary substantially across suppliers, maintenance, connectivity, and staffing structure. Thus, we interpret the curve primarily as a **relative trade-off shape** (how quickly protection saturates), rather than a procurement-accurate forecast.

Model parameter	Meaning	Value
`DEFAULT_PATROL_COST`	cost of one ranger patrol team	65,000
`DEFAULT_MIN_RANGERS_REQUIRED`	minimum number of teams $K_{\min}$	2
`DEFAULT_GPS_FULL_COST`	all-or-nothing GPS package cost	90,000
`DEFAULT_SOUND_TRACKER_COST_PER_KM`	sound monitoring cost per km of border	750

Table 4: Baseline unit-cost assumptions used in the budget sweep experiments.

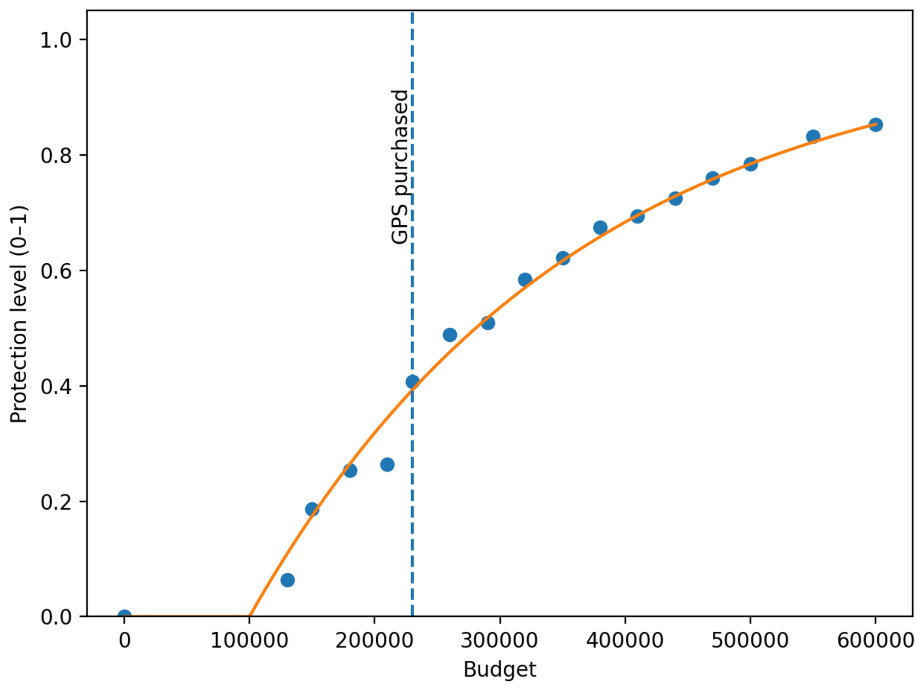


Figure 5: Budget vs. protection level (points + smoothed saturating trend). The curve illustrates diminishing returns: each additional budget increment yields a smaller protection gain.

Representative patrol plan (map). Figure 9 (Appendix A.1) shows one representative route output for $K = 3$ (three teams) over the priority map. It illustrates how the heuristic splits the park into three large patrol areas and increases unique coverage of high-priority blocks within the same shift-time budget.

5 Strengths and Limitations

Strengths

Our approach clearly separates priorities in the field of nature conservation and the seasonality of operational activities by decomposing threats into an internal impact component W_i^{impact} and a probability component specific to a particular regime $w_i(g)$, later combining them into

$$\text{RiskCoeff}_i(g) = W_i^{\text{impact}}\, w_i(g).$$

This makes the model interpretable and flexible.

The weighting procedure is transparent and auditable, as qualitative judgments are converted into quantitative weights by using a deterministic AHP-style mapping with consistency checks and fixed rating scales from 1 to 5, facilitating a uniform assessment of heterogeneous threats. Reliability is ensured through sensitivity analysis by perturbing criterion weights, ordinal ratings, and rank mapping, as well as re-evaluating subsequent rankings and recommendations.

The basic mathematical model supports the practical allocation of resources in conditions of scarcity and represents the fleet as a spatial decision-making structure with layers of risk specific to threats. Concretely, a single mobility graph (travel-time model) is used consistently to connect the priority map, budgeted resource allocation, daily route construction, and Monte Carlo evaluation. This allows for the optimization of patrol deployment towards high-value areas rather than unachievable full coverage, while explicitly considering constraints such as travel time, accessibility, staffing capacity, route length limitations, and shift feasibility.

This structure may include randomization of patrol routes and diversity constraints to reduce predictability for adaptive adversaries, and it generates quantitative measures of protection that meet the task requirement to define and measure protection in a realistic and practical manner. The staged portfolio logic also prioritizes controls that are harder to bypass at scale (e.g., full-coverage GPS auditability on the road network) while using boundary sensing as a complementary layer.

Limitations

The results of the model are only as reliable as its structural assumptions and input layers, and several simplifications may affect its realism.

First, because incident-level data are incomplete or inconsistently reported, regime probability weights are derived from structured expert ratings and therefore represent relative probabilities within a regime rather than statistically estimated event probabilities. The results should be interpreted as guidance for prioritization rather than as precise predictions.

Second, the deterministic translation of rating gaps into the Saaty scale fixes the curvature of differences in weights, and alternative expert rankings may produce different numerical vectors. This motivates sensitivity testing but does not eliminate dependence on these choices.

Third, the spatial representation of the park inevitably discretizes continuous terrain and may omit micro-hot spots, off-road mobility, and small-scale habitat features. As a result, optimization may lead to excessive concentration of patrol efforts on modeled corridors while underestimating unmodeled access routes.

Fourth, several dynamic aspects of threats are simplified: spreading hazards are represented without full physical calibration or contact network calibration, and deliberate human threats can behave strategically and adapt to patrol patterns more strongly than captured, unless an explicit adversary model is included.

Fifth, resource allocation typically optimizes expected risk reduction at assumed detection or response effectiveness. In practice, however, anti-poaching effectiveness depends on an end-to-end operational pipeline that can be hampered by communication issues, reporting delays, limited vehicle availability, or insufficient analytical capabilities. If these delays are not parameterized, technologically complex measures may be overestimated.

Sixth, staffing estimates depend on assumed patrol speed, route feasibility, coverage definitions, and compliance constraints. Small errors in these operational parameters can significantly alter required staffing levels, especially when coverage targets approach maximum capacity. In particular, we assume a single constant travel speed (e.g., 40 km/h) for all moves; deviations due to terrain, road condition, and seasonality would change feasibility and staffing requirements. We also assume that each ranger team is anchored to a single base and returns to it (no overnight relocation between bases), which simplifies logistics but can reduce flexibility in multi-day operations.

Seventh, normalizing weights across all threats produces relative assessments, even if absolute risk levels are simultaneously increasing. Therefore, recalculation or absolute calibration is required for year-over-year comparisons.

Finally, although this framework is transferable, its application to other fleets requires recalibration of regime calendars, risk levels, mobility restrictions, detection capabilities, and threat typologies. Without local data and stakeholder validation, direct transfer of parameter values may result in misplaced priorities.

6 Adaptation to other protected areas

The first protected area to be considered is the Altai State Nature Biosphere Reserve. This reserve contains several mountain ranges within and along its borders, as well as more than 1,000 lakes, including Lake Teletskoye. At the same time, there are no roads in the Altai Reserve, and the territory is practically impassable unless you use the rare trails laid out by foresters and reserve employees. Visiting the reserve is only possible with the permission of the administration and requires a special pass.

The climate is mountainous and continental, so the summers are short and cool and the winters are cold and snowy, making it even more difficult to move around the reserve.

The main threats to animal safety include poaching (often illegal fishing and hunting of musk deer and reindeer), illegal extraction of natural resources, fires, climate change, and tourism pressure in border areas. However, the difficult terrain makes it much more difficult to patrol the territory. Usually, a patrol lasts 2-2.5 weeks, and violators of the reserve's rules are caught by chance.

The following list of species can also be identified as requiring more careful protection due to their vulnerability or value: snow leopard, Altai mountain sheep, musk deer, wolverine, golden eagle, peregrine falcon, grayling, Rhodiola rosea.

Thus, the following points from the model can be identified as needing to be changed: the relief map, the list of animals and plants that need to be more carefully protected, changing the way patrols move around (from cars to air/foot), and adding seasonality.

Next, let's take a look at Wood Buffalo National Park. Most of the park is covered by northern coniferous forests interspersed with swamps and peat bogs. In the centre is the delta of the Pees and Athabasca rivers. It contains extensive floodplains, wetlands and a network of channels and temporary lakes. Unlike the previous park, Wood Buffalo National Park is predominantly flat terrain with high marshiness. This makes it difficult to move around the park. The climate is sharply continental, subarctic. Winter lasts from October to April, and temperatures often drop below -30 °C, which makes it easier to travel across dried-up swamps and rivers. Summers are short, not too warm, with frequent thunderstorms. There is not much precipitation, mainly in the form of snow, but there are spring floods. The main threats to animal safety include the following: disease (brucellosis and tuberculosis), industry (there are oil production sites nearby and leaks can occur), poaching (illegal hunting of bison, moose and reindeer, illegal fishing). Poaching is not as widespread as in Etosha National Park, as horns and tusks do not have a high commercial value, and there is active cooperation with indigenous peoples, but the vast and impassable terrain makes patrolling very difficult. The following list of species that require more careful protection due to their vulnerability or value can also be highlighted: wood bison, wolves, grizzly bears, black bears, moose, reindeer, and whooping cranes. Thus, the following points from the model need to be changed: the relief map, the list of species under close observation, patrol mobility (replace cars with foot patrols in the summer), and create two different graphs for two different seasons (because in winter the territory becomes more accessible due to the freezing of swamps).

References

[1] Outdoor Tech Lab. 4 Best Garmin InReach Devices in 2026: Ultimate Buyer's Guide. 2025.

[2] Altzapovednik. 18-Brakonierov. 2012.

[3] Legacy Landscapes Fund. 50+ Years of Funding Ahead for Skeleton Coast-Etosha. 2024.

[4] Admin. Suspected Poachers Shot in Etosha. Namibian.org, 2025.

[5] Ongava Research Centre. Greater Etosha Carnivore Programme. 2024.

[6] Africa Geographic. African Safaris with Africa Geographic. 2025.

[7] IZW Cheetah Research Project. Spatial Ecology and Distribution. 2019.

[8] Balfour, D., et al. African Rhino Conservation 2025–2035: A Contemporary Strategic Framework.

[9] Beasley, J. Studying and Conserving Carnivores in a Changing World. Conservation Namibia, 2025.

[10] Berezin, J. L., et al. A Snapshot into the Lives of Elephants. *Diversity*, 15(11), 1146, 2023. doi:10.3390/d15111146

[11] Berry, H. H. Surveillance and Control of Anthrax and Rabies in Namibia. *Rev. Sci. Tech. OIE*, 12(1), 137–146, 1993. doi:10.20506/rst.12.1.675

[12] Borges, J., et al. Assessing Habitat Suitability. *Remote Sensing*, 16(15), 2855, 2024. doi:10.3390/rs16152855

[13] Chan, A. N., et al. Landscape Connectivity for Namibian Elephants. *Landscape Ecology*, 39(8), 2024. doi:10.1007/s10980-024-01957-2

[14] TourRadar. Etosha Animals: Check What Animals to Spot on Safari. 2019.

[15] SafariBookings. Etosha Weather and Climate.

[16] Hamalwa, F. Risking All for Mopane Worms at Etosha. New Era, 2025.

[17] X-Kom. DJI Inspire 1 Pro Product Page. 2026.

[18] Hansson, L., et al. Trip Report Southern Africa. 2001.

[19] Ashodia, H. Exploring Microsoft's Vision: Planetary Computer. 2024.

[20] Natural Selection. World Ranger Day. 2025.

[21] Needhi, J. The Role of AI in Wildlife Conservation. Medium, 2024.

[22] WWF. Kenya's Black Rhinos Get Microchips. 2024.

[23] Voice of America. Kenya's Rhinos Get Anti-Poaching Microchips. 2013.

[24] Mansfield, K., et al. Rabies Epizootics in Kudu. *BMC Veterinary Research*, 2006. doi:10.1186/1746-6148-2-2

[25] McKendrick, E. Acing a Self-Drive Safari in Etosha. 2017.

[26] Microsoft. Microsoft Planetary Computer.

[27] Ministry of Environment, Forestry and Tourism. Annual Report. 2021.

[28] Naha, D., et al. Fencing Affects Movement Patterns. *Frontiers in Ecology and Evolution*, 2023. doi:10.3389/fevo.2023.1031321

[29] Associated Press. Namibia Sends Hundreds of Soldiers to Battle Wildfire. 2025.

[30] Reuters. Namibia Projects Tourism Decline after Wildfire. 2025.

[31] Oloo, R., et al. Seasonal Variation in Feeding Ecology of Black Rhinoceros. 1994.

[32] Ongava Research Centre. Counting Carnivores in the Greater Etosha Landscape. 2024.

[33] Ongava Research Centre. Counting Spots: Citizen Science Survey. 2025.

[34] Ongava Research Centre. Large-Scale Surveying of Etosha's Wildlife. 2023.

[35] Etosha National Park. Can Drones Save Etosha's Wildlife? 2014.

[36] Patterson, J. R., et al. Tourism Effects on Lions and Hyenas. *Global Ecology and Conservation*, 2025. doi:10.1016/j.gecco.2025.e03681

[37] Portas, R., et al. Zebra with Anthrax as Cause of Death for Cheetahs. *Frontiers in Veterinary Science*, 2021. doi:10.3389/fvets.2021.714758

[38] PAWS. Protection Assistant for Wildlife Security. 2018.

[39] Seidel, D. P. RhinoRecursion Dataset. Zenodo, 2019. doi:10.5281/zenodo.2783972

[40] Van Coeverden de Groot, P. J., et al. Conservation Genetics of the Black Rhinoceros. *Conservation Genetics*, 2011. doi:10.1007/s10592-011-0185-1

[41] Wato, Y. A., et al. Movement Patterns of African Elephants. *Frontiers in Ecology and Evolution*, 2018. doi:10.3389/fevo.2018.00167

[42] World Health Organization. Rabies Fact Sheet. 2024.

[43] Zidon, R., et al. Zebra Migration Strategies and Anthrax. *Ecosphere*, 2017. doi:10.1002/ecs2.1925

[44] Siegfried, W. R. The incidence of veld-fire in the Etosha National Park, 1970–1979. *Madoqua*, 12(4), 1981.

[45] Ministry of Environment and Tourism. Etosha National Park Aerial Survey 2005. First Draft, 14 October 2005.

[46] Ministry of Environment and Tourism. Etosha Management Plan. 2007.

[47] ResearchGate. Map of Etosha National Park showing locations of sampled *D. bicornis* individuals (figure).

[48] EUMETSAT. *Active Fire Monitoring: Product Guide*. 2015. `https://user.eumetsat.int/s3/eup-strapi-media/pdf_fir_pg_73dd73799f.pdf`

[49] EUMETSAT. *Fire Radiative Power and Fire Radiative Energy Factsheet*. 2020. `https://www-cdn.eumetsat.int/files/2020-04/pdf_frp_fre_factsheet.pdf`

[50] Ministry of Environment, Forestry and Tourism (Namibia). Notice Board Advert: Vacant Positions – Department of Wildlife and National Parks (includes Ranger Grade 11 salary scale and allowances). PDF. `https://www.meft.gov.na/files/files/Final%20Notice%20board%20advert%20for%20vacant%20positions%20-%20DWNP%201.pdf`

[51] Made-in-China (supplier listing). Universal 4G OBD2 Car GPS Tracker (low-end bulk price range listing). https://tmarktracker.en.made-in-china.com/product/qAQRSeHYIGVJ/China-Universal-4G-OBD2-Car-GPS-Tracker-Best-Price-Locator-History-Playback-Logo-Customize-Easy-Transfer-of-Platform.html

[52] CHOOVIO IoT Solutions. Browan TBSL100 LoRaWAN Sound Level Sensor (price listing). https://www.choovio.com/product/tbsl100-lorawan-sound-level-sensor/

Report on Use of AI Tools

This report documents how AI tools were used in preparing this submission. AI tools were used for web-search assistance and translation purposes of publicly available sources.

Tool: Perplexity
Dates used: 10–11 Feb 2026
Purpose: Find public sources (news, reports, datasets) and collect stable links; produce brief summaries of what each source contains.

Query 1

Date: 10 Feb 2026
User query:

```
Etosha snares confiscated "wire snare" operations statistics location Okaukuejo report
in recent years.  Give ALL links
```

AI output (cleaned, links preserved):

- Observer 24: article discussing wire-snare removals around Okaukuejo (mentions 62 active snares retrieved in three days; 2023).
  `https://www.observer24.com.na/illegal-wire-snares-are-decimating-wildlife-in-etosha/`
- Namibian.org: article on intensified patrols to curb wire traps in Etosha (Okaukuejo area; 2023).
  `https://namibian.org/news/nature-and-environment/more-patrols-in-etosha-national-park-to-curb-wire-traps`
- NBC Namibia: item reporting the same operational statistic (62 active wire snares retrieved; 2023).
  `https://nbcnews.na/index.php/node/102275`
- The EIS (Environmental Information Service) eLibrary: index page containing a PDF item (Informanté) referencing the same operation.
  `https://the-eis.com/elibrary/taxonomy/term/105484?page=11`
- MEFT annual report (general context; not always location-specific):
  `https://www.meft.gov.na/files/downloads/MEFT%20Annual%20Report%202021-2022.pdf`

Query 2

Date: 11 Feb 2026
User query:

```
Etosha wildlife telemetry dataset Movebank OR Zenodo "Etosha" elephant GPS collar
data shapefile.  Give links
```

AI output (cleaned, links preserved):

- Movebank Data Repository: African elephants in Etosha National Park (data package / metadata).
  `https://datarepository.movebank.org/handle/10255/move.812`

- DOI record for the same Movebank dataset:
  `https://doi.org/10.5441/001/1.3nj3qj45/1`

- Versioned item page (Movebank repository listing):
  `https://datarepository.movebank.org/items/bd98ee38-55c8-468b-8271-da3f2caf3acf`

Query 3

Date: 11 Feb 2026
User query:

```
Etosha National Park all poaching incidents with links
```

AI output (cleaned, links preserved):

- MEFT (Namibia) poaching statistics update (national-level reporting, includes Etosha context):
  `https://www.meft.gov.na/files/files/poaching%20update%202022%2023.pdf`

- Namibia Financial Intelligence Centre (typology report; includes case examples / context):
  `https://www.fic.na/uploads/TrendsandTypologies/FICTrendsandTypologyReports/Namibias%20Wildlife%20Poaching%20and%20related%20Money%20Laundering%20Activities%20FINAL.pdf`

- Save the Rhino: discussion of Namibia rhino poaching totals with Etosha highlighted (secondary reporting):
  `https://www.savetherhino.org/rhino-species/black-rhino/poaching-in-namibia-almost-doubles-in-2022/`

- Namibian.org: Etosha rhino-poaching incident reporting and government-quote summaries (incident anchors):
  `https://namibian.org/news/nature-and-environment/rhino-protection-intensified-after-recent-poaching-in-etosha-national-park`
  `https://namibian.org/news/nature-and-environment/suspected-poachers-shot-dead-in-etosha`

- The Namibian: reporting on Etosha poaching incident counts under an anti-poaching operation (incident-count statement):
  `https://www.namibian.com.na/police-commander-reports-decrease-in-etosha-poaching/`

- ABC News: reporting citing official data and statements on record rhino-poaching levels (secondary reporting):
  `https://abcnews.go.com/International/namibia-reports-record-rise-rhino-poaching/story?id=96842444`

- EtoshaNationalPark.org (non-official aggregation site; used only as a discovery aid, not as a primary source):
  `https://www.etoshanationalpark.org/news/tag/poaching`

A Additional Materials

A.1 Additional Spatial Maps

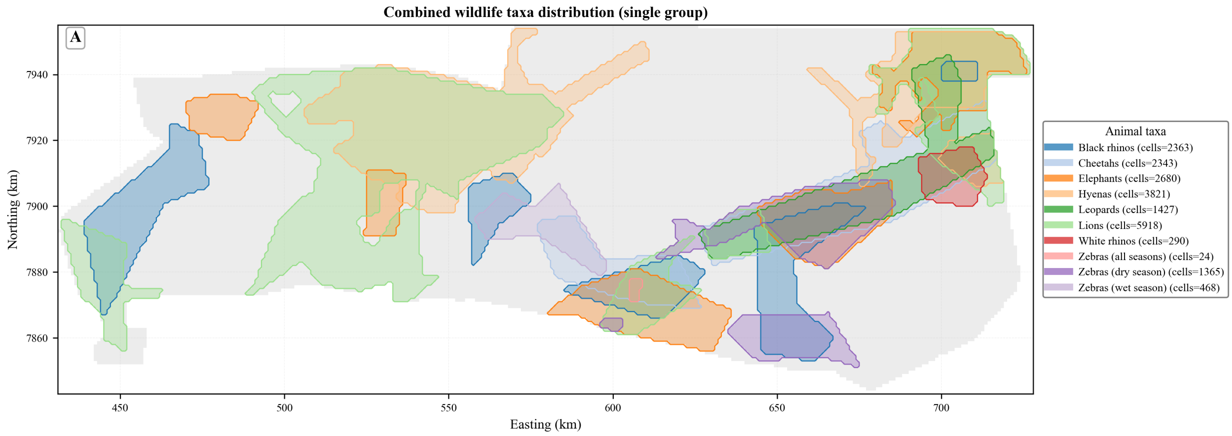


Figure 6: Combined spatial distribution of key wildlife taxa.

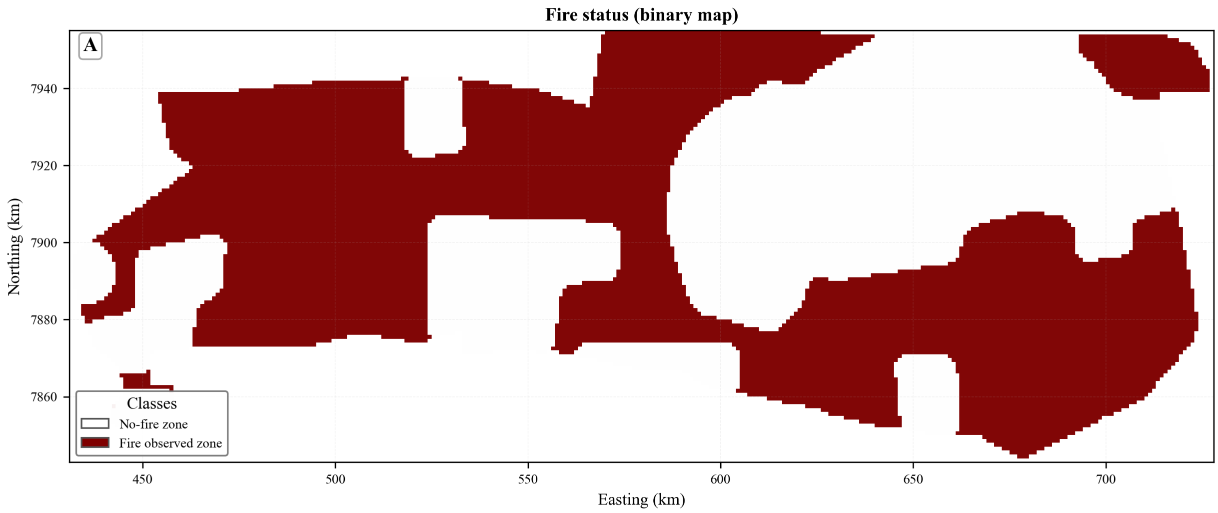


Figure 7: Fire status map (observed fire vs. no-fire zones).

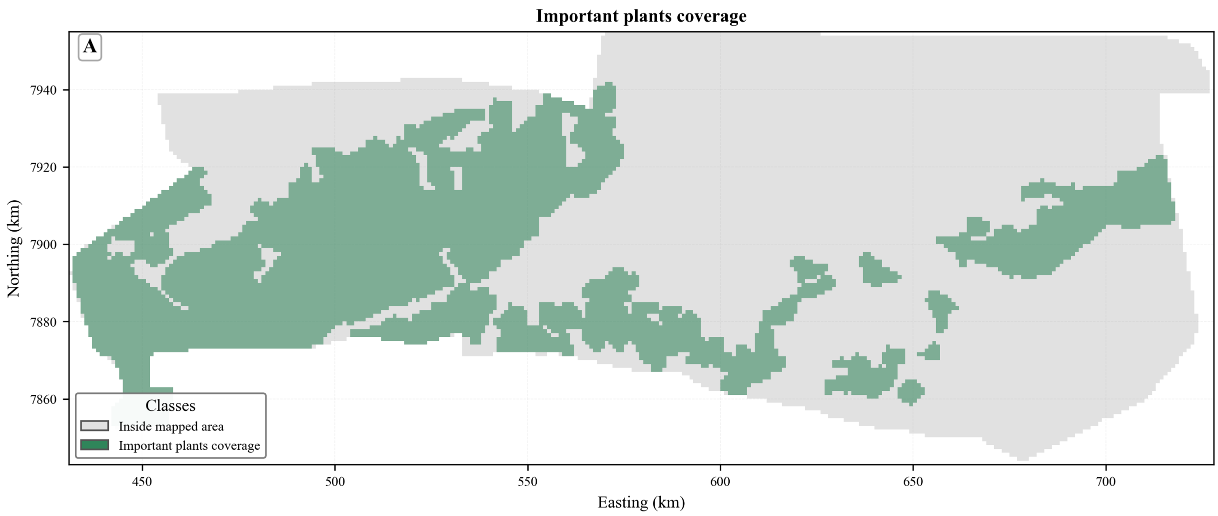


Figure 8: Coverage of important plant areas.

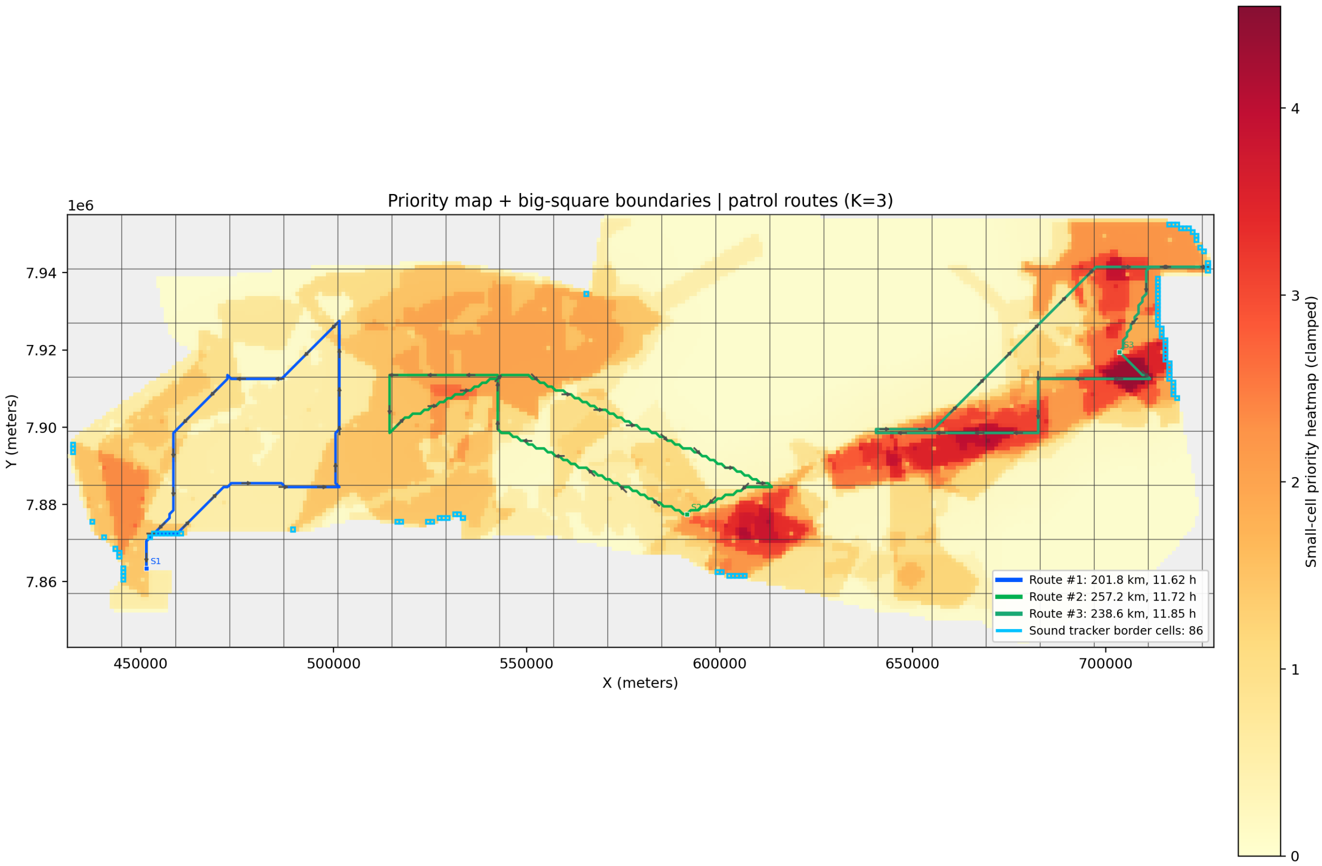


Figure 9: Representative patrol routing output for $K = 3$ teams (priority heatmap with big-cell boundaries).

A.2 Implementation Excerpts (Python)

A.2.1 Budget allocation policy (minimum rangers + full GPS rule)

This excerpt implements the staged budget policy used in Section 3.3.

Listing 1: Budget allocation with minimum ranger capacity and all-or-nothing GPS.

```
def allocate_budget(
    total_budget: float,
    patrol_cost: float,
    min_rangers_required: int,
    gps_full_cost: float,
) -> BudgetPlan:
    if total_budget < 0.0:
        raise ValueError("total_budget must be >= 0")
    if patrol_cost <= 0.0:
        raise ValueError("patrol_cost must be > 0")
    if min_rangers_required < 1:
        raise ValueError("min_rangers_required must be >= 1")
    if gps_full_cost < 0.0:
        raise ValueError("gps_full_cost must be >= 0")

    affordable_total = int(total_budget // patrol_cost)
    mandatory = min(min_rangers_required, affordable_total)
    spent_mandatory = mandatory * patrol_cost
    minimum_met = mandatory >= min_rangers_required

    remaining = total_budget - spent_mandatory
    gps_bought = minimum_met and (remaining >= gps_full_cost)
    spent_gps = gps_full_cost if gps_bought else 0.0
    remaining_after_gps = remaining - spent_gps
```

```python

    if minimum_met:
        k_max_by_budget = int((total_budget - spent_gps) // patrol_cost)
    else:
        k_max_by_budget = affordable_total

    return BudgetPlan(
        total_budget=total_budget,
        patrol_cost=patrol_cost,
        min_rangers_required=min_rangers_required,
        mandatory_rangers=mandatory,
        spent_on_mandatory=spent_mandatory,
        minimum_met=minimum_met,
        gps_full_cost=gps_full_cost,
        gps_bought=gps_bought,
        spent_on_gps=spent_gps,
        remaining_after_gps=remaining_after_gps,
        k_min=mandatory,
        k_max_by_budget=k_max_by_budget,
    )
```

A.2.2 Selecting acoustic sensors on the true park border (priority-ranked)

This excerpt extracts border cells (inside cell with at least one outside 4-neighbor) and ranks them by node priority.

Listing 2: Border cell extraction and ranking by priority.

```python
out: List[BorderCellScore] = []
for cell_id, node_feat in nodes.items():
    if not isinstance(node_feat, dict):
        continue
    if node_feat.get("median_elevation_m") is None:
        continue
    rr = int(node_feat.get("row", -1))
    cc = int(node_feat.get("col", -1))
    if rr < 0 or cc < 0:
        continue
    has_outside_neighbor = False
    for dr, dc in ((1, 0), (-1, 0), (0, 1), (0, -1)):
        nb_id = by_rc.get((rr + dr, cc + dc))
        if nb_id is None:
            has_outside_neighbor = True
            break
        if not inside_mask.get(nb_id, False):
            has_outside_neighbor = True
            break
    if not has_outside_neighbor:
        continue

    if cell_id not in node_priority:
        continue

    centroid = node_feat.get("centroid_m") or [0.0, 0.0]
    if not isinstance(centroid, list) or len(centroid) != 2:
        centroid = [0.0, 0.0]

    out.append(
        BorderCellScore(
            cell_id=cell_id,
            priority=float(node_priority[cell_id]),
            row=int(node_feat.get("row", -1)),
```

```
            col=int(node_feat.get("col", -1)),
            centroid_m=[float(centroid[0]), float(centroid[1])],
        )
    )

out.sort(key=lambda x: (-x.priority, x.cell_id))
return out


def select_border_cells_for_sound(
    graph: Dict[str, Any],
    priority_data: Dict[str, Any],
    sound_km: float,
    *,
    score_field: str = DEFAULT_SCORE_FIELD,
) -> Dict[str, Any]:
    if sound_km < 0:
        raise ValueError("sound_km must be non-negative")

    cell_size_m = float(graph.get("meta", {}).get("cell_size_m", 1000.0))
    if cell_size_m <= 0:
        raise ValueError("Invalid graph.meta.cell_size_m, expected > 0")

    km_per_border_cell = cell_size_m / 1000.0
    requested_cell_count = int(math.floor(sound_km / km_per_border_cell + 1e-12))

    border_cells = _extract_border_cells(graph, priority_data, score_field=score_field)
    selected_count = min(requested_cell_count, len(border_cells))
    selected = border_cells[:selected_count]
    all_selected_are_border = True
```

A.2.3 Hourly patrol position traces (one year)

This excerpt generates hourly patrol locations from daily randomized routes over assigned key blocks.

Listing 3: Hourly patrol position generation from daily randomized keypoints.

```
def _build_hourly_patrol_positions(
    ctx: SimulationContext,
    patrol_plan: dict,
    seed: int,
    n_days: int = 365,
) -> Tuple[List[List[str]], List[List[str]]]:
    rng = random.Random(seed)
    patrols = patrol_plan.get("patrols", [])
    if not isinstance(patrols, list):
        return [], []

    n_hours = n_days * 24
    hourly_big: List[List[str]] = []
    hourly_nodes: List[List[str]] = []

    for p in patrols:
        base_big = str(p.get("base", ""))
        if base_big not in ctx.rep_node_by_big:
            continue

        assigned = p.get("assigned", [])
        way = p.get("way", [])

        key_bigs: List[str] = []
        for b in assigned:
            sb = str(b)
```

```
            if sb != base_big and sb in ctx.rep_node_by_big and sb not in key_bigs:
                key_bigs.append(sb)
        if not key_bigs:
            for b in way:
                sb = str(b)
                if sb != base_big and sb in ctx.rep_node_by_big and sb not in key_bigs:
                    key_bigs.append(sb)

        one_big = [base_big for _ in range(n_hours)]
        one_node = [ctx.rep_node_by_big[base_big] for _ in range(n_hours)]

        for day in range(n_days):
            day_start = day * 24
            if not key_bigs:
                continue

            shuffled = list(key_bigs)
            rng.shuffle(shuffled)
            shuffled = shuffled[:DAILY_KEYPOINT_LIMIT]

            route = [base_big]
            cur = base_big
            remain = set(shuffled)
            while remain:
                nxt = min(
                    remain,
                    key=lambda b: ctx.big_distance_m.get(cur, {}).get(b, float("inf")),
                )
                route.append(nxt)
                remain.remove(nxt)
                cur = nxt
            route.append(base_big)

            seg_durations_h: List[float] = []
            total_h = 0.0
            for a, b in zip(route, route[1:]):
                da = ctx.rep_node_by_big[a]
                db = ctx.rep_node_by_big[b]
                d_m = _safe_distance_m(da, db, ctx.node_xy, ctx.node_big, ctx.big_distance_m)
                if not math.isfinite(d_m):
                    d_m = 0.0
                dt_h = d_m / (RANGER_SPEED_KMH * 1000.0)
                seg_durations_h.append(dt_h)
                total_h += dt_h

            for hh in range(24):
                global_h = day_start + hh
                if total_h <= 1e-9:
                    one_big[global_h] = base_big
                    one_node[global_h] = ctx.rep_node_by_big[base_big]
                    continue

                shift_h = float(hh % PATROL_ACTIVE_HOURS_PER_DAY)
                t_h = (shift_h / float(PATROL_ACTIVE_HOURS_PER_DAY)) * total_h

                acc = 0.0
                chosen_big = base_big
                for seg_i, dt_h in enumerate(seg_durations_h):
                    if t_h <= acc + dt_h or seg_i == len(seg_durations_h) - 1:
                        a = route[seg_i]
                        b = route[seg_i + 1]
                        if dt_h <= 1e-9:
                            chosen_big = b
```

```
                        else:
                            frac = max(0.0, min(1.0, (t_h - acc) / dt_h))
                            chosen_big = a if frac < 0.5 else b
                        break
                    acc += dt_h

                one_big[global_h] = chosen_big
                one_node[global_h] = ctx.rep_node_by_big.get(chosen_big, ctx.rep_node_by_big[base_big])

        hourly_big.append(one_big)
        hourly_nodes.append(one_node)
```

A.2.4 OP/IH events: spawning, movement, detection and loss evaluation

This excerpt shows how OP/IH events are instantiated and how detection triggers loss computation.

Listing 4: OP/IH event generation with detection triggers and loss computation.

```
e_type = _sample_event_type(rng, season)

if e_type in ("OP", "IH"):
    spawn_mode = "border" if rng.random() < 0.5 else "road"
    if spawn_mode == "border":
        spawn_pool = ctx.border_nodes if ctx.border_nodes else ctx.animal_nodes
    else:
        spawn_pool = ctx.road_nodes if ctx.road_nodes else ctx.animal_nodes

    if not spawn_pool:
        spawn_pool = tuple(ctx.node_xy.keys())

    if spawn_mode == "border" and ctx.border_nodes:
        spawn_node = _sample_border_node_priority_quadratic(
            rng=rng,
            border_nodes=spawn_pool,
            node_priority=ctx.node_priority,
        )
    else:
        spawn_node = rng.choice(spawn_pool)
    target_node = _pick_target_node(rng, ctx, spawn_node)

    d_to_target_m = _safe_distance_m(spawn_node, target_node, ctx.node_xy, ctx.node_big, ctx.big_distance_m)
    if not math.isfinite(d_to_target_m):
        d_to_target_m = 0.0
    t_to_hab_h = d_to_target_m / (INTRUDER_SPEED_KMH * 1000.0)
    t_hab_h = hour + t_to_hab_h

    ev = {
        "type": e_type,
        "start_h": float(hour),
        "spawn_node": spawn_node,
        "target_node": target_node,
        "spawn_mode": spawn_mode,
        "t_hab_h": float(t_hab_h),
        "end_h": float(hour + OP_IH_MAX_DURATION_H),
    }

    auto_detect = False
    detect_node = spawn_node
    if spawn_mode == "road" and gps_enabled:
        auto_detect = True
    elif spawn_mode == "border" and spawn_node in sound_nodes:
```

```
        auto_detect = True
    elif _nearest_ranger_distance_m_at_hour(hour, spawn_node, hourly_patrol_nodes, ctx) <= RANGER_DETECT_RADIUS_M:
        auto_detect = True
    elif _photo_near(spawn_node, ctx, photo_cache):
        auto_detect = True

    if auto_detect:
        loss = _loss_from_detection(
            event_type=e_type,
            t_det_h=float(hour),
            t_hab_h=float(t_hab_h),
            detect_node=detect_node,
            hourly_patrol_nodes=hourly_patrol_nodes,
            ctx=ctx,
            max_event_end_h=float(hour + OP_IH_MAX_DURATION_H),
        )
        total_loss += loss
```

A.2.5 Spatial graph node and edge construction

This excerpt builds the spatial graph from a regular grid: it creates cell nodes with environmental features and adds 8-neighborhood edges with distance, elevation difference, and road connectivity.

Listing 5: Forming nodes and edges of the spatial graph.

```
node_features = {}
total_cells = n_rows * n_cols
for linear_idx in range(total_cells):
    rr, cc = linear_to_row_col(linear_idx, n_cols)
    x0 = min_x + cc * CELL_SIZE_M
    y0 = min_y + rr * CELL_SIZE_M
    x1 = x0 + CELL_SIZE_M
    y1 = y0 + CELL_SIZE_M

    cid = cell_id(rr, cc)
    poi_counter = poi_counts.get(linear_idx, Counter())
    road_length = float(road_len.get(linear_idx, 0.0))
    animals_here = sorted(list(animals_by_cell.get(linear_idx, set())))
    no_fire_here = bool(no_fire_final_mask[rr, cc])
    has_plant_here = bool(linear_idx in plant_cells)

    node_features[cid] = {
        "row": int(rr),
        "col": int(cc),
        "is_boarder": bool(rr == 0 or cc == 0 or rr == n_rows - 1 or cc == n_cols - 1),
        "bbox_m": [float(x0), float(y0), float(x1), float(y1)],
        "centroid_m": [float(x0 + 0.5 * CELL_SIZE_M), float(y0 + 0.5 * CELL_SIZE_M)],
        "median_elevation_m": (
            float(elev_median[linear_idx]) if linear_idx in elev_median else None
        ),
        "poi_type_counts": {k: int(v) for k, v in sorted(poi_counter.items())},
        "road_total_length_m": round(road_length, 3),
        "road_segment_count": int(road_cnt.get(linear_idx, 0)),
        "animals_present": animals_here,
        "no_fire_zone": bool(no_fire_here),
        "has_plant": bool(has_plant_here),
    }

edge_features = {cid: {} for cid in node_features}
active_lookup = {(node_features[cid]["row"], node_features[cid]["col"]): cid for cid in node_features}
```

```
roads_sindex = roads_metric.sindex
road_geometries = roads_metric.geometry.values
edge_road_cache = {}

for cid, feats in node_features.items():
    rr = feats["row"]
    cc = feats["col"]
    for dr, dc in (
        (-1, 0), (1, 0), (0, -1), (0, 1),
        (-1, -1), (-1, 1), (1, -1), (1, 1),
    ):
        nr = rr + dr
        nc = cc + dc
        neighbor_id = active_lookup.get((nr, nc))
        if neighbor_id is None:
            continue

        z1 = feats["median_elevation_m"]
        z2 = node_features[neighbor_id]["median_elevation_m"]
        zdiff = None if z1 is None or z2 is None else abs(float(z1) - float(z2))
        is_diagonal = dr != 0 and dc != 0

        if is_diagonal:
            edge_road_stats = {
                "road_between_cells": False,
                "roads_between_cells_count": 0,
                "roads_between_cells_length_m": 0.0,
            }
            distance_m = DIAGONAL_DISTANCE_M
            shared_border_m = 0.0
        else:
            edge_key = ((rr, cc), (nr, nc)) if (rr, cc) < (nr, nc) else ((nr, nc), (rr, cc))
            if edge_key not in edge_road_cache:
                edge_road_cache[edge_key] = road_between_cells_stats(
                    roads_sindex=roads_sindex,
                    road_geometries=road_geometries,
                    min_x=min_x,
                    min_y=min_y,
                    cell_size_m=CELL_SIZE_M,
                    n_cols=n_cols,
                    road_len_by_cell=road_len,
                    r1=rr, c1=cc, r2=nr, c2=nc,
                )
            edge_road_stats = edge_road_cache[edge_key]
            distance_m = CELL_SIZE_M
            shared_border_m = CELL_SIZE_M

        edge_features[cid][neighbor_id] = {
            "distance_m": distance_m,
            "shared_border_m": shared_border_m,
            "elevation_median_diff_m": zdiff,
            **edge_road_stats,
        }
```

A.2.6 Node priority computation pipeline

This excerpt computes each cell priority as a weighted mixture of its local importance and the importance of its near and far graph context, with exponential decay by graph distance.

Listing 6: Node priority computation pipeline.

```
import heapq
```

```
import math
from typing import Any, Dict, Iterable, List, Tuple

TOP_K_NEIGHBORS = 1000

W_ANIMAL = 1.00
W_WATER = 0.50
W_PLANT = 0.50
W_COVERAGE_PENALTY = 0.50

HOP_DECAY_ALPHA = 1.0

DEFAULT_UNKNOWN_ANIMAL_WEIGHT = 0.0
ANIMAL_WEIGHTS: Dict[str, float] = {
    "all_lions": 1.00,
    "all_black_rhyno": 0.95,
    "all_white_rhino": 0.90,
    "all_elephants": 0.75,
    "all_cheetah": 0.80,
    "all_leopard": 0.85,
    "all_hyena": 0.50,
    "all_zebra": 0.35,
    "dry_zebra": 0.30,
    "wet_zebra": 0.30,
}

def edge_travel_time_hours(edge_feat: Dict[str, Any]) -> float:
    distance_m = float(edge_feat.get("distance_m", 1000.0))
    road_between_cells = bool(edge_feat.get("road_between_cells", False))
    speed_kmh = 60.0 if road_between_cells else 30.0
    return distance_m / (speed_kmh * 1000.0)

def avg(values: Iterable[float]) -> float:
    data = list(values)
    if not data:
        return 0.0
    return sum(data) / len(data)

def _normalized_layer_weights(use_hop1: bool, use_hop2: bool) -> Tuple[float, float, float]:
    raw_self = math.exp(-HOP_DECAY_ALPHA * 0.0)
    raw_hop1 = math.exp(-HOP_DECAY_ALPHA * 1.0) if use_hop1 else 0.0
    raw_hop2 = math.exp(-HOP_DECAY_ALPHA * 2.0) if use_hop2 else 0.0
    raw_sum = raw_self + raw_hop1 + raw_hop2
    if raw_sum <= 0.0:
        return 1.0, 0.0, 0.0
    return raw_self / raw_sum, raw_hop1 / raw_sum, raw_hop2 / raw_sum

def top_k_by_time(candidates: List[str], time_by_node: Dict[str, float], k: int) -> List[str]:
    return sorted(candidates, key=lambda cid: (time_by_node.get(cid, float("inf")), cid))[:k]

def compute_local_score(node_feat: Dict[str, Any]) -> Tuple[float, Dict[str, float]]:
    animals_present = node_feat.get("animals_present") or []
    poi_counts = node_feat.get("poi_type_counts") or {}

    an_i = 0.0
    for animal in animals_present:
        an_i += float(ANIMAL_WEIGHTS.get(animal, DEFAULT_UNKNOWN_ANIMAL_WEIGHT))

    wa_i = 1.0 if int(poi_counts.get("waterhole", 0)) > 0 or int(poi_counts.get("waterhole_dry",
        0)) > 0 else 0.0
    pl_i = 1.0 if bool(node_feat.get("has_plant", False)) else 0.0
    cov_i = 1.0 if int(poi_counts.get("patrol_house", 0)) > 0 or int(poi_counts.get("photo_trap",
         0)) > 0 else 0.0
```

```python

    local_raw = W_ANIMAL * an_i + W_WATER * wa_i + W_PLANT * pl_i
    s_i = local_raw * (1.0 - W_COVERAGE_PENALTY * cov_i)

    return s_i, {
        "AN_i": an_i,
        "WA_i": wa_i,
        "PL_i": pl_i,
        "COV_i": cov_i,
    }

def build_hop_sets_and_times(
    node_id: str,
    weighted_adj: Dict[str, List[Tuple[str, float]]],
    top_k_neighbors: int,
) -> Tuple[List[str], Dict[str, float], List[str], Dict[str, float]]:
    n1_time: Dict[str, float] = {}
    for n1, t_1 in weighted_adj.get(node_id, []):
        n1_time[n1] = t_1
    n1_nodes = sorted(n1_time.keys())

    n1_set = set(n1_nodes)
    forward_time: Dict[str, float] = {}
    best_time: Dict[str, float] = {node_id: 0.0}
    pq: List[Tuple[float, str]] = [(0.0, node_id)]

    while pq and len(forward_time) < top_k_neighbors:
        t_cur, cur = heapq.heappop(pq)
        if t_cur > best_time.get(cur, float("inf")):
            continue

        if cur != node_id and cur not in n1_set and cur not in forward_time:
            forward_time[cur] = t_cur
            if len(forward_time) >= top_k_neighbors:
                break

        for nxt, edge_t in weighted_adj.get(cur, []):
            cand = t_cur + edge_t
            old = best_time.get(nxt)
            if old is None or cand < old:
                best_time[nxt] = cand
                heapq.heappush(pq, (cand, nxt))

    forward_nodes = sorted(forward_time.keys(), key=lambda cid: (forward_time[cid], cid))
    return n1_nodes, n1_time, forward_nodes, forward_time

def compute_priorities(graph: Dict[str, Any], top_k_neighbors: int = TOP_K_NEIGHBORS) -> Dict[str
    , Any]:
    nodes: Dict[str, Dict[str, Any]] = graph["node_features"]
    edges: Dict[str, Dict[str, Dict[str, Any]]] = graph["edge_features"]

    weighted_adj = {
        node_id: [(nxt, edge_travel_time_hours(edge_feat)) for nxt, edge_feat in neis.items()]
        for node_id, neis in edges.items()
    }

    local_score: Dict[str, float] = {}
    local_parts: Dict[str, Dict[str, float]] = {}
    for node_id, feat in nodes.items():
        s_i, parts = compute_local_score(feat)
        local_score[node_id] = s_i
        local_parts[node_id] = parts

```

```
    priority_out: Dict[str, Dict[str, Any]] = {}
    for node_id in nodes.keys():
        n1_nodes, n1_time, n2_nodes, n2_time = build_hop_sets_and_times(
            node_id=node_id,
            weighted_adj=weighted_adj,
            top_k_neighbors=top_k_neighbors,
        )

        n1_top = top_k_by_time(n1_nodes, n1_time, top_k_neighbors)
        n2_top = top_k_by_time(n2_nodes, n2_time, top_k_neighbors)

        n1_avg = avg(local_score[n] for n in n1_top)
        n2_avg = avg(local_score[n] for n in n2_top)

        w_self, w_n1, w_n2 = _normalized_layer_weights(bool(n1_top), bool(n2_top))
        p_i = w_self * local_score[node_id] + w_n1 * n1_avg + w_n2 * n2_avg

        priority_out[node_id] = {
            "local_score_S_i": local_score[node_id],
            "priority_P_i": p_i,
            **local_parts[node_id],
            "n1_used_count": len(n1_top),
            "n2_used_count": len(n2_top),
            "n1_used_nodes": n1_top,
            "n2_used_nodes": n2_top,
            "n1_avg_S": n1_avg,
            "n2_avg_S": n2_avg,
            "w_self": w_self,
            "w_n1": w_n1,
            "w_n2": w_n2,
        }

    return {
        "meta": {
            "method": "local_score_plus_neighbors",
            "formula_local": "S_i=(1.00*sum_{t in animals_present(i)}w(t)+0.50*WA_i+0.50*PL_i)*(1-0.50*COV_i)",
            "formula_priority": "P_i=w0*S_i+w1*avg(N1)+w2*avg(FWD_topK), raw(h)=exp(-alpha*h), h in {0,1,2}, then normalize",
            "neighbor_scope": "N1=direct neighbors; FWD_topK=up to top_k shortest-path-time nodes excluding self and N1",
            "top_k_neighbors": int(top_k_neighbors),
            "hop_decay_alpha": HOP_DECAY_ALPHA,
        },
        "node_priority": priority_out,
    }
```