Optimal Route Planning for Orienteering: Branch-and-Cut with Terrain Cost Surfaces and Fatigue

Pablo Borrego Ramos[a,*]

[a] *Independent Researcher, Cáceres, 10003, Spain*

Abstract

We address the problem of optimal route planning for competitive orienteering on real terrain. A Geographic Information System (GIS) pipeline transforms orienteering map data and digital terrain models into a fully asymmetric cost matrix that captures directional slope costs (via the Minetti metabolic model) and cumulative athlete fatigue. The resulting problem, the Asymmetric Orienteering Problem with Fatigue (AOPF), is a new variant of the Orienteering Problem in which travel cost depends on both direction and elapsed effort. We propose two mixed-integer linear programming formulations: a Miller–Tucker–Zemlin (MTZ) model with McCormick linearisation and a single-commodity flow model where the fatigue budget is natively linear and a Branch-and-Cut solver strengthened with tailored valid inequalities, including a novel family based on the assignment relaxation. We validate the framework on two Spanish terrains (14 instances) and release a benchmark suite of 21 synthetic instances. The flow-based solver certifies global optimality on 9 of 14 real-terrain instances within 15 minutes, reducing the mean optimality gap from 16.0% to 5.4% compared to the MTZ formulation.

*Corresponding author.

Email address: `paborrego@alumnos.unex.es` (Pablo Borrego Ramos)

Keywords: Branch-and-cut, Orienteering problem, Valid inequalities, Asymmetric routing, Terrain cost surface

1. Introduction

The Orienteering Problem (OP) asks a competitor to select a subset of control points and determine a route starting and ending at a depot, maximising the total score collected while respecting a travel budget. Formally introduced by Tsiligirides (1984), the OP is NP-hard and has been studied extensively in operations research. However, most formulations assume symmetric, Euclidean distances, which poorly approximate real-world orienteering where terrain, vegetation, slope direction, and fatigue create highly asymmetric travel costs.

This paper bridges two research streams: (i) GIS-based cost surface modelling for orienteering maps, following Albert and Sárközy (2021), and (ii) exact optimisation of the OP via integer programming. The terrain realities captured by the cost pipeline—directional slope costs, surface heterogeneity, and cumulative fatigue—produce a problem structure not addressed by existing OP formulations, motivating both a new problem variant and tailored solution methods. Our contributions are:

1. A terrain cost model combining IOF map symbols (ACR), DEM-derived morphometry (HCR), directional metabolic cost (Minetti), and cumulative fatigue into a single asymmetric cost matrix.
2. An IQR-based scaling method for merging ACR and HCR that adapts to terrain characteristics without manual calibration.
3. The Asymmetric Orienteering Problem with Fatigue (AOPF), a problem variant arising naturally from the terrain pipeline, formalising the

OP with directed travel costs and a cumulative fatigue multiplier. To the best of our knowledge, this is the first exact formulation for the OP with position-dependent fatigue costs.

4. Two LP formulations for the AOPF: an MTZ-based model with McCormick linearisation and a flow formulation where the fatigue budget is natively linear.
5. Valid inequalities tailored to the AOPF: tightened flow-arc coupling, fatigue-aware lifted covers, routing infeasibility cuts (a novel family based on the assignment relaxation), and directed adaptations of the cycle cover and path inequalities of Fischetti et al. (1998).
6. A parameterised benchmark suite of 21 synthetic instances for the asymmetric OP with fatigue, varying node count, asymmetry, fatigue rate, and budget tightness—the first benchmark for this problem variant.
7. Computational experiments on two real terrains comparing the MTZ and flow formulations, and an ablation study on the benchmark suite quantifying the marginal contribution of each cut family.

The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 describes the cost surface model. Section 4 presents the mathematical formulation. Section 5 details the solution methods. Section 6 outlines the computational pipeline. Section 7 describes the experimental setup, and section 8 presents the results, including a synthetic benchmark suite and ablation study followed by real-terrain experiments. Section 9 discusses findings and limitations, and section 10 concludes.

2. Related Work

The Orienteering Problem. The OP was introduced by Tsiligirides (1984); Vansteenwegen et al. (2011) and Gunawan et al. (2016) survey exact methods, heuristics, and variants. Exact approaches typically employ branch-and-cut (Fischetti et al., 1998; Gendreau et al., 1998), but most formulations assume symmetric Euclidean distances. Kobeaga et al. (2021) proposed a revisited B&C for large-scale symmetric instances, whose component evaluation methodology informs our ablation study. In this work, we introduce a new problem variant—the Asymmetric Orienteering Problem with Fatigue (AOPF)—derived from real terrain data, and develop a tailored Branch-and-Cut solver with novel valid inequalities. Our formulation addresses asymmetric costs and cumulative fatigue, which preclude undirected comb inequalities and require the flow-based linearisation developed here.

Terrain cost modelling and asymmetric routing. Tobler (1993) proposed the hiking function relating slope to speed; Minetti et al. (2002) measured metabolic cost across slopes via a fifth-degree polynomial that better captures ascent and descent energetics. Albert and Sárközy (2021) introduced a GIS-based approach to OP route planning using the Hypsometry Cost Raster (HCR); we extend their work by replacing Tobler's function with the Minetti model, proposing IQR-based raster fusion, and coupling the cost surface with an exact solver. The ATSP polyhedral theory (Fischetti, 1991; Fischetti and Toth, 1997) underpins our directed SECs. The fatigue model makes arc costs state-dependent, analogous to load-dependent (Bektaş and Laporte, 2011) and time-dependent VRP (Malandraki and Daskin, 1992); to the best of our knowledge, no prior work addresses position-dependent cumulative fatigue with exact methods.

3. Cost Surface Model

Let the terrain be discretised into a regular grid of cells with resolution δ (metres per cell), where each cell is indexed by its row r and column c. Each cell (r, c) is assigned a base traversal cost that integrates map-derived and DEM-derived information.

3.1. Areal Cost Raster (ACR)

The ACR assigns a dimensionless weight $w_{rc} \in (0, 10]$ to each cell based on the IOF map symbol classification. Following Albert and Sárközy (2021), each symbol category maps to a running speed, and the weight is inversely proportional:

$$w_{rc} = \frac{\delta}{v_{\text{base}} \cdot s_{\text{cat}(r,c)}} \tag{1}$$

where $v_{\text{base}} = 2.5\,\text{m/s}$ is the reference flat-terrain speed, s_{cat} is the speed factor for the symbol category (e.g., $s = 1.0$ for paved road, $s = 0.133$ for fight vegetation), and δ is the cell size. Impassable features (water, cliffs, buildings) receive $w = 10$. The ACR is rasterised from the orienteering map file (OMAP format) by rendering symbol objects onto the UTM grid.

3.2. Hypsometry Cost Raster (HCR)

The HCR captures terrain morphometry not represented by map symbols, derived from a Digital Terrain Model (DTM). Following Albert and Sárközy (2021), three morphometric variables are computed from the DTM: the Terrain Ruggedness Index (TRI, standard deviation of elevation in a 3×3 neighbourhood (Riley et al., 1999)), the absolute Plan Curvature (PC, detecting ridges and valleys (Zevenbergen and Thorne, 1987)), and the Slope Magnitude (SLO, in degrees). Each variable is normalised to $[0.1, 1.0]$ via robust percentile clipping at the 1st and 99th percentiles. The HCR is then:

$$\mathrm{HCR}_{rc} = \widehat{\mathrm{TRI}}_{rc}^{\,\alpha} \cdot |\widehat{\mathrm{PC}}_{rc}|^{\,\beta} \cdot \widehat{\mathrm{SLO}}_{rc}^{\,\gamma} \tag{2}$$

with $\alpha = 1$, $\beta = 0.5$, $\gamma = 1$ as specified by Albert and Sárközy (2021).

3.3. IQR-Based Raster Fusion

To combine ACR and HCR, the HCR must be scaled into ACR's numerical space. Albert and Sárközy (2021) proposed histogram mode matching, which is sensitive to distribution shape. We propose an Interquartile Range (IQR) ratio, selected over five alternatives (median, mean, standard deviation ratios, percentile matching, quantile mapping) for its robustness to outliers and automatic adaptation to terrain characteristics ($c = 0.87$ for Torremocha, $c = 2.18$ for La Muela):

$$c = \frac{\mathrm{IQR(ACR)}}{\mathrm{IQR(HCR)}} = \frac{Q_{75}^{\mathrm{ACR}} - Q_{25}^{\mathrm{ACR}}}{Q_{75}^{\mathrm{HCR}} - Q_{25}^{\mathrm{HCR}}} \tag{3}$$

The scaled HCR is added to ACR with path protection:

$$\mathrm{Cost}_{rc} = w_{rc} + c \cdot \mathrm{HCR}_{rc} \cdot \pi_{rc} \tag{4}$$

where π_{rc} attenuates HCR on paths/roads ($\pi = 0.1$ for $w_{rc} < 0.15$), tracks ($\pi = 0.3$ for $0.15 \leq w_{rc} < 0.25$), and impassable features ($\pi = 0.0$ for $w_{rc} \geq 5.0$); all other terrain receives $\pi = 1.0$.

3.4. Directional Metabolic Cost (Minetti Model)

The metabolic cost of locomotion on a slope gradient i (rise/run) is modelled by the fifth-degree polynomial of Minetti et al. (2002):

$$C(i) = 280.5\, i^5 - 58.7\, i^4 - 76.8\, i^3 + 51.9\, i^2 + 19.6\, i + 2.5 \quad [\mathrm{J/kg/m}] \tag{5}$$

valid for $i \in [-0.45, +0.45]$ (slopes outside this range are clamped to ± 0.45), with cost multiplier $\phi(i) = \max(C(i), 0.5)/C(0)$. The traversal cost of a Dijkstra step between adjacent cells is:

$$d_{12} = \|\Delta\| \cdot \frac{\mathrm{Cost}_{r_1 c_1} + \mathrm{Cost}_{r_2 c_2}}{2} \cdot \phi(i_{12}) \tag{6}$$

where i_{12} is the signed slope gradient along the direction of travel (computed by projecting the DEM-derived gradient onto the movement vector), Cost_{rc} is from eq. (4), and $\|\Delta\|$ is the Euclidean step length. Since $\phi(i) \neq \phi(-i)$ for non-zero slopes, this produces asymmetric traversal costs.

3.5. Asymmetric Cost Matrix Construction

Given n control points plus the depot (node 0), the pairwise cost matrix $\mathbf{C} \in \mathbb{R}^{(n+1)\times(n+1)}$ is computed by running $n+1$ single-source anisotropic Dijkstra computations on the combined cost grid with Minetti slope factors, using eq. (6) as edge weights. The grid is optionally downsampled by factor s, reducing complexity to $O(n \frac{HW}{s^2} \log \frac{HW}{s^2})$. Impassable cells (Cost ≥ 9.0) act as hard barriers. The resulting matrix satisfies $C_{ij} \neq C_{ji}$ in general, with asymmetry $\mathcal{A} = \mathrm{mean}_{i\neq j}|C_{ij} - C_{ji}|/\mathrm{mean}_{i\neq j} C_{ij}$.

3.6. Cumulative Fatigue Model

Athlete performance degrades over time. We model fatigue as a linear multiplier on travel cost:

$$f(t) = 1 + \lambda \cdot \frac{t}{B} \tag{7}$$

where t is the elapsed base travel time, B is the raw time budget, and λ is the fatigue rate (calibrated in section 7).

The effective (fatigue-adjusted) cost of traversing arc (i, j) after elapsed time t_i is:

$$C^f_{ij}(t_i) = C_{ij} \cdot \left(1 + \lambda \cdot \frac{t_i}{B}\right) \tag{8}$$

The total fatigue-adjusted route cost for a route $\sigma = (0, v_1, v_2, \ldots, v_k, 0)$ is:

$$D^f(\sigma) = \sum_{\ell=0}^{k} C_{v_\ell, v_{\ell+1}} \cdot \left(1 + \lambda \cdot \frac{T_\ell}{B}\right) \tag{9}$$

where $T_\ell = \sum_{m=0}^{\ell-1} C_{v_m, v_{m+1}}$ is the cumulative base cost up to leg ℓ, and $v_0 = v_{k+1} = 0$ (depot).

4. Mathematical Formulation

We formulate the AOPF as a mixed-integer linear program. The core challenge is the bilinear fatigue term $t_i \cdot x_{ij}$ in the budget constraint; we present an MTZ-based formulation that linearises it via McCormick envelopes and a flow-based formulation where the budget is natively linear.

Let $V = \{0, 1, \ldots, n\}$ be the set of nodes (node 0 is the depot) and $A = \{(i, j) : i, j \in V, i \neq j\}$ the set of directed arcs, with cost matrix C_{ij}, budget B, fatigue rate λ, and node scores $p_i \geq 0$ ($p_0 = 0$) as defined in section 3. Both formulations share arc variables $x_{ij} \in \{0, 1\}$ and node-visit variables $y_i \in \{0, 1\}$, with objective $\max \sum_{i \in V} p_i \, y_i$.

4.1. Shared Constraints

Flow conservation. Each visited node has exactly one incoming and one outgoing arc:

$$\sum_{j \in V \setminus \{i\}} x_{ji} = y_i \qquad \forall\, i \in V \tag{10}$$

$$\sum_{j \in V \setminus \{i\}} x_{ij} = y_i \qquad \forall\, i \in V \tag{11}$$

Depot. The depot is always visited: $y_0 = 1$.

Arc elimination. Arcs that are structurally infeasible are fixed to zero a priori. An arc (i, j) is eliminated if the base cost alone exceeds the budget:

$$x_{ij} = 0 \quad \text{if } C_{ij} + C_{j0} > B \;\text{ or }\; C_{0i} + C_{ij} + C_{j0} > B \tag{12}$$

The base-cost test (12) is applied first as a computationally inexpensive pre-filter (no multiplications); arcs that survive are then tested with the fatigue-aware criterion. Even on the shortest possible path using arc (i, j)—the direct route $0 \to i \to j \to 0$—the fatigue-adjusted cost must not exceed the budget:

$$x_{ij} = 0 \quad \text{if } C_{0i} + C_{ij}\left(1 + \frac{\lambda\, C_{0i}}{B}\right) + C_{j0}\left(1 + \frac{\lambda\,(C_{0i} + C_{ij})}{B}\right) > B \tag{13}$$

4.2. Formulation A: MTZ with McCormick Linearisation

This formulation introduces arrival-time variables $t_i \in [0, B]$ at each node and auxiliary variables $w_{ij} \geq 0$ to linearise the bilinear fatigue term $t_i \cdot x_{ij}$.

Time propagation (MTZ-type). For each arc (i, j) with $j \neq 0$:

$$t_j \geq t_i + C_{ij} - M_{ij}(1 - x_{ij}) \tag{14}$$

where the big-M constant is tightened to:

$$M_{ij} = \max\bigl(B - C_{0i} - C_{j0},\ C_{ij}\bigr) \tag{15}$$

The value $B - C_{0i} - C_{j0}$ is the maximum time that could have elapsed at node i on any feasible route that still uses arc (i, j) and returns to the depot. This is tighter than the standard MTZ constant $M = n$ and reduces the LP relaxation gap.

McCormick linearisation. The fatigue budget involves the bilinear term $t_i \cdot x_{ij}$. We introduce $w_{ij} \approx t_i \cdot x_{ij}$ with McCormick envelope constraints. Let $\bar{w}_{ij} = \min(B - C_{ij} - C_{j0},\ B - C_{i0})$ be the tightest upper bound on w_{ij}:

$$t_i - \bar{w}_{ij}(1 - x_{ij}) \ \leq\ w_{ij}\ \leq\ \min\bigl(\bar{w}_{ij}\, x_{ij},\ t_i\bigr) \tag{16}$$

These force $w_{ij} = t_i$ when $x_{ij} = 1$ and $w_{ij} = 0$ when $x_{ij} = 0$.

Fatigue budget (MTZ).

$$\sum_{(i,j)\in A} C_{ij}\, x_{ij} + \frac{\lambda}{B} \sum_{(i,j)\in A} C_{ij}\, w_{ij} \leq B \tag{17}$$

The McCormick envelope is an outer approximation of the true bilinear set, introducing LP relaxation looseness that grows with the fatigue rate λ.

4.3. Formulation B: Single-Commodity Flow

This formulation replaces the MTZ time variables and McCormick auxiliary variables with flow variables $f_{ij} \geq 0$. We define f_{ij} so that, by construction of the constraints below, $f_{ij} = t_i \cdot x_{ij}$ at any integer-feasible solution. At fractional LP solutions, the flow constraints enforce a tighter relaxation of this product than the McCormick envelope, eliminating the relaxation looseness introduced by Formulation A and making the fatigue budget a natively linear constraint.

Flow-arc coupling. Flow is only permitted on active arcs, with tightened bounds exploiting minimum arrival times:

$$C_{0i} \cdot x_{ij} \leq f_{ij} \leq (B - C_{ij} - C_{j0}) \cdot x_{ij} \qquad \forall\,(i,j) \in A,\; i \neq 0 \tag{18}$$

The lower bound reflects the minimum elapsed cost at node i (direct from depot); the upper bound reflects the maximum that still allows returning to the depot. This reduces the LP relaxation's freedom to assign unrealistic flow values to fractional arcs.

Time flow conservation. At each non-depot node j, incoming flow plus arc cost equals outgoing flow:

$$\sum_{i \in V} \big(f_{ij} + C_{ij}\, x_{ij}\big) = \sum_{k \in V} f_{jk} \qquad \forall\, j \in V \setminus \{0\} \tag{19}$$

with $f_{0j} = 0$ for all j. At integer feasibility, $f_{ij} = t_i \cdot x_{ij}$ holds exactly; at fractional solutions, the flow structure provides a tighter relaxation than the McCormick envelope.

Fatigue budget (flow). Because $f_{ij} = t_i \cdot x_{ij}$ by construction, the fatigue budget is natively linear:

$$\sum_{(i,j)\in A} C_{ij}\, x_{ij} + \frac{\lambda}{B} \sum_{(i,j)\in A} C_{ij}\, f_{ij} \leq B \tag{20}$$

The flow formulation has $O(n^2)$ additional continuous variables (one f_{ij} per arc) compared to $O(n)$ time variables in the MTZ formulation, but produces a tighter LP relaxation by eliminating the McCormick approximation gap.

4.4. Subtour Elimination and Connectivity Cuts

Both formulations use the same cutting planes, added dynamically. Because $C_{ij} \neq C_{ji}$, all cuts use directed arc variables.

Subtour elimination constraints (SECs). For a subset $S \subseteq V \setminus \{0\}$ (identified by the separation routines in section 4.6), the directed SEC prevents cycles disconnected from the depot:

$$\sum_{i\in S} \sum_{j\in S} x_{ij} \leq |S| - 1 \tag{21}$$

Connectivity cuts. These ensure that every visited node can reach and be reached from the depot:

$$\sum_{i\in S} \sum_{j\notin S} x_{ij} \geq y_k \qquad \forall\, k \in S \tag{22}$$

$$\sum_{i\notin S} \sum_{j\in S} x_{ij} \geq y_k \qquad \forall\, k \in S \tag{23}$$

A combined two-way cut further tightens the relaxation:

$$\sum_{i \in S} \sum_{j \notin S} x_{ij} + \sum_{i \notin S} \sum_{j \in S} x_{ij} \geq 2\, y_k \quad \text{for some } k \in S \tag{24}$$

4.5. Budget-Based Cutting Planes

The following four cut families exploit the budget constraint to eliminate LP solutions with excessive routing cost.

(B1) Lifted cover cuts. A subset $\mathcal{C} \subseteq A$ with $\sum_{(i,j)\in\mathcal{C}} C_{ij} > B$ is a *cover*: no feasible solution can use all arcs in $\mathcal{C}$. The cover inequality $\sum_{(i,j)\in\mathcal{C}} x_{ij} \leq |\mathcal{C}| - 1$ is valid and can be strengthened via sequential lifting.

(B2) Routing infeasibility cuts. We introduce inequalities defined purely on node-visit variables y_i. For $Q \subseteq V \setminus \{0\}$, let $C_{\text{assign}}(Q)$ denote the optimal value of the assignment relaxation on the subgraph induced by $Q \cup \{0\}$, i.e., the minimum total base cost of a set of arcs where every node has exactly one incoming and one outgoing arc, computed via the Hungarian algorithm in $O(|Q|^3)$. Since fatigue only increases travel cost, $C_{\text{assign}}(Q)$ is a lower bound on the fatigue-adjusted cost of any cycle through $Q \cup \{0\}$.

Proposition 1 (Routing infeasibility)**.** *For any $Q \subseteq V\setminus\{0\}$ with $C_{assign}(Q) > B$, the inequality*

$$\sum_{i \in Q} y_i \leq |Q| - 1 \tag{25}$$

is satisfied by every integer-feasible solution of the AOPF.

The proof follows by contradiction on budget feasibility (see Supplementary Material). Unlike cover cuts, routing infeasibility cuts constrain y_i directly; the LP cannot circumvent them by redistributing arc flow.

(B3) Directed cycle cover cuts. Adapting Fischetti et al. (1998) to the directed setting: for a directed cycle $F \subset A$ with nodes $V(F)$ whose base traversal cost exceeds B:

Proposition 2 (Directed cycle cover)**.** *For any directed cycle $F \subset A$ whose base cost $\sum_{(i,j)\in F} C_{ij} > B$, the inequality*

$$\sum_{(i,j)\in F} x_{ij} \leq \sum_{v\in V(F)} y_v - 1 \tag{26}$$

is satisfied by every integer-feasible solution of the AOPF.

The proof follows by contradiction: if all arcs in F are used, the route cost exceeds B (see Supplementary Material).

This is stronger than B1 because the right-hand side involves y_v: when $y_v < 1$, fewer arcs are permitted.

(B4) Directed path inequality cuts. Adapting Fischetti et al. (1998): for a directed path $P = (v_1 \to \cdots \to v_k)$ with feasible closing set $W(P) = \{v \in V \setminus V(P) : \text{path } P \text{ can be closed through } v \text{ within budget}\}$: The path inequality states:

$$\sum_{\ell=1}^{k-1} x_{v_\ell,v_{\ell+1}} - \sum_{v\in V(P)} y_v + y_{v_1} + y_{v_k} - \sum_{v\in W(P)} x_{v_k,v} \leq 0 \tag{27}$$

The fatigue model makes $W(P)$ smaller than in the symmetric case, producing tighter cuts on high-fatigue instances.

4.6. Separation Routines

The valid inequalities above are added dynamically during the B&C search. At each node of the search tree, the solver runs a cut loop of up to $K_{\max} = 20$ iterations, attempting to find violated inequalities in the current LP solution (x^*, y^*, f^*). The loop terminates when no violated cuts are found. Below we describe how each cut family is separated.

Subtour and connectivity separation. A directed support graph $G^* = (V^*, A^*)$ is constructed from arcs with $x^*_{ij} > 0.5$. Two complementary procedures identify violated SECs and connectivity cuts:

1. *Kosaraju SCC detection.* Kosaraju's algorithm identifies all strongly connected components of G^* in $O(|V^*|+|A^*|)$ time. Any SCC not containing the depot yields a violated subset S, and all four cut types (eqs. (21) to (24)) are added for S.
2. *Reverse BFS.* A reverse BFS from the depot identifies nodes that can reach the depot in G^*. Any visited node ($y^*_i > 0.5$) not in this reachable set requires an outgoing connectivity cut.

Max-flow connectivity separation. The rounding-based separation above may miss violated cuts when fractional arc values are spread thinly. For each node k with $y^*_k > 0.1$, a max-flow computation (Dinic's algorithm) from depot to k on the fractional support graph (using x^*_{ij} as capacities) identifies violated connectivity cuts when the max-flow value is less than y^*_k.

Budget-based cut separation. The following procedures identify violated instances of the budget-based cuts B1–B4:

- *(B1) Cover cuts.* Arcs with $x^*_{ij} > 0$ are accumulated greedily by decreasing x^*_{ij} until their fatigue-inflated cost exceeds B (arc weights inflated to $C_{ij}(1 + \lambda C_{0i}/B)$). Up to 3 covers are added per iteration.
- *(B2) Routing infeasibility cuts.* Nodes with $y^*_i > 0.5$ are added greedily to a candidate set Q by decreasing y^*_i. When $C_{\text{assign}}(Q) > B$ and

$\sum_{i \in Q} y_i^* > |Q| - 1$, the set is minimised by removing nodes with smallest y_i^* while $C_{\text{assign}}(Q)$ remains above B.

- *(B3)–(B4) Cycle cover and path cuts.* Short directed cycles (length 3–4) and paths (length 2–3) are enumerated in the support graph. Violated instances of (26) and (27) are added when the fatigue-adjusted cost exceeds B.

5. Solution Methods

5.1. Simulated Annealing (SA)

Method selection. The SA metaheuristic was selected after comparing four methods under equal 2-second budgets across 21 benchmark instances (table 1).

Table 1: Heuristic comparison: mean gap below SA (%) across 21 benchmark instances (equal 2 s budget, 10 seeds). All differences significant (Wilcoxon, $p < 0.001$).

Method	Mean gap	Std	Min gap	Max gap
Greedy	23.0%	9.4%	3.7%	40.0%
GA	49.2%	11.0%	24.6%	71.9%
ACO	8.1%	4.9%	1.3%	24.1%
SA	(ref.)	(ref.)	(ref.)	(ref.)

Algorithm. Starting from a greedy insertion solution, the SA applies Remove ($p = 0.30$), Insert ($p = 0.30$), Or-opt on segments of 1–3 nodes ($p = 0.20$), and Swap ($p = 0.20$). Moves violating the fatigue budget are rejected. Parameters ($T_0 = 100$, $T_{\text{end}} = 0.1$, geometric cooling) were tuned via grid search over 1,000 synthetic instances. Iterations and restarts are auto-calibrated: $N_{\text{iter}} = \text{clip}(2500n, 10000, 120000)$, $N_{\text{restart}} = \text{clip}(3000/n, 40, 200)$. The search terminates early on stagnation (no improvement for $N_{\text{iter}}/5$ iterations).

5.2. Branch-and-Cut (B&C)

The B&C solver uses the LP relaxation of the AOPF formulation (section 4), solved via the HiGHS dual simplex (Huangfu and Hall, 2018). HiGHS was chosen as it is open-source, provides a C API suitable for embedding in a custom B&C loop, and its dual simplex implementation supports efficient warm-starting after cut addition. The algorithm proceeds as follows:

Algorithm 1 Branch-and-Cut for AOPF

```
Build root LP relaxation
Warm-start incumbent $z^* \leftarrow$ SA solution value
Push root node onto stack
while stack non-empty and time $< T_{\max}$ do
    Pop node, apply variable fixings
    for $k = 1, \ldots, K_{\max}$ do                    ▷ Cut loop, $K_{\max} = 20$
        Solve LP relaxation
        if $\bar{z} \leq z^* + \varepsilon$ then prune and continue
        end if
        Find violated SECs via Kosaraju SCC
        Find depot-unreachable nodes via reverse BFS
        Find violated max-flow connectivity cuts
        Find violated budget-based cuts (B1–B4, section 4.6)
        if no violations found then break
        end if
        Add all violated cuts to LP
    end for
    if solution is integer-feasible then
        Extract route, validate fatigue feasibility
        if score $> z^*$ then update incumbent
        end if
    else
        Branch on most fractional $y_i$
        Push child nodes (DFS order)
    end if
end while
if stack empty then solution is provably optimal
```

Key implementation details: branching on y_i (eliminating $O(n)$ arcs per fixing), LP deep-copy via `Highs_passLp`, duplicate SEC tracking via hash set, maximum depth $D_{\max} = 15$, and time limit $T_{\max} = 900\,\mathrm{s}$.

5.3. Node Selection Strategy

The B&C algorithm admits two complementary node selection strategies. *Depth-first search* (DFS) descends quickly into promising subtrees, finds integer-feasible improvements early, and prunes aggressively once a strong incumbent is established. *Best-first search* (BFS) processes the node with the best LP upper bound first, minimising the number of nodes examined to close the optimality gap at the cost of slower incumbent discovery. This tradeoff is well studied in the MIP literature (Linderoth and Savelsbergh, 1999).

For the real-terrain instances ($n \leq 40$) we use DFS, and for the benchmark suite (up to $n = 100$) we use BFS. Both configurations share the same formulation, cut families, and warm-start heuristic; only the node selection rule differs.

6. Computational Pipeline

The full pipeline consists of two stages:

Stage 1: Cost Surface Generation (Python).

1. Parse OMAP file → rasterise symbols onto UTM grid → ACR.
2. Load DTM, compute TRI, PC, SLO → normalise → HCR (eq. (2)).
3. Scale HCR via IQR ratio (eq. (3)), merge with ACR (eq. (4)).
4. Compute directional slopes from DTM.
5. Run anisotropic Dijkstra from each node with Minetti factors.
6. Export asymmetric cost matrix, scores, and budget as JSON.

Stage 2: Route Optimisation (C++).

1. Parse JSON input.
2. Run iterated SA $\rightarrow$ warm-start solution.
3. Build LP model with HiGHS, run B&C (algorithm 1).
4. Report solution, optimality status, and timing.

7. Experimental Setup

7.1. Study Areas

Torremocha is a gently undulating terrain in Extremadura, Spain (2.1×2.9 km, 55 m relief, median slope 3.2°, cost asymmetry $\mathcal{A} = 16.1\%$, IQR scaling $c = 0.87$), with moderate forest cover and extensive rock features. La Muela is a mountainous area (2.8×3.2 km, 464 m relief, median slope 15.9°, cost asymmetry $\mathcal{A} = 50.0\%$, IQR scaling $c = 2.18$), with steep slopes, cliffs, and significant elevation variation. Both terrains use 2.0 m resolution DTMs.

7.2. Control Distributions

Seven control-point distributions (standard, clustered, ring, path-biased, elevation-biased, sparse-far, mixed-density) with 25–40 controls each are generated for each terrain to test solver robustness across varying spatial structures. Point values (10–50) are assigned based on terrain difficulty.

7.3. Parameters

Race duration: 1 hour. Fatigue rate: $\lambda = 0.2$. Base speed: 2.5 m/s. Dijkstra downsampling: $s = 8$. B&C time limit: 900 s. SA iterations: auto-calibrated per instance. These values reflect the typical duration of a middle-distance orienteering race. The fatigue rate $\lambda = 0.2$ reflects the approximately doubled degradation rate in orienteering versus road running, accounting for terrain roughness (Voloshina et al., 2015), higher oxygen cost (Creagh and Reilly, 1997), and cognitive load (Millet et al., 2010). The benchmark suite (section 8.1) evaluates sensitivity across $\lambda \in \{0.0, 0.1, 0.2, 0.3\}$.

8. Results

We evaluate the framework on the two real-terrain study areas (Torremocha, 16.1% cost asymmetry; La Muela, 50.0% cost asymmetry) and on the synthetic benchmark suite. For each instance, the SA warm-start solution provides the incumbent; the B&C then attempts to prove optimality or tighten the LP relaxation gap within a 15-minute time limit. Both formulations (MTZ and flow) receive the same SA incumbent. The *gap* reported in each table is the relative difference between the best LP upper bound and the incumbent: gap $= 100 \times (\bar{z} - z^*)/z^*$, where $\bar{z}$ is the LP bound and z^* is the best integer solution found. A gap of 0.0% with "YES" indicates that the B&C has proven the SA solution to be globally optimal.

8.1. Synthetic Benchmark Instances

To the best of our knowledge, no standard benchmark exists for the asymmetric OP with fatigue. We generate a parameterized benchmark suite by varying four dimensions independently from a baseline configuration ($n = 40$, $\alpha = 0.25$, $\lambda = 0.2$, medium budget):

- Node count $n \in \{20, 30, 40, 50, 75, 100\}$
- Asymmetry parameter $\alpha \in \{0.0, 0.1, 0.25, 0.5\}$, controlling directional cost perturbation
- Fatigue rate $\lambda \in \{0.0, 0.1, 0.2, 0.3\}$

- Budget tightness: the budget B is set to 70% (loose), 50% (medium), or 30% (tight) of the nearest-neighbour tour cost

Additionally, four extreme-case instances test boundary conditions. Nodes are placed uniformly in a 1000×1000 m domain with the depot at the centre; asymmetric costs are generated by perturbing Euclidean distances with a directional slope factor proportional to α plus 10% noise. All instances and the solver are publicly released.

8.2. Ablation Study: Effect of Cut Families

To quantify the marginal contribution of each cut family introduced in section 4.6, we evaluate five cumulative configurations of the flow-based B&C solver on the full benchmark suite of 21 instances, following the component evaluation methodology of Kobeaga et al. (2021). Each configuration adds one cut family to the previous one, using the same SA warm start and 900 s time limit. Table 2 reports the mean optimality gap, number of instances proven optimal, and mean solving time for each configuration.

Table 2: Ablation study on the benchmark suite (21 instances, 900 s time limit). Each row adds one cut family to the previous configuration. B&C uses best-first search (see section 5.3). Gap is $100 \times (\bar{z} - z^*)/z^*$.

	Cuts enabled	**Mean gap**	**Max gap**	**Opt.**	**Mean time (s)**
1	Base (SECs + conn. + tight. + fat.)	5.54%	20.32%	6/21	727
2	+ B1 (lifted covers)	5.85%	19.88%	7/21	741
3	+ B2 (routing infeas.)	5.41%	19.69%	7/21	723
4	+ B3 (cycle covers)	5.35%	20.19%	7/21	718
5	+ B4 (path ineq.)	5.42%	19.81%	7/21	718

The base configuration proves 6 of 21 instances optimal with a mean gap of 5.54%. Adding B1 (lifted cover cuts) proves one additional instance but slightly increases the mean gap due to separation overhead. B2 (routing infeasibility cuts) reduces the mean gap to 5.41% without changing the number of optimal solutions. B3 (cycle cover cuts) yields the lowest mean gap (5.35%). B4 (path inequalities) provides no further improvement, with the mean gap increasing slightly to 5.42%.

8.3. Benchmark Results

Both formulations (MTZ and flow) are evaluated on the full benchmark suite under identical conditions (same SA warm start, 900 s time limit).

Table 3: Benchmark results: effect of instance size ($\alpha = 0.25$, $\lambda = 0.2$, medium budget, top panel) and effect of asymmetry and fatigue ($n = 40$, medium budget, bottom panel). B&C uses best-first search (see section 5.3).

			MTZ		**Flow**	
Parameter	**Value**	SA pts	gap	optimal?	gap	optimal?
Size n	20	350	0.0%	**YES**	0.0%	**YES**
	30	540	0.0%	**YES**	0.0%	**YES**
	40	660	18.5%	no	7.5%	no
	50	940	16.5%	no	5.9%	no
	75	1370	19.4%	no	6.5%	no
	100	1840	19.9%	no	7.8%	no
Asymmetry α	0.00	690	15.8%	no	6.2%	no
	0.10	730	14.1%	no	2.8%	no
	0.25	670	20.2%	no	5.9%	no
	0.50	630	22.7%	no	7.5%	no
Fatigue λ	0.00	790	6.3%	no	0.0%	**YES**
	0.10	730	8.5%	no	3.0%	no
	0.20	730	15.4%	no	7.7%	no
	0.30	750	30.5%	no	10.5%	no

The benchmark reveals that the flow formulation consistently produces tighter LP bounds than MTZ, with gap reductions growing with instance size (11.0 pp at $n = 40$ to 12.1 pp at $n = 100$). The fatigue dimension confirms the McCormick looseness: the MTZ gap grows from 6.3% ($\lambda = 0$) to 30.5% ($\lambda = 0.3$), while the flow gap increases only from 0.0% to 10.5%. Across all

21 instances, the flow formulation reduces the mean gap from 16.0% to 5.4% and proves 7 instances optimal versus 4 for MTZ.

Table 4: Benchmark results: budget tightness and extreme cases (n = 40 unless noted). B&C uses best-first search (see section 5.3).

		MTZ		**Flow**	
Instance	SA pts	gap	opt?	gap	opt?
Loose budget	960	18.2%	no	8.0%	no
Medium budget	680	25.5%	no	5.3%	no
Tight budget	450	0.0%	**YES**	0.0%	**YES**
Sym. easy (n=20, α=0, λ=0)	470	0.0%	**YES**	0.0%	**YES**
Asym. hard (n=100, α=.5, λ=.3)	1160	42.0%	no	10.5%	no
High asym., no fat. (α=.5, λ=0)	970	9.1%	no	1.1%	no
Sym., high fat. (α=0, λ=.3)	850	24.9%	no	9.9%	no

The extreme cases highlight the flow formulation's strengths: the hardest instance improves from 42.0% to 10.5%, and the no-fatigue instance ($\lambda = 0$) is newly proven optimal by the flow formulation.

Proposition 3 (Optimality certificate)**.** *If the B&C algorithm terminates with an empty node stack, the incumbent z^*_{SA} is the globally optimal value of the AOPF.*

This follows from the standard B&C correctness argument: every node is either pruned by bound, infeasible, or resolved to an integer solution, and the union of all subtrees covers the entire feasible region (see Supplementary Material for the full proof).

Based on the benchmark analysis, we apply both formulations to the real-terrain instances using DFS node selection (**??**

8.4. Torremocha Results

Table 5: Torremocha results: SA and B&C under both formulations (IQR scaling $c = 0.87$, cost asymmetry $\mathcal{A} = 16.1\%$). B&C uses depth-first search (see section 5.3).

		SA		**MTZ**		**Flow**	
Distribution	ctrls	pts	visited	gap	optimal?	gap	optimal?
Standard	40	940	25	14.1%	no	12.2%	no
Clustered	35	510	29	0.0%	**YES**	0.0%	**YES**
Ring	30	650	26	0.0%	**YES**	0.0%	**YES**
Path-biased	35	630	30	0.0%	**YES**	0.0%	**YES**
Elev-biased	35	380	29	7.4%	no	6.3%	no
Sparse-far	25	660	19	0.0%	**YES**	0.0%	**YES**
Mixed-density	40	680	32	8.6%	no	7.1%	no

The flow formulation proves 4 of 7 instances optimal, matching MTZ (4/7) on the same subset (Clustered, Ring, Path-biased, Sparse-far), with tighter gaps on the unproven instances: Standard (14.1% $\rightarrow$ 12.2%), Elev-biased (7.4% $\rightarrow$ 6.3%), and Mixed-density (8.6% $\rightarrow$ 7.1%).

8.5. La Muela Results

Table 6: La Muela results: SA and B&C under both formulations (IQR scaling $c = 2.18$, cost asymmetry $\mathcal{A} = 50.0\%$). B&C uses depth-first search (see section 5.3).

		SA		**MTZ**		**Flow**	
Distribution	ctrls	pts	visited	gap	optimal?	gap	optimal?
Standard	31	610	14	0.0%	**YES**	0.0%	**YES**
Clustered	35	800	24	0.0%	**YES**	0.0%	**YES**
Ring	30	470	21	6.3%	no	0.0%	**YES**
Path-biased	35	570	17	0.0%	**YES**	0.0%	**YES**
Elev-biased	35	730	22	17.0%	no	12.4%	no
Sparse-far	25	600	13	0.0%	**YES**	0.0%	**YES**
Mixed-density	40	680	25	14.6%	no	8.6%	no

The flow formulation proves 5 of 7 instances optimal versus 4 for MTZ, additionally proving Ring optimal (6.3% $\rightarrow$ 0.0%), and reduces gaps on the remaining instances: Elev-biased (17.0% $\rightarrow$ 12.4%) and Mixed-density (14.6% $\rightarrow$ 8.6%). The harder terrain (464 m relief, 50% cost asymmetry) produces larger gaps overall, but the flow formulation's advantage is consistent across both terrains. Across both terrains (14 instances total), the flow formulation proves 9 of 14 instances optimal (vs. 8 for MTZ) and reduces the mean gap on unproven instances from 11.3% to 9.3%.

9. Discussion

SA effectiveness. On all 9 instances where the B&C proves optimality, the incumbent at termination is the original SA warm-start solution, confirming SA is highly effective for OP instances of this size ($n \leq 40$) (Vansteenwegen et al., 2011).

Fatigue linearisation. The comparison between the MTZ and flow formulations confirms that the McCormick envelope is the primary source of LP relaxation looseness in the MTZ approach. The flow formulation, combined with max-flow separation and reliability branching, reduces the mean benchmark gap from 16.0% to 5.4% and proves 7 of 21 instances optimal. On real-terrain instances, the flow formulation reduces the La Muela Mixed-density gap from 14.6% (MTZ) to 8.6%, demonstrating the practical impact of this tighter relaxation.

Computational complexity and runtime. Cost matrix construction runs n anisotropic Dijkstra passes on a $(H/s)(W/s)$ grid, giving $O\big(nHWs^{-2}\log(HWs^{-2})\big)$; with Numba JIT and $s = 8$, this completes in under 4 s for 41 nodes. SA runs in $O(N_{\text{restart}} \cdot N_{\text{iter}} \cdot n)$ and completes in under 1 s for all tested instances. The B&C has worst-case exponential complexity; in practice, runtime varies dramatically with instance structure (e.g., 14 s for a 24-node clustered instance vs. 900 s for a 31-node dense instance), motivating the SA warm start for aggressive early pruning.

Limitations. The framework has not been validated against GPS tracks from real orienteers. The fatigue model is linear; real fatigue may follow a more complex curve. The B&C does not scale well beyond ∼50 nodes; for larger instances, the SA solution should be treated as a heuristic. Dijkstra downsampling ($s = 8$) introduces path approximation error ($< 2\%$ median deviation vs. $s = 4$).

10. Conclusion

We presented an integrated framework for optimal route planning in orienteering that bridges GIS-based terrain cost modelling with exact combinatorial optimisation. The main contributions are the AOPF problem formulation with cumulative fatigue, a flow-based B&C formulation that eliminates the McCormick relaxation gap, novel routing infeasibility cuts based on the assignment relaxation, and a 21-instance benchmark suite. The flow formulation proves 9 of 14 real-terrain instances optimal and reduces the mean benchmark gap from 16.0% to 5.4%. Future work includes validation against real GPS data, extension to multi-day events, and weather-dependent traversability.

Declaration of generative AI and AI-assisted technologies in the manuscript preparation process

During the preparation of this work the author used Kiro (Amazon Web Services) in order to assist with code development, literature search, and manuscript editing. After using this tool, the author reviewed and edited the content as needed and takes full responsibility for the content of the published article.

References

G. Albert and Z. Sárközy. Route planning on orienteering maps with least-cost path analysis. *Proceedings of the ICA*, 4:4, 2021. `doi:10.5194/ica-proc-4-4-2021`.

T. Bektaş and G. Laporte. The pollution-routing problem. *Transportation Research Part B: Methodological*, 45(8):1232–1250, 2011. `doi:10.1016/j.trb.2011.02.004`.

U. Creagh and T. Reilly. Physiological and biomechanical aspects of orienteering. *Sports Medicine*, 24(6):409–418, 1997. `doi:10.2165/00007256-199724060-00005`.

M. Fischetti. Facets of the asymmetric traveling salesman polytope. *Mathematics of Operations Research*, 16(1):42–56, 1991. `doi:10.1287/moor.16.1.42`.

M. Fischetti and P. Toth. A polyhedral approach to the asymmetric traveling salesman problem. *Management Science*, 43(11):1520–1536, 1997. `doi:10.1287/mnsc.43.11.1520`.

M. Fischetti, J. J. Salazar González, and P. Toth. Solving the orienteering problem through branch-and-cut. *INFORMS Journal on Computing*, 10(2):133–148, 1998. `doi:10.1287/ijoc.10.2.133`.

M. Gendreau, G. Laporte, and F. Semet. A branch-and-cut algorithm for the undirected selective travelling salesman problem. *Networks*, 32(4):263–273, 1998. doi:10.1002/(SICI)1097-0037(199812)32:4<263::AID-NET3>3.0.CO;2-Q.

A. Gunawan, H. C. Lau, and P. Vansteenwegen. Orienteering problem: A survey of recent variants, solution approaches and applications. *European Journal of Operational Research*, 255(2):315–332, 2016. doi:10.1016/j.ejor.2016.04.059.

Q. Huangfu and J. A. J. Hall. Parallelizing the dual revised simplex method. *Mathematical Programming Computation*, 10(1):119–142, 2018. doi:10.1007/s12532-017-0130-5.

G. Kobeaga, M. Merino, and J. A. Lozano. A revisited branch-and-cut algorithm for large-scale orienteering problems. *European Journal of Operational Research*, 290(3):865–880, 2021. doi:10.1016/j.ejor.2020.08.040.

J. T. Linderoth and M. W. P. Savelsbergh. A computational study of search strategies for mixed integer programming. *INFORMS Journal on Computing*, 11(2):173–187, 1999. doi:10.1287/ijoc.11.2.173.

C. Malandraki and M. S. Daskin. Time dependent vehicle routing problems: Formulations, properties and heuristic algorithms. *Transportation Science*, 26(3):185–200, 1992. doi:10.1287/trsc.26.3.185.

G. Y. Millet, C. Divert, T. Banizette, and J.-B. Morin. Changes in running pattern due to fatigue and cognitive load in orienteering. *Journal of Sports Sciences*, 28(2):153–160, 2010. doi:10.1080/02640410903406182.

A. E. Minetti, C. Moia, G. S. Roi, D. Susta, and G. Ferretti. Energy cost of walking and running at extreme uphill and downhill slopes. *Journal of Applied Physiology*, 93(3):1039–1046, 2002. doi:10.1152/japplphysiol.01177.2001.

S. J. Riley, S. D. DeGloria, and R. Elliot. A terrain ruggedness index that quantifies topographic heterogeneity. *Intermountain Journal of Sciences*, 5(1–4):23–27, 1999.

W. Tobler. Three presentations on geographical analysis and modeling. Technical Report 93-1, National Center for Geographic Information and Analysis, 1993.

T. Tsiligirides. Heuristic methods applied to orienteering. *Journal of the Operational Research Society*, 35(9):797–809, 1984. `doi:10.1057/jors.1984.162`.

P. Vansteenwegen, W. Souffriau, and D. Van Oudheusden. The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1–10, 2011. `doi:10.1016/j.ejor.2010.03.045`.

A. S. Voloshina, A. D. Kuo, M. A. Daley, and D. P. Ferris. Biomechanics and energetics of running on uneven terrain. *Journal of Experimental Biology*, 218(5):711–719, 2015. `doi:10.1242/jeb.106518`.

L. W. Zevenbergen and C. R. Thorne. Quantitative analysis of land surface topography. *Earth Surface Processes and Landforms*, 12(1):47–56, 1987. `doi:10.1002/esp.3290120107`.

CRediT authorship contribution statement

Pablo Borrego Ramos: Conceptualization, Methodology, Software, Formal analysis, Investigation, Data curation, Writing – original draft, Writing – review & editing, Visualization.

Declaration of competing interest

The author declares that he has no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability

The source code, benchmark instances, and supplementary data are publicly available at `https://github.com/PabloAlmendralejo/Orienteering-Problem`.