

Applications of Operations Research in Real-Life Scenarios: A Literature-Based Analytical Study

Topic: Tourist Route Planning (Project Management - PERT/CPM and Network Models)

Author(s) name: **Kaashvi Chandak, Krishna Kongalla, Karm Thakkar, Kapish Bansal, Kavya Mohite**

Course & College: BSc Finance, NMIMS Mumbai



<https://doi.org/10.55041/ijst.v2i10.030>

Cite this Article: Chandak, K., Kongalla, K., Thakkar, K., Bansal, K. & Mohite, K. (2026). Applications of Operations Research in Real-Life Scenarios: A Literature-Based Analytical Study. *International Journal of Science, Strategic Management and Technology*, 02(10), 1-9.
<https://doi.org/10.55041/ijst.v2i10.030>

License:  This article is published under the Creative Commons Attribution 4.0 International License (CC BY 4.0), permitting use, distribution, and reproduction in any medium, provided the original author(s) and source are properly credited.

Abstract

Spending a day in a city you do not know means making a string of decisions: what to see, in which sequence, and when to arrive, while opening hours, travel times and the limited daylight all get in the way. This report looks at how Operations Research (OR) has been used on this problem, through a close reading of two journal articles written by Vansteenwegen and his co-authors. The first, “The City Trip Planner: An expert system for tourists” (*Expert Systems with Applications*, 2011), shows an orienteering model working inside a live web application. The second, “Iterated local search for the team orienteering problem with time windows” (*Computers & Operations Research*, 2009), presents the metaheuristic used to solve the model behind it. For both articles we set out the real-world setting, the aim of the model, its constraints and assumptions, how it is solved, and where it is strong or weak. After that, a general mathematical statement of the TOPTW (team orienteering problem with time windows) is presented. Last, a small PERT/CPM example built by us shows how a chosen itinerary can be scheduled, and checked against a deadline, once the selection step is done.

1. Introduction to OR and the Problem Type

1.1 Operations Research in brief

Operations Research is the field that applies mathematical models, optimisation and data analysis to decisions on how limited resources should be used. Most OR studies move through the same stages. The real problem is defined first; a model is then written in terms of the variables to be decided, something to maximise or minimise, and the limits that apply; the model is solved either exactly or with a heuristic; the result is checked against what happens in practice; and finally it is put to use. Well-known tools include linear programming, transportation and assignment problems, network flows, PERT/CPM and metaheuristics. Whatever the tool, the idea is the same: guesswork gives way to a clear and repeatable way of comparing options.

1.2 Tourist route planning as an OR problem

Planning a tourist’s route can be treated as an Operations Research problem that combines network optimisation, routing and project scheduling, and it splits naturally into two layers. The first deals with selection and routing. Nobody can see everything in a city, so a subset of attractions has to be picked and put in order so that overall satisfaction is as high as possible within the time available. This is the *Orienteering Problem* (OP), introduced by Golden, Levy and Vohra (1987), and its variants: the *Team OP* (several routes, such as one for each day; Chao et al., 1996), together with the OP

with time windows (opening hours; Kantor & Rosenwein, 1992). All of these are combinatorial integer programs that are NP-hard, so in practice heuristics are what people actually use.

The second is a scheduling layer. Once the day's attractions are fixed, the day behaves like a small project: activities with durations and precedence relations, a hard deadline, and uncertainty. The Critical Path Method (Kelley & Walker, 1959) and the Program Evaluation and Review Technique (Malcolm et al., 1959) answer when each activity should start, which ones are critical and how likely the plan is to finish on time. The two layers are complementary: selection decides *what*, scheduling decides *when and how safely*.

1.3 Method and structure

Two journal articles were selected and analysed under the same headings: real-life context, modelling, solution methodology, and critical analysis. The analysis is based on the published abstracts, the bibliographic record and the secondary literature that describes the models. Section 2 introduces the papers, Sections 3 and 4 analyse them, Section 5 explains the underlying mathematical model, Section 6 adds the PERT/CPM illustration, Section 7 reflects critically on the approaches, and the report closes with Section 8.

2. Selected Research Articles

Table 1. Articles analysed in this report

Paper	Reference	Domain	OR technique
A	Vansteenkoven, P., Souffriau, W., Vanden Berghe, G., & Van Oudheusden, D. (2011). The City Trip Planner: An expert system for tourists. <i>Expert Systems with Applications</i> , 38(6), 6540-6546.	Tourism; decision-support (expert) system	Orienteering model with time windows; fast heuristic (GRASP-based) inside a web application
B	Vansteenkoven, P., Souffriau, W., Vanden Berghe, G., & Van Oudheusden, D. (2009). Iterated local search for the team orienteering problem with time windows. <i>Computers & Operations Research</i> , 36(12), 3281-3290.	Tourism / vehicle-routing type problems	Iterated local search (ILS), a metaheuristic applied to the TOPTW

Both are journal articles from the same research group. They complement each other: Paper A shows the OR model working as a usable product, Paper B shows how such a model can be solved fast enough for interactive use.

3. Analysis of Paper A: The City Trip Planner

3.1 Real-life context

A visitor to a city has only so much time but a huge number of sights to choose from. Digital tourist applications had started to offer personalised routes, and Paper A presents the City Trip Planner, a web application that builds itineraries for five Belgian cities. Users enter their interests and trip limits; the system compares these with a database of locations to estimate how much the person would like each attraction, and then builds a trip for the number of days asked for, taking opening hours into account and leaving room for lunch. The domain is tourism and information systems. OR methods fit well here because picking and ordering attractions within a time limit is a combinatorial optimisation problem, where the number of possible plans explodes as attractions are added, and because the answer has to come back immediately while the user waits.

3.2 Modelling and problem formulation

- **Objective:** maximise the total (predicted) personal-interest score of the attractions included in the trip.
- **Constraints:** a time budget per day; opening hours of each attraction; each attraction visited at most once; travel time between locations; a time slot reserved for a lunch break; the requested number of days.
- **Type of problem:** a TOPTW (team orienteering problem with time windows), which is a selective routing problem in which each member of the “team” is one day of the trip.
- **Assumptions:** interest scores are known in advance (predicted from the user’s profile); visit and travel times are fixed, deterministic values; the value of a trip is the sum of the scores of its attractions.

3.3 Solution methodology

The trip is produced by a quick planning algorithm that works from what the user has asked for. The paper is indexed under the Greedy Randomised Adaptive Search Procedure (GRASP), a constructive metaheuristic that builds a plan over and over by adding attractions drawn at random from a short list of the best candidates, and retains the best plan it has found. The algorithm runs inside a web application that draws on a database of locations. It was assessed in real use: the paper gives usage statistics and user feedback which indicate that tourists value the tool highly.

3.4 Findings, impact and critical analysis

The main finding is practical rather than purely mathematical: an orienteering-type model can be solved quickly enough to power a consumer-facing planner, and users value the result. The impact is that personalised, time-aware itineraries become available without a human travel agent.

- **Strengths:** personalisation through predicted interests; handling of opening hours and a lunch break; response time suitable for interactive use; evidence from a real deployment.
- **Limitations:** a heuristic gives no guarantee of optimality; times are deterministic, so queues, traffic or weather are not modelled; the system covers five cities; evidence of quality is largely user feedback.
- **Extensions:** stochastic or time-dependent travel times, real-time re-planning during the trip, additional constraints such as a money budget, and public-transport legs.

4. Analysis of Paper B: Iterated Local Search for the TOPTW

4.1 Real-life context

Electronic tourist guides that personalise routes have to find a route that gives the tourist the most satisfaction while staying within attraction opening hours and the time at hand. Paper B looks at this problem in its general form, the TOPTW, where a set of locations is supplied, each with a score, a service time and a time window, and a fixed number of routes (for example, days) has to be built. Exact methods only cope with small instances, so a fast heuristic is the obvious OR tool for realistic sizes. The domain is tourism and, more broadly, routing and logistics, because selective vehicle routing has the same structure.

4.2 Modelling and problem formulation

The aim is to maximise the total score gathered across every route. The limits are the time budget of each route, the time window of each location visited (service has to begin inside the window, though waiting for a location to open is permitted), the rule that a location can be visited no more than once, and the rule that each route begins and finishes at the depot or hotel. The problem is an NP-hard integer program. It assumes deterministic travel and service times and fixed, additive scores. Section 5 gives the full formulation.

4.3 Solution methodology

The paper puts forward an iterated local search (ILS) heuristic that alternates between two steps. In the insertion step, unvisited locations are added to the current routes, at the route and position that offer the best balance between the score gained and the extra time used; a quick feasibility check makes sure no time window is ever broken. In the shake step,

a few consecutive visits are taken out of the routes so that the search can move away from the current local optimum, and the insertion step is then run again. The best solution found so far is kept, and the loop repeats until the stopping rule is reached. The method is a custom-coded heuristic and not a call to a general-purpose solver. It was tested on standard TOPTW benchmark instances, and its results were compared against the best ones known. These computational experiments give evidence of how good and how fast the algorithm is.

4.4 Findings, impact and critical analysis

The main result is that even a simple ILS gives good TOPTW solutions in very little computation time, which is exactly what an interactive tourist application needs. Later work shows both how influential the algorithm has been and that it can still be improved: Gunawan et al. (2017) showed that careful parameter tuning of ILS-type algorithms improved 50 of the best known solution values on the TOPTW benchmark instances.

- **Strengths:** simplicity, speed, easy extension to further constraints, and direct usefulness for tourist guides.
- **Limitations:** no optimality guarantee; solution quality depends on parameter settings; deterministic data; benchmark instances are generic and not drawn from real city data.
- **Extensions:** automated parameter tuning, hybridisation with other metaheuristics such as simulated annealing, multiple constraints (budget, categories), and stochastic or time-dependent travel times.

5. Mathematical Model: The TOPTW Formulation

The model below follows the standard TOPTW formulation found in this literature, adjusted to a tourist setting; both papers rest on it. Nodes 1 and n mark the starting and ending points (the hotel). Each route d stands for one day.

Figure 1. TOPTW network: selecting the best itinerary from a hotel depot

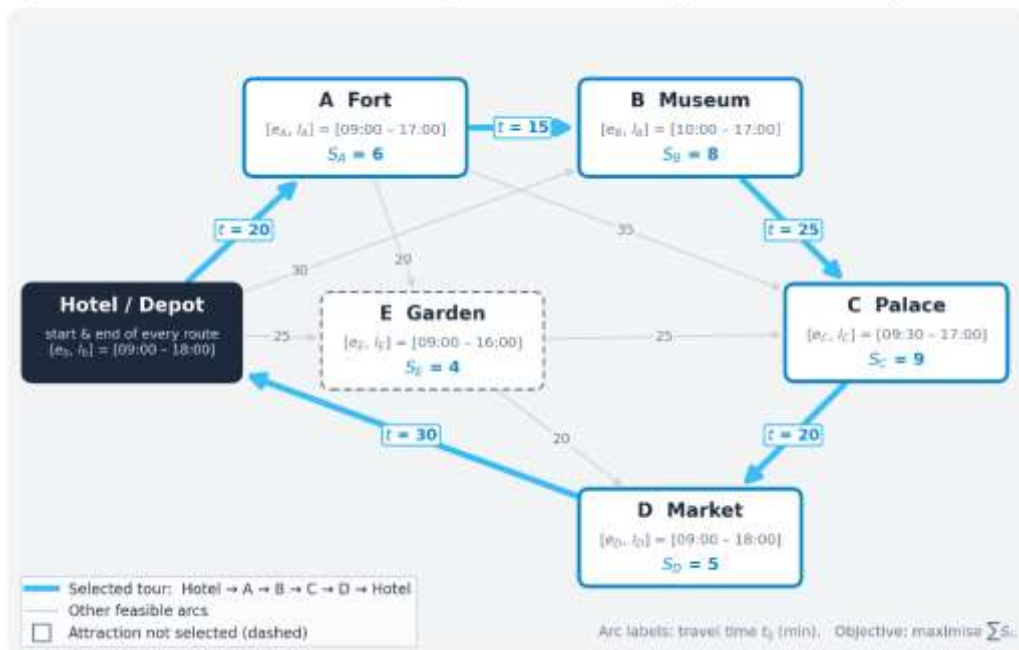


Table 2. Notation

Symbol	Meaning
i, j	Nodes (attractions); 1 = start, n = end, $2 \dots n-1$ = attractions
$d = 1 \dots m$	Routes (days)
S_i	Score (predicted interest) of attraction i ; $S_1 = S_n = 0$
T_i	Visit (service) duration at i
t_{ij}	Time needed to travel from node i to node j
$[O_i, C_i]$	Opening and closing times of the time window; $O_1 = 0$, $C_n = T_{\max}$ (daily time budget)
x_{ijd}	Binary variable: equals 1 when j is visited immediately after i on route d
y_{id}	Binary variable: equals 1 when i is part of route in route d
s_{id}	Continuous variable: the time at which the visit to i begins on route d

Formulation

$$\text{Maximise } Z = \sum_{d=1}^m \sum_{i=2}^{n-1} S_i y_{id} \quad (1)$$

subject to:

$$\sum_{j=2}^n x_{1jd} = \sum_{i=1}^{n-1} x_{ind} = 1, \text{ for each } d \quad (2)$$

$$\sum_{i=1}^{n-1} x_{ikd} = y_{kd} = \sum_{j=2}^n x_{kjd}, \text{ for each } k = 2 \dots n-1 \text{ and each } d \quad (3)$$

$$\sum_{d=1}^m y_{kd} \leq 1, \text{ for each } k = 2 \dots n-1 \quad (4)$$

$$s_{id} + T_i + t_{ij} - s_{jd} \leq M_{ij}(1 - x_{ijd}), \text{ for all } i, j, d \quad (5)$$

$$O_i \leq s_{id} \leq C_i, \text{ for all } i, d \quad (6)$$

$$x_{ijd}, y_{id} \in \{0, 1\}; \quad s_{id} \geq 0 \quad (7)$$

Equation (1) is the objective: it maximises the total score collected. Constraint (2) forces every route to depart from the start node and to finish at the end node. Constraint (3) is flow conservation: an attraction is visited in a route only if the route enters and leaves it. Constraint (4) allows each attraction to be visited at most once over all days. Constraint (5) propagates time along a route (M_{ij} is a sufficiently large constant, for example $\max\{C_i + T_i + t_{ij} - O_j, 0\}$), which also eliminates sub-tours. Constraint (6) enforces opening hours, and because $C_n = T_{\max}$ it also enforces the daily time budget. A lunch break can be represented as an extra mandatory node with its own time window. Because the number of binary variables grows with $n^2 m$, exact solution is feasible only for small instances, which is the reason Papers A and B use heuristics.

6. Extension: Scheduling a Chosen Itinerary with CPM and PERT

The two papers decide *what* to visit. To complement the route-selection models discussed in the two studies, this report applies CPM and PERT to a tourist itinerary. The data are hypothetical. The tourist leaves the hotel at 09:00, must be back by 18:00 (540 minutes), and visits a Fort, a museum (opens 10:00), a Palace (closes 17:00) and a Market, with a lunch break and two bookings (entry tickets and return transport) that can be done in parallel.

Taking the earliest start as $ES(j) = \max EF(i)$ over its predecessors, $EF(j) = ES(j) + d(j)$, then a backward pass gives LF and LS, and total float is $TF = LS - ES$; these give the CPM schedule in Table 3. PERT brings in three-point estimates, with expected duration $t_e = (o + 4m + p)/6$ and variance $((p - o)/6)^2$.

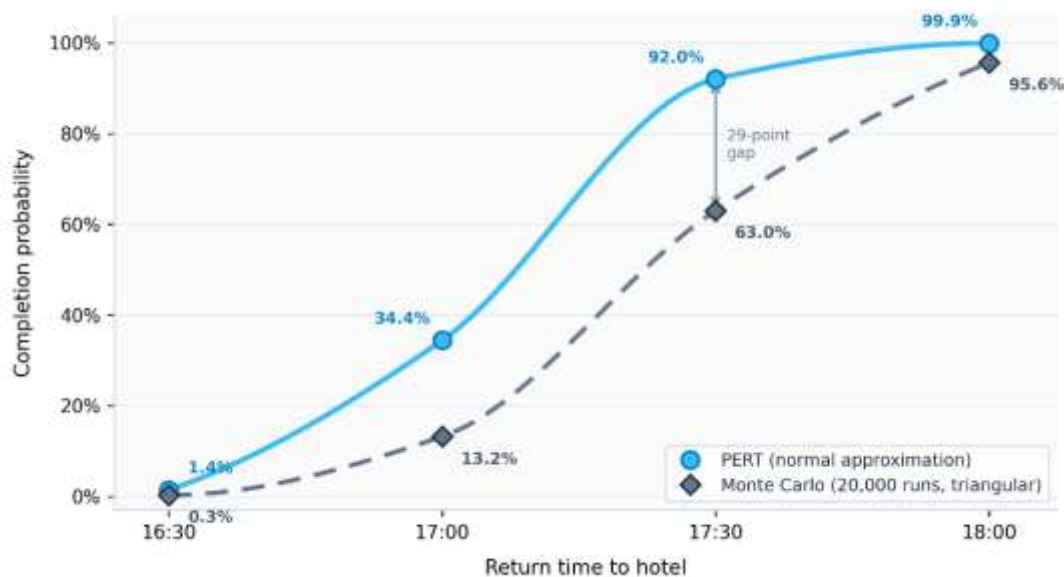
Table 3. CPM schedule with most likely durations (minutes from 09:00)

ID	Activity	Dur.	ES	EF	Float	Critical
A	Travel hotel → Fort	20	0	20	0	Yes
B	Fort visit	75	20	95	0	Yes
C	Pre-book tickets	15	0	15	95	No
D	Travel Fort → Museum	15	95	110	0	Yes
E	Museum visit	105	110	215	0	Yes
F	Lunch	45	215	260	0	Yes
G	Travel to Palace	25	260	285	0	Yes
H	Palace visit	90	285	375	0	Yes
I	Travel to Market	20	375	395	0	Yes
J	Market visit	45	395	440	0	Yes
K	Return to hotel	30	440	470	0	Yes
L	Book return transport	10	0	10	430	No

The critical path runs through A-B-D-E-F-G-H-I-J-K. The itinerary takes 470 minutes, so the tourist is back at 16:50 with 70 minutes in hand. Booking the tickets has 95 minutes of float and booking the return transport has 430 minutes, so either can be done at any convenient moment. The museum visit starts at 10:50, after it opens at 10:00, and the Palace visit finishes at 15:15, well before the 17:00 closing time, so every time window is respected.

Using the expected durations, the critical path has a mean of 486.7 minutes (return about 17:07) and a standard deviation of 16.6 minutes. A Monte Carlo simulation with 20,000 runs and triangular distributions gives a more cautious picture (Table 4).

Probability of Completing the Itinerary by Return Time



Comparison of PERT and Monte Carlo completion probabilities from the illustrative itinerary analysis.

Table 4. Probability of being back at the hotel by a given time

Target return time	PERT approximation)	(normal Monte Carlo (triangular)
16:30	1.4%	0.3%
17:00	34.4%	13.2%
17:30	92.0%	63.0%
18:00	99.9%	95.6%

The 18:00 deadline is safe under both methods, but a 17:30 target is only moderately safe, and the two methods disagree by almost 30 percentage points there, which shows how sensitive the advice is to the assumed distribution (MacCrimmon & Ryavec, 1964). The model also supports scope decisions: adding a Garden (20-minute transfer plus 60-minute visit) pushes the deterministic return to 18:10 and breaks the deadline, whereas swapping it for the Market keeps the return near 17:05. Deciding which swap yields the most satisfaction is exactly the orienteering problem of Sections 3 to 5.

7. Critical Reflection

Table 5. Comparison of the approaches

	Paper A (City Trip Planner)	Paper B (ILS for TOPTW)	CPM/PERT illustration
Decides	What to visit, personalised, as a deployed tool	What to visit and in which order, at algorithm level	When each activity starts and whether the day finishes on time
Objective	Maximise predicted interest	Maximise total score	Minimise duration; estimate probability of meeting a deadline
Method	Fast constructive heuristic (GRASP) in a web app	Iterated local search (insertion + shake)	Forward/backward pass, three-point estimates, simulation
Main weakness	Deterministic times, limited coverage	No optimality guarantee, parameter dependence	Needs a fixed activity set; assumes independent durations

7.1 Strengths of the modelling approach

The OP family captures the core trade-off of a tourist day in a compact model, and the heuristics make it usable in real time. Treating the result as a project adds transparency: the critical path shows where delay hurts, float shows where flexibility exists, and probabilities turn a nominal schedule into a risk statement.

7.2 Limitations and assumptions

All three approaches rely on deterministic or independently distributed times, whereas real delays are correlated: one traffic jam delays several transfers at once. Scores are assumed to be known and additive, although the real value of a visit depends on what came before it. Heuristics trade optimality for speed, and the CPM/PERT illustration uses assumed, not measured, data and an almost serial network, which limits what it can show about parallel activities.

7.3 Possibilities for extension

Natural improvements are stochastic and time-dependent travel times, real-time re-planning when the tourist falls behind, additional constraints such as money and stamina (a multi-constraint TOPTW, which links to resource-constrained scheduling), automated parameter tuning of the heuristics, and a single tool that couples a selection layer

with a scheduling layer that computes the critical path and the on-time probability. Validation with real destination data and tourist trials would turn the illustration into evidence.

8. Conclusion

Tourist route planning shows how OR can turn an everyday problem into a precise model. Paper A shows that a planner based on orienteering can be built into a web application that tourists find useful, and Paper B shows that a simple iterated local search can solve the underlying TOPTW both quickly and well. The mathematical model (objective, flow, visit-once, time-propagation and time-window constraints) is common to both. Adding CPM and PERT shows that, once the visits have been chosen, scheduling tools reveal the critical activities, the float available and the risk of missing the deadline. The main lessons are that selection and scheduling are two halves of one planning problem, that heuristics are the practical choice at realistic sizes, and that the most valuable next step is to model uncertainty explicitly.

References

- Chao, I.-M., Golden, B. L., & Wasil, E. A. (1996). The team orienteering problem. *European Journal of Operational Research*, 88(3), 464-474.
- Golden, B. L., Levy, L., & Vohra, R. (1987). The orienteering problem. *Naval Research Logistics*, 34(3), 307-318.
- Gunawan, A., Lau, H. C., Vansteenwegen, P., & Lu, K. (2017). Well-tuned algorithms for the team orienteering problem with time windows. *Journal of the Operational Research Society*, 68(8).
- Kantor, M. G., & Rosenwein, M. B. (1992). The orienteering problem with time windows. *Journal of the Operational Research Society*, 43(6), 629-635.
- Kelley, J. E., & Walker, M. R. (1959). Critical-path planning and scheduling. *Proceedings of the Eastern Joint Computer Conference*, 160-173.
- MacCrimmon, K. R., & Ryavec, C. A. (1964). An analytical study of the PERT assumptions. *Operations Research*, 12(1), 16-37.
- Malcolm, D. G., Roseboom, J. H., Clark, C. E., & Fazar, W. (1959). Application of a technique for research and development program evaluation. *Operations Research*, 7(5), 646-669.
- Vansteenwegen, P., Souffriau, W., Vanden Berghe, G., & Van Oudheusden, D. (2009). Iterated local search for the team orienteering problem with time windows. *Computers & Operations Research*, 36(12), 3281-3290.
- Vansteenwegen, P., Souffriau, W., Vanden Berghe, G., & Van Oudheusden, D. (2011). The City Trip Planner: An expert system for tourists. *Expert Systems with Applications*, 38(6), 6540-6546.