Ant Colony Optimization applied to the Travelling Santa Problem

FISHER, Elliot and SMITH, Robin <http://orcid.org/0000-0002-9671-8599>

frontiers in Applied Mathematics and Statistics

TYPE Brief Research Report
PUBLISHED 23 July 2026
DOI 10.3389/fams.2026.1773581

Check for updates

OPEN ACCESS

EDITED BY
Frank Werner,
Otto-von-Guericke University, Germany

REVIEWED BY
Mohd Ashraf Ahmad,
Universiti Malaysia Pahang Al-Sultan Abdullah, Malaysia
Emilio Singh,
University of Johannesburg, South Africa

*CORRESPONDENCE
Robin Smith
✉ robin.smith@shu.ac.uk

RECEIVED 22 December 2025
REVISED 16 June 2026
ACCEPTED 29 June 2026
PUBLISHED 23 July 2026

CITATION
Fisher E and Smith R (2026) Ant Colony Optimization applied to the Traveling Santa Problem.
Front. Appl. Math. Stat. 12:1773581.
doi: 10.3389/fams.2026.1773581

Ant Colony Optimization applied to the Traveling Santa Problem

Elliot Fisher[1] and Robin Smith[1,2]*

[1]School of Engineering and Built Environment, Sheffield Hallam University, Sheffield, United Kingdom,
[2]Laboratory for Nuclear Science at Avery Point, University of Connecticut, Groton, CT, United States

The hypothetical global delivery schedule of Santa Claus must follow strict rolling night-time windows that vary with the Earth's rotation and obey an energy budget that depends on payload size and cruising speed. Given the vast number of permutations of city-to-city legs, different payload drop-offs at each location, and the necessity of delivering gifts during the hours of darkness, to minimize energy expenditure, a heuristic-based optimization was applied. To design this schedule, the Traveling-Santa Ant-Colony Optimization (TSaP–ACO) framework was developed. This heuristic framework constructs potential routes via a population of artificial ants that iteratively extend partial paths. Ants make their decisions much like they do in nature, following pheromones left by other ants, but with a degree of permitted exploration. This approach: (i) embeds local darkness feasibility directly into the pheromone heuristic, (ii) seeks to minimize aerodynamic work via a shrinking sleigh cross-sectional area depending on current payload, (iii) uses a low-cost "rogue-ant" reversal to capture direction-sensitive time-zones, and (iv) tunes leg-specific cruise speeds on the fly. On benchmark sets of 15 and 30 capital cities, the TSaP–ACO eliminates all daylight violations and reduces total work by up to 10% compared to a distance-only Ant-Colony Optimization (ACO). In a 40-capital-city stress test, it cuts energy use by 89% and shortens tour length by 60%. Population-first routing emerges naturally from work minimization (50% served by leg 11 of 40). These results demonstrate that rolling-window, energy-aware ACO has potential applications in more realistic global delivery scenarios.

KEYWORDS
Ant-Colony Optimization, energy-aware routing, heuristic optimization, high-performance computing, Traveling Santa Problem, Traveling Salesman Problem

1 Introduction

The classical Traveling Salesman Problem (TSP) asks for the shortest route that visits a set of locations once and returns to the start. Although deceptively simple, it is NP-hard and underpins many real-world logistics tasks [1]. The related Traveling Santa Problem (TSaP) revisits this puzzle with a twist: rather than just minimizing the distance traveled, Santa must deliver gifts worldwide only during local hours of darkness to maintain his air of mystery and anonymity. Each destination has a strict night-time window that varies with latitude and longitude, making the feasibility of a delivery route depend on timing relative to the Earth's rotation. Furthermore, the necessity for a high-efficiency, energy-saving route adds complexity, leading to constraints on flight speed (to minimize aerodynamic drag) and drop-off schedules that maximise early present deliveries (lowering payload). These rolling darkness windows and other constraints convert the purely spatial problem of the TSP into a spatio-temporal routing challenge for the TSaP; distance-only optimizations inevitably lead to unacceptable daylight deliveries and fail to minimize energy use.

Time-window routing is important in transport and logistics across many areas, such as agriculture [2], healthcare [3], and waste collection [4, 5]. As such, this problem has been studied for decades [6], but prior studies typically assume fixed or regionally synchronized windows. However, global delivery schedules are sometimes restricted to sending or receiving goods during typical local working hours. Hence, determining optimal delivery routes that explicitly incorporate a moving delivery time window is important. To meet the net-zero goals of governments around the world, "green" effective route-finding logistics now optimize distance, speed, and load to minimize energy consumption and emissions [7, 8]. Therefore, handling both timing and energy constraints calls for a flexible approach.

Early exact solvers and heuristics for the TSP laid essential groundwork [1]. Incorporating service intervals produces the Vehicle Routing Problem with Time Windows (VRPTW) [9], first formalized by Solomon [6]. Exact branch-and-bound methods solve small VRPTW problems, but meta-heuristics dominate at realistic scales [10, 11]. The majority of the VRPTW research assumes time windows on a single global clock; the rolling night-time constraint of TSaP is largely unexplored. Approaches to NP-hard routing such as the Traveling Santa Problem typically include meta-heuristics like evolutionary algorithms [12] and tabu search [13, 14], deep learning methods such as reinforcement learning [15], and hybrid approaches that combine multiple techniques [16]. These methods aim to quickly find near-optimal solutions to problems that are computationally too complex for exact algorithms to solve.

General-purpose meta-heuristics, genetic algorithms, simulated annealing, and others offer robust VRPTW solutions [17]. Among swarm techniques, Ant-Colony Optimization (ACO) is notable for its distributed search and positive feedback via pheromone trails [18]. The Multiple Ant Colony System extended ACO to VRPTW by separating routing and scheduling colonies [19], yet still assumed synchronous time windows. In the modern scientific literature, there are numerous examples of applying the ACO to vehicle routing problems [20–23].

A key advantage of ACO is its ability to naturally incorporate dynamic decision-making into the construction of a tour. Compared to standard population-based methods such as genetic algorithms, which rely on recombination of complete candidate solutions, ACO incrementally builds routes and can have feasibility checks, such as compliance with darkness windows at each decision step. This is particularly important for the TSaP, where temporal feasibility is not fixed but changes continuously with the Earth's rotation, requiring route construction to remain adaptive rather than globally pre-defined. Pheromone updates provide an efficient mechanism for determining both the spatial and temporal solution quality, allowing ants to implicitly learn favorable trade-offs among distance, timing, and energy expenditure.

In contrast to methods that require explicit scheduling layers or *post-hoc* repair mechanisms, ACO offers a unified framework in which routing, timing, and load-dependent cost considerations can be jointly optimized during solution construction. These properties make it particularly well-suited to the Traveling Santa Problem, where feasibility, energy efficiency, and temporal alignment are tightly coupled. Hence, the ACO was chosen as a potential approach to the Traveling Santa Problem, which is the focus of this study. Only a handful of novelty projects have addressed the TSaP directly, such as Strutz [24], which, on a million-node Christmas tour, minimized distance only, ignoring night-time feasibility. No prior study jointly addresses (i) worldwide darkness windows, (ii) load-dependent energy expenditure, and (iii) a single-vehicle schedule.

This brief research report presents a preliminary ACO framework that constructs night-feasible, energy-aware Santa Claus delivery routes that visit capital cities across the globe. The main unique contributions are: (i) integration of rolling-darkness windows into pheromone-guided search; (ii) a work-based cost function that scales with sleigh load and speed; and (iii) an adaptive cruise-speed model reflecting realistic travel times. The Materials and Methods section formalizes the TSaP and details the ACO framework, including its implementation. The Results section critiques the performance of preliminary optimized routes, explores convergence behavior, and quantifies energy savings. The discussion section further highlights real-world applications of this technique and the advances needed to extend its use to more complex delivery systems. Possible alternative algorithms beyond the ACO are also discussed.

2 Materials and methods

2.1 Formalizing the problem

Santa's delivery task can be expressed as vertices, labeled as $V = \{0, 1, ..., N, 0\}$, where vertex 0 represents Santa's base at the North Pole. and each $i > 0$ is a delivery location. To simplify the problem, each vertex corresponds to a world capital city. Each of these is characterized by geographic coordinates (ϕ_i, λ_i) expressed in degrees, a population P_i, and a sunrise and sunset, in local time, $[t_i^{dusk}, t_i^{dawn}]$, defined on 24–25 December 2025. The night-time windows were shortened by a 15-min buffer. To reduce the computational cost of the problem to a level that can be computed on a laptop, the number of cities selected was capped at 40 (one-fifth of all world capitals) for this scoping study. For low numbers of cities ≤ 30, the number of iterations required to find a viable night-time-only delivery route initially increased exponentially with the number of cities.

A tour (π) is an ordered list of edges that: (i) starts and ends at 0, (ii) visits every city exactly once, and (iii) reaches each city during its darkness window. The Traveling Santa Problem seeks the tour (π) that minimizes energy expenditure subject to the feasibility rules stated above.

2.2 Classic Ant-Colony Optimization

The power of biological systems in addressing path-finding problems is no secret, with well-popularized systems utilizing slime mold-inspired algorithms [25]. With a similar basic philosophy, the ACO framework [26] optimizes a route by simulating tours of a population of artificial ants that iteratively extend partial paths. These digital ants make their decisions much like they do in nature, by following pheromones left previously by other ants, while maintaining a degree of exploration. This is incorporated into the algorithm during the construction of the tour. At the construction step at time t, an individual ant (k) is located at a city (i) with a set $\mathcal{N}_i^{(k)}$ of yet-unvisited cities. The probability of this ant choosing the next city j is calculated as

$$p_{ij}^{(k)}(t) = \frac{\tau_{ij}^{\alpha}(t)\,\eta_{ij}^{\beta}}{\sum\limits_{l\in\mathcal{N}_i^{(k)}} \tau_{il}^{\alpha}(t)\,\eta_{il}^{\beta}}, \tag{1}$$

Here, $\tau_{ij}(t)$ is the pheromone level on path (i,j) at time t and $\eta_{ij} = 1/J_\pi$ is the heuristic, using the inverse of the effective cost function for that part of the journey, J_π. The α and β are powers (> 0) that weight the influence of the pheromone against heuristic information. In the zeroth iteration, equal pheromone levels are applied to all paths, a certain number of ants set out on their journeys, and complete their tours.

In the next iteration, the ant pheromone levels along each possible path are updated to weight the choices of the next set of ants, such that more favorable paths are chosen. This is achieved with two updates.

(1) Uniform evaporation reduces every trail by a factor $(1-\rho)$, where $\rho \in (0, 1)$ is the evaporation rate.

(2) Deposit adds new pheromone in proportion to tour quality.

Combined, these two effects modify the pheromone level as shown in Equation 2 as

$$\tau_{ij}(t+1) = (1-\rho)\,\tau_{ij}(t) \;+\; \sum_{k=1}^{m} \Delta\tau_{ij}^{(k)}(t), \tag{2}$$

with

$$\Delta\tau_{ij}^{(k)}(t) = \begin{cases} Q/J_k, & \text{if edge } (i,j) \text{ belongs to ant } k\text{'s tour,} \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

Here, J_k is the total "cost" incurred during ant k's tour (Equation 4) and Q is a user-defined deposit factor. The best tour found so far receives an extra reinforcement pass, biasing future ants toward high-quality edges. The level at which to weigh preferential edges with pheromone is a delicate balance between following the wisdom gained by earlier ants and allowing further exploration.

2.3 Traveling-Santa implementation

To evaluate a tour, two cost components were introduced into the heuristic and are depicted in Figure 1:

(1) Aerodynamic work W_{ij} incurred while traveling from city i to city j

(2) A daylight penalty Ω that activates whenever the sleigh lands in daylight. This is done by introducing an indicator variable

$$\gamma_{ij} \;=\; \begin{cases} 1, & \text{if arrival at } j \text{ is in daylight,} \\ 0, & \text{otherwise.} \end{cases}$$

The constant $\Omega = 2.1 \times 10^{100}$J was chosen to outweigh any plausible work term, making daylight arrivals hideously unattractive. The total effective tour "energy" is therefore defined as

$$J(\pi) \;=\; \sum_{(i,j)\in\pi} \Big[\, W_{ij} \;+\; \gamma_{ij}\,\Omega \,\Big]. \tag{4}$$

2.3.1 Dynamic cruise speeds

Each ant begins iteration 0 with a default cruise speed $\bar{v}_{\text{default}} = 11,500$ km h^{-1}, which was chosen as a sufficient value to complete a naïve, distance-only world tour in 24 h. At departure time t_{dep} (in UTC), before committing to an edge (i,j), the algorithm checks whether the earliest possible arrival under this speed would violate the local darkness window at city j. The arrival time is computed as

$$t_{\text{arr}}(i,j) = t_{\text{dep}} + \frac{d_{ij}}{\bar{v}_{\text{default}}}.$$

If t_{arr} lies within the hours of darkness at destination j, the ant may proceed as is. However, if t_{arr} lies before dusk or after dawn, a *feasible cruise speed* that just meets the nearest boundary is computed as

$$v_{ij}(t) = \begin{cases} \dfrac{d_{ij}}{t_j^{\text{dusk}}(\text{UTC}) - t_{\text{dep}}}, & t_{\text{dep}} < t_j^{\text{dusk}} \\[2ex] \dfrac{d_{ij}}{t_j^{\text{dawn}}(\text{UTC}) - t_{\text{dep}}}, & t_{\text{dep}} \ge t_j^{\text{dusk}}. \end{cases}$$

To keep the model physically plausible, speeds are restricted to $v_{\min} = 10$ km h^{-1} and $v_{\max} = 15,000$ km h^{-1}. An object will burn up in the atmosphere due to extreme heating from air compression and friction at supersonic speeds, with meteors often vaporizing at 24, 000 km h^{-1} [27]. Therefore, the sleigh's maximum velocity was chosen to be sufficiently below this limit. While even slower speeds above 2, 400 km h^{-1} can ignite flammable materials like clothing, Santa cannot complete his delivery schedule at such low speeds, so it is assumed that his sleigh and reindeer possess some advanced heat shield. If the required speed exceeds $v_{\max}$ or the remaining darkness window is negative, the edge is marked infeasible and receives a costly penalty in the heuristic.

Every 1% of iterations, v_{default} is further refined, since from an aerodynamic work perspective, slower speeds are preferable. The program calculates v_{avg} – the mean cruise speed of the current best tour – and a smoothing update is then applied as $\bar{v}_{\text{new}} = 0.1\, v_{\text{avg}} + 0.9\, \bar{v}_{\text{default}}$. This $\bar{v}_{\text{new}}$ as the reduced default speed is adopted *only* if that tour remains fully night-feasible. This feedback loop tightens travel-time estimates without destabilizing pheromone learning.

2.3.2 Aerodynamic work and payload-driven cross-section

Like in many simplified physics calculations, here Santa's sleigh is modeled as a perfect sphere. The sleigh's energy consumption while traveling between cities (i,j) is modeled as aerodynamic work using the formula:

$$W_{ij} = \tfrac{1}{2}\rho\, A(t)\, v_{ij}^2(t)\, d_{ij},$$

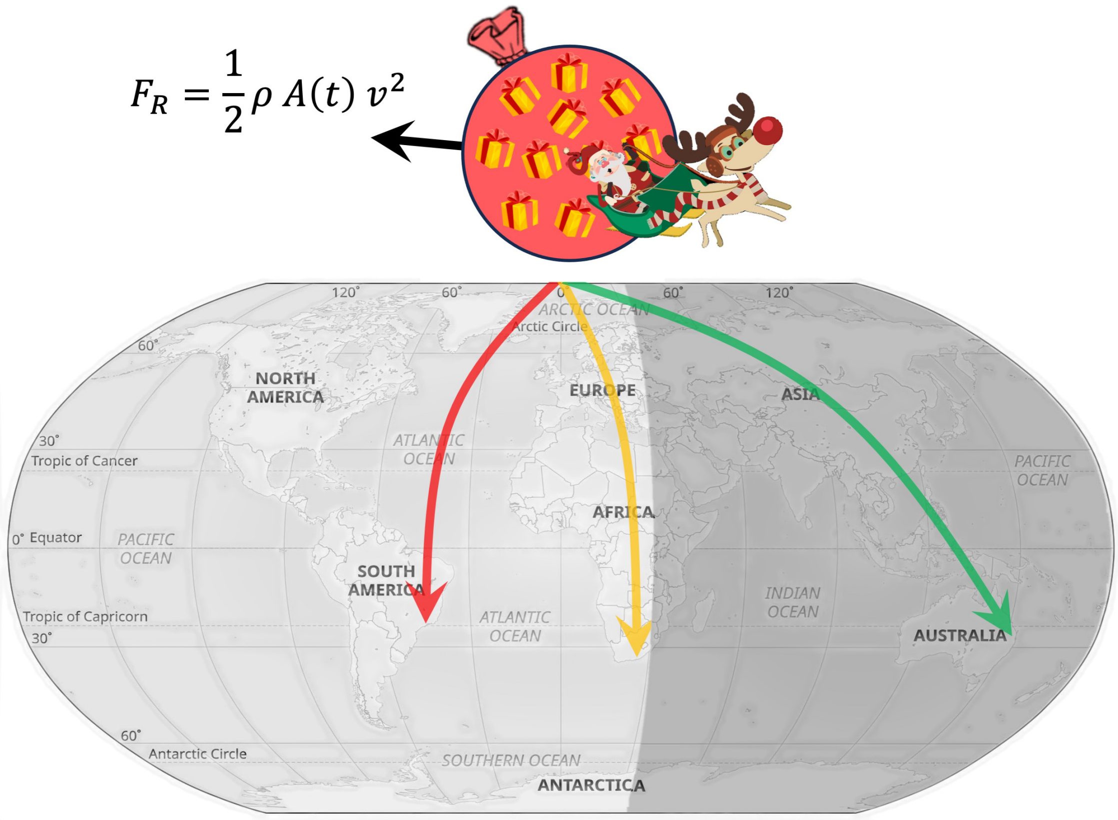

FIGURE 1
A schematic diagram depicting the set-up of the TSaP. In this diagram, Santa, who is currently situated at the North Pole, is assessing three potential cities to visit on his first stop. The green route to Sydney is permitted because the city is in darkness and will remain so when he arrives. The red route to Rio de Janeiro is not permitted and would incur a costly daylight penalty. The amber route to Pretoria is permitted but requires a reasonable change in Santa's speed to ensure arrival during the hours of darkness. The sleigh is modeled as a spherical sack, which shrinks as presents are delivered. This is subject to a resistive aerodynamic drag force F_R, which depends on the air density ρ, sack area $A(t)$, and velocity v. Cartoon image of Santa and Rudolph is taken from reference [42].

where $\rho = 1.225\,\mathrm{kg\,m^{-3}}$ is the density of air at sea level, d_{ij} is the geodesic distance in kilometers, and $v_{ij}(t)$ is the cruise speed selected by the dynamic-speed module, described earlier. The cross-sectional area of Santa's sleigh, $A(t)$, decreases as presents are delivered, reflecting a decreasing payload. As presents are delivered, his sack's cross-sectional area decreases according to the equation

$$A(t) = 0.01 + 0.99\big[1 - \sum_{k\in\text{visited}} P_k/P_{\text{tot}}\big].$$

The P_k/P_{tot} is the fraction of gifts delivered upon visiting a location. The constant term (0.01m^2) prevents the area from collapsing to zero after the final present has been delivered. This work term feeds directly into the ACO heuristic through Equation 4, biasing ants toward energy-efficient segments. It encourages low-velocity routes that serve high-population cities early in the journey, rapidly reducing the payload size and, hence, aerodynamic drag.

2.3.3 Adaptive exploration

The degree of ant exploration balances the exploitation of strong pheromone trails with occasional exploration of random moves. A "ε-greedy" policy selects a random feasible city with probability ε_r or chooses the city with the highest transition probability (Equation 1) otherwise. The pheromone decay schedule maintains exploration early on while encouraging exploitation later [3]. The exploration probability begins at $\varepsilon_0 = 0.45$ and decays as the number of iterations increases. A baseline value of $\varepsilon_{\min} = 0.05$ makes sure that even during later iterations, there is still a chance for some exploration, rather than just following the pheromone trails. To ensure that the exploration probability first meets $\varepsilon_{\min}$ exactly at the final iteration R, the per-step decay factor, d_{tuned} was set to

$$d_{\text{tuned}} = \left(\varepsilon_{\min}/\varepsilon_0\right)^{1/R}.$$

2.3.4 Rogue-ant re-evaluation

The darkness constraint makes the cost surface direction-sensitive; a perfectly feasible tour East-to-West may incur heavy daylight penalties when traveling West-to-East and vice versa. To discover this, each completed route is subjected to a *rogue-ant check*. Once an ant finishes its forward tour π^{fwd} a "rogue" copy instantly computes the reversed sequence $\pi^{\text{rev}} = (0, v_N, \ldots, v_1, 0)$. Both directions are simulated with the same departure time, calculating energies $J(\pi^{\text{fwd}})$ and $J(\pi^{\text{rev}})$ using Equation 4. The direction with the lower cost is retained for pheromone deposit, while the more expensive one is discarded.

Because reversal costs only $O(N)$ time per ant, the extra computational time is negligible relative to the full tour construction. However, the inclusion of the rogue ant was found to lead to a more rapid reduction in daylight penalties in early iterations of pilot tests. Intuitively, the additional "look-behind" step widens the search neighborhood without requiring new construction heuristics. It is especially valuable when night windows are long in one hemisphere and short in the other — a pattern that naturally skews edge desirability by direction.

2.3.5 Ant colony size

Initial smaller-scale calculations showed that although the overall ability for the TSaP-ACO to find a low-energy viable night-time only route was independent of the number of ants chosen, more rapid convergence was observed when the number of ants was set to around 35% of the number of cities, in broad agreement with the literature [28]. Therefore, 14 ants were chosen.

2.3.6 Objective and fitness functions

In this study, the objective function of the TSaP is to minimize the total effective energy cost of the delivery route while ensuring arrivals occur during the local darkness window. For each leg $i \to j$ the base cost is estimated from the aerodynamic work required for travel. This cost is then modified with a small reward for arrivals during darkness and a large penalty for daylight arrivals. This effective cost is used directly as the fitness function in the ACO framework. The heuristic term guiding city selection is defined in Equation 3, meaning routes with lower effective cost correspond to higher fitness. They are therefore more likely to be selected and reinforced through pheromone updates. The main tuned parameters in the algorithm are the pheromone influence α, heuristic influence β, evaporation rate ρ, pheromone deposit factor Q, pheromone bounds, number of ants, and the exploration parameter ϵ.

2.3.7 Model parameters summary

The overall model parameters used in the presented calculations are given in the list below:

- Number of iterations, $R = 60{,}000$
- Colony size, $m = 14$
- Pheromone bias, $\alpha = 3$
- Heuristic bias, $\beta = 2$
- Deposit factor, $Q = 1.0$
- Pheromone limits, $\tau_{\min} = 0.1$, $\tau_{\max} = 10$
- Exploration schedule, $\epsilon = 0.45 \searrow 0.05$
- Cruise-speed bounds, $v_{\min} = 10\ \text{km h}^{-1}$, $v_{\max} = 15{,}000\ \text{km h}^{-1}$
- Daylight penalty, $\Omega = 2.1 \times 10^{100}$ J

3 Results

3.1 Comparative analysis of traditional ACO vs. TSaP

The performance of the full TSaP-ACO approach described in Section 2 was measured against the traditional distance-only ACO approach. The convergence behavior and final energy outcomes of the two algorithms were compared across varying subsets of world capital cities ($N = 15, 30, 45$) during a 3000-iteration testing phase.

Convergence was measured by plotting the total energy expenditure of a route (Equation 4) as a function of the number of iterations. Both approaches appear to converge in approximately 500 iterations for $N = 15$, with incremental improvements becoming negligible thereafter. Notably, the TSaP-ACO achieves lower final energies sooner than the distance-only method, as expected, demonstrating its ability to account for additional constraints such as aerodynamic drag and rolling-darkness windows.

The total energy expenditure was calculated for the traditional distance-only ACO and the full TSaP-ACO approach over 15 trials. The full TSaP-ACO demonstrated not only a lower median energy expenditure but also a narrower distribution.

Convergence curves reveal how each algorithm handles increasing spatial and temporal complexity. For the smallest instance ($N = 15$) the trajectories of the distance-only and full TSaP-ACO heuristics are almost identical until roughly iteration 400. After that point, the full model edges ahead by a few percent and both methods stabilize before iteration 1,000.

A different story is told for $N = 30$. Here, the two curves diverge much earlier, crossing at around iteration 800, and the distance-only run continues to exhibit low-amplitude oscillations well past iteration 2,500. At the same time, the full TSaP-ACO has already stabilized at a lower energy by around 1,200 iterations. These results suggest that the richer pheromone landscape created by the constraints helps the ants to identify high-quality edges sooner, and that this advantage grows with the number of cities.

3.2 Large-scale validation: 40-capital instance

A final stress-test was run on a set of $N = 40$ national capitals drawn from all inhabited continents except Antarctica. To perform a statistical analysis, the optimization was run ten times independently. Figure 2 overlays tour length and total work on a common timeline. Distinct "stairs" are visible, each marking the discovery of a more efficient valid tour. Over the run, the objective falls from $8.08 \pm 0.81 \times 10^{14}$J to $8.86 \pm 0.58 \times 10^{13}$J, (errors quoted as standard deviations over ten measurements), while the tour length shrinks by 60% on average. It is worth noting here that a valid near-optimal tour length is found after 35,000 iterations. However, after that point, energy expenditure continues to fall sharply, further emphasizing that a route's energy efficiency is not determined solely

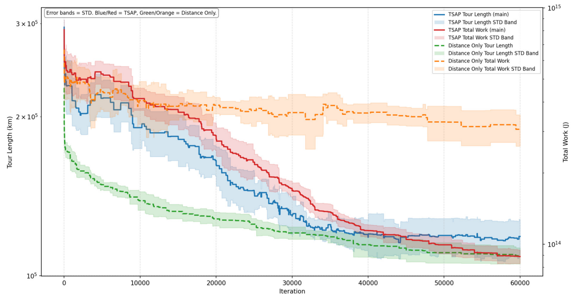

FIGURE 2
The convergence of distance traveled and energy expenditure (work) for the 40-capital-city calculation over 60, 000 iterations. Solid lines show the average convergence behavior of ten calculations for the full TSaP, and the dashed lines show the distance-only optimization. The shaded regions show the error bands defined as the standard deviation of the ten calculations.

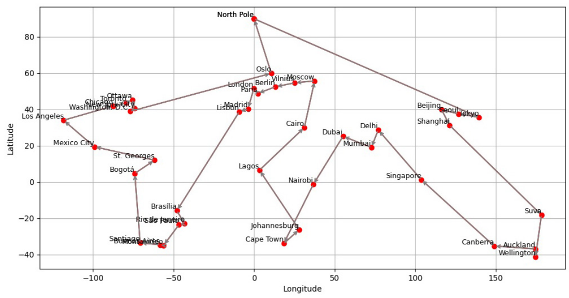

FIGURE 3
The great-circle view of the optimum 40-city route as determined with the full TSaP-ACO after 60, 000 iterations.

by length. This fact is also highlighted by the distance-only ACO optimization results, displayed by the dashed lines in Figure 2. Here, it is seen that although the tour length initially converges more rapidly than in the full TSaP-ACO results, the total work remains high. The work only falls from $6.65 \pm 0.73 \times 10^{14}$J to $3.06 \pm 0.47 \times 10^{14}$J, with the standard deviation remaining high.

The best delivery route is shown in Figure 3. It covers 99, 286 km in 32.86 hours with zero daylight violations. In general, the route tracks East-West, following the Earth's rotation, with some exceptions that appear to facilitate early visits to high-population cities.

An alternative way to visualize the journey is depicted in Figure 4. Here, time runs along the horizontal axis and the index of each stop is on the vertical axis. The night-time windows of each considered capital city are shown by the gray rectangles. Santa's route is shown by the black line and reveals a near-diagonal progression across the gray bars. This demonstrates that the darkness windows line up in a rough chronological order as the leg index increases. It is also worth noting that at the start of Santa's route, he arrives in the early evening, whereas after this he seems to progress into the late night. He begins rounding out his tour in North America, with the final stop in Oslo just before sunrise.

3.3 Load, speed and work distribution

By tracking the per-leg energy expenditure along this route, it was found that the first 10 delivery stops account for the majority of the energy expenditure. This is as expected, as the sack of presents is largest at the beginning of the journey and subject to larger aerodynamic drag. Further analysis of the route confirms the expected "heaviest-first" delivery policy: 50% of the entire population is served by leg 11 of 40. This is beneficial because, when delivering to larger-population cities first, Santa's sleigh will have a smaller cross-sectional area and therefore experience less aerodynamic drag. This further highlights the goal of minimizing energy expenditure rather than distance traveled.

Cruise speeds settle on a tuned ceiling of $\approx 3,800\,km\,h^{-1}$ with a small reduction for the first few legs, $\approx 1,000\,km\,h^{-1}$. The slowest first leg was found such that arrival in Tokyo (the first stop) happened just as the sun was setting.

4 Discussion

The present calculations represent a paradigm shift toward *timing* rather than geography in the modern "green" Traveling-Santa landscape. Placing rolling-darkness windows directly inside the pheromone heuristic removed *all* daylight violations *without* adding any holding patterns. In practical terms, the sleigh never wastes fuel while "circling till nightfall", a weakness that distance-only heuristics possess.

Energy-aware routing delivers a second, possibly more useful benefit. When aerodynamic work, payload decay, and leg-specific cruise-speed tuning are jointly optimized, total work falls significantly while the geometric tour length is cut by two-thirds. These gains are because the model continually trades a small increase in trip time for a larger reduction in drag ($\propto v^2$). The algorithm then uses the saved energy to better align with the moving darkness band due to Earth's rotation.

Despite these successes, Figure 4 indicates that the current solution does not represent a true optimal tour in terms of energy minimization. A better solution would fill a larger portion of each city's darkness window, indicating that Santa is traveling more slowly and thus expending less energy. The presence of underutilized time windows indicates potential inefficiencies, likely stemming from applying a constant default speed across all legs of the journey, changing only when circumstances demand.

To address this, a future study could extend the existing adaptive speed model to dynamically calculate optimal cruise speeds for each leg of the journey. Tailoring speeds to individual legs would facilitate a more efficient distribution of travel times, maximizing the utilization of available darkness windows, thereby reducing overall aerodynamic work and improving energy performance.

Scalability to larger numbers of cities shows promise. Although for small numbers of cities the computational cost initially grows exponentially, the 40-node journey required only 1.67 times as many iterations as the 30-node case to reach a convergence plateau. This supports the idea that richer pheromone landscapes highlight valuable edges more quickly. Unfortunately, the calculation

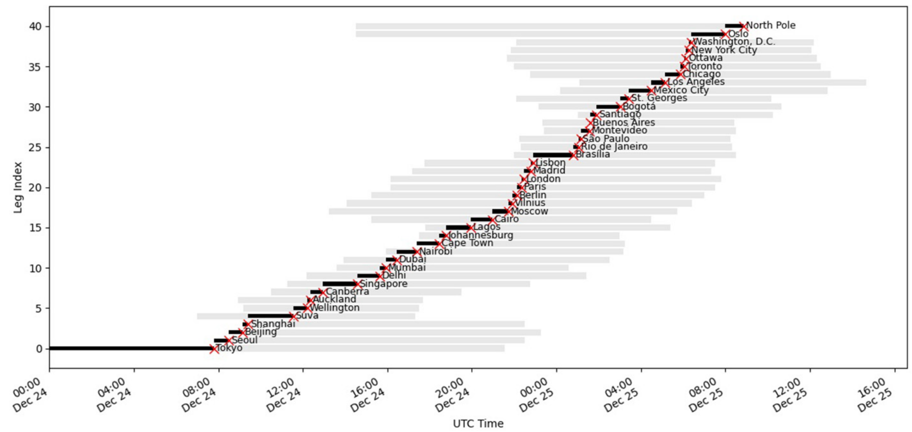

FIGURE 4
The dark window of the optimized 40-city solution as determined with the full TSaP-ACO after 60, 000 iterations.

required to explore routes for the full 195-capital-city delivery was beyond the computational scope of the laptop computer used for this study. However, access to larger computer clusters would resolve this issue. One improvement to scalability would be to use a GPU to run the individual ants in parallel, which, if well-designed, will in turn make the algorithm much faster, allowing even more cities to be added to the framework with relative ease.

The framework adopts a useful *heaviest-first* strategy: half the population is served by leg 11. Deliveries to high-population hubs naturally occur early, rapidly reducing sack size and diminishing drag, thereby enhancing energy efficiency. Finally, the *rogue-ant* reversal proves cost-effective. A single, $O(N)$ reverse-tour check per ant slashes early daylight penalties. It sharpens pheromone clusters along the dominant Europe-Americas corridor, adding little computational cost yet expanding the search neighborhood.

5 Conclusions

This study has demonstrated the applicability of the ACO Algorithm to the Traveling Santa Problem. Although the TSaP is niche, the overall aims of Santa's delivery route have strong synergies with common vehicle routing problems. It is a single-vehicle delivery schedule in which multiple cities must be serviced within strict dynamic time windows while minimizing energy expenditure with a variable payload. Viable night-time-only delivery routes were found for a 40-city itinerary, although scalability to larger numbers of cities is challenging without further computational resources.

The ACO algorithm was applied here based on the experience of the authors with this method, along with the numerous previous examples of it being successfully applied to discrete vehicle routing problems [20–23]. However, the ACO does have some well-documented downfalls [29]. As described in Section 2, the ACO relies on generating a large number of randomized paths and, as demonstrated in Figure 2, finding a globally optimized route within the rigid delivery windows requires a significant run-time, despite the flexibility afforded by the adaptive cruise speeds. As with all implementations of the ACO, there was a risk of converging prematurely on suboptimal routes if the pheromone/exploration weighting was not correctly set, requiring careful consideration and adaptation during set-up.

Other tried-and-tested algorithms applicable to vehicle routing optimization and the Traveling Salesman Problems do exist [30] and could outperform ACO in several areas when applied to the TSaP. The Discrete Marine Predators Algorithm has previously been applied to the classical symmetric TSP [31] and has also been used to develop a more unusual thermal-aware routing scheme for wireless body area networks, demonstrating its applicability beyond simple distance optimization [32]. Marine Predators-based approaches are more robust in avoiding local optima than the ACO, because they leverage a combination of Lévy flights (long, random, unpredictable jumps) to explore the broader search space, and Brownian motion for more localized refinements [33]. They are also effective in multi-objective optimizations and could therefore be useful for more complex TSaP constructions than those currently presented, in which energy expenditure and delivery time could be minimized concurrently.

Furthermore, compact sine-cosine algorithm approaches have successfully tackled vehicle routing problems with time windows [34, 35]. These approaches update a probabilistic model using sine and cosine trigonometric functions to fluctuate between *exploration* (broad search) and *exploitation* (fine-tuning) [36]. These methods typically require less memory and computation time than the ACO because they only store and update a probability vector rather than tracking large numbers of individual paths — a property that would be highly advantageous when attempting to optimize a delivery schedule incorporating more cities.

Game theory-based approaches have previously been applied to route planning in dynamic road networks [37, 38], although the majority of the literature focuses on multi-vehicle systems, unlike the system examined in the present study. In recent years, researchers have explored a hybrid game theory/ACO approach, in which game theory is employed to develop a more strategic and informed selection of the initial population [39]. Intelligently choosing a strong initial delivery route would partially mitigate the downfalls of the ACO mentioned above.

The tunicate swarm algorithm also shows promise for high-dimensional optimization problems [40], especially for dynamic problems where random factors can interfere with scheduling decisions and execution [41]. Such random factors would likely need to be considered in a real-world routing problem, but lie beyond the scope of the simplified TSaP picture presented here.

Clearly, scalability to a very large number of delivery locations is a key criterion for the TSaP, and we encourage a comparative analysis of the convergence behavior of the various algorithms listed above in a future study. The compact sine cosine algorithm approach shows significant promise for extension to larger numbers of delivery locations.

Data availability statement

Source codes, written in python, are available in this data repository: https://github.com/elliotfisher1/Travelling-Santa-Problem/tree/main.

Author contributions

EF: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. RS: Conceptualization, Funding acquisition, Methodology, Project administration, Resources, Supervision, Writing – original draft, Writing – review & editing.

Funding

The author(s) declared that financial support was received for this work and/or its publication. The research was internally funded by Sheffield Hallam University. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) license to any author accepted manuscript version of this paper arising from this submission.

Conflict of interest

The author(s) declared that this work was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The author(s) declared that Generative AI was used in the creation of this manuscript. Generative AI was used very sparingly to correct grammar and simplify sentence structures.

Any alternative text (alt text) provided alongside figures in this article has been generated by Frontiers with the support of artificial intelligence and reasonable efforts have been made to ensure accuracy, including review by the authors wherever possible. If you identify any issues, please contact us.

Publisher's note

References

1. Applegate DL, Bixby RE, Chvátal V, Cook WJ. The traveling salesman problem: a computational study. In: *The Traveling Salesman Problem*. Princeton, NJ: Princeton University Press (2011).

2. Wu D, Wu C. Research on the time-dependent split delivery green vehicle routing problem for fresh agricultural products with multiple time windows. *Agriculture*. (2022) 12:793. doi: 10.3390/agriculture12060793

3. Liu R, Xie X, Augusto V, Rodriguez C. Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care. *Eur J Oper Res*. (2013) 230:475–86. doi: 10.1016/j.ejor.2013.04.044

4. Kim BI, Kim S, Sahoo S. Waste collection vehicle routing problem with time windows. *Comput Operat Res*. (2006) 33:3624–42. doi: 10.1016/j.cor.2005.02.045

5. Singh A, Mittal M, Verma R, Sharma M, Sharma S, Singh G. Optimizing municipal waste collection routes using ant colony optimization. In: *2025 Eighth international conference on image information processing (ICIIP)*. New York, NY: IEEE (2025). p. 595–602. doi: 10.1109/ICIIP68302.2025.11346162

6. Solomon MM. Algorithms for the vehicle routing and scheduling problems with time window constraints. *Oper Res*. (1987) 35:254–65. doi: 10.1287/opre.35.2.254

7. Kara I, Kara BY, Yetis MK. Energy minimizing vehicle routing problem. In: *International conference on combinatorial optimization and applications*. London: Springer (2007). p. 62–71. doi: 10.1007/978-3-540-73556-4_9

8. Ramamoorthy R, Thangavelu M. An enhanced distance and residual energy-based congestion aware ant colony optimization routing for vehicular *ad hoc* networks. *Int J Commun Syst*. (2022) 35:e5179. doi: 10.1002/dac.5179

9. Liu X, Chen YL, Por LY, Ku CS. A systematic literature review of vehicle routing problems with time windows. *Sustainability*. (2023) 15:12004. doi: 10.3390/su151512004

10. Bräysy O, Gendreau M. Vehicle routing problem with time windows, Part II: metaheuristics. *Transp Sci*. (2005) 39:119–39. doi: 10.1287/trsc.1030.0057

11. Pereira VG, Alves-Junior OC, Baldo F. An approach to solve the heterogeneous fixed fleet vehicle routing problem with time window based on adaptive large neighborhood search meta-heuristic. *IEEE Trans Intell Transp Syst*. (2024) 25:8148–57. doi: 10.1109/TITS.2024.3386457

12. Almufti SM, Shaban AA, Ali ZA, Ali RI, Fuente JD. Overview of metaheuristic algorithms. *Polaris Glob J Schol Res Trends*. (2023) 2:10–32. doi: 10.58429/pgjsrt.v2n2a144

13. Barbarosoglu G, Ozgur D, A. tabu search algorithm for the vehicle routing problem. *Comput Operat Res*. (1999) 26:255–70. doi: 10.1016/S0305-0548(98)00047-1

14. Ahmed G, Sheltami T, Yasar A. Optimal path recommendation in dynamic traffic networks using the hybrid Tabu-A* algorithm. *Transp Res Part E: Logist Transp Rev*. (2025) 204:104414. doi: 10.1016/j.tre.2025.104414

15. Szepesvári C. *Algorithms for Reinforcement Learning*. London: Springer Nature (2022).

16. Li P, Hao J, Tang H, Fu X, Zhen Y, Tang K. Bridging evolutionary algorithms and reinforcement learning: a comprehensive survey on hybrid algorithms. *IEEE Trans Evol Comput*. (2024). doi: 10.1109/TEVC.2024.3443913

17. Gendreau M, Tarantilis CD. *Solving Large-Scale Vehicle Routing Problems with Time Windows: The State-of-The-Art*. Montréal: Cirrelt Montreal (2010).

18. Dorigo M, Gambardella LM. Ant colonies for the travelling salesman problem. *Biosystems*. (1997) 43:73–81. doi: 10.1016/S0303-2647(97)01708-5

19. Gambardella LM, Taillard É, Agazzi G. *MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows*. New York, NY: Istituto Dalle Molle Di Studi Sull Intelligenza Artificiale (1999).

20. Ma X, Liu C. Improved ant colony algorithm for the split delivery vehicle routing problem. *Appl Sci*. (2024) 14:5090. doi: 10.3390/app14125090

21. Wang C, Zhang H, Wang J, Zhang L. Ant colony optimization with deep heuristic information for solving electric vehicle routing problems. *IEEE Internet Things J*. (2025) 54436–54449. doi: 10.1109/JIOT.2025.3620353

22. Chen G, Gao J, Chen D. Research on vehicle routing problem with time windows based on improved genetic algorithm and ant colony algorithm. *Electronics*. (2025) 14:647. doi: 10.3390/electronics14040647

23. Ibrahim MF, Mustofa M, Meilanitasari P, Wijaya S. An improved ant colony optimization algorithm for vehicle routing problem with time windows. *Jurnal Teknik Industri*. (2022) 23:105–20. doi: 10.22219/JTIUMM.Vol23.No2.105-120

24. Strutz T. Travelling santa problem: optimization of a million-households tour within one hour. *Front Robot AI*. (2021) 8:652417. doi: 10.3389/frobt.2021.652417

25. Tero A, Kobayashi R, Nakagaki T. A mathematical model for adaptive transport network in path finding by true slime mold. *J Theor Biol*. (2007) 244:553–64. doi: 10.1016/j.jtbi.2006.07.015

26. Blum C. Ant colony optimization: a bibliometric review. *Phys Life Rev*. (2024) 51:87–95. doi: 10.1016/j.plrev.2024.09.014

27. Romig MF. Physics of meteor entry. *AIAA J*. (1965) 3:385–94.

28. Pettersson L, Johansson C. *Ant Colony Optimization - Optimal Number of Ants* (2018). Available online at: https://api.semanticscholar.org/CorpusID:224066194 (Accessed June 2, 2026).

29. Mulani M, Desai VL. Design and implementation issues in ant colony optimization. *Int J Appl Eng Res*. (2018) 13:1–11. doi: 10.1016/j.sca.2023.100056

30. Lähdeaho O, Hilmola OP. An exploration of quantitative models and algorithms for vehicle routing optimization and traveling salesman problems. *Supply Chain Anal*. (2024) 5:100056. doi: 10.1016/j.sca.2023.100056

31. Kumar M, Panwar K, Deep K. Discrete marine predators algorithm for symmetric travelling salesman problem. *Evol Intell*. (2024) 17:3833–48. doi: 10.1007/s12065-024-00960-5

32. Hai T, Zhou J, Masdari M, Marhoon HA. A hybrid marine predator algorithm for thermal-aware routing scheme in wireless body area networks. *J Bionic Eng*. (2023) 20:81–104. doi: 10.1007/s42235-022-00263-4

33. Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH. Marine predators algorithm: a nature-inspired metaheuristic. *Expert Syst Appl*. (2020) 152:113377. doi: 10.1016/j.eswa.2020.113377

34. Pan JS. Yang Qy, Chu SC, Chang KC. Compact sine cosine algorithm applied in vehicle routing problem with time window. *Telecommun Syst*. (2021) 78:609–28. doi: 10.1007/s11235-021-00833-7

35. Pham VHS, Nguyen VN. Cement transport vehicle routing with a hybrid sine cosine optimization algorithm. *Adv Civil Eng*. (2023) 2023:2728039. doi: 10.1155/2023/2728039

36. Mirjalili S. SCA a sine cosine algorithm for solving optimization problems. *Knowl-Based Syst*. (2016) 96:120–33. doi: 10.1016/j.knosys.2015.12.022

37. Belhaiza S. A game theoretic approach for the real-life multiple-criterion vehicle routing problem with multiple time windows. *IEEE Syst J*. (2016) 12:1251–62. doi: 10.1109/JSYST.2016.2601058

38. Zhang D, Zhou Y, Wang J. Game Theory-based approach for massive route planning in dynamic road networks. *IEEE Trans Consumer Electr*. (2024) 71:2131–41. doi: 10.1109/TCE.2024.3449285

39. Jafari FR, Fard ND, Zadeh BS, Ghaffari A. Game theory and ant colony optimization for efficient routing in wireless multi-hop networks. *Comput Netw*. (2025) 275:111902. doi: 10.1016/j.comnet.2025.111902

40. Kaur S, Awasthi LK, Sangal AL, Dhiman G. Tunicate Swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. *Eng Appl Artif Intell*. (2020) 90:103541. doi: 10.1016/j.engappai.2020.103541

41. Si T, Miranda PB, Nandi U, Jana ND, Mallik S, Maulik U, et al. Opposition-based chaotic tunicate swarm algorithms for global optimization. *IEEE Access*. (2024) 12:18168–88. doi: 10.1109/ACCESS.2024.3359587

42. *Cartoon Image of Sleigh* (2016). Available online at: https://www.vexels.com/png-svg/preview/128698/santa-claus-on-sleigh-cartoon (Accessed April 16, 2026).