

Learning road distances for combinatorial optimization

T. Varol, O. Arslan

G-2026-57

September 2026

La collection *Les Cahiers du GERAD* est constituée des travaux de recherche menés par nos membres. La plupart de ces documents de travail a été soumis à des revues avec comité de révision. Lorsqu'un document est accepté et publié, le pdf original est retiré si c'est nécessaire et un lien vers l'article publié est ajouté.

Citation suggérée : T. Varol, O. Arslan (Septembre 2026). Learning road distances for combinatorial optimization, Rapport technique, Les Cahiers du GERAD G- 2026-57, GERAD, HEC Montréal, Canada.

Avant de citer ce rapport technique, veuillez visiter notre site Web (<https://www.gerad.ca/fr/papers/G-2026-57>) afin de mettre à jour vos données de référence, s'il a été publié dans une revue scientifique.

The series *Les Cahiers du GERAD* consists of working papers carried out by our members. Most of these pre-prints have been submitted to peer-reviewed journals. When accepted and published, if necessary, the original pdf is removed and a link to the published article is added.

Suggested citation: T. Varol, O. Arslan (September 2026). Learning road distances for combinatorial optimization, Technical report, Les Cahiers du GERAD G-2026-57, GERAD, HEC Montréal, Canada.

Before citing this technical report, please visit our website (<https://www.gerad.ca/en/papers/G-2026-57>) to update your reference data, if it has been published in a scientific journal.

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2026
– Bibliothèque et Archives Canada, 2026

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2026
– Library and Archives Canada, 2026

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Télec. : 514 340-5665
info@gerad.ca
www.gerad.ca

Learning road distances for combinatorial optimization

Taha Varol
Okan Arslan

*Department of Decision Sciences, HEC Montréal
& GERAD & CIRRELT, Montréal (Qc), Canada,
H3T 2A7*

taha.varol@hec.ca
okan.arslan@gerad.ca

September 2026
Les Cahiers du GERAD
G–2026–57

Copyright © 2026 Varol, Arslan

Les textes publiés dans la série des rapports de recherche *Les Cahiers du GERAD* n'engagent que la responsabilité de leurs auteurs. Les auteurs conservent leur droit d'auteur et leurs droits moraux sur leurs publications et les utilisateurs s'engagent à reconnaître et respecter les exigences légales associées à ces droits. Ainsi, les utilisateurs:

- Peuvent télécharger et imprimer une copie de toute publication du portail public aux fins d'étude ou de recherche privée;
- Ne peuvent pas distribuer le matériel ou l'utiliser pour une activité à but lucratif ou pour un gain commercial;
- Peuvent distribuer gratuitement l'URL identifiant la publication.

Si vous pensez que ce document enfreint le droit d'auteur, contactez-nous en fournissant des détails. Nous supprimerons immédiatement l'accès au travail et enquêterons sur votre demande.

The authors are exclusively responsible for the content of their research papers published in the series *Les Cahiers du GERAD*. Copyright and moral rights for the publications are retained by the authors and the users must commit themselves to recognize and abide the legal requirements associated with these rights. Thus, users:

- May download and print one copy of any publication from the public portal for the purpose of private study or research;
- May not further distribute the material or use it for any profit-making activity or commercial gain;
- May freely distribute the URL identifying the publication.

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Abstract : We study methods for approximating road-network distances on dense urban maps, with a focus on downstream optimization tasks such as route planning. In many applications, distances are obtained on demand from a shortest-path engine or mapping service, generating millions to billions of origin–destination queries and making distance retrieval a major computational bottleneck. We treat these systems as distance oracles and build surrogate models that approximate these distances while preserving downstream solution quality. We train neural networks using decision-agnostic and decision-focused learning by representing the map as a grid and by using network-level features. Our theoretical analysis establishes convergence guarantees for the grid-based approximation and derives bounds linking distance-prediction errors to downstream solution quality. We consider the asymmetric traveling salesman problem (ATSP) and the p -median problem in a real last-mile service area in Montréal with the Open Source Routing Machine (OSRM) as the shortest-path oracle. Our computational study evaluates grid resolutions ranging from 50 to 250 cells along each dimension, compares multiple learning models, and tests downstream performance on optimization instances with up to 250 locations for the ATSP and 500 locations for the p -median problem. On the largest ATSP and p -median benchmarks, the strongest learned surrogates reduce the downstream gaps by about 95% relative to the Euclidean baseline. Decision-focused models trained on ATSP instances achieve the strongest ATSP performance, whereas decision-agnostic learning is stronger overall on the p -median benchmarks and is more robust across the two optimization problems.

Keywords : Distance estimation; combinatorial optimization; machine learning; networks; transportation

1 Introduction

We study the road network distance estimation problem that arises in large-scale combinatorial optimization when an origin–destination (OD) distance matrix is needed, but each OD entry must be obtained through a shortest-path engine or mapping service. Motivated by last-mile logistics and urban distribution planning, we develop surrogate distance models that replace routing queries with fast predictions while preserving the accuracy required for downstream optimization. We use the term *oracle* for a shortest-path engine or mapping service that returns a road-network distance for a requested OD pair.

Many operational optimization models assume that model parameters are easy to obtain, yet this assumption may not necessarily hold for road network distances. When a distance matrix is not already available, each entry is produced by querying an oracle such as Open Source Routing Machine (OSRM) (Luxen and Vetter 2011) or a commercial mapping API. This setting creates an immediate quadratic bottleneck: for N locations, a directed OD matrix contains $N(N - 1)$ distances, so even moderately sized instances require millions of queries and large instances require orders of magnitude more. Querying dense distance matrices at this scale is problematic for both latency and monetary cost. Commercial APIs charge per request and impose rate limits, so generating a full matrix can become expensive, and repeating this process during reoptimization can multiply the cost. For example, Makowski et al. (2025) report that, as of March 1, 2025, Google Maps Platform pricing for distance estimation and route generation starts at \$450 per 100,000 queries and rises substantially at higher volumes and tiers. Because full OD queries are costly, practitioners often fall back on geometric proxies such as Euclidean or haversine distances, sometimes adjusted by circuitry factors (Ballou et al. 2002). These approximations are fast but they ignore road topology. Empirically, such geometric approximations can be poor surrogates for routing decisions on road networks (Boyacı et al. 2021). More broadly, road distance estimation has a long history, from early parametric models (Love and Morris 1979) to partition-based estimation methods such as vector quantization (Altnel et al. 1997), but these can be inaccurate in dense urban environments with irregular connectivity and one-way constraints. Furthermore, existing approaches evaluate accuracy primarily at the distance level and do not explicitly connect distance error to the quality of downstream combinatorial optimization decisions.

We address this gap with a novel feature construction and learning pipeline designed for repeated optimization under distance query latency. Our main contributions are as follows:

- We discretize the service region into a regular grid and precompute major components of the proposed pipeline only once, turning repeated distance estimation on the same service region into a feature-extraction and prediction task.
- Building on this grid representation, we introduce a new set of road-network-aware features that capture both local accessibility patterns of the map and broader connectivity structures.
- We train feed-forward surrogate models that map these features to road-network distances, providing fast approximations to expensive oracle queries.
- We consider both decision-agnostic learning and decision-focused learning, where the training objective is aligned with downstream decision quality through regret-based surrogates (Elmachtoub and Grigas 2022, Mandi et al. 2024, Sadana et al. 2025).
- We establish approximation and decision-stability guarantees that quantify how surrogate distance errors propagate to TSP and p -median objective values and regret, and identify conditions under which the approximation error approaches zero.

The remainder of the paper is organized as follows. Section 2 reviews related work on road distance estimation and learning-based methods. Section 3 presents our grid representation, feature construction, and estimation models, and develops approximation and decision-quality bounds. Section 4 describes the computational study, Section 5 reports the resulting performance on the ATSP and the p -median problem, and Section 6 concludes the study.

2 Literature review

We review three streams relevant to our study: classical road-distance approximation, efficient shortest-path computation, and learning-based optimization.

The task of estimating road distances between two locations using only their coordinates has a long history in operations research and transportation. Early work proposed parametric functional forms that map spatial coordinate differences to road travel distances, typically balancing computational simplicity and interpretability against the irregularity of real transportation networks. For example, Love and Morris (1979) develop mathematical models for road travel distances and show how functional approximations can be calibrated to capture systematic deviations from straight-line geometry. Building on this tradition, Üster and Love (2001a) study the weighted sum-of-order- p , or ℓ_{bp} -norm, which generalizes the conventional weighted ℓ_p -norm by introducing separate directional weights and an axis-rotation parameter. Complementary theoretical analysis by Üster and Love (2001b) examines the directional properties of the ℓ_{bp} -norm.

Classical distance-estimation studies have also examined alternative fitting criteria. Love and Üster (2001) compare several error criteria and find limited differences in their predictive performance. This stream is closely related to our comparison of alternative regression losses. These classical studies evaluate fitting criteria primarily through distance-level prediction errors, while we examine whether the resulting estimates preserve solution quality when embedded in downstream combinatorial optimization problems.

In practice, a widely used baseline is to begin with Euclidean distance and then apply a multiplicative “circuitry” or detour adjustment. Several studies show that circuitry varies by region and network structure and provide empirical guidance for such adjustments (Ballou et al. 2002). While these approaches are computationally attractive and may be adequate at coarse geographic scales, they can be brittle at the urban level, where one-way streets, barriers, bridges, rivers, turn restrictions, and disconnected sub-networks induce complex and highly non-uniform deviations from geometric distance. Parametric norms can account for broad regional characteristics such as rectangularity, axis orientation, and directional impedance, but they still describe the entire region through a small number of global parameters. This limitation motivates methods that can learn more localized and heterogeneous road-network structure from observed distances.

A second line of work treats the true distance function as a black box that can be queried at a cost and aims to construct a surrogate that generalizes from a limited set of oracle evaluations. In this spirit, Altnel et al. (1997) propose vector-quantization-based methods for arbitrary distance-function estimation. Their method is notable in that it can approximate distances without enforcing strict metric properties such as the triangle inequality, which is important when the “distance” corresponds to travel time, toll-based costs, or other generalized impedances. At the same time, their setting is symmetric and macro-scale, whereas urban road networks produce asymmetric OD costs due to directed arcs and turn restrictions, and downstream decision problems such as last-mile delivery require highly localized fidelity. These considerations motivate surrogate designs that can incorporate local network structure while remaining compatible with asymmetric, oracle-defined costs. Consistent with this limitation, Boyacı et al. (2021) show that the quality of Euclidean approximations for vehicle routing depends strongly on the road network and operational setting.

A third stream concerns computing shortest-path distances directly using algorithmic and software advances. Classical algorithms such as Dijkstra’s method and A* search remain fundamental building blocks, while modern speedups exploit road-network structure to enable fast point-to-point queries at scale, such as contraction hierarchies (Geisberger et al. 2012). In practice, many distance oracles use shortest-path engines built on OpenStreetMap data, including high-performance engines such as OSRM (Luxen and Vetter 2011). Most recently, the SCGraph project provides an offline, dependency-free Python toolkit for multimodal routing and network-based distance estimation, reporting substantial runtime improvements over common Python stacks while maintaining close agreement with commer-

cial map distances (Makowski et al. 2025). These contributions highlight two important realities for our setting. First, obtaining accurate network distances is computationally feasible, but it remains burdensome when an optimization model requires *many* OD pairs, such as a full or near-full distance matrix, because even fast individual queries compound under $\Theta(N^2)$ pairwise requirements. Second, software-level speedups and learned surrogates are complementary: a fast oracle such as OSRM or SCGraph can reduce the marginal cost of generating training labels, while a learned model can reduce repeated oracle use and amortize distance retrieval across optimization instances.

Related work also estimates complete routing objective values. Kwon et al. (1995) develop regression and neural-network estimators of optimal TSP tour lengths, while Kou et al. (2023) use Sammon maps for non-Euclidean TSP tour-length estimation. Varol et al. (2024) combine neural networks with node-, instance-, and solution-level features to estimate optimal Euclidean TSP tour lengths under diverse node distributions and integrate these estimates into a generalized TSP solution framework. Our study estimates directed OD road distances for reuse across optimization instances and evaluates their effect on downstream decisions.

Finally, our work is closely connected to the broader literature on integrating learning and optimization. Related work in this direction uses learned surrogates or predictive components to reduce computational burden in optimization, including machine-learning-based warm starts for column generation (Sugishita et al. 2024). Surveys of contextual optimization emphasize the use of covariates to prescribe decisions under uncertainty, unifying terminology across data-driven optimization, policy learning, and task-based learning (Sadana et al. 2025). In parallel, decision-focused learning has emerged as a systematic paradigm for training predictive models to improve downstream decision quality rather than minimizing only statistical prediction loss (Mandi et al. 2024). Within predict-then-optimize, the Smart-Predict-then-Optimize and SPO+ frameworks of Elmachoub and Grigas (2022) provide principled loss constructions for learning cost vectors that induce high-quality decisions in linear optimization problems and enable tractable learning formulations through convex analysis and duality. Recent work in contextual stochastic optimization further develops theory and methodology for learning decision policies with statistical guarantees and improved rates in structured settings (Hu et al. 2022, Kallus and Mao 2023). These frameworks justify evaluating learned distance surrogates not only by approximation accuracy but also by induced optimality gaps or regret in downstream combinatorial objectives. Our paper adopts this perspective: the distance matrix is a query-generated optimization parameter, and we design and train surrogates explicitly to preserve solution quality in large-scale combinatorial optimization while reducing dependence on repeated shortest-path queries.

3 Methodology

We first introduce the map representation and precomputed objects used to construct the features, then develop the feature sets, analyze their approximation properties, and present the estimation models. The main notation is summarized in Online Appendix A.

3.1 Map representation and precomputed objects

A natural challenge in distance estimation is that the map contains many possible OD pairs, making it impractical to compute or store all true pairwise distances. We denote by $\Omega \subset \mathbb{R}^2$ a compact region. For a grid resolution $g \geq 1$, we partition the bounding box of Ω into a $g \times g$ grid of axis-aligned rectangular cells and let $\mathcal{C}_g$ denote the resulting set of g^2 cells. For each cell $\alpha \in \mathcal{C}_g$, let $c(\alpha)$ denote its centroid, and for each point $u \in \Omega$, let $\gamma_g(u) \in \mathcal{C}_g$ denote the cell containing u . Throughout, for a generic ordered OD pair $u, v \in \Omega$, the parameter $r(u, v)$ denotes the true road-network distance returned by the oracle. This discretization provides a finite representation of the map that can be constructed once and reused when forming the features described below. We use an axis-aligned rectangular grid because containing cells can be identified directly from coordinates and neighborhood features can be organized in a fixed row-column structure. Alternative representations, such as Uber’s H3 hexagonal

indexing (Brodsky 2018), do not provide the same natural row-column structure used by our tiled features and would require additional indexing and neighborhood-mapping operations. For our fixed service region and feature construction, the rectangular grid therefore provides a simpler and more direct representation.

The *snapped* (centroid) distance replaces each endpoint by the centroid of its grid cell:

$$\tilde{r}_g(u, v) := r(c(\gamma_g(u)), c(\gamma_g(v))), \quad u, v \in \Omega. \quad (1)$$

We use the term snapping because each endpoint is mapped to its cell centroid before the road-network distance is evaluated.

The map-based features use this grid representation together with representative points for buildings and street-direction attributes associated with those buildings.

Centroid distance matrix. For each pair of grid cells, we define

$$D_{\alpha\beta}^{(g)} := r(c(\alpha), c(\beta)), \quad \alpha, \beta \in \mathcal{C}_g.$$

The values $D_{\alpha\beta}^{(g)}$ form the directed $g^2 \times g^2$ centroid distance matrix. In other words, $D_{\alpha\beta}^{(g)}$ is the road-network distance from the centroid of cell α to the centroid of cell β . For a given grid resolution, this matrix is constructed once and reused across subsequent training, prediction, and optimization instances.

Building-to-grid K -nearest assignment. Let b denote a building representative point obtained by averaging the coordinates of the vertices of its footprint on the map. For a fixed integer $K \geq 1$ and for each grid cell $\alpha \in \mathcal{C}_g$, we compute the Euclidean distance from $c(\alpha)$ to every building representative point and let $B_K^{(g)}(\alpha)$ denote the ordered collection of the K closest building representative points in Euclidean norm. Thus, $B_K^{(g)}(\alpha)$ identifies the buildings assigned to cell α based solely on Euclidean proximity.

Reference building distances. For all representative buildings selected by at least one cell, we precompute the directed road-network distances $r(b, b')$ between every pair of selected representatives. These values are reused when constructing the building-based features below.

Building-street direction indicator. For each building representative point b , we identify the street whose geometric representative point (e.g., midpoint) is closest to b and associate a binary indicator equal to 1 if that street is one-way and 0 otherwise.

3.2 Features

In this section, we construct two groups of features: reusable map-based features and local/OD features associated with the queried origin and destination.

3.2.1 Map-based features

We now construct three map-based feature groups from the reusable objects introduced in Section 3.1.

Tiled OD distance features. To capture map structure around both endpoints, for a given integer $t \geq 0$, let $\mathcal{T}_t(\gamma_g(u)) \subseteq \mathcal{C}_g$ denote the $(2t+1) \times (2t+1)$ tile centered at the cell containing u of the $g \times g$ grid, and define $\mathcal{T}_t(\gamma_g(v))$ analogously. For example, for $t = 1$, we consider a set of nine cells, including $\gamma_g(u)$ and its neighboring cells (as in Figure 1). We include the centroid-to-centroid road distances $D_{\alpha\beta}^{(g)}$ for every $\alpha \in \mathcal{T}_t(\gamma_g(u))$ and $\beta \in \mathcal{T}_t(\gamma_g(v))$, yielding $(2t+1)^4$ distance features. A fixed row-column ordering of the cells (from top left to bottom right) within each tile is used when forming the feature vector. The experimental value of t , together with the building-grid assignment parameter K , is specified in Section 4.

Tiled OD building distance features. We extend the grid-grid tiling by incorporating buildings through the building-to-grid K -nearest assignment. For every $\alpha \in \mathcal{T}_t(\gamma_g(u))$ and $\beta \in \mathcal{T}_t(\gamma_g(v))$, we include the K^2 road-network distances $r(b, b')$ for $b \in B_K^{(g)}(\alpha)$ and $b' \in B_K^{(g)}(\beta)$. Using a consistent ordering of cells and assigned buildings, this yields $K^2(2t+1)^4$ building-distance features.

Tiled street-direction features. For each of the K buildings assigned to every cell in the origin and destination tiles, we include the corresponding binary street-direction indicator, using the same consistent cell and building ordering as above. This yields $2K(2t+1)^2$ binary features.

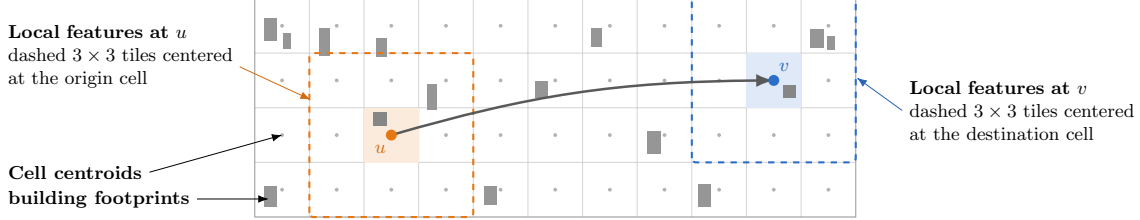


Figure 1: Illustration of the feature construction for an ordered origin-destination pair (u, v) . The dashed boxes show the local 3×3 tiles centered at the origin and destination cells. Cell centroids and building footprints represent reusable map-level objects, while the highlighted origin and destination cells indicate the local OD-specific information used to construct the feature vector.

3.2.2 Local/OD features

These features describe the immediate spatial context of each OD pair. For a pair $(u, v) \in \Omega \times \Omega$, let $u = (\lambda_u, \phi_u)$ and $v = (\lambda_v, \phi_v)$ denote the endpoint coordinates. Let $r_{\text{euc}}(u, v)$ be the Euclidean distance between OD pair (u, v) . This one-dimensional feature provides a simple geometric proxy for the true driving distance. We also use the raw endpoint coordinates $(\lambda_u, \phi_u, \lambda_v, \phi_v)$, the row and column indices of $\gamma_g(u)$ and $\gamma_g(v)$ within the $g \times g$ grid as features, and the absolute coordinate differences $\kappa|\lambda_v - \lambda_u|$ and $\kappa|\phi_v - \phi_u|$, where $\kappa > 0$ is a fixed scaling factor used to avoid working with very small coordinate-difference values. To capture endpoint neighborhoods, we use the tiles $\mathcal{T}_t(\gamma_g(u))$ and $\mathcal{T}_t(\gamma_g(v))$. For each $\alpha \in \mathcal{T}_t(\gamma_g(u))$, we include the Euclidean distance $r_{\text{euc}}(u, c(\alpha))$, and analogously include $r_{\text{euc}}(v, c(\alpha))$ for $\alpha \in \mathcal{T}_t(\gamma_g(v))$. Using the building-to-grid K -nearest assignment, for each $\alpha \in \mathcal{T}_t(\gamma_g(u))$ and each $b \in B_K^{(g)}(\alpha)$, we also include $r_{\text{euc}}(u, b)$; the corresponding distances $r_{\text{euc}}(v, b)$ are included for the destination tile. The same consistent cell and building ordering is used as for the map-based features.

We refer to the compact collection consisting of the snapped distance, Euclidean distance, endpoint coordinates, grid-cell indices, and scaled coordinate differences as the *small feature set*. The *large feature set* augments these features with the tiled centroid-distance, building-distance, street-direction, and endpoint-neighborhood features described above.

3.3 Theoretical analysis of the features

The grid structure is the computational backbone of several map-based features introduced above. Replacing arbitrary locations with their grid-cell centroids enables road-distance information to be stored in a finite matrix and to be reused efficiently in tiled feature constructions. For simplicity of exposition, the theoretical analysis parameterizes grid refinement by a common side length h for axis-aligned square cells, rather than by the computational resolution g and the resulting rectangular cells. With a slight abuse of notation, we write $\tilde{r}_h$ for the corresponding snapped distance. We first develop the results for square grids and symmetric distances. The extensions to rectangular grids and directed distances are straightforward and are discussed subsequently.

The purpose of this section is to determine whether the discrepancy between $r(u, v)$ and $\tilde{r}_h(u, v)$ can be uniformly controlled and whether decisions based on snapped distances remain close to those

obtained using true distances. We first define the worst-case point-to-centroid snapping error $\rho(h)$, and derive a uniform bound on $|r(u, v) - \tilde{r}_h(u, v)|$. We then show that this bound converges to zero as the grid is refined; that is, as $h \rightarrow 0$. Finally, we propagate the resulting per-distance error bound to the traveling salesman and the p -median problems, obtaining guarantees on their optimal objective function values. These results establish the grid representation as a controllable approximation and clarify the trade-off between grid resolution and approximation accuracy.

3.3.1 Uniform snapping error bound.

Proposition 1. Consider a compact region $\Omega \subset \mathbb{R}^2$. Assume that the distance function r is nonnegative, is defined and continuous for all pairs of points in a compact domain containing Ω and all grid-cell centroids, that r is *symmetric*, and satisfies the triangle inequality and $r(u, u) = 0$ for all u in its domain. Consider an axis-aligned square grid with cell side length $h > 0$ covering Ω , and let $\gamma_h(u)$ denote the cell containing u and $\bar{u}_h := c(\gamma_h(u))$ be its centroid. We define $\tilde{r}_h(u, v) := r(\bar{u}_h, \bar{v}_h)$, and the snap radius as

$$\rho(h) := \sup_{u \in \Omega} r(u, \bar{u}_h), \quad (2)$$

which is the worst-case distance from a point in Ω to the centroid of its grid cell. Then, for every OD pair $(u, v) \in \Omega \times \Omega$,

$$|r(u, v) - \tilde{r}_h(u, v)| \leq 2\rho(h). \quad (3)$$

Proof. Given $(u, v) \in \Omega \times \Omega$, the triangle inequality gives $r(u, v) \leq r(u, \bar{u}_h) + r(\bar{u}_h, \bar{v}_h) + r(\bar{v}_h, v)$ and $r(\bar{u}_h, \bar{v}_h) \leq r(\bar{u}_h, u) + r(u, v) + r(v, \bar{v}_h)$. Rearranging and using symmetry gives $|r(u, v) - r(\bar{u}_h, \bar{v}_h)| \leq r(u, \bar{u}_h) + r(v, \bar{v}_h)$. By Equation 2, both terms on the right-hand side of this inequality are at most $\rho(h)$; hence $|r(u, v) - \tilde{r}_h(u, v)| \leq 2\rho(h)$, as desired. $\square$

Proposition 2. We have $\rho(h) \rightarrow 0$ as $h \downarrow 0$, and hence

$$\sup_{u, v \in \Omega} |r(u, v) - \tilde{r}_h(u, v)| \xrightarrow{h \downarrow 0} 0. \quad (4)$$

Proof. Note that within a square of side length h , any point is within Euclidean distance at most $h/\sqrt{2}$ of the centroid. By the continuity assumption in Proposition 1, r is uniformly continuous on the relevant compact Cartesian square. Since $r(u, u) = 0$, it follows that $r(u, \bar{u}_h) \rightarrow 0$ uniformly over $u \in \Omega$ as $h \downarrow 0$. Therefore, $\rho(h) \rightarrow 0$, and Equation 4 follows. $\square$

Corollary 1. For any $\varepsilon > 0$, there exists $h_\varepsilon > 0$ such that, for all $0 < h \leq h_\varepsilon$,

$$\sup_{u, v \in \Omega} |r(u, v) - \tilde{r}_h(u, v)| \leq \varepsilon.$$

In other words, for a sufficiently small grid-cell size, the snapped distance $\tilde{r}_h$ is uniformly ε -accurate over the map.

Remark 1. The uniform bound in Equation (3) implies that the centroid distance matrix can serve as a surrogate for the full continuous map-distance structure: every point-to-point road distance is uniformly close to its snapped centroid-to-centroid counterpart, and the discrepancy vanishes as the grid is refined.

Remark 2. Proposition 1 is stated for symmetric distances and square grid cells to keep the presentation and proof concise. These assumptions are not essential. The same argument extends directly to rectangular grid cells, provided that the maximum cell diameter tends to zero as the grid is refined. In that case, the Euclidean distance from any point in a cell to its centroid is bounded by one half of the cell diagonal, so the uniform-continuity argument above remains unchanged.

Remark 3. The symmetry assumption can also be relaxed. For asymmetric distances, such as directed road-network distances or travel times, define the forward and backward snapping errors as $\rho^+(h) := \sup_{u \in \Omega} r(u, \bar{u}_h)$ and $\rho^-(h) := \sup_{u \in \Omega} r(\bar{u}_h, u)$. The same triangle-inequality argument then gives $|r(u, v) - \tilde{r}_h(u, v)| \leq \rho^+(h) + \rho^-(h)$ for all $u, v \in \Omega$. Under continuity and uniformly shrinking cells, both $\rho^+(h)$ and $\rho^-(h)$ vanish as $h \downarrow 0$. Therefore, all subsequent stability results presented below remain valid after replacing the symmetric error term $\varepsilon_h = 2\rho(h)$ with the directed error term $\rho^+(h) + \rho^-(h)$. Thus, the square and symmetric setting can be understood as a notationally simpler representative case rather than a structural limitation of the approximation argument.

3.3.2 TSP stability under snapped distances.

Consider solving the Traveling Salesman Problem (TSP) on a symmetric graph using the precomputed snapped distances $\tilde{r}_h$. Set $\varepsilon_h = 2\rho(h)$. Proposition 1 gives the uniform per-edge additive bound

$$|r(u, v) - \tilde{r}_h(u, v)| \leq \varepsilon_h, \quad \forall u, v \in \Omega, \quad (5)$$

and Proposition 2 implies $\varepsilon_h \rightarrow 0$ as $h \downarrow 0$. Let $V = \{v_1, \dots, v_N\} \subset \Omega$ be the set of TSP cities, and let T denote a Hamiltonian cycle on V . We define the tour costs under r and $\tilde{r}_h$ as

$$\text{cost}_r(T) := \sum_{(v_a, v_b) \in T} r(v_a, v_b) \quad \text{and} \quad \text{cost}_{\tilde{r}_h}(T) := \sum_{(v_a, v_b) \in T} \tilde{r}_h(v_a, v_b),$$

respectively, and the corresponding optimal tour values as

$$\text{OPT}_r := \min_T \text{cost}_r(T) \quad \text{and} \quad \text{OPT}_{\tilde{r}_h} := \min_T \text{cost}_{\tilde{r}_h}(T),$$

respectively, where the minimum is taken over all Hamiltonian cycles on V .

Lemma 1. For any Hamiltonian cycle T on N cities, $|\text{cost}_r(T) - \text{cost}_{\tilde{r}_h}(T)| \leq N \varepsilon_h$.

Proof. A Hamiltonian cycle has exactly N edges. Due to Inequality (5) and the triangle inequality for absolute values, we have

$$|\text{cost}_r(T) - \text{cost}_{\tilde{r}_h}(T)| \leq \sum_{(v_a, v_b) \in T} |r(v_a, v_b) - \tilde{r}_h(v_a, v_b)| \leq N \varepsilon_h,$$

as desired. □

Proposition 3. The optimal TSP objective values under r and $\tilde{r}_h$ satisfy

$$|\text{OPT}_r - \text{OPT}_{\tilde{r}_h}| \leq N \varepsilon_h. \quad (6)$$

Proof. Let $T_r^* \in \arg \min_T \text{cost}_r(T)$. By Lemma 1,

$$\text{OPT}_{\tilde{r}_h} \leq \text{cost}_{\tilde{r}_h}(T_r^*) \leq \text{cost}_r(T_r^*) + N \varepsilon_h = \text{OPT}_r + N \varepsilon_h.$$

This gives $\text{OPT}_{\tilde{r}_h} - \text{OPT}_r \leq N \varepsilon_h$. Now let $T_{\tilde{r}_h}^* \in \arg \min_T \text{cost}_{\tilde{r}_h}(T)$ be an optimal tour under the snapped distances. Applying Lemma 1 again, but now to $T_{\tilde{r}_h}^*$, gives

$$\text{OPT}_r \leq \text{cost}_r(T_{\tilde{r}_h}^*) \leq \text{cost}_{\tilde{r}_h}(T_{\tilde{r}_h}^*) + N \varepsilon_h = \text{OPT}_{\tilde{r}_h} + N \varepsilon_h.$$

Therefore, $\text{OPT}_r - \text{OPT}_{\tilde{r}_h} \leq N \varepsilon_h$. By combining the two inequalities, $\text{OPT}_{\tilde{r}_h} - \text{OPT}_r \leq N \varepsilon_h$ and $\text{OPT}_r - \text{OPT}_{\tilde{r}_h} \leq N \varepsilon_h$, we obtain (6). □

Proposition 4. Given $T_{\tilde{r}_h}^* \in \arg \min_T \text{cost}_{\tilde{r}_h}(T)$, we have

$$\text{cost}_r(T_{\tilde{r}_h}^*) \leq \text{OPT}_r + 2N \varepsilon_h. \quad (7)$$

Proof. By Lemma 1, $\text{cost}_r(T_{\tilde{r}_h}^*) \leq \text{cost}_{\tilde{r}_h}(T_{\tilde{r}_h}^*) + N\varepsilon_h = \text{OPT}_{\tilde{r}_h} + N\varepsilon_h$. Applying Proposition 3, we have $\text{cost}_r(T_{\tilde{r}_h}^*) \leq (\text{OPT}_r + N\varepsilon_h) + N\varepsilon_h = \text{OPT}_r + 2N\varepsilon_h$, as desired. $\square$

Corollary 2. For a given set of cities V , we have $\varepsilon_h = 2\rho(h) \rightarrow 0$ as $h \downarrow 0$. Thus, $|\text{OPT}_r - \text{OPT}_{\tilde{r}_h}| \rightarrow 0$.

For any fixed city set V , refining the grid drives the optimal-value gap to zero. For ATSP instances, the same stability bounds follow from Remark 3 after replacing the symmetric per-edge error term $\varepsilon_h = 2\rho(h)$ by the directed error term $\rho^+(h) + \rho^-(h)$.

3.3.3 Stability of the p -median problem under snapped distances.

The p -median problem is a classical optimization problem in which we select p facility locations from a set of candidate sites to minimize the total demand-weighted distance between each demand point and its assigned facility (Laporte et al. 2015). The stability argument developed for the TSP extends directly to this problem. Under the uniform per-pair error bound in Equation (5), the difference between the true and snapped assignment costs of any fixed facility set is at most $W\varepsilon_h$, where W is the total demand weight. Consequently, the optimal objective values under r and $\tilde{r}_h$ differ by at most $W\varepsilon_h$, while a facility set that is optimal under the snapped distances has true cost at most $2W\varepsilon_h$ above the true optimum. In the unweighted case with N demand points, we have $W = N$, and the resulting bounds have the same form as the TSP bounds. For any fixed p -median instance with finite positive total demand weight W , $\varepsilon_h \rightarrow 0$ as the grid is refined, so both the optimal-value discrepancy and the regret of the snapped-optimal facility plan disappear. Formal statements and proofs are provided in Online Appendix B.

We finally note that the results in this section are stated for the snapped-distance approximation because it admits a direct uniform error bound. After introducing the learned distance models in Section 3.4, we explain how the same perturbation argument extends to general surrogate cost vectors and yields corresponding average-error regret guarantees.

3.4 Estimation models

We now describe the predictive models used to estimate road-network distances from the feature representations introduced in Section 3.2. We consider a training sample $\mathcal{D}$ of OD observations, whose labels are the oracle distances $r(u, v)$. For an OD pair (u, v) , let $\Phi(u, v)$ denote the corresponding feature vector under either the small or large feature configuration described in Section 3.2.

All estimation models used in this work are implemented as *feed-forward neural networks* (Goodfellow et al. 2016). The learned surrogate is written compactly as $\hat{r}_\theta(u, v) = f_\theta(\Phi(u, v))$, where f_θ is a nonlinear parametric mapping and θ denotes all trainable parameters (weights and biases). We adopt neural networks because the relationship between the constructed features and the true road distance is highly nonlinear, involving complex geometric interactions between local tiles, building neighborhoods, and global topological patterns.

Training proceeds under two distinct paradigms. In the first, which we refer to as *decision-agnostic learning* (DAL), the network is trained only to minimize a pointwise prediction loss between $\hat{r}_\theta(u, v)$ and the oracle distance $r(u, v)$. In the second, referred to as *decision-focused learning* (DFL), the network is trained so that its predictions lead to high-quality decisions when used as cost inputs in downstream combinatorial optimization problems. These paradigms are described next.

3.4.1 Decision-agnostic learning

Decision-agnostic training treats distance estimation as a supervised regression problem. Training is conducted by minimizing an empirical risk of the form

$$\hat{L}^{\text{DAL}}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(u,v) \in \mathcal{D}} \ell(\hat{r}_\theta(u, v), r(u, v)), \quad (8)$$

where $\ell(\cdot, \cdot)$ is a pointwise loss function. The specific choice of ℓ (e.g., squared, absolute, or relative errors) does not affect the structure of (8); different losses simply reflect different error penalties and scaling behaviors. The parameter vector θ is obtained by solving the unconstrained optimization problem $\min_{\theta} \hat{L}_{\text{DAL}}(\theta)$ using stochastic gradient-based updates over mini-batches drawn from $\mathcal{D}$ (Goodfellow et al. 2016). This decision-agnostic objective emphasizes pointwise predictive accuracy and does not differentiate between OD pairs according to their influence on the downstream decisions. The computational specifications of the DAL implementations are presented in Section 4, where we systematically evaluate multiple variants under multiple settings.

3.4.2 Decision-focused learning

While decision-agnostic learning aims to minimize pointwise prediction error, many operational settings require predicted distances to serve as cost coefficients in a downstream optimization problem. In such cases, the primary objective is not necessarily to predict every individual distance accurately, but rather to ensure that the resulting decisions such as tours, facility selections, or assignments, perform well under the true road distances. DFL makes this connection explicit by aligning the training objective of the predictive model with the quality of the decisions induced by its outputs. For each training instance i , we consider a downstream combinatorial optimization problem of the form $\min_{s \in \mathcal{S}_i} c_i^\top s$, where $\mathcal{S}_i$ is the feasible region (e.g., all tours or facility-location decisions), and c_i denotes the vector of true costs derived from the road network. Let s_i^* denote an optimal decision under c_i . The prediction model parametrized by θ produces an estimated cost vector $\hat{c}_i(\theta)$, which induces its own decision $\hat{s}_i(\theta) \in \arg \min_{s \in \mathcal{S}_i} \hat{c}_i(\theta)^\top s$. The quality of $\hat{s}_i(\theta)$ is evaluated under the true costs via the *regret*, defined as $c_i^\top \hat{s}_i(\theta) - c_i^\top s_i^*$. A perfectly accurate predictor yields zero regret. More generally, small errors in distances that do not affect the optimal decision structure may incur little or no regret, while small errors on pivotal edges can produce large decision errors. DFL therefore seeks to minimize an empirical decision-focused loss of the form

$$\hat{L}^{\text{DFL}}(\theta) = \frac{1}{Q} \sum_{i=1}^Q \ell^{\text{DFL}}(\hat{c}_i(\theta), c_i, \mathcal{S}_i), \quad (9)$$

where Q is the number of training instances and $\ell^{\text{DFL}}(\cdot, \cdot, \cdot)$ is a surrogate loss that upper bounds or approximates the regret and accounts for the structure of the downstream optimization problem. Different surrogate losses have been proposed in the decision-focused learning literature (Mandi et al. 2024), generally requiring the solution of one or more optimization problems for each training instance under modified cost vectors. The specific surrogate loss used in our study, together with its implementation details, is described in Section 4.

Unlike DAL losses, which treat all prediction errors independently and uniformly, the DFL objective concentrates learning effort on cost components that materially influence the downstream optimization results. This often yields models whose pointwise prediction error may be larger but whose induced decisions achieve significantly better true objective values, especially in high-dimensional or combinatorially structured settings.

3.4.3 Extension of stability results to learned surrogates.

The stability arguments in Section 3.3 are stated for the snapped-distance surrogate. Online Appendix C extends the analysis to arbitrary surrogate cost vectors and shows that the true regret of the decision optimized under a surrogate $\hat{c}$ is controlled by the discrepancy between $\hat{c}$ and the true cost vector c . Under a uniform error bound, this recovers the finite-sample snapped-distance decision-regret guarantee. Under an average absolute error bound, it yields an asymptotic regret guarantee: for dense OD-matrix problems such as ATSP and unweighted p -median, if the average absolute prediction error of a learned surrogate is $o(1/N)$ on N -location instances, then its regret per selected edge or per demand point converges to zero. This is a sufficient condition on the prediction errors of a learned

surrogate; it does not imply that a particular training procedure attains the required rate. Online Appendix C provides the corresponding MAE- and RMSE-based bounds and distinguishes this condition from the finite-sample uniform guarantee in Proposition 4.

4 Computational study

This section presents the experimental framework used to evaluate the proposed distance surrogates. We first describe the data generation process in Section 4.1, model specification and feature construction in Section 4.2, model inventory in Section 4.3, and the benchmarking in Section 4.4. The objective of the computational study is to understand not only which models predict road distances accurately, but also which models produce high-quality decisions when embedded in operational optimization problems. We therefore evaluate all surrogates through both statistical accuracy metrics and downstream optimization performance.

4.1 Data generation

All datasets are constructed from a real last-mile delivery service region in the Greater Montréal area. We begin by sampling 1800 customer locations uniformly over the service area. For each OD pair, we query OSRM under a shortest-distance routing profile and record the resulting directed driving distance. These values serve as ground-truth labels for all learned and nonparametric surrogate models. The resulting dataset contains approximately 3.2 million OD pairs. We partition this set into training and validation subsets using an 80%–20% split, yielding approximately 2.56 million training pairs and 0.64 million validation pairs. To obtain an unbiased test set, we independently sample an additional 2000 locations from the same region and compute all pairwise OSRM distances among them. These test distances are never used during training or hyperparameter tuning and are reserved exclusively for out-of-sample evaluation. All distance queries and computations are performed on a MacBook Pro equipped with a 12-core Apple M4 Pro CPU and 48 GB of RAM. All code required to reproduce the reported results as well as the dataset are publicly available.¹

4.2 Model specification and feature construction

All learned surrogates are implemented as feed-forward neural networks with a common architecture across decision-agnostic and decision-focused settings. Each network is a 6-layer multilayer perceptron with 1024 neurons per hidden layer with ReLU activations (Goodfellow et al. 2016), and a single nonnegative output representing the predicted road distance. Depending on the specification, each model is trained using either a *small* feature set or a *large* feature set (see Section 3.2 for their definitions). All learned models are trained for a fixed wall-clock budget of one hour using Adam (Goodfellow et al. 2016) with scheduled learning-rate decay, where learning rate (10^{-3}) and decay rate (10^{-4}) are kept fixed. For each model, we store the model checkpoint attaining the lowest validation metric (loss function value for DAL models and regret for DFL models).

Across all model specifications, the tiled feature blocks are constructed using 3×3 neighborhoods around the origin and destination grid cells, corresponding to $t = 1$. For the building-based feature blocks, each grid cell is assigned its $K = 1$ nearest building representative points. These parameter values are held fixed wherever the corresponding features are used.

For each grid resolution, the grid-based distance and feature objects are built once for the fixed service region and reused across all subsequent training, prediction, and downstream optimization runs. The largest grid used in our experiments, 250×250 , contains 62,500 centroids and therefore induces a large directed centroid-distance matrix. In particular, at $g = 250$, the centroid matrix contains approximately 3.91×10^9 directed entries and is constructed only once for the service region. This

¹<https://github.com/tahavarol/road-distance-estimation-for-CO>

preprocessing cost is nontrivial, but it is an offline map-indexing step rather than a per-instance cost. After preprocessing, each learned-surrogate OD prediction uses only fixed-size local feature blocks and a neural-network forward pass. This setting matches repeated-planning applications in which many OD matrices are generated over the same urban region.

4.3 Model inventory

We carry out a structured ablation analysis along multiple dimensions: grid resolution, feature-set size, regression loss, learning-task type, learning-task size, and learning paradigm. Let $\mathcal{M}$ denote the set of all benchmarked surrogate models. The set $\mathcal{M}$ consists of 1 Euclidean baseline, 5 snapped distance matrix models, 30 DAL models, and 60 DFL models, a total of 96 models. We now present the details of this model inventory.

4.3.1 Euclidean baseline

An additional geometric benchmark we use is the Euclidean distance. This model requires no training and provides a simple reference point between raw geometric proximity and learning-based surrogates.

4.3.2 Snapped distance matrix model

The second model is a nonparametric baseline based on the centroid distance matrix introduced in Section 3.1. This snapped distance matrix model provides an efficient lookup-based approximation requiring no training.

The snapped model predicts $\tilde{r}_g(u, v) = D_{\gamma_g(u), \gamma_g(v)}^{(g)}$ for any ordered OD pair $(u, v) \in \Omega \times \Omega$. Once the centroid distances have been precomputed, each prediction is a constant-time table lookup. We evaluate $g \in \{50, 100, 150, 200, 250\}$, resulting in 5 snapped-distance baseline models.

4.3.3 DAL models

DAL models are trained to minimize pointwise prediction error relative to OSRM ground-truth distances over the training set described in Section 4.1. For an OD pair (u, v) , the losses compare the predicted distance $\hat{r}_\theta(u, v)$ with the oracle distance $r(u, v)$. We consider three regression losses: mean squared error (MSE), mean absolute error (MAE), and mean absolute percentage error (MAPE), defined as

$$\begin{aligned} \ell_{\text{MSE}}(u, v; \theta) &= (\hat{r}_\theta(u, v) - r(u, v))^2, \\ \ell_{\text{MAE}}(u, v; \theta) &= |\hat{r}_\theta(u, v) - r(u, v)|, \quad \ell_{\text{MAPE}}(u, v; \theta) = \frac{|\hat{r}_\theta(u, v) - r(u, v)|}{\max\{r(u, v), \tau\}}, \end{aligned}$$

where $\tau > 0$ is a small constant introduced for numerical stability. For DAL, we train a separate network for every combination of three regression loss functions, two feature configurations, small and large, and five grid resolutions, $g \in \{50, 100, 150, 200, 250\}$. Thus, the DAL inventory contains $3 \times 2 \times 5 = 30$ distinct models.

4.3.4 DFL models

DFL trains the surrogate to improve downstream decision quality rather than pointwise distance accuracy. In this study, we base DFL training on the SPO+ surrogate loss of Elmachroub and Grigas (2022). For each training instance i , let c_i denote the true cost vector and let $\hat{c}_i(\theta)$ denote the cost vector predicted by the surrogate model. The SPO+ loss is defined as

$$\mathcal{L}_{\text{SPO+}}(\hat{c}_i(\theta), c_i) = \max_{s \in \mathcal{S}_i} (c_i - 2\hat{c}_i(\theta))^\top s + 2\hat{c}_i(\theta)^\top s_i^* - c_i^\top s_i^*.$$

We consider three DFL training regimes: ATSP-based training, p -median-based training, and hybrid training combining both task classes. Here, a training regime refers to the type of downstream optimization problem used to generate the decision-focused training signal. For each of the three regimes, we train models across two feature-set sizes, five grid resolutions $g \in \{50, 100, 150, 200, 250\}$, and two training-instance sizes corresponding to 20-location and 50-location problems. The full DFL design therefore contains 60 models.

4.4 Benchmarking

All 96 models are benchmarked against OSRM within the same ablation framework. OSRM is treated throughout as the reference distance oracle, while model performance is compared using both predictive error metrics and downstream optimization gaps. The goal is to quantify, within a unified ablation framework, how modeling and training choices affect both statistical fidelity and operational usefulness.

4.4.1 Evaluation metrics

We evaluate pointwise prediction accuracy on the held-out OD pairs using mean absolute error (MAE), root mean squared error (RMSE), mean absolute percentage error (MAPE), and prediction bias. The mathematical definitions of these metrics are provided in Online Appendix D. We additionally evaluate whether surrogate distances preserve solution quality when embedded in downstream optimization models. We consider two representative problem classes: ATSP and the p -median problem. For each model and each problem setting, we solve 100 independently sampled optimization instances and report average gaps. In the p -median experiments, an instance labeled N - p contains N candidate/customer locations, from which p locations must be selected as facilities; for example, 500-50 denotes 500 locations with 50 facilities to be opened. For each sampled instance, we construct both the true OSRM cost matrix C and the surrogate-induced cost matrix $\hat{C}^{(a)}$ associated with model $a \in \mathcal{M}$.

For ATSP, let T denote the tour returned by LKH (Helsgaun 2017) under C , and let $T^{(a)}$ denote the tour obtained under $\hat{C}^{(a)}$. Evaluating both tours under the true OSRM costs, we define

$$z_{\text{ATSP}}^{\text{LKH}} := \text{cost}(C, T), \quad z_{\text{ATSP}}^{(a)} := \text{cost}(C, T^{(a)}), \quad \text{and} \quad \text{Gap}_{\text{ATSP}}^{\text{LKH},(a)} := 100 \frac{z_{\text{ATSP}}^{(a)} - z_{\text{ATSP}}^{\text{LKH}}}{z_{\text{ATSP}}^{\text{LKH}}}.$$

Because T is produced by the heuristic LKH solver, this is an LKH-referenced gap rather than a proven optimality gap. For the p -median problem, on the other hand, let S^* denote an optimal facility set under C , obtained with Gurobi, and let $S^{(a)}$ denote the facility set obtained under $\hat{C}^{(a)}$. Evaluating both under OSRM costs, we define

$$z_{\text{PM}}^* := \sum_{i=1}^N \min_{j \in S^*} C_{ij}, \quad z_{\text{PM}}^{(a)} := \sum_{i=1}^N \min_{j \in S^{(a)}} C_{ij}, \quad \text{and} \quad \text{Gap}_{\text{PM}}^{(a)} := 100 \frac{z_{\text{PM}}^{(a)} - z_{\text{PM}}^*}{z_{\text{PM}}^*}.$$

Since the OSRM p -median benchmark instances are solved to proven optimality, $\text{Gap}_{\text{PM}}^{(a)}$ is a relative optimality gap.

4.4.2 Training data construction for DFL.

Unlike DAL, which is trained on individual OD pairs, DFL is trained on task-level optimization instances. To keep the amount of information processed per epoch comparable with DAL, we choose the number of training instances so that the total number of predicted cost entries per epoch is approximately the same. The DAL training set contains about 2.56×10^6 OD pairs (Section 4.1). A DFL instance with N locations requires the model to predict a dense $N \times N$ cost matrix before evaluating the SPO+ loss. Accordingly, we use approximately $2.56 \times 10^6 / 50^2 = 1024$ training problems for $N = 50$ and $2.56 \times 10^6 / 20^2 = 6400$ training problems for $N = 20$. This controls the number of predicted cost entries per epoch in training among different methods. All such instances are generated

by subsampling from the original pool of 1800 locations, ensuring that DAL and DFL are trained over the same geographic distribution. Lastly, the ATSP optimization subproblems used in DFL training are solved with LKH (Helsgaun 2017), whereas the p -median subproblems are solved optimally with Gurobi. Because LKH is heuristic, the ATSP training signal should be understood as an LKH-based approximation to the exact optimization step required by SPO+.

4.4.3 Ablation dimensions

The experimental design varies across models. In particular, we compare DAL and DFL; MSE, MAE, and MAPE training losses for DAL models; ATSP, p -median, and hybrid training regimes for DFL models; small and large feature configurations; five grid resolutions, $g \in \{50, 100, 150, 200, 250\}$; and 20-location versus 50-location training instances for DFL. This design enables us to isolate the marginal effect of each modeling choice on both held-out prediction quality and downstream decision quality.

5 Results and discussion

We now present the computational results and discuss the performance of the proposed distance-approximation functions. We first examine the effect of grid resolution, then evaluate the role of feature size, and compare learning models across downstream optimization tasks.

5.1 Effect of grid resolution

We begin the ablation study with a nonlearned baseline comparison, the Euclidean baseline and snapped-distance surrogates at grid resolutions $g \in \{50, 100, 150, 200, 250\}$. This experiment isolates the effect of spatial discretization, without learned features or training objectives.

The results show a clear advantage for snapped distances. As reported in Table 1, the Euclidean baseline has large pointwise errors, with an MAE of 586.18 meters and an RMSE of 671.79 meters. Its bias of -586.14 meters also shows that it systematically underestimates true road distances, which is expected in dense urban networks where straight-line distance ignores detours, one-way streets, turns, and block structure. In contrast, even the coarsest snapped grid substantially reduces error: Snap-50 lowers the MAE to 160.96 meters and the MAPE to 10.74%. Refining the grid further improves accuracy, with Snap-250 achieving the lowest MAE and MAPE, at 43.02 meters and 3.20%, respectively. Note that the improvement is not strictly monotone in every metric. For instance, Snap-200 has a slightly larger RMSE than Snap-150, indicating that some larger errors remain at that resolution. Nevertheless, the overall trend is consistent: finer snapped grids reduce average error, percentage error, and bias relative to Euclidean distance. They also largely remove the severe negative bias of the Euclidean baseline, producing only small positive biases.

Table 1: Pointwise prediction errors for the grid-resolution ablation.

Model	MAE (m)	RMSE (m)	MAPE (%)	Bias (m)
Euclidean	586.18	671.79	26.52	-586.14
Snap-50	160.96	353.81	10.74	18.93
Snap-100	88.96	221.35	6.07	6.37
Snap-150	62.45	147.00	4.61	8.61
Snap-200	51.02	171.10	3.61	7.34
Snap-250	43.02	146.88	3.20	7.51

The downstream optimization results lead to a similar conclusion. Tables 2–4 show that snapped grids generally outperform Euclidean distance and that performance generally improves as the grid is refined. The only exception to this pattern is Snap-50 on the 500–5 p -median instances, where its gap is 11.97% compared with 11.46% for Euclidean distance. For ATSP, the Euclidean baseline deteriorates rapidly with instance size, reaching a relative LKH-referenced gap of 58.28% at 250 locations, whereas

Snap-250 reduces this gap to 10.90%. For p -median, the relative optimality gap on the 100–50 instances falls from 89.03% under Euclidean distance to 13.31% under Snap-250, and on the 500–50 instances it falls from 61.72% to 9.11%. Taken together, the 250×250 grid provides the strongest overall performance among the five grid resolutions considered.

Table 2: ATSP benchmarking ($\text{Gap}_{\text{ATSP}}^{\text{LKH}}$) results for the grid-resolution ablation.

Model	20 locations	50 locations	100 locations	250 locations
Euclidean	8.61	18.33	28.85	58.28
Snap-50	5.68	14.10	24.16	51.03
Snap-100	2.91	6.69	12.37	27.03
Snap-150	1.55	3.96	8.06	17.02
Snap-200	1.29	3.27	5.97	13.92
Snap-250	1.13	2.57	4.62	10.90

Table 3: The p -median problem benchmarking (Gap_{PM}) results for small- and medium-scale instances in the grid-resolution ablation. Column names indicate instance parameters (number of nodes– p value).

Model	20–2	20–4	50–5	50–10	100–5	100–10	100–25	100–50
Euclidean	6.49	11.14	12.57	23.81	12.34	23.33	46.23	89.03
Snap-50	3.61	8.16	10.07	16.34	10.32	16.57	33.68	68.17
Snap-100	2.25	3.25	4.86	8.44	3.77	7.28	15.83	32.55
Snap-150	0.77	2.12	2.95	5.13	2.19	5.20	10.18	21.99
Snap-200	1.26	1.51	2.49	3.80	2.96	4.74	8.34	17.81
Snap-250	0.46	1.13	1.45	3.04	1.50	2.73	5.82	13.31

Table 4: The p -median problem benchmarking (Gap_{PM}) results for large-scale instances in the grid-resolution ablation.

Model	250–5	250–10	250–25	250–50	500–5	500–10	500–25	500–50
Euclidean	12.55	23.85	43.54	67.97	11.46	23.20	43.21	61.72
Snap-50	9.77	16.36	30.16	52.99	11.97	20.39	31.93	52.89
Snap-100	3.05	6.58	12.70	25.64	4.15	6.69	12.22	22.30
Snap-150	2.03	4.27	8.53	16.87	1.94	4.73	7.98	15.21
Snap-200	2.74	5.36	7.02	13.29	4.44	7.95	7.27	12.95
Snap-250	1.39	2.76	5.15	10.46	0.99	2.82	5.19	9.11

These findings are consistent with the theoretical results in Section 3. Under the assumptions of Propositions 1–2, together with the rectangular-grid extension in Remark 2 and the directed-distance extension in Remark 3 relevant to our experiments, snapped distances converge uniformly to the true distance as the grid is refined, while the TSP and p -median stability results imply corresponding convergence in downstream optimization quality. We provide empirical checks related to the ATSP stability bound in Online Appendix E. Overall, Euclidean distance is too crude for city-scale operational decisions, whereas snapped distance matrices provide a simple, efficient, and substantially stronger nonparametric baseline.

5.2 Effect of feature size

We next examine the effect of feature size. Since the 250×250 grid performs best, we fix this resolution and compare the small and large feature sets defined in Section 3.2. The large-feature models reduce average gaps in all ATSP settings and all but one p -median setting, indicating that the additional road-network features generally improve downstream performance. Detailed results are reported in Online Appendix F.

5.3 Comparison of learning models

Having established that the 250×250 grid is the strongest discretization setting and that the large feature set is generally stronger, we restrict the rest of the computational study to this specification. This leaves nine learned models: six DFL models, obtained from the three training tasks (ATSP, p -Median, and Hybrid) and two training-instance sizes (20 and 50), and three DAL models, trained with MSE, MAE, and MAPE losses. We denote DFL models by DFL-[task]-[size], where task indicates the downstream problem used for training (ATSP, p -Median, or Hybrid) and $\text{size} \in \{20, 50\}$ is the number of locations in each training instance; DAL models are denoted by DAL-[loss], where $\text{loss} \in \{\text{MSE}, \text{MAE}, \text{MAPE}\}$.

Table 5 shows that DAL models are substantially better point predictors than DFL models. Their MAE values are between 17.38 and 18.98 meters, their MAPE values are at most 1.66%, and their biases are close to zero. In contrast, DFL models have much larger pointwise errors and, in several cases, strong systematic bias. This is expected: DAL directly optimizes prediction accuracy, whereas DFL optimizes downstream decision quality. The downstream results in Table 6, on the other hand, show a more nuanced pattern. For ATSP, task-aligned DFL performs the best: DFL-ATSP-20 achieves the lowest or tied-lowest gap at every ATSP size, ranging from 0.20% at 20 locations (tied with DAL-MAPE) to 2.96% at 250 locations. Hybrid DFL models are generally the next strongest DFL variants, while p -Median-trained DFL models are weaker for routing. Thus, DFL can be highly effective when the training and evaluation tasks are aligned.

Table 5: Pointwise prediction errors at large feature size and 250×250 grid resolution.

Learning Model	MAE (m)	RMSE (m)	MAPE (%)	Bias (m)
DFL-Hybrid-20	236.01	281.52	10.37	-233.64
DFL-Hybrid-50	210.01	250.49	9.37	195.59
DFL-ATSP-20	941.04	1097.82	38.34	939.76
DFL-ATSP-50	248.68	347.34	10.60	217.25
DFL- p -Median-20	169.71	212.16	8.09	-165.52
DFL- p -Median-50	194.86	241.09	9.98	-50.10
DAL-MAE	17.38	64.47	1.61	-1.67
DAL-MSE	18.98	65.05	1.66	1.83
DAL-MAPE	18.73	64.25	1.39	-1.56

Table 6: ATSP benchmarking ($\text{Gap}_{\text{ATSP}}^{\text{LKH}}$) results at large feature size and 250×250 grid resolution.

Learning Model	20 locations	50 locations	100 locations	250 locations
DFL-Hybrid-20	0.22	0.73	1.62	3.69
DFL-Hybrid-50	0.31	0.83	1.54	3.45
DFL-ATSP-20	0.20	0.66	1.31	2.96
DFL-ATSP-50	0.31	0.85	1.38	3.25
DFL- p -Median-20	0.28	0.94	1.95	4.64
DFL- p -Median-50	0.39	0.98	1.99	4.76
DAL-MAE	0.25	0.96	1.68	3.96
DAL-MSE	0.32	0.80	1.41	3.61
DAL-MAPE	0.20	1.03	1.76	5.19

For the p -median benchmarks in Tables 7 and 8, however, DAL models are the strongest overall. DAL-MSE and DAL-MAE outperform all DFL variants in nearly every setting, including the largest instances. For example, at 500-50, DAL-MSE achieves a gap of 3.34%, compared with 4.42% for DFL-Hybrid-20 and 4.96% for DFL- p -Median-50. The ATSP-trained DFL models transfer poorly to p -median, confirming that DFL performance is strongly task dependent. Within the DAL family, the three losses produce similar pointwise accuracy, but MSE and MAE are more stable for downstream optimization. DAL-MAPE achieves the lowest percentage prediction error, but it is weaker on several ATSP and p -median benchmarks. Among these results, DAL-MSE provides the strongest all-around performance, with DAL-MAE as a close alternative.

Table 7: The p -median problem benchmarking (G_{appM}) results for small- and medium-scale instances at large feature size and 250×250 grid resolution.

Learning Model	20-2	20-4	50-5	50-10	100-5	100-10	100-25	100-50
DFL-Hybrid-20	0.35	0.66	0.83	1.19	0.73	1.07	2.63	7.64
DFL-Hybrid-50	0.32	0.76	0.87	1.46	0.87	1.17	2.66	7.51
DFL-ATSP-20	1.50	2.20	3.43	4.65	3.46	4.81	8.00	14.45
DFL-ATSP-50	2.38	3.46	4.55	6.20	4.98	6.22	10.24	19.01
DFL- p -Median-20	0.35	0.94	0.84	1.54	0.93	1.34	3.31	9.58
DFL- p -Median-50	0.45	0.83	0.81	1.46	0.82	1.17	3.33	9.36
DAL-MAE	0.17	0.31	0.38	0.90	0.40	0.70	1.51	4.50
DAL-MSE	0.10	0.30	0.49	0.96	0.42	0.71	1.47	4.20
DAL-MAPE	0.18	0.39	0.43	1.14	0.48	0.94	2.30	5.84

Table 8: The p -median problem benchmarking (G_{appM}) results for large-scale instances in the exhaustive comparison at large feature size and 250×250 grid resolution.

Learning Model	250-5	250-10	250-25	250-50	500-5	500-10	500-25	500-50
DFL-Hybrid-20	0.87	1.12	2.37	5.01	1.08	1.32	2.42	4.42
DFL-Hybrid-50	1.16	1.56	2.85	5.33	1.58	1.90	3.00	5.13
DFL-ATSP-20	3.69	5.32	8.03	12.49	3.41	5.56	8.07	11.94
DFL-ATSP-50	5.19	7.57	10.68	16.06	5.29	7.36	10.65	15.49
DFL- p -Median-20	0.87	1.29	3.00	6.01	1.16	1.73	2.86	5.32
DFL- p -Median-50	0.76	1.27	2.88	5.53	0.95	1.40	2.58	4.96
DAL-MAE	0.39	0.80	1.65	3.65	0.39	0.84	1.59	3.75
DAL-MSE	0.51	0.67	1.45	3.28	0.59	0.84	1.50	3.34
DAL-MAPE	0.48	0.93	2.07	5.24	0.45	0.99	2.08	5.32

Overall, DFL is useful for specialized, task-aligned optimization, especially for ATSP. DAL is more accurate and more robust across problem classes. Therefore, when the surrogate is intended as a general-purpose road-distance estimator, DAL is the stronger choice; when the downstream task is fixed and known in advance, task-specific DFL can provide additional benefits.

In addition to solution quality, the learned surrogates can reduce the time required to generate large OD distance matrices. In our Montréal setting, constructing a 2000×2000 matrix through sequential point-to-point OSRM queries took 16,200 seconds, whereas the current prediction pipeline required 700 seconds, corresponding to an approximately $23\times$ speedup over this sequential-query baseline. This sequential comparison is distinct from the one-time preprocessing of the grid-based feature objects, which is performed using OSRM’s parallelized and batched table-generation functionality. Moreover, as a highly optimized C++-based shortest-path engine, OSRM can exploit preprocessing, batching, chunking, and parallel execution and can outperform our current Python- and PyTorch-based implementation under such configurations. The learned surrogate, on the other hand, provides a complementary approximation option whose feature computation and neural-network inference are highly vectorizable and offer substantial potential for further acceleration through GPU inference and systems-level optimization.

6 Conclusion

We have studied how to approximate road-network distances for repeated use in combinatorial optimization, with the aim of reducing costly shortest-path queries while preserving downstream solution quality. We have developed a grid-based framework with reusable road-network features, trained both decision-agnostic and decision-focused surrogate models, derived approximation and decision-stability results, and evaluated the approach on a real Montréal road network using ATSP and the p -median problem.

We draw three main conclusions. First, road-network information is essential. The Euclidean distance becomes increasingly unreliable as the size of the optimization instances grows, whereas snapped centroid distances provide a substantially stronger nonparametric baseline. Second, richer feature representations improve downstream performance in nearly all tested settings, showing that map-based and local road-network descriptors generally add useful information beyond the snapped distance alone. Third, the comparison between decision-agnostic learning (DAL) and decision-focused learning (DFL) reveals a clear trade-off between general-purpose accuracy and task specialization. In the model comparison, DAL models achieve highly accurate distance predictions, with mean absolute errors below 20 meters, and perform robustly across both ATSP and the p -median problem. DFL models are more task dependent, but can outperform DAL when the training and evaluation problems are closely aligned. Overall, the results show that the best surrogate depends not only on predictive accuracy, but also on how its errors affect downstream optimization. More broadly, road-distance estimation should be viewed not merely as a prediction task, but as a decision-support layer between routing engines and large-scale optimization.

A Notation summary

Table A1: Summary of main notation.

Symbol	Meaning
Ω	Compact service region in $\mathbb{R}^2$.
$\mathcal{D}$	Training sample of ordered OD pairs (u, v) , with oracle labels $r(u, v)$.
$\mathcal{M}$	Set of benchmarked surrogate models.
N	Number of locations in a downstream optimization instance.
p	Number of facilities selected in a p -median instance.
g	Computational grid resolution; the bounding box of Ω is partitioned into a $g \times g$ grid.
h	Square-cell side length used to parameterize grid refinement in the theoretical analysis.
t	Tile radius; $\mathcal{T}_t(\cdot)$ contains a $(2t + 1) \times (2t + 1)$ neighborhood.
K	Number of nearest building representative points assigned to each grid cell.
$u, v \in \Omega$	Generic origin and destination points.
$\mathcal{C}_g$	Set of the g^2 grid cells at resolution g .
$\alpha, \beta \in \mathcal{C}_g$	Generic grid cells at resolution g .
$D_{\alpha\beta}^{(g)}$	Road-network distance from $c(\alpha)$ to $c(\beta)$ for cells $\alpha, \beta \in \mathcal{C}_g$.
$\gamma_g(u)$	Grid cell in $\mathcal{C}_g$ containing point u .
$c(\alpha)$	Centroid of grid cell $\alpha \in \mathcal{C}_g$.
b	Building representative point.
$\mathcal{T}_t(\gamma_g(u))$	$(2t + 1) \times (2t + 1)$ tile of grid cells centered at $\gamma_g(u)$; defined analogously for v .
$B_K^{(g)}(\alpha)$	Ordered collection of the K nearest building representative points assigned to cell α at resolution g .
$\Phi(u, v)$	Feature vector for OD pair (u, v) .
$r(u, v)$	True road-network distance returned by the oracle for the ordered OD pair (u, v) .
$\tilde{r}_g(u, v)$	Snapped distance obtained by replacing u and v with the centroids of their cells at resolution g .
$\tilde{r}_h(u, v)$	Snapped distance used in the theoretical analysis with square-cell side length h .
$\hat{r}_\theta(u, v)$	Road-network distance predicted by the learned surrogate with parameter vector θ .
$\hat{r}^{(a)}(u, v)$	Distance predicted for (u, v) by benchmark model $a \in \mathcal{M}$.
$r_{\text{euc}}(u, v)$	Euclidean distance between points u and v .

B Stability of the p -median problem under snapped distances

Consider the classic p -median problem. As with routing, it is natural to plan using snapped distances $\tilde{r}_h$, with the uniform additive bound (5). Let $V = \{v_1, \dots, v_N\} \subset \Omega$ be demand points with weights $w_i \geq 0$ and positive total weight $W := \sum_{i=1}^N w_i > 0$. Let $J \subset \Omega$ be a finite set of candidate facility sites. For any $S \subseteq J$ with $|S| = p$, we define the costs and the optimal solutions of a p -median problem

under r and $\tilde{r}_h$ as follows:

$$\text{PM}_r(S) := \sum_{i=1}^N w_i \min_{j \in S} r(v_i, j) \quad \text{and} \quad \text{PM}_{\tilde{r}_h}(S) := \sum_{i=1}^N w_i \min_{j \in S} \tilde{r}_h(v_i, j),$$

$$\text{OPT}_r^{\text{PM}} := \min_{\substack{S \subseteq J \\ |S|=p}} \text{PM}_r(S) \quad \text{and} \quad \text{OPT}_{\tilde{r}_h}^{\text{PM}} := \min_{\substack{S \subseteq J \\ |S|=p}} \text{PM}_{\tilde{r}_h}(S).$$

Using binary assignment variables x_{ij} and facility opening variables y_j , the Integer Programming (IP) formulation for solving the p -median problem is:

$$\begin{aligned} \min_{x, y} \quad & \sum_{i=1}^N \sum_{j \in J} w_i r(v_i, j) x_{ij} \\ \text{s.t.} \quad & \sum_{j \in J} x_{ij} = 1, & i = 1, \dots, N, \\ & x_{ij} \leq y_j, & i = 1, \dots, N, \forall j \in J, \\ & \sum_{j \in J} y_j = p, \\ & x_{ij} \in \{0, 1\}, y_j \in \{0, 1\}, & i = 1, \dots, N, \forall j \in J. \end{aligned}$$

When solving with snapped distances, $r(v_i, j)$ is replaced by $\tilde{r}_h(v_i, j)$ in the objective function.

Lemma B1. For any $v \in V$ and any $S \subseteq J$ with $|S| = p$,

$$\left| \min_{j \in S} r(v, j) - \min_{j \in S} \tilde{r}_h(v, j) \right| \leq \varepsilon_h.$$

Proof. From (5), for every $j \in S$, $r(v, j) \leq \tilde{r}_h(v, j) + \varepsilon_h$. Taking the minimum over j yields $\min_{j \in S} r(v, j) \leq \min_{j \in S} \tilde{r}_h(v, j) + \varepsilon_h$. The reverse direction follows by swapping r and $\tilde{r}_h$. Combining the two inequalities completes the proof. $\square$

Lemma B2. For any $S \subseteq J$ with $|S| = p$,

$$|\text{PM}_r(S) - \text{PM}_{\tilde{r}_h}(S)| \leq W\varepsilon_h.$$

Proof. Applying Lemma B1 to each v_i , multiplying by w_i , and summing over $i = 1, \dots, N$ gives the desired result. $\square$

Proposition B1. The optimal p -median values under r and $\tilde{r}_h$ satisfy

$$|\text{OPT}_r^{\text{PM}} - \text{OPT}_{\tilde{r}_h}^{\text{PM}}| \leq W\varepsilon_h.$$

Proof. Let $S_r^* \in \arg \min_{\substack{S \subseteq J \\ |S|=p}} \text{PM}_r(S)$. By Lemma B2,

$$\text{OPT}_{\tilde{r}_h}^{\text{PM}} \leq \text{PM}_{\tilde{r}_h}(S_r^*) \leq \text{PM}_r(S_r^*) + W\varepsilon_h = \text{OPT}_r^{\text{PM}} + W\varepsilon_h.$$

Swapping r and $\tilde{r}_h$ yields the reverse inequality; combining the two gives the desired result. $\square$

Proposition B2. Let $\hat{S} \in \arg \min_{\substack{S \subseteq J \\ |S|=p}} \text{PM}_{\tilde{r}_h}(S)$. Then

$$\text{PM}_r(\hat{S}) \leq \text{OPT}_r^{\text{PM}} + 2W\varepsilon_h.$$

Proof. By Lemma B2,

$$\text{PM}_r(\hat{S}) \leq \text{PM}_{\hat{r}_h}(\hat{S}) + W\varepsilon_h = \text{OPT}_{\hat{r}_h}^{\text{PM}} + W\varepsilon_h \leq (\text{OPT}_r^{\text{PM}} + W\varepsilon_h) + W\varepsilon_h,$$

as desired. $\square$

Corollary B1. For any fixed p -median instance with finite positive total demand weight W , as $h \downarrow 0$, we have $\varepsilon_h = 2\rho(h) \rightarrow 0$, hence $|\text{OPT}_r^{\text{PM}} - \text{OPT}_{\hat{r}_h}^{\text{PM}}| \rightarrow 0$.

In other words, for any fixed p -median instance with finite positive W , refining the grid drives both the optimal value gap and the average per-unit-weight regret of the snapped-optimal facility plan to zero.

Remark B1. In the unweighted p -median case (with $J = V$ and $w_i = 1$ for $i = 1, \dots, N$), we have $W = N$, so Proposition B2 gives a $2N\varepsilon_h$ worst-case deviation for the snapped-optimal plan. The same conclusions apply to directed road-network distances by replacing $\varepsilon_h = 2\rho(h)$ with $\varepsilon_h = \rho^+(h) + \rho^-(h)$.

C Average-error decision bounds for learned surrogates

The bounds in Section 3 use a uniform error bound for the snapped-distance surrogate. For the learned surrogates, our computational evaluation uses average-error measures such as MAE and RMSE. The following finite-instance bound relates these measures directly to decision regret and requires neither metric properties nor the grid assumptions of Section 3.

Consider an N -location optimization instance. Let $\mathcal{A}_N$ index the cost entries represented by the true and surrogate cost vectors, and let $\mathcal{X}_N \subseteq \{0, 1\}^{\mathcal{A}_N}$ be the projection of the feasible set onto the corresponding binary decisions. Variables that do not appear in the objective, such as the facility-opening variables in our p -median formulation, affect feasibility through the definition of $\mathcal{X}_N$. Let $c_N, \hat{c}_N \in \mathbb{R}^{\mathcal{A}_N}$ be the true and surrogate cost vectors, and define

$$x_N^* \in \arg \min_{x \in \mathcal{X}_N} c_N^\top x, \quad \hat{x}_N \in \arg \min_{x \in \mathcal{X}_N} \hat{c}_N^\top x.$$

The regret of the surrogate-induced decision is

$$\mathcal{R}_N := c_N^\top \hat{x}_N - c_N^\top x_N^*.$$

For the error vector $e_N := \hat{c}_N - c_N$, define

$$\Delta_N := \|e_N\|_\infty, \quad \bar{e}_N := \frac{\|e_N\|_1}{|\mathcal{A}_N|}, \quad q_N := \frac{\|e_N\|_2}{\sqrt{|\mathcal{A}_N|}}, \quad B_N := \max_{x \in \mathcal{X}_N} \|x\|_1.$$

Thus, $\bar{e}_N$ and q_N are the MAE and RMSE, respectively, on the cost vector of the downstream instance.

Proposition C1. For any surrogate cost vector $\hat{c}_N$,

$$0 \leq \mathcal{R}_N \leq \min \left\{ 2B_N\Delta_N, |\mathcal{A}_N|\bar{e}_N, \sqrt{2B_N|\mathcal{A}_N|}q_N \right\}.$$

Proof. Surrogate optimality gives $\hat{c}_N^\top \hat{x}_N \leq \hat{c}_N^\top x_N^*$. Hence,

$$\mathcal{R}_N = \hat{c}_N^\top (\hat{x}_N - x_N^*) + e_N^\top (x_N^* - \hat{x}_N) \leq e_N^\top (x_N^* - \hat{x}_N).$$

Let $d_N := x_N^* - \hat{x}_N$. Because $x_N^*, \hat{x}_N \in \{0, 1\}^{\mathcal{A}_N}$,

$$\|d_N\|_1 \leq 2B_N, \quad \|d_N\|_\infty \leq 1, \quad \|d_N\|_2 \leq \sqrt{2B_N}.$$

Applying Hölder's and Cauchy-Schwarz inequalities gives

$$\mathcal{R}_N \leq |e_N^\top d_N| \leq \min \{ \|e_N\|_\infty \|d_N\|_1, \|e_N\|_1 \|d_N\|_\infty, \|e_N\|_2 \|d_N\|_2 \},$$

which yields the stated upper bound. Nonnegativity follows from the optimality of x_N^* under c_N . $\square$

For an N -location ATSP, $\mathcal{A}_N$ contains the $N(N-1)$ directed arcs and every tour uses exactly N arcs. Therefore, $|\mathcal{A}_N| = N(N-1)$ and $B_N = N$, so Proposition C1 gives

$$\frac{\mathcal{R}_N}{N} \leq \min \left\{ 2\Delta_N, (N-1)\bar{e}_N, \sqrt{2(N-1)}q_N \right\}.$$

For the unweighted p -median problem with N demand points and N candidate sites, let x contain only the assignment variables; facility-opening variables enter through feasibility. Then $|\mathcal{A}_N| = N^2$ and every feasible assignment uses exactly N cost entries, yielding

$$\frac{\mathcal{R}_N}{N} \leq \min \left\{ 2\Delta_N, N\bar{e}_N, \sqrt{2N}q_N \right\}.$$

If the facility set selected under surrogate costs is reassigned optimally under the true costs, as in our computational evaluation, its true cost can only decrease, so the same bound remains valid.

Consequently, along any sequence of N -location instances with $N \rightarrow \infty$, any one of the sufficient conditions

$$\Delta_N = o(1), \quad \bar{e}_N = o(N^{-1}), \quad \text{or} \quad q_N = o(N^{-1/2})$$

implies $\mathcal{R}_N/N \rightarrow 0$, corresponding to vanishing regret per tour edge or per demand point. Note that these conditions are sufficient but not necessary. They guarantee vanishing regret through the bounds above, although larger prediction errors may still preserve the optimal decision and hence yield zero or vanishing regret.

The uniform-error decision-regret bound in Section 3 is recovered as a special case. For the snapped-distance surrogate, $\Delta_N \leq \varepsilon_h$ implies $\mathcal{R}_N \leq 2N\varepsilon_h$ for ATSP and unweighted p -median, recovering the corresponding decision-regret bounds in Proposition 4 and Online Appendix B. For weighted p -median, the assignment-cost argument in Online Appendix B gives the sharper weight-dependent bound $2W\varepsilon_h$.

The proposition is deterministic and applies to any surrogate cost vector, including DAL and DFL, once its errors on the downstream cost matrix are known. In particular, test set MAE or RMSE from Online Appendix D does not by itself certify the corresponding bound for every optimization instance.

Finally, Proposition C1 assumes that the surrogate problem is solved exactly. If an algorithm instead returns $x_N^{\text{alg}} \in \mathcal{X}_N$ satisfying

$$\hat{c}_N^\top x_N^{\text{alg}} \leq \min_{x \in \mathcal{X}_N} \hat{c}_N^\top x + \delta_N,$$

then the same proof bounds its regret with δ_N added to the right-hand side. Thus, together with any of the sufficient error conditions above, $\delta_N/N \rightarrow 0$ is sufficient for vanishing regret per edge or demand point. Note that this is relevant to the LKH-based ATSP experiments, whose reported gaps are LKH-referenced rather than exact optimality gaps.

D Pointwise prediction metrics

Recall that $r(u, v)$ denotes the OSRM distance for the ordered OD pair (u, v) . Let $\mathcal{D}_{\text{test}}$ denote the set of OD pairs induced by the held-out test locations, and let $\hat{r}^{(a)}(u, v)$ denote the distance predicted by model $a \in \mathcal{M}$. Pointwise prediction performance is evaluated using

$$\begin{aligned} \text{MAE}(a) &:= \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(u,v) \in \mathcal{D}_{\text{test}}} \left| \hat{r}^{(a)}(u, v) - r(u, v) \right|, \\ \text{RMSE}(a) &:= \sqrt{\frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(u,v) \in \mathcal{D}_{\text{test}}} \left(\hat{r}^{(a)}(u, v) - r(u, v) \right)^2}, \end{aligned}$$

$$\text{MAPE}(a) := \frac{100}{|\mathcal{D}_{\text{test}}|} \sum_{(u,v) \in \mathcal{D}_{\text{test}}} \frac{|\hat{r}^{(a)}(u,v) - r(u,v)|}{\max\{r(u,v), \tau\}},$$

and

$$\text{Bias}(a) := \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(u,v) \in \mathcal{D}_{\text{test}}} \left(\hat{r}^{(a)}(u,v) - r(u,v) \right).$$

E Empirical check of the ATSP stability bound

We next provide an empirical check of the ATSP stability bound derived in Section 3. For the actual $g \times g$ computational grid, the rectangular-grid and directed extensions in Remarks 2 and 3 use worst-case forward and backward snapping errors ρ_g^+ and ρ_g^- , with $\varepsilon_g := \rho_g^+ + \rho_g^-$. If these quantities upper-bound the corresponding errors over the full region, the same argument as in Proposition 4 gives the formal $2N\varepsilon_g$ bound for an N -node TSP under the single-cell snapping rule. The numerical comparisons below use the corresponding empirical error values and should therefore be read as empirical checks rather than as independent proofs of the uniform bound.

For single-cell snapping, we compute the directed road distances from each sampled location to the centroid of its containing grid cell and from that centroid back to the location. The maxima of these distances over all sampled locations define ρ_g^+ and ρ_g^- , respectively. For the 3×3 and 5×5 neighborhood comparisons, we first determine, for each sampled location, the minimum point-to-centroid and centroid-to-point distances over the centroids in the corresponding neighborhood. These two minima may be attained at different centroids. We then take the maximum of each collection of per-location minima over all sampled locations to obtain ρ_g^+ and ρ_g^- , respectively. In each comparison, we set $\varepsilon_g = \rho_g^+ + \rho_g^-$ and report the threshold $2N\varepsilon_g$.

Under the theoretical worst-case definition of the snapping errors, snapping each point to the centroid of its own grid cell gives a valid bound, but the corresponding empirical threshold is very conservative in dense road networks. This is expected: a point and its cell centroid may be geographically close but separated by one-way restrictions, turn constraints, or local connectivity barriers, leading to large values of ρ_g^+ and ρ_g^- . To assess how much of this conservatism is caused by the single-centroid snapping rule, we also compute alternative empirical thresholds in which each point is allowed to snap to the best centroid in its surrounding 3×3 or 5×5 neighborhood. For each snapping rule, we compute the resulting ε_g and report the threshold $2N\varepsilon_g$ for $N \in \{20, 50, 100, 250\}$. We then compare this threshold with the observed excess true tour cost across the ATSP test instances and record both the number of exceedances and the average exceedance amount conditional on an exceedance.

Tables E1–E3 summarize the results for the Snap-250 surrogate and the DAL-MSE learned surrogate. In these tables, “Excd.” denotes the number of test instances exceeding the empirical threshold, and “Avg.” denotes the average exceedance amount conditional on an exceedance. For Snap-250, the absence of exceedances under the single-cell comparison is consistent with the finite-sample stability bound, whereas the tighter 3×3 comparison is an empirical diagnostic. For DAL-MSE, the comparisons should likewise be interpreted as empirical diagnostics rather than as a formal validation of Proposition 4, because Proposition 4 is based on a uniform error bound for the surrogate being optimized. However, the single-cell threshold is extremely loose, while the 3×3 neighborhood rule produces a substantially tighter empirical threshold without any observed exceedances. The 5×5 rule gives much smaller values of $2N\varepsilon_g$, but these values are not theorem-based bounds for the Snap-250 model because the actual snapped surrogate still uses the own-cell centroid. Accordingly, a small number of exceedances appear for Snap-250 at smaller ATSP sizes; they disappear at the 250-location scale. In contrast, the DAL-MSE model remains below the empirical 5×5 thresholds in all tested cases, despite using snapped distances only as input features rather than as final predictions.

Table E1: Empirical ATSP stability comparison under single-cell snapping.

Model	20 locations			50 locations			100 locations			250 locations		
	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.
SNAP	296408	0	0	741020	0	0	1482040	0	0	3705100	0	0
DAL-MSE	296408	0	0	741020	0	0	1482040	0	0	3705100	0	0

Table E2: Empirical ATSP stability comparison under 3×3 neighborhood snapping.

Model	20 locations			50 locations			100 locations			250 locations		
	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.
SNAP	33534	0	0	83835	0	0	167670	0	0	419175	0	0
DAL-MSE	33534	0	0	83835	0	0	167670	0	0	419175	0	0

Table E3: Empirical ATSP stability comparison under 5×5 neighborhood snapping.

Model	20 locations			50 locations			100 locations			250 locations		
	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.	$2N\epsilon_g$	Excd.	Avg.
SNAP	1670	2	3266	4175	1	6107	8350	1	8647	20875	0	0
DAL-MSE	1670	0	0	4175	0	0	8350	0	0	20875	0	0

These results support two conclusions. First, for Snap-250 the single-cell empirical threshold is respected and is consistent with the finite-sample stability result, but it is highly conservative; the neighborhood comparisons and the DAL-MSE comparisons are diagnostic rather than theorem-based. Second, allowing local neighborhood snapping provides a useful diagnostic for the effective approximation error of the grid representation: it shows that much tighter empirical ATSP stability thresholds are obtained when local centroid mismatch is reduced. This is consistent with the role of the forward and backward snapping errors in the theory, and it helps explain why the learned large-feature models can deliver strong ATSP performance even though their feature construction is rooted in snapped grid distances.

F Additional ablation results

This appendix presents two additional ablation analyses. Section F.1 reports the detailed feature-size comparison summarized in Section 5.2, while Section F.2 examines the joint effects of grid resolution and feature size within the DAL-MSE family.

F.1 Feature-size ablation

All models in this comparison use the 250×250 grid resolution. The reported values are averaged over the corresponding training configurations to isolate the effect of feature size.

Table F1 shows that the large feature set improves ATSP performance across all instance sizes. The average LKH-referenced gap decreases from 1.19% to 0.28% at 20 locations, from 4.69% to 1.63% at 100 locations, and from 11.06% to 3.95% at 250 locations. This indicates that richer features reduce routing errors more effectively, especially as the number of locations grows. Tables F2 and F3 show the same pattern for the p -median problem. For example, the average optimality gap decreases from 15.67% to 9.12% in the 100–50 setting, and from 11.36% to 6.63% in the 500–50 setting. Although the small feature set already provides a useful map-aware representation, the additional local and global road-network descriptors in the large feature set lead to consistently stronger downstream decisions.

Table F1: ATSP benchmarking results ($\text{Gap}_{\text{ATSP}}^{\text{LKH}}$) for the feature-size ablation, averaged over all training configurations.

Feature set	20 locations	50 locations	100 locations	250 locations
Small features	1.19	2.60	4.69	11.06
Large features	0.28	0.86	1.63	3.95

Table F2: p -median benchmarking results (Gap_{PM}) for small- and medium-scale instances in the feature-size ablation, averaged over all training configurations.

Feature set	20-2	20-4	50-5	50-10	100-5	100-10	100-25	100-50
Small features	0.72	1.32	1.76	3.58	1.80	3.17	6.92	15.67
Large features	0.64	1.09	1.40	2.17	1.45	2.01	3.94	9.12

Table F3: p -median benchmarking results (Gap_{PM}) for large-scale instances in the feature-size ablation, averaged over all training configurations.

Feature set	250-5	250-10	250-25	250-50	500-5	500-10	500-25	500-50
Small features	1.65	3.12	6.11	12.14	1.32	3.05	5.99	11.36
Large features	1.55	2.28	3.89	6.96	1.66	2.44	3.86	6.63

F.2 DAL-MSE results across grid resolutions and feature sizes

The complete computational study contains 90 trained models, making it impractical to report every model across all downstream optimization settings. To provide a focused examination of how grid resolution and feature size interact, this appendix reports the complete factorial comparison within the DAL-MSE family. Specifically, we evaluate the small and large feature sets at grid resolutions of 50×50 , 100×100 , 150×150 , 200×200 , and 250×250 , resulting in ten DAL-MSE models.

Table F4 shows that increasing the grid resolution consistently reduces MAE, RMSE, and MAPE within both feature-size settings. At every grid resolution, the large feature set also improves all three error measures relative to the corresponding small feature set. DAL-MSE-250-large provides the strongest pointwise performance, with an MAE of 18.98 meters, an RMSE of 65.05 meters, and a MAPE of 1.66%. Its bias is also close to zero at 1.83 meters.

Table F4: Pointwise prediction errors for DAL-MSE models across grid resolutions and feature sizes.

Learning Model	MAE(m)	RMSE(m)	MAPE (%)	Bias(m)
DAL-MSE-50-small	139.61	224.26	9.77	-20.24
DAL-MSE-50-large	98.28	169.75	7.32	-12.92
DAL-MSE-100-small	99.84	198.98	7.19	9.71
DAL-MSE-100-large	49.20	108.26	4.04	4.89
DAL-MSE-150-small	72.73	164.23	5.44	-5.60
DAL-MSE-150-large	32.07	89.59	2.74	2.63
DAL-MSE-200-small	58.11	157.83	4.30	-3.60
DAL-MSE-200-large	24.64	77.31	2.12	3.62
DAL-MSE-250-small	54.96	148.09	4.05	-3.23
DAL-MSE-250-large	18.98	65.05	1.66	1.83

The ATSP results in Table F5 show the same overall pattern. The large-feature model outperforms the corresponding small-feature model at every grid resolution and problem size. DAL-MSE-250-large achieves the lowest gap in every column, ranging from 0.32% at 20 locations to 3.61% at 250 locations. Although the improvement with grid resolution is not strictly monotone for every small-feature result, the overall reduction is substantial. For example, the 250-location gap decreases from 43.81% for DAL-MSE-50-small to 3.61% for DAL-MSE-250-large.

Tables F6 and F7 provide consistent evidence for the p -median problem. At every grid resolution, the large-feature model outperforms its small-feature counterpart across all reported instances. DAL-

MSE-250-large is the best-performing model in every p -median configuration. For example, its gap is 4.20% on the 100-50 instance and 3.34% on the 500-50 instance, compared with 60.73% and 44.46%, respectively, for DAL-MSE-50-small.

Table F5: ATSP benchmarking ($\text{Gap}_{\text{ATSP}}^{\text{LKH}}$) results for DAL-MSE models across grid resolutions and feature sizes.

Learning Model	20 locations	50 locations	100 locations	250 locations
DAL-MSE-50-small	4.60	10.97	20.85	43.81
DAL-MSE-50-large	3.17	7.50	13.61	32.37
DAL-MSE-100-small	2.79	6.22	12.02	25.19
DAL-MSE-100-large	1.24	2.93	5.46	12.84
DAL-MSE-150-small	1.63	4.08	7.66	16.61
DAL-MSE-150-large	0.89	1.64	3.37	7.62
DAL-MSE-200-small	1.27	3.12	6.03	13.04
DAL-MSE-200-large	0.60	0.98	2.33	4.94
DAL-MSE-250-small	1.34	2.69	4.63	11.06
DAL-MSE-250-large	0.32	0.80	1.41	3.61

Table F6: The p -median problem benchmarking (Gap_{PM}) results for small- and medium-scale instances for DAL-MSE models across grid resolutions and feature sizes. Column names indicate instance parameters (number of nodes- p value).

Learning Model	20-2	20-4	50-5	50-10	100-5	100-10	100-25	100-50
DAL-MSE-50-small	3.88	7.41	8.24	14.17	7.71	14.94	28.85	60.73
DAL-MSE-50-large	2.03	4.48	4.95	9.61	4.47	7.78	18.89	41.31
DAL-MSE-100-small	2.12	3.41	4.93	7.98	3.62	6.92	15.46	31.63
DAL-MSE-100-large	0.62	1.14	1.94	3.16	1.60	2.74	6.60	16.03
DAL-MSE-150-small	0.80	2.39	2.93	4.87	2.29	5.04	10.09	22.97
DAL-MSE-150-large	0.35	1.29	0.99	2.00	1.46	2.03	4.40	10.83
DAL-MSE-200-small	1.13	2.26	2.54	3.86	3.49	5.38	8.41	18.54
DAL-MSE-200-large	0.57	0.72	1.08	1.44	0.68	1.35	2.79	6.91
DAL-MSE-250-small	0.70	1.27	2.12	3.74	1.96	3.25	6.91	14.57
DAL-MSE-250-large	0.10	0.30	0.49	0.96	0.42	0.71	1.47	4.20

Table F7: The p -median problem benchmarking (Gap_{PM}) results for large-scale instances for DAL-MSE models across grid resolutions and feature sizes.

Learning Model	250-5	250-10	250-25	250-50	500-5	500-10	500-25	500-50
DAL-MSE-50-small	7.65	12.40	25.41	49.46	8.42	14.31	23.58	44.46
DAL-MSE-50-large	6.46	7.64	16.65	33.53	6.18	6.45	15.50	31.29
DAL-MSE-100-small	3.10	6.89	13.71	26.04	4.81	8.01	14.13	23.41
DAL-MSE-100-large	1.86	2.95	6.39	12.97	2.98	3.58	7.14	13.36
DAL-MSE-150-small	2.30	4.29	8.86	17.27	1.71	4.54	7.80	15.96
DAL-MSE-150-large	1.13	1.68	3.80	7.94	1.50	2.08	4.18	7.76
DAL-MSE-200-small	3.14	5.52	7.80	13.55	5.08	7.90	8.21	13.22
DAL-MSE-200-large	0.76	1.25	2.49	5.06	1.21	1.76	2.65	4.90
DAL-MSE-250-small	2.16	3.18	6.09	11.23	2.00	3.62	6.22	11.26
DAL-MSE-250-large	0.51	0.67	1.45	3.28	0.59	0.84	1.50	3.34

The comparison also shows that feature richness cannot be replaced entirely by grid refinement. DAL-MSE-200-large has lower MAE, RMSE, and MAPE than DAL-MSE-250-small and outperforms it across every ATSP and p -median benchmark. Thus, both a finer grid and the additional map-aware features contribute materially to performance. These ten DAL-MSE variants provide a focused robustness check supporting the grid-resolution and feature-size ablations in the main body without requiring all 90 trained models to be reported under every optimization setting.

References

Altinel IK, Oommen J, Aras N (1997) Vector quantization for arbitrary distance function estimation. *INFORMS Journal on Computing* 9(4):439–451.

- Ballou RH, Rahardja H, Sakai N (2002) Selected country circuitry factors for road travel distance estimation. *Transportation Research Part A: Policy and Practice* 36(9):843–848.
- Boyacı B, Dang TH, Letchford AN (2021) Vehicle routing on road networks: How good is Euclidean approximation? *Computers & Operations Research* 129:105197.
- Brodsky I (2018) H3: Uber’s hexagonal hierarchical spatial index. Uber Engineering, URL <https://www.uber.com/us/en/blog/h3/>, published June 27, 2018; accessed September 8, 2026.
- Elmachtoub AN, Grigas P (2022) Smart “predict, then optimize”. *Management Science* 68(1):9–26.
- Geisberger R, Sanders P, Schultes D, Vetter C (2012) Exact routing in large road networks using contraction hierarchies. *Transportation Science* 46(3):388–404.
- Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning* (Cambridge, MA: MIT Press).
- Helsgaun K (2017) An extension of the Lin-Kernighan-Helsgaun TSP solver for constrained traveling salesman and vehicle routing problems. Technical report, Roskilde University.
- Hu Y, Kallus N, Mao X (2022) Fast rates for contextual linear optimization. *Management Science* 68(6):4236–4245.
- Kallus N, Mao X (2023) Stochastic optimization forests. *Management Science* 69(4):1975–1994.
- Kou S, Golden BL, Poikonen S (2023) Optimal TSP tour length estimation using Sammon maps. *Optimization Letters* 17(1):89–105.
- Kwon O, Golden BL, Wasil EA (1995) Estimating the length of the optimal TSP tour: An empirical study using regression and neural networks. *Computers & Operations Research* 22(10):1039–1046.
- Laporte G, Nickel S, da Gama FS (2015) Introduction to location science. *Location Science*, 1–18 (Springer).
- Love RF, Morris JG (1979) Mathematical models of road travel distances. *Management Science* 25(2):130–139.
- Love RF, Üster H (2001) A statistical comparison of three goodness-of-fit criteria used in modelling distances. *Journal of Applied Mathematics and Decision Sciences* 5(4):235–251.
- Luxen D, Vetter C (2011) Real-time routing with openstreetmap data. *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 513–516, GIS ’11 (New York, NY, USA: ACM), ISBN 978-1-4503-1031-4.
- Macowski C, Saragih A, Guter W, Russell T, Heinold A, Lekkakos S (2025) SCGraph: A dependency-free python package for road, rail, and maritime shortest path routing generation and distance estimation. Technical report, Center for Transportation & Logistics, Massachusetts Institute of Technology, working paper.
- Mandi J, Kotary J, Berden S, Mulamba M, Bucarey V, Guns T, Fioretto F (2024) Decision-focused learning: Foundations, state of the art, benchmark and future opportunities. *Journal of Artificial Intelligence Research (JAIR)* 80:1623–1701.
- Sadana U, Chenreddy A, Delage E, Forel A, Frejinger E, Vidal T (2025) A survey of contextual optimization methods for decision-making under uncertainty. *European Journal of Operational Research* 320(2):271–289.
- Sugishita N, Grothey A, McKinnon K (2024) Use of machine learning models to warmstart column generation for unit commitment. *INFORMS Journal on Computing* 36(4):1129–1146.
- Üster H, Love RF (2001a) Application of a weighted sum of order p to distance estimation. *IIE Transactions* 33(8):675–684.
- Üster H, Love RF (2001b) On the directional bias of the ℓ_{bp} -norm. *European Journal of Operational Research* 128(3):664–673.
- Varol T, Özener OÖ, Albey E (2024) Neural network estimators for optimal tour lengths of traveling salesperson problem instances with arbitrary node distributions. *Transportation Science* 58(1):45–66.