TripSnek: A Memetic Algorithm for Expert-Guided Multi-Week European Travel Itinerary Optimization

Daniel R. Tuohy
Independent Researcher
Atlanta, Georgia, USA
danielr2e@gmail.com

Abstract

We present TripSnek, a memetic algorithm for multi-week European itinerary optimization spanning up to 60 nights across 27 countries. The system encodes expert knowledge from a widely recognized authority on European travel (Rick Steves) into a fitness function that models destinations—which can be either major cities or geographic regions—as units with quality ratings, recommended durations, and reward saturation.

A key finding is that *destination-level reward aggregation* induces effective hierarchical search behavior without requiring explicit search space decomposition—expanding the search space from 264 representative cities to a more comprehensive set of 526 yields similar solution quality because the fitness landscape, not the representation, constrains the search. Analysis of 135,030 real user requests validates this primarily expert-curated model: city preferences follow a power law ($\alpha \approx 2$), with the top 100 cities covering 87% of user requirements and 96.85% falling within our 526-city model.

Through ablation studies on 1,000 user trip specifications, we identify components critical to performance: crowding-based replacement, geographically-constrained neighborhoods, and local search operators. The system has been deployed to over 100,000 users, running entirely in-browser with typical completion in 1–2 seconds.

CCS Concepts

• **Computing methodologies → Genetic algorithms**; *Heuristic function construction.*

Keywords

memetic algorithms, travel itinerary optimization, tourist trip design problem, fitness function design

ACM Reference Format:
Daniel R. Tuohy. 2026. TripSnek: A Memetic Algorithm for Expert-Guided Multi-Week European Travel Itinerary Optimization. In *Genetic and Evolutionary Computation Conference (GECCO Companion '26), July 13–17, 2026, San Jose, Costa Rica.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3795101.3805316

1 Introduction

Planning multi-destination European travel presents a complex optimization challenge. Travelers must balance destination quality, transit efficiency, logistical constraints like rental car cross-border penalties, and sustainable pacing—all within limited vacation time. A common approach is to construct "bucket list" itineraries connecting major cities via short-hop flights, but this squanders Europe's exceptional rail infrastructure and carries significant overhead: airport transfers, security delays, and carbon emissions an order of magnitude higher than rail [1]. More fundamentally, flight-heavy itineraries bypass the medieval hilltop towns, landscapes, and regional character that represent much of the continent's appeal. The alternative—ground-based travel connecting diverse destinations—is the optimization problem we address.

This problem is typically formulated as a Tourist Trip Design Problem (TTDP) [3, 8, 11], where individual points of interest (POIs) are selected and sequenced to maximize collected score within a time budget. Standard TTDP formulations derive POI scores from user ratings, which present two fundamental limitations. First, reviews reflect satisfaction relative to expectation rather than absolute value—a hidden gem that exceeds expectations may rate higher than a world-class attraction that merely meets them. Second, reviews cannot capture experiential overlap: the châteaux of the Loire Valley might each deserve high ratings individually, but a traveler who has toured several gains progressively less from the next—not because it is less worthy, but because the architectural styles and furnishings grow familiar. Expert travel guides address both challenges: a knowledgeable guide assesses destinations against a consistent standard, identifies which sights offer similar experiences, and estimates how long it takes to experience a satisfying dose of the most compelling offerings in each area. This last point suggests a natural representation: model destinations with quality ratings and recommended durations that implicitly encode how much there is to do.

We explore this expert-curated representation in TripSnek[1], encoding destination ratings and duration recommendations from Rick Steves [9]—whose guidebooks have sold millions of copies—into a fitness function with reward saturation. TripSnek differs from typical TTDP systems in both the source of reward information and problem scale (multi-week continental itineraries vs. single-day urban tours), though memetic approaches have been applied to TTDP variants at smaller scales [2]. These approaches are complementary: TripSnek determines destinations and durations, while fine-grained TTDP methods could plan daily activities within each.

The most comparable prior formulation is the Vacation Planning Problem (VPP) [12], which addresses regional-scale travel with destination selection and nights allocation. VPP builds destinations bottom-up through algorithmic clustering of POIs; we use top-down expert-defined destinations, reflecting different assumptions about what constitutes reliable scoring.

[1] https://www.tripsnek.com

ACM ISBN 979-8-4007-2488-6/2026/07
https://doi.org/10.1145/3795101.3805316

Contributions. (1) We demonstrate that encoding expert knowledge with *reward aggregation* within destinations induces hierarchical search behavior without requiring explicit decomposition—the fitness landscape constrains the search. (2) Through ablation on 1,000 real-world trip specifications, we identify the components critical to performance and report that crossover provides negligible benefit in this configuration. (3) Analysis of 135,030 user specifications validates the expert-curated model, showing destination preferences follow a power law ($\alpha \approx 2$) with 96.85% coverage. The system has served over 100,000 users.

2 Problem and Representation

Problem. Given a start city, end city, total nights N (up to 60), optional required locations with minimum stays and date locks, transportation mode, and pace preference, the system produces an ordered itinerary maximizing per-night fitness. Hard constraints fix endpoints, enforce exact total nights, and include all required locations. Soft constraints penalize excessive travel time, under-allocation at high-quality destinations, revisiting locations, and cross-border rental car returns. Transportation mode includes a mixed rail/car option: many compelling destinations (remote ruins, Tuscan hill towns, rural coastlines) are poorly served by public transit, making strategic rental car use important. The algorithm determines where to pick up and drop off cars, accounting for cross-border surcharges imposed by rental companies.

Destination model. The system operates on 264 expert-curated *destinations* containing 526 *locations* across 27 countries. A destination may be a major city (Paris, Rome) or a region represented by a home-base city (Nice for the French Riviera). Each destination encodes an expert rating (0–3, corresponding to unrated/worthwhile/try hard to see/don't miss), recommended nights at three pace levels, day-trip suitability, and 17 categorical features enabling preference-based reward adjustment.

Reward aggregation. All locations within a destination share that destination's total reward. A traveler visiting multiple French Riviera cities receives the same reward as one staying at a single base—the fitness function scores "French Riviera: 3 nights" as an atomic decision. Without aggregation, the optimizer exploits the landscape by distributing single nights across adjacent cities within high-rated destinations, producing degenerate itineraries. Figure 1 illustrates this for a 21-night Paris→Nice trip: without aggregation, the algorithm clusters 9 cities in Provence; with aggregation, it produces a geographically diverse route.

Emergent hierarchy. An initially surprising finding is that expanding the search from 264 destination-level home bases to all 526 locations yields similar fitness (0.955 vs. 0.970 normalized), confirming that the aggregated reward structure—not explicit search space restriction—prevents degenerate clustering. The hierarchical behavior emerges from optimization pressure on the fitness landscape rather than being imposed architecturally.

3 Algorithm

TripSnek employs a steady-state memetic algorithm [5, 6] with population size 600 and 75,000 evaluations. An individual encodes an itinerary as an ordered sequence of stops, each specifying a location, number of nights, and rental car flags.

Fitness function. The objective is to maximize a fitness function combining destination rewards with travel penalties, normalized per night (Table 1). This per-night formulation enables comparison across trips of different durations and reflects that travelers seek to maximize quality per unit of limited vacation time. User preferences dynamically adjust rewards: pace preference shifts recommended-night thresholds, while interest categories scale destination ratings based on feature alignment. Coefficients were manually iteratively tuned so that generated itineraries closely resemble those published by the domain expert.

Table 1: Fitness function components. Rewards are positive; penalties negative.

Component	Description
Rewards	
Base visit	Reward for visiting location (scaled by rating and user interest alignment)
Per-night	Full reward up to recommended nights; 0.4× with 10% decay beyond
Day trip	+2.0 (ideal), +1.0 (ok), −1.0 (bad)
Penalties	
Travel time	$0.28 \times (\text{hours} - 0.5)^2$ per day
Shortchanged	−1 to −12 per night below recommendation
One-night stay	−1.0 for locations meriting longer visits
Multi-visit	$-1.5 \times (\text{visits} - 1)$ for revisits
Cross-border car	−25.0 standard; −80.0 for prohibitive pairs
Flight	−8.0 per flight (hassle factor)

Initialization. Individuals are constructed by creating segments between date-locked requirements, inserting non-locked requirements via best-first search, allocating recommended nights, then filling remaining capacity with random locations from a *geographically constrained neighborhood* (5–15 closest locations).

Selection and replacement. Binary tournament selection with an elite of 2. Offspring replace existing individuals using **WAMS (Worst Among Most Similar)** crowding [4]: form k=2 groups of m=3 random individuals, select the most similar to the offspring in each group (by Jaccard overlap of high-value locations), and replace the least fit. This focuses diversity pressure on scarce "don't miss" destinations that dominate fitness.

Mutation. Five operators are applied per-stop: add stop (prob. 0.040), transfer night (0.042), delete stop (0.009), replace stop (0.006), and swap stops (0.005). Critically, add and replace operations select new locations from geographically constrained neighborhoods, preventing incoherent jumps across the continent.

Local search. Two operators applied probabilistically to offspring make this a memetic algorithm: (1) *Night redistribution* (83.1% of offspring) transfers nights from surplus locations to deficit locations, improving alignment with expert duration recommendations. (2) *Adjacent pairwise interchange (API) with rental boundary shift* (67.3% of offspring), a 2-opt variant that reduces travel time and cross-border penalties.

On the memetic framing. Ablation reveals that crossover provides negligible benefit ($d = 0.05$, $p = 0.099$) and is disabled in production, where it consumed 21% of runtime. The population-based

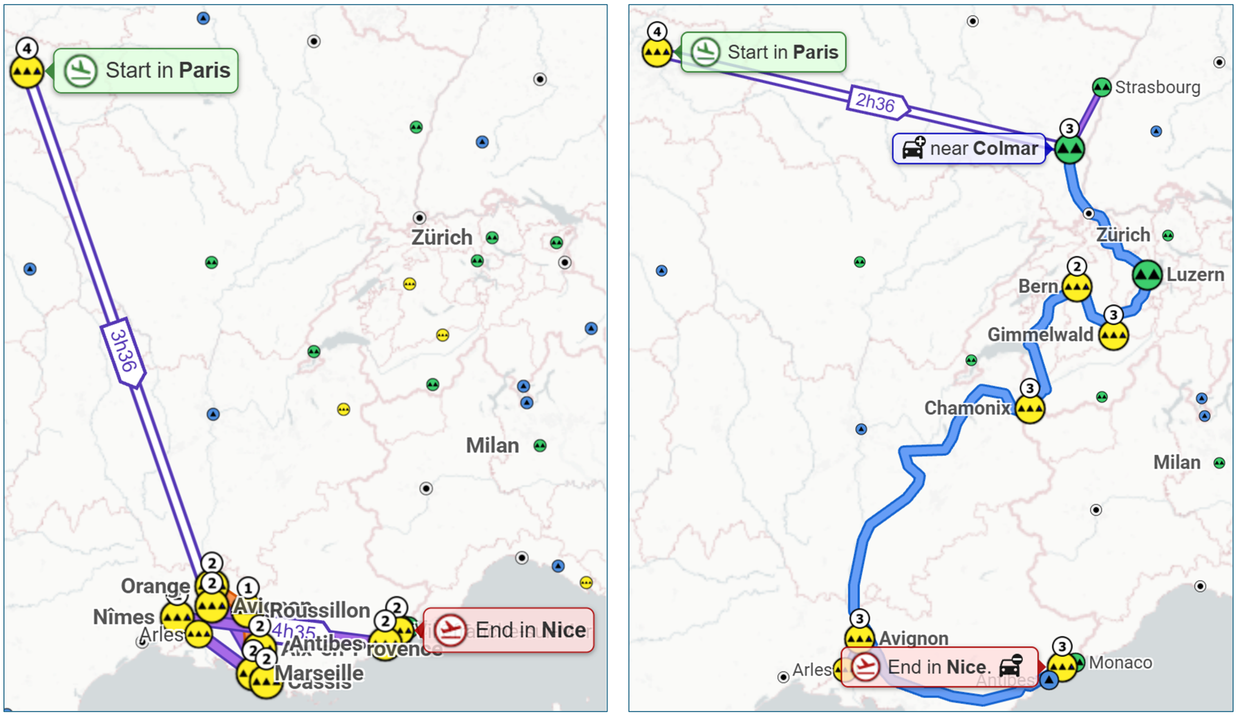

Figure 1: Impact of destination-level reward aggregation (screenshots from the deployed system). A 21-night Paris→Nice trip: without aggregation (left), the optimizer clusters 9 adjacent Provence/Riviera cities; with aggregation (right), it generates a diverse route through Alsace, Switzerland, and the French Alps.

framework nonetheless earns its keep: WAMS crowding—which requires a population to maintain diversity across solution niches—is the single most impactful component ($d = 0.48$). The algorithm is memetic by definition (population-based search with local improvement [5]); that crossover is ineffective in this configuration is an empirical finding consistent with prior work showing crossover effectiveness is highly problem-dependent [7], particularly in the presence of strong local search.

4 Experiments

4.1 Methodology

We sampled 1,000 trip specifications from 135,030 real user itineraries, filtering to 21–60 night trips with at most 10 required cities. Sampling first covered all 257 user-requested destinations (161 selections), then drew 839 uniformly at random, ensuring evaluation across the full diversity of European geography.

Fitness is normalized per-specification: 0 = worst initial population best across experiments; 1 = highest fitness achieved by any experiment during evolution. Each configuration ran for 75,000 evaluations with population 600. We report Cohen's d effect sizes and paired t-test significance across all 1,000 trips, confirmed by Wilcoxon signed-rank tests.

4.2 Ablation Results

Table 2 presents results for the four components with the largest individual impact.

Table 2: Ablation study. Control achieves 0.972 normalized fitness. All differences significant at $p < 0.001$ except API + boundary shift ($p = 0.016$).

Configuration	Score	Δ	d
Control (full system)	0.972	—	—
No WAMS replacement	0.922	−0.050	0.48
Unconstrained neighborhoods	0.919	−0.053	0.49
No night redistribution	0.938	−0.034	0.36
No API + rental boundary shift	0.966	−0.006	0.08
All four disabled	0.635	−0.337	1.44

WAMS replacement and geographically constrained neighborhoods are the most critical individual components, each contributing ≈0.05 to normalized fitness with medium effect sizes. Night redistribution provides a medium effect ($d = 0.36$), while API + boundary shift shows a small but significant effect ($d = 0.08$, $p = 0.016$). Disabling all four reduces fitness to 0.635 ($d = 1.44$)—substantially larger than the sum of individual effects (0.143), indicating significant interaction between components.

We additionally ablated mutation operators and crossover (Table 3). Only **add-stop** shows substantial impact ($d = 0.68$, $p < 0.001$); other mutation operators are partially redundant with local search. **Crossover** shows no significant effect ($p = 0.099$, $d = 0.05$); given that it consumed 21% of runtime, it is disabled in production.

Table 3: Mutation and crossover ablation. Only add-stop is significant ($p < 0.001$, $d = 0.68$); crossover is not ($p = 0.099$).

Ablation	Score	Δ	d
Control	0.937	—	—
No crossover	0.932	−0.005	0.05
No add stop	0.838	−0.099	0.68
No delete stop	0.931	−0.006	0.05
No swap stops	0.937	−0.000	−0.01
No replace stop	0.935	−0.002	0.01
No transfer night	0.929	−0.008	0.07

4.3 Validation of Expert-Curated Coverage

To validate that the expert-curated location set reflects real user interests, we analyzed 727,061 city requirements from 135,030 user specifications. Users can specify any of over 170,000 European locations via autocomplete; 96.85% of requirements match our 526-city model directly. Destination demand follows a power law ($\alpha \approx 2$, $R^2 > 0.97$), as shown in Figure 2: Paris alone accounts for 7% of requirements, the top 10 cities for 40%, and the top 100 for 87%. This concentration, consistent with Zipf-like distributions observed in tourism demand [10], validates the expert-curated approach: a well-modeled set of destinations covers the vast majority of user needs.

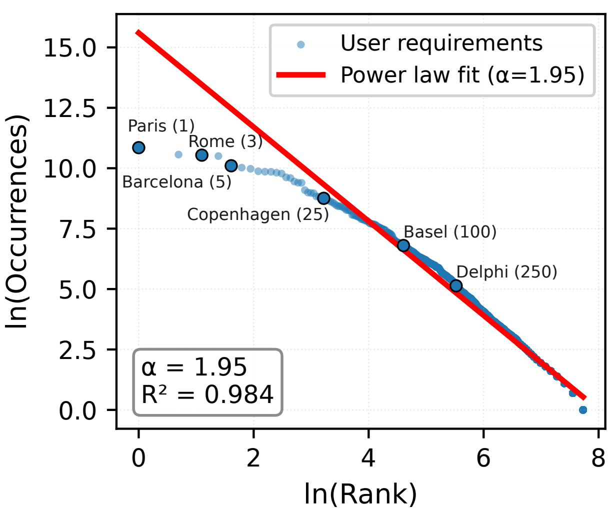

Figure 2: User destination requirements follow a power law with exponent $\alpha \approx 2$ ($R^2 > 0.97$).

5 Discussion and Future Work

The central finding is that reward aggregation within expert-defined destinations induces hierarchical search behavior without explicit decomposition. The representation can include fine-grained locations for travel time accuracy and user flexibility without fragmenting the reward structure; the fitness function, not the encoding, determines effective decision granularity. WAMS replacement maintains diversity specifically among high-value destinations, preventing premature convergence, while geographically constrained neighborhoods ensure mutations produce locally coherent changes reflecting European transportation geography.

This explicit fitness function design offers practical advantages over learned reward functions: hard constraints (required cities, date locks) are enforced by operator design; user preferences directly modulate reward coefficients for predictable adjustments; and when generated itineraries exhibit undesirable characteristics, the fitness components enable root-cause diagnosis and correction.

Limitations and future work. This paper validates internal design decisions through ablation but does not compare against alternative algorithms (simulated annealing, large neighborhood search, greedy heuristics). Fitness coefficients were manually tuned against published expert itineraries without quantitative alignment measures, and we rely on expert guidance as a proxy for itinerary quality without direct user studies measuring traveler satisfaction. Future work will address these gaps through baseline algorithm comparisons, parameter sensitivity analysis, and user validation studies. Results are specific to European travel and may not transfer to regions with different transportation infrastructure.

Acknowledgments

Manuscript preparation was assisted by a large language model for editing and data analysis scripting. All scientific content and claims are the sole responsibility of the author.

References

[1] Jens Borken-Kleefeld, Jan Fuglestvedt, and Terje Berntsen. 2010. Mode, Load, and Specific Climate Impact from Passenger Trips. *Environmental Science & Technology* 44, 9 (2010), 3230–3237.

[2] Ali Divsalar, Pieter Vansteenwegen, Kenneth Sörensen, and Dirk Cattrysse. 2014. A Memetic Algorithm for the Orienteering Problem with Hotel Selection. *European Journal of Operational Research* 237, 1 (2014), 29–49.

[3] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, and Grammati Pantziou. 2014. A Survey on Algorithmic Approaches for Solving Tourist Trip Design Problems. *Journal of Heuristics* 20, 3 (2014), 291–328.

[4] Samir W. Mahfoud. 1995. *Niching Methods for Genetic Algorithms.* Technical Report 95001. University of Illinois at Urbana-Champaign.

[5] Pablo Moscato. 1989. *On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms.* Technical Report C3P Report 826. Caltech Concurrent Computation Program.

[6] Ferrante Neri and Carlos Cotta. 2012. Memetic Algorithms and Memetic Computing Optimization: A Literature Review. *Swarm and Evolutionary Computation* 2 (2012), 1–14.

[7] Krunoslav Puljić and Robert Manger. 2013. Comparison of eight evolutionary crossover operators for the vehicle routing problem. *Mathematical Communications* 18, 2 (2013), 359–375.

[8] José Ruiz-Meza and Jairo R. Montoya-Torres. 2022. A Systematic Literature Review for the Tourist Trip Design Problem. *Operations Research Perspectives* 9 (2022), 100228.

[9] Rick Steves. 2025. Rick Steves' Europe. https://www.ricksteves.com/europe. Destination ratings and duration recommendations accessed 2022–2025.

[10] Mehmet A. Ulubaşoğlu and Bharat R. Hazari. 2004. Zipf's law strikes again: the case of tourism. *Journal of Economic Geography* 4, 4 (2004), 459–472.

[11] Pieter Vansteenwegen and Dirk Van Oudheusden. 2007. The Mobile Tourist Guide: An OR Opportunity. *OR Insight* 20, 3 (2007), 21–27.

[12] Nikolaos Vathis, Charalampos Konstantopoulos, Grammati Pantziou, and Damianos Gavalas. 2023. The Vacation Planning Problem: A Multi-Level Clustering-Based Metaheuristic Approach. *Computers & Operations Research* 150 (2023), 106083.